How to Choose an Free and Open Source Integrated Library System

Tristan Müller
Fondation pour une Bibliothèque Globale, Québec, Canada

Published on OCLC Systems & Services: International digital library perspectives. Vol. 27, no. 1, 2011, pp. 57-78.

Emerald Group Publishing Limited / 1065-075X / DOI 10.1108/10650751111106573

The current issue and full text archive of this journal is available at
www.emeraldinsight.com/1065-075X.htm

Abstract

Purpose: This paper presents the results of an analysis of 20 free and open source ILS platforms offered to the library community. These software platforms were subjected to a three step analysis, whereby the results aim to assist librarians and decision makers in selecting an open source ILS, based on objective criteria.

Methodology: The methodology applied involves three broad steps. The first step consists of evaluating all the available ILS and keeping only those that qualify as truly open source or freely licensed software. During this step, we measured the correlation between the practices within the community and the terms associated to the free or open software license. The second step involves 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 entails subjecting the remaining ILS to an analysis of almost 800 functions and features to determine which ILS are most suited to the needs of libraries. The final score is used to identify strengths, weaknesses and differentiating or similar features of each ILS.

Findings: More than 20 open source ILS's were submitted to this methodology but only 3 passed all the steps: Evergreen, Koha, and PMB. The main goal is not to identify the best open source ILS, but rather to highlight which, from the batch of dozen open source ILS, librarians and decision makers can choose from without worrying about how perennial or sustainable each open or free project is, as well as understanding which ILS provides them with the functionalities to meet the needs of their institutions.

Practical implications: This paper offers a basic model so that librarians and decision makers can make their own analysis and adapt it to the needs of their libraries.

Originality/value: This methodology meets the best practices in technology selection, with a multiple criteria decision analysis. It can also be easily adapted to the needs of all libraries.

Keywords: Free and Open Source ILS, software evaluation, computer software, Evergreen, Koha, PMB

Paper type: Research paper

I. Introduction

Integrated library systems (ILS) are multifunction, adaptable software applications that allow libraries to manage, catalog and circulate their materials to patrons. In choosing ILS software, libraries must base their decision not only on the performance and efficiency of the system, but also on its fundamental flexibility to readily adapt to the future demands and needs of their patrons.

With the advent of the Internet, the World Wide Web and open source technologies, developers, contributors and open source software users have turned more and more to free and open source software (FOSS) library solutions. Since the emergence of such technology ten years ago, developers have continuously increased the offerings of fast, freely-available ILS software. Increasingly, the quality of FOSS products is easily comparable to that of proprietary ILS products developed by commercial software vendors.

However, before the current explosion of FOSS ILS offerings, how could librarians who wanted to switch to FOSS be sure they were getting a truly turnkey, free and open-source solution? Knowing also that many FOSS become inactive after a few years, are these ILS solutions sustainable? How do libraries determine which FOSS solution best fits their needs? Unfortunately a lack of good information often keeps library information professionals from embracing FOSS technologies. Many harbor doubts with regards to the functionality and performance of such software, given the complex nature of library systems management. Because FOSS ILS solutions are free and relatively new, many libraries consider such options too risky to consider and undertake.

II. Object of this Article

This article is intended to be a reference tool for library policy makers interested in acquiring an FOSS ILS for their institution. Using the proposed methodology, policymakers can significantly shorten lead times and improve the efficiency of the decision making process with regards to selecting the best FOSS ILS solution for their needs. The prescribed methodology meets the best practices standards in selecting and implementing new technologies and can easily be adapted to the needs of all libraries.

The focus of this article's assessment method is not to identify the best FOSS ILS software, but rather to highlight the best products available to libraries without having to worrying about future sustainability and functionality issues.

Respecting that every library has its own particular needs, it is not possible to tailor an analysis to every type of existing library. However, our proposed analysis model is one that policy makers can rely upon for making a decision. The methodology has been developed to assist in the decision making process and the resulting analysis is presented below.

III. Selection and Evaluation Guidelines

In order to better filter down appropriate FOSS ILS software for large libraries, software products aimed at smaller libraries or even personal collections were not considered for this research.

It is not uncommon that FOSS may be updated several times per year and for the purpose of this study, the most recent versions were selected.

To further limit our analysis, only the default functionality, available without changing or personalizing settings was considered.

IV. Methodology

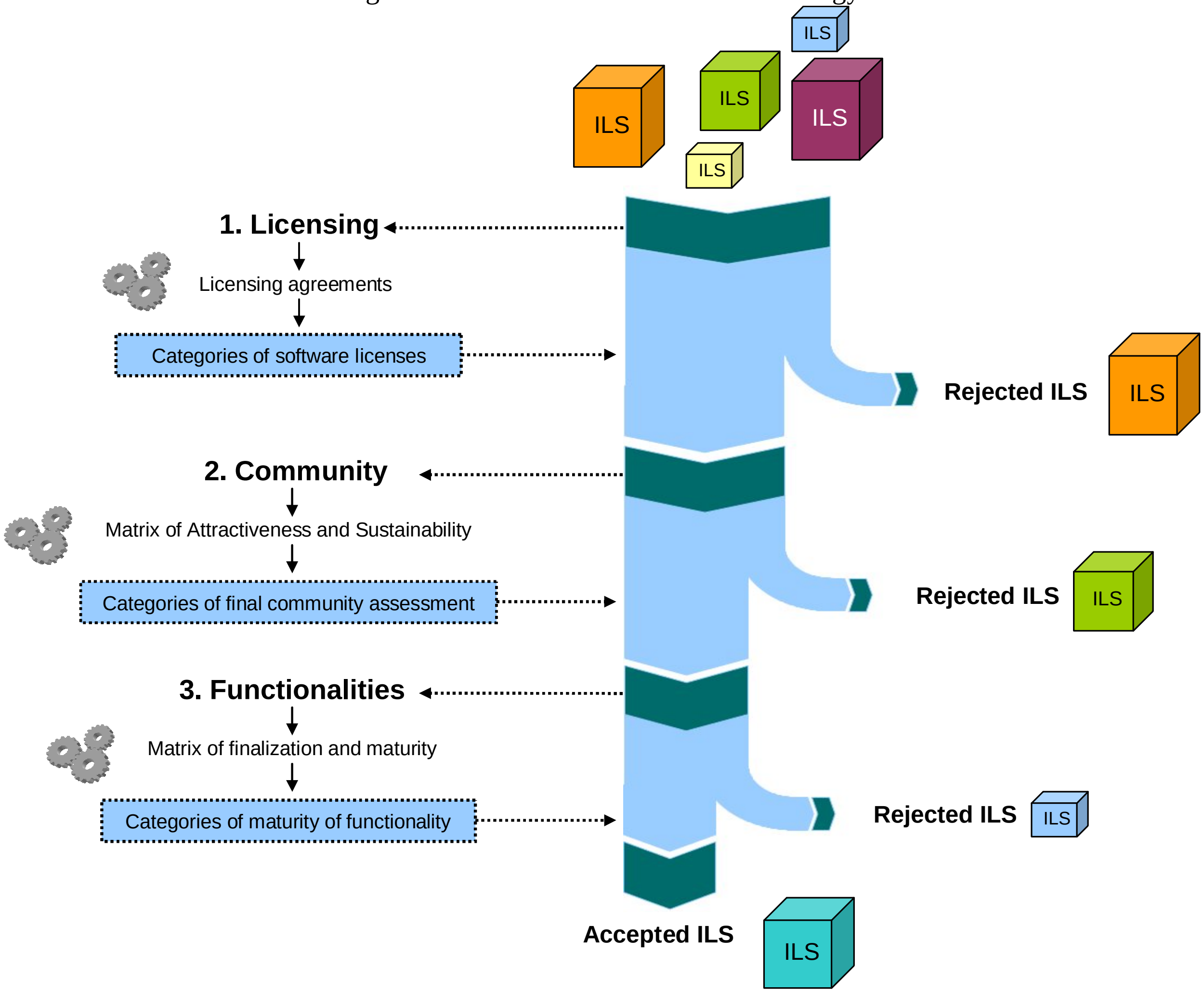
The proposed methodology follows the principle of observing and analyzing the flow of energy (or wave flow) through a filter. In the case of our study, three filters or steps were set up through which only the most suitable FOSS ILS will pass through and be recommended. This process is designed to uncover the true free and open software options that incorporate attractive and sustainable development communities and usable features that meet the needs of libraries. Each step of the study will analyze a specific tool by implementing its own evaluation criteria. At each stage the functionality of a specific tools is evaluated and then ranked by category. These ranked positions indicate whether an FOSS ILS will continue to be considered in the course of the study or rather rejected from further consideration.

Initially, a select grouping of FOSS ILS were assembled together, forming the metaphorical tree trunk of our study. The width of this trunk's core was equal to the total number of ILS under consideration. As the results of each stage of assessment were completed, branches emanating away from the trunk indicate rejected FOSS products while the continuing trunk represent products still under consideration.

The three analyses that software underwent are the following:

- Evaluation of software licensing
- Evaluation of the community
- Evaluation of functionalities

Figure 1 : FOSS ILS Selection Methodology


ILS
ILS
ILS
ILS
ILS
1. Licensing
Licensing agreements
Categories of software licenses
Rejected ILS
ILS
2. Community
Matrix of Attractiveness and Sustainability
Categories of final community assessment
Rejected ILS
ILS
3. Functionalities
Matrix of finalization and maturity
Categories of maturity of functionality
Rejected ILS
ILS
Accepted ILS
ILS

1. Evaluation of ILS licensing agreements

Is the application truly an free or open source software as defined by license agreements? At this stage, we assess the relationship between the development community's usage privileges and defined usage restrictions of the software based upon licensing contracts. FOSS are then sorted and ranked based upon usage rights.

Evaluation Criteria

Just because software are offered under the designation of “free license”, as with the example of GPL, does not necessarily mean that all aspects of the product are free and open to use. By a close analysis of FOSS ILS communities it becomes possible to assess user permissions and rights and to best determine the definition of free.

For example, free licensing generally requires a software's source code to be freely available. A community that does not make its source code available inadvertently inhibits the necessary freedom needed by the community to study, change and improve the program. So in addition to weakening the basic usage of such restricted free software, it equally diminishes the dynamics of the attached development community. Developers, users and potential contributors are all denied access to more closely study the FOSS code. A community that agrees in advance to a truly open application essentially demonstrates a higher level of transparency, openness and user confidence in its software product.

Software Categories

By studying software licenses we were able to place each of the FOSS ILS into one of the seven below categories of software. The FOSS ILS that fall into the categories of “Free Software” or “Open Source Software” was kept.

Table 2 : Categories of software licenses by usage rights

Categories of software licenses
Public domain
Free Software
Open Source Software
Freeware
Shareware
Proprietary Software
Patented Software

2. Evaluation of FOSS ILS Community

Is a FOSS community engaged and dynamic? At this stage we began to analyze the development communities affiliated with each software product, studying both their attractiveness and sustainability. Each of the communities was analyzed based upon a series of criteria whose results were then compiled and sorted into a table matrix. The final scoring of these ILS allowed them to be placed into one of the four categories of free software communities (see table 3).

When choosing a free software, you are not only choosing functionalities, but also choosing to become part of the affiliated FOSS community. It's best to always carefully consider the software community first before selecting a product, because it is this community that ultimately supports usability and determines future development. The size and level of activity of a software community has a direct impact on all aspects of FOSS development. Therefore, assessing a community is actually assessing the usability, attractiveness and sustainability of a free software product.

Attractiveness and Sustainability

Attractiveness and user interest levels in a software product help to assess the influence of users on any given project and also predict the levels of community support. Sustainability is in many ways a measure of broad community support and interest (of time, energy or money) that is likely to continue into the future. Sustainability is not measured only at a given time and must logically be assessed over extended periods of time.

The attractiveness and sustainability of a FOSS ILS community can be assessed by measuring the best practices implemented by these communities to ensure continuing development and constant evolution of their software. Initially, this is determined by the existence of a critical mass of developers, contributors and users all working on the software. The natural establishment of procedures, methods, practices and tools developed by these groups, with the issues of performance, stability and adequate levels of security in mind, allow for innovations in the process of FOSS development innovation while also ensuring sustainability.

When a community scores highly in this assessment stage, one cannot necessarily assume it ensures a more reliable, sustainable, interoperable, relevant or effective FOSS project. But, the score is indicative of an opportunity for reliability, durability and interoperability.

Evaluation Criteria

For assessing the FOSS communities, we designed a table matrix of 40 community assessment criteria grouped into 18 categories organized into two axes (see appendix). For each of these criteria and categories within the table matrix we specified a weight and then assessed each criterion by assigning them a rating.

Final Community Assessment

The final score resulting from the table matrix of the community is used to position the ILS into one of four categories of final community assessment (sustainable, emerging, just released or inactive). The ILS that ranked in the categories “sustainable” or “emerging” was selected for further analysis. The software that was placed into the categories of “just released” or “inactive” was excluded from the remainder of the study.

Table 3 : Categories of final community assessment

Score	Category
5 to 7	Sustainable
3.5 to 5	Emerging
0 to 3.5	Just released
n/a	Inactive

3. Evaluation of FOSS ILS Functionality

Does the functionality of FOSS ILS meet users' needs? At this stage of the research, we analyzed the functionality of each software with regard to efficiency and usability. We evaluated the features of each ILS based on a series of criteria and then compiled the results into a table matrix. The final score was then used to identify strengths, weaknesses and the distinctive features of each product. The resulting analysis is specifically meant for use by library decision makers of bodies in identifying the FOSS ILS most relevant to the specific needs of their library.

Maturity

Based on the criteria of user satisfaction and usability, the area of maturity is the component that measures the degree of integration of processes or functionality that make up part of the software. It is really the success of this particular element that allows a thorough assessment of overall functionality and comparison of integrated modules in the various ILS under consideration.

Maturity is also associated with growth of a software over time. When a child moves from childhood to adulthood, it becomes increasingly independent and takes on more responsibility at each stage of life, becoming increasingly more mature and refined. Software also passes through stages of maturation. Early versions of software often offer relatively few features and can be filled with defects. Subsequent versions typically improve and offer additional features and greater stability.

Without measuring a feature's maturity it would be difficult to compare the similar functionality of the various software products under consideration. The truth is that the implementation of a particular functionality or application within an ILS software has little to do with the quality of its performance or whether it even works. However, a functionality that is barely implemented in one software product may be truly remarkable in another. It is precisely this difference that is reflected in the results of each criteria and category ranking.

To determine the maturity of software functionality, we created the necessary circumstances so that targeted ILS functions were properly expressed and demonstrated. We then evaluated whether a particular function responded well enough to the insinuated need to determine usability, which is remains the key measure of quality when comparing system interfaces.

Finalization

This is simply the measure of the amount of features available for each of the FOSS ILS software based upon the total amount of functionality expected in a typical ILS software. The resulting score was then expressed as a percentage, with 100% representing the ideal score.

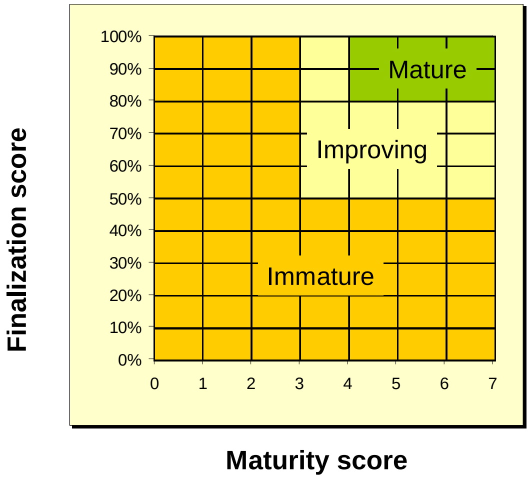
Evaluation Criteria

For assessing the functionality of the FOSS ILS, we designed a table containing 799 different criteria, grouped into 50 categories and 8 modules. These criteria represented the typical automation needs of a standard library. For each of these criteria, categories and modules within the table matrix we specified a weight and then assessed each criterion by assigning them a rating.

Categories of Maturity of Functionality

The final scores of the resulting table matrix for finalization and maturity were used to rank the ILS into one of three categories of maturity of function; “mature”, “improving” or “immature”. The FOSS ILS that ranked as “mature” was then selected for further consideration. The software placed in the categories of “improving” or “immature” was excluded from the remainder of the study.

Figure 4 : Categories of maturity of functionality



V. Decision matrices

Using decision matrices to help organize the decision making process is commonly used by libraries when choosing a new automation software. The exercise allows them to select the ILS that is most suited to the needs and operations of their library.

In this study, we used two decision matrices that incorporated the method of weighted sums. One matrix for the evaluation of the FOSS ILS communities and the other for the evaluation of software functionality. For each of these two matrices, we proceeded as follows:

1. Established evaluation criteria.

2. Assigned a weight to each criteria based upon its importance.
3. Created an L-shaped table matrix to organize data and show results.
4. Assigned a rating for each ILS according to the established criteria.
5. Calculate the scores.

Weight

We have defined a scale of three weights. Each of these weights indicates the importance of a particular criteria based upon certain situations. The same scale of weights was used in both the table matrix of evaluation for ILS communities and for the table for the evaluation of functionality.

Table 5 : Weights

Weight	Meaning
3	Mandatory criteria: criterion essential or very important requirements.
2	Desirable criteria: major criterion or highly desirable.
1	Optional criteria: criterion desirable, but not important.

Rating

The points scale defined in the following rating system was used to arrive at the scores found in the evaluation table matrix of ILS communities as well as for the criteria used to rank maturity and finalization in the evaluation of functionality table matrix.

Table 6 : Rating

Rating	Meaning
7	Remarkable - no fault found
6	Excellent – almost no improvements needed
5	Very good
4	Good
3	Satisfactory, but with considerable room for improvement
2	Barely acceptable
1	Very poor - not desirable
0	Not available

Table 7 : Matrix example

			OPTIONS					
			Option A		Option B		Option C	
Categories and criteria	Weight	Weights sum	Rating	Score	Rating	Score	Rating	Score
Category 1	3	6		1.7		2.3		2.2
Criterion C1	1		3	3	3	3	3	3
Criterion C2	2		2	4	1	2	2	4
Criterion C3	3		1	3	3	9	2	6
Category 2	1	5		2		1.4		2.6
Criterion C4	3		2	6	1	3	3	9
Criterion C5	2		2	4	2	4	2	4
Total		4		1.9		1.9		2.4

1. Calculation of Scores Criteria

In order to properly calculate the score for each of the criteria, the ratings of each criterion must be first multiplied by the weight.

Score = Rating x Weight

2. Calculation of Scores Categories

In the case where the criteria are grouped into categories, a weight is also assigned to each category to indicate its relative importance in the evaluation process. The score of each category is calculated by adding up all the scores of all criteria in the category and then dividing that by the total amount of the weights for all criteria in the category.

Score category = Sum of scores of criteria for a category / Sum of weights of criteria for a category

3. Calculation of Final Score

The final score is calculated by adding up the multiplied totals by its weight and then dividing by the sum of all weighted categories.

Final score = (Score x Weight Category 1) + (Score x Weight category 2) + (Score x Weight of category n) / Sum of weighted categories

VI. Results

1. Evaluation of ILS licensing agreements

Table 8 : Classification of ILS Software Based on categories of software licence

Software category	ILS name
Public domain	
Free Software	Emilda, EspaBiblio, Evergreen, Gnuteca, InfoCID, Jayuya, Koha, NewGenLib. oBiblio, OPALS, OpenAmapthèque, OpenBiblio, PhpMyLibrary, PMB, Senayan
Open source software	
Freeware	Avanti 1.0, BiblioteQ, Open MarcoPolo, Weblis, LearningAccessILS
Shareware	
Proprietary software	
Patented software	

Avanti 1.0 is considered freeware because it is possible to download the software free of charge. However, it is not considered an Open source software because its source code is protected by its licensing agreement. It should also be noted that Avanti 2.0 has become a proprietary software. This software was not selected for further study.

For the software LearningAccessILS, licensing terms could not be determined. There does not appear to be any links to development information on its website. Users must obtain a formal authorization to download and run the software. It does not appear under be distributed under a recognized free license agreement. This software was not selected for further study.

BiblioteQ and WebLis are both considered to be freeware since both can be downloaded for free. However, they are not subject to licensing and their source codes are also not available. These programs were not selected for further study.

Open MarcoPolo, although subject to the GPL is not a free software. As well, its source code is unavailable to developers. It was determined to be freeware but not selected for further study.

All other ILS including Emilda, Evergreen, Gnuteca, InfoCid, Jayuya, Koha, NewGenLib, oBiblio, OPALS, OpenAmapthèque, OpenBiblio, PhpMyLibrary, PMB and Senayan, are considered free software based on licensing agreements. All of these programs were selected for further study.

2. Evaluation of FOSS ILS Communities

Table 9 : Classification of FOSS ILS based on Community

Score	Category	FOSS ILS name
5 to 7	Sustainable	Evergreen, Koha
3.5 to 5	Emerging	PMB
0 to 3.5	Just Released	Gnuteca, InfoCID, NewGenLib, oBiblio, OPALS, OpenAmapthèque, Senayan
n/a	Inactive	Emilda, EspaBiblio, Jayuya, OpenBiblio, PhpMyLibrary

Inactive Communities

Emilda, Jayuya, OpenBiblio and PhpMyLibrary are considered inactive because no development activity has been noted in their source codes or on their discussion lists for several years. These programs were not selected for further study.

Just Released Communities

Gnuteca, InfoCid, NewGenLib, oBiblio, OPALS, OpenAmapthèque, and Senayan are considered to be just released because their communities do not currently have a critical mass of developers, contributors and users. Some procedures, methods, practices and tools remain unimplemented and are not freely available to ensure sustainability. These programs were not selected for further study.

Emerging Communities

PMB is considered emerging because the software has a growing community of developers, contributors and users. However, the community has failed to implement procedures, methods, practices and tools to ensure sustainability. For example there is currently a lack of good access to the software infrastructure and codes hindering the development community from pushing forward towards future development goals. Also, there is no clear renewal or development of leadership, no existence of control procedures or support, no list of bug tracking or schedule of further development, etc. Even though this software has a minimal qualification, it was sufficient to stay in the study.

Sustainable Communities

Koha and Evergreen are considered sustainable communities because they both have obtained a critical mass of interested developers, contributors and users. Both communities have managed to put in place a very solid collaborative infrastructure made of development tools to help manage goals, function, architecture and design, outlining the responsibilities of developers and contributors - concurrent versioning system (CVS), bug tracking, translation, development planning, etc. These procedures, methods, practices and tools help to reinforce sustainability. These software were retained for further study.

About ILS retained

Although fairly dynamic and responsive, the community of PMB did not demonstrate some of the following basic best practices of attractiveness and sustainability:

- the unavailability of a strong infrastructure for public collaboration and commitment to future access and transparency. There is no public CVS, bug tracking or development schedule, only discussion lists;

- the geographical concentration of developers and contributors, about 10 people mainly from PMB Services, is based in France;
- the presence of a single development corporation (PMB Services).

The Koha community shows great attractiveness and sustainability because of the following:

- a good number of companies and business consultants providing worldwide support for the product. More than 19 companies throughout the world can be identified each offering software counselling, support, training, accommodation, etc. ;
- a very high degree of integration by adopting two Marc formats (Marc21 and Unimarc) and system availability in more than 25 languages;
- its ability to effectively structure their community and its contributions;
- the development and renewal of its leadership. There is a different administrator for each version;
- the size of its community of developers and contributors (over 50 people) is spread across dozens of countries.

The Evergreen community also shows great potential based on its vitality and sustainability:

- the rapid growth of its community of developers and contributors;
- the high caliber of its developers and contributors. They come from consortia of public libraries and academic library systems in the United States and Canada;
- the high levels of user satisfaction as demonstrated by the annual user survey and journals publications.

3. Evaluation of FOSS ILS Functionality

From the previous of products, only three FOSS ILS (Evergreen, Koha and PMB) were selected for the evaluation of functions. Depending on the priorities of each library, these results may vary.

Figure 10 : Summary of finalization and maturity of FOSS ILS

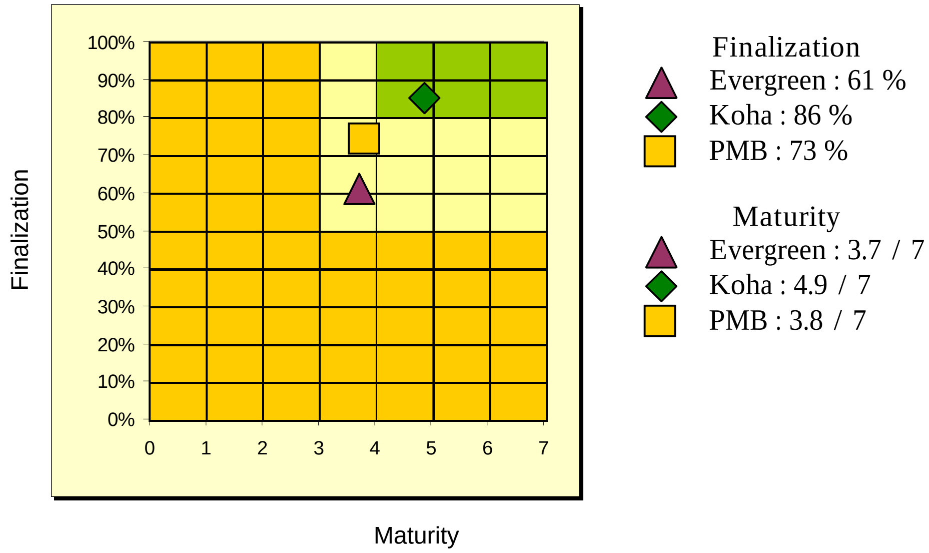


Table 11 : Classification of FOSS ILS based on Maturity of Functionality

Categories	FOSS ILS name
Mature	Koha
Improving	Evergreen, PMB
Immature	

Networking and sharing

Evergreen and Koha were both born out of consortial administration of resources spread among multiple libraries and this is reflected in the software's internal operations by the number of options and its overall ease of implementation. Shared administration is not limited to the catalog and applies also to bibliographic records, authority records, categories of documents, patron records, patron preferences, budgetary funds, etc. It is also possible to treat all branches as independent, but still share a common catalog.

Settings

These three ILS systems offer a myriad of options that allow the system to be set up in accordance with existing library policies: eg. circulation policy, locations, documents types, budgets, customer classification, data entry, Z39.50 servers, etc.

Koha is the ILS that offers the most international options: choice of date format (U.S., metric, iso), choose by type of MARC record (UNIMARC or MARC21) and choice of 25 languages. Koha is the only one to offer options related to FRBR, OAI, event logs and reporting based on OPAC usage.

PMB provides options for SDI (Selective Dissemination of Information Service), reindexing, an interface for database back up and import/export of bibliographical records, the determination of default values (eg. when cataloging materials), choice of interface languages (9 languages), choice of interfaces and friendly Web-based OPAC.

Settings for Evergreen are not directly accessible to librarians because it is coded in the system programming. This ensures security especially important to the project as this ILS is often used by very large library networks containing between 15 and 270 branches. There are two levels of administration settings, one for the system administrator and another accessible only to the director of library operations.

Report Generating

Evergreen stands by the quality and scope of its report generator. Built intuitively, it is easy to use. It is easy to query all desired data within the software database even without knowing any programming language. Other FOSS ILS possess, in addition to the default reports, less sophisticated report generators than Evergreen's and which overall tend to be more difficult to manipulate.

Koha has its own report generator to query the data and produce reports. This does requires some experimentation in the choice of data options in order to obtain the desired end result.

PMB also has, in addition to a default report generator, a module to generate specific reports. It is thus possible to query all data together, but this requires a basic knowledge in SQL query language, which may be a limiting factor for library staff.

Managing Privileges and Permissions

The three FOSS ILS software all have a system to assign user privileges based upon the user. In all cases, you must first create a user account and then select the rights and privileges of that particular user.

Evergreen has the most comprehensive security system of all the FOSS ILS that were considered. After creating a user account, you can select a predefined library security group, according to the profile of the employee (for example: cataloger, managing cataloger, etc.). It is also possible to refine the security and permissions for 150 other actions and functions. The assignment of permissions can also be defined by library bus, branch library, library network, library system and even consortium. Lastly, permissions can also be selected by branch locations.

The security systems of Koha and PMB offer only groups of features rather than individual functions. In a library containing a large number of employees and, consequently, specialized tasks, these system limitations may not be sufficient.

MARC Format

Koha offers the possibility to set, add and modify zones and subzones.

Evergreen works only with the standard MARC 21 and it is possible to catalog directly into the MARC format field.

PMB is based on the Unimarc format and did not allow direct cataloging in any format.

Authority Control

Koha and PMB both have modules for authority control, institutions, names, subject headings, authors or other type of data. Authority control in Koha is based on the MARC21 standard. PMB does not handle authority control based on an established standard.

Evergreen has no module for the authority control.

Search Options

The three FOSS ILS all offer excellent research options when searching the OPAC, allowing the use of Boolean operators as well as truncation of characters.

Evergreen and Koha offer by default a simple search in a streamlined interface similar to Google's search engine interface. It is also possible to access advanced search options if you want to refine the search by combining multiple fields with Boolean operators, limiting the search and sorting the results.

PMB offers five types of research: simple (default), combined (combination of several fields with Boolean operators), terms (specified by the library), by tag (specified by the patron) and external (searching several external databases or catalogs outside of PMB). This variety of research options may even serve to confuse researchers. Even though it is only possible to limit searches by document type, PMB is capable of sorting the results by several different parameters.

Koha and PMB both offer the ability to save searches queries by RSS.

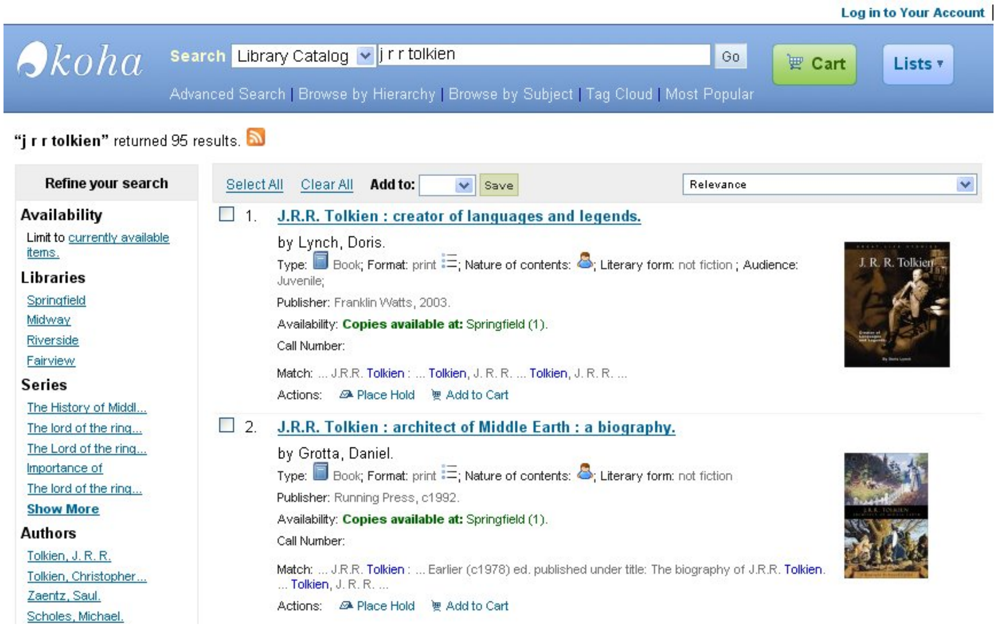
Presentation of Research Results

All FOSS ILS included in the study presented results initially in a simple format, and by clicking on them, it then accesses the results in a full record format. It is possible to navigate through the results by foraging and to conduct another search by clicking, for example, the name of the author, which will display all records by that author.

In Koha and Evergreen, the results were displayed on the screen similar to that of the Amazon.com format. Both present default results in order of relevance. Both also offer the possibility to browse the results by facets: availability of documents, libraries, authors, subjects, collections, etc.

In PMB, the default is limited to 10 possible results. You must click on the button “Show all records” to be able to view and navigate through the results pages.

Figure 12 : Displaying Search Results in Koha



Log in to Your Account

koha
Search
Library Catalog
Go
Cart
Lists ▾
Advanced Search | Browse by Hierarchy | Browse by Subject | Tag Cloud | Most Popular

j r r tolkien

"j r r tolkien" returned 95 results.

Refine your search

Availability

Limit to currently available items.

Libraries

Springfield
Midway
Riverside
Fairview

Series

The History of Middl...
The lord of the ring...
The Lord of the ring...
Importance of
The lord of the ring...
Show More

Authors

Tolkien, J. R. R.
Tolkien, Christopher...
Zaentz, Saul.
Scholes, Michael.

Relevance

□ 1. J.R.R. Tolkien : creator of languages and legends.

by Lynch, Doris.

Type: Book; Format: print; Nature of contents: ; Literary form: not fiction ; Audience: Juvenile;

Publisher: Franklin Watts, 2003.

Availability: **Copies available at:** Springfield (1).

Call Number:

Match: ... J.R.R. Tolkien : ... Tolkien, J. R. R. ... Tolkien, J. R. R. ...

Actions: Place Hold Add to Cart

□ 2. J.R.R. Tolkien : architect of Middle Earth : a biography.

by Grotta, Daniel.

Type: Book; Format: print; Nature of contents: ; Literary form: not fiction

Publisher: Running Press, c1992.

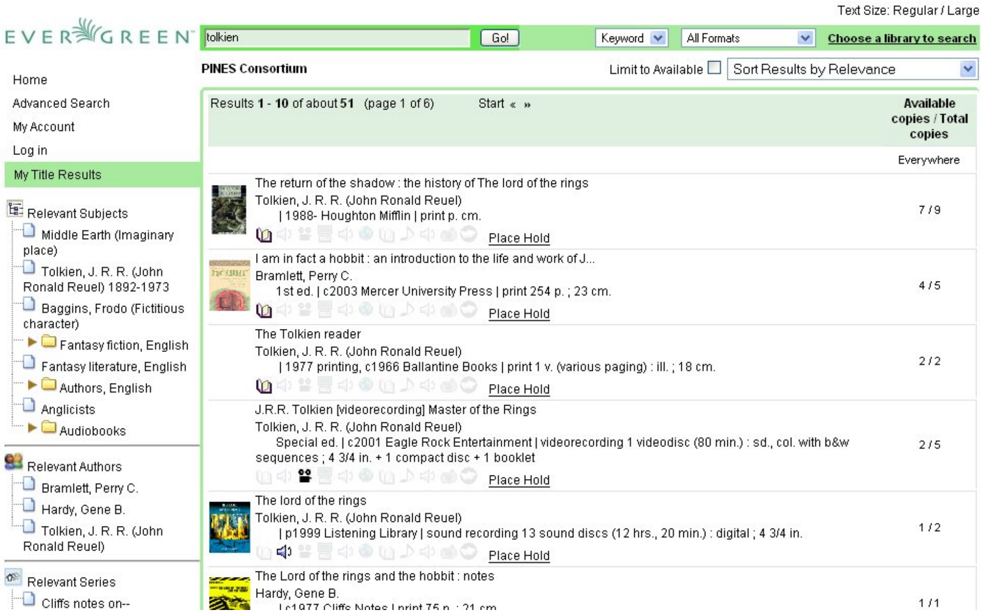
Availability: **Copies available at:** Springfield (1).

Call Number:

Match: ... J.R.R. Tolkien : ... Earlier (c1978) ed. published under title: The biography of J.R.R. Tolkien. ... Tolkien, J. R. R. ...

Actions: Place Hold Add to Cart

Figure 13 : Displaying Search Results in Evergreen



tolkien

PINES Consortium

Limit to Available □ Sort Results by Relevance

Results **1** - **10** of about **51** (page 1 of 6) Start « »

	Available copies / Total copies
	Everywhere
The return of the shadow : the history of The lord of the rings Tolkien, J. R. R. (John Ronald Reuel) | 1988- Houghton Mifflin | print p. cm. Place Hold	7 / 9
I am in fact a hobbit : an introduction to the life and work of J... Bramlett, Perry C. 1st ed. | c2003 Mercer University Press | print 254 p. ; 23 cm. Place Hold	4 / 5
The Tolkien reader Tolkien, J. R. R. (John Ronald Reuel) | 1977 printing, c1966 Ballantine Books | print 1 v. (various paging) : ill. ; 18 cm. Place Hold	2 / 2
J.R.R. Tolkien [videorecording] Master of the Rings Tolkien, J. R. R. (John Ronald Reuel) Special ed. | c2001 Eagle Rock Entertainment | videorecording 1 videodisc (80 min.) : sd., col. with b&w sequences ; 4 3/4 in. + 1 compact disc + 1 booklet Place Hold	2 / 5
The lord of the rings Tolkien, J. R. R. (John Ronald Reuel) | p1999 Listening Library | sound recording 13 sound discs (12 hrs., 20 min.) : digital ; 4 3/4 in. Place Hold	1 / 2
The Lord of the rings and the hobbit : notes Hardy, Gene B. | c1977 Cliffs Notes | print 75 p. ; 21 cm.	1 / 1

Relevant Subjects

- Middle Earth (Imaginary place)
- Tolkien, J. R. R. (John Ronald Reuel) 1892-1973
- Baggins, Frodo (Fictitious character)
- ▶ Fantasy fiction, English
- Fantasy literature, English
- ▶ Authors, English
- Anglicists
- ▶ Audiobooks

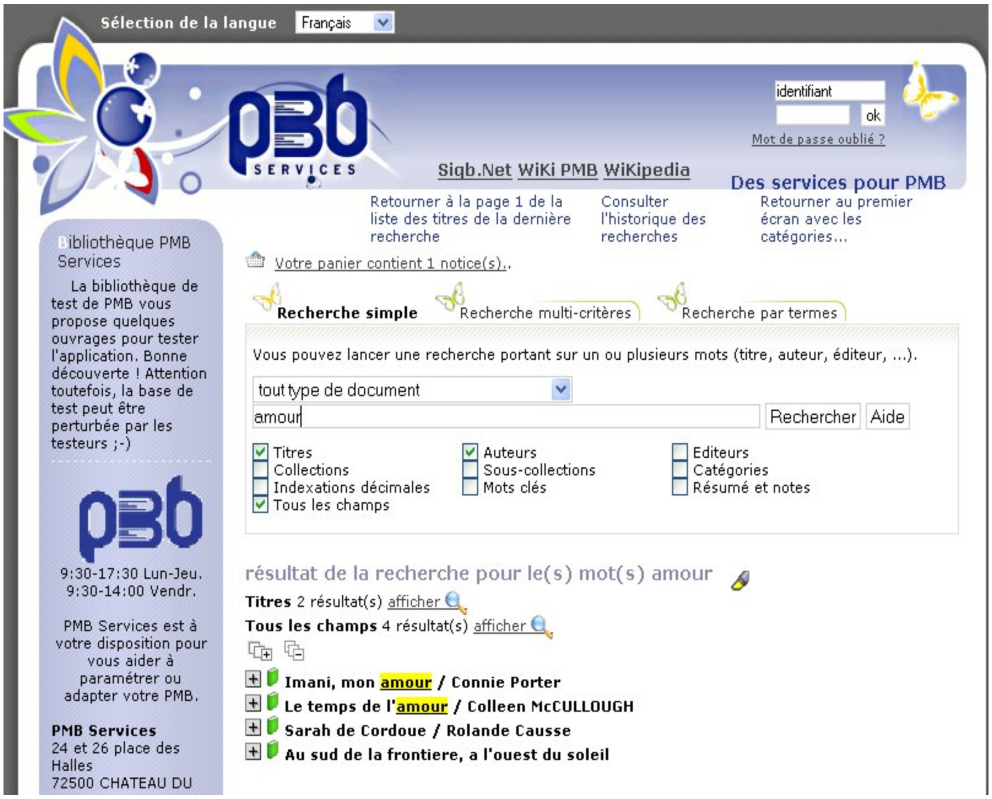
Relevant Authors

- Bramlett, Perry C.
- Hardy, Gene B.
- Tolkien, J. R. R. (John Ronald Reuel)

Relevant Series

- Cliffs notes on--

Figure 14 : Displaying Search Results in PMB



Sélection de la langue Français

identifiant
ok
Mot de passe oublié ?

Siqb.Net WiKi PMB WiKipedia

Des services pour PMB

Retourner à la page 1 de la liste des titres de la dernière recherche

Consulter l'historique des recherches

Retourner au premier écran avec les catégories...

ibliothèque PMB Services

La bibliothèque de test de PMB vous propose quelques ouvrages pour tester l'application. Bonne découverte ! Attention toutefois, la base de test peut être perturbée par les testeurs ;-)

9:30-17:30 Lun-Jeu.
9:30-14:00 Vendr.

PMB Services est à votre disposition pour vous aider à paramétrer ou adapter votre PMB.

PMB Services
24 et 26 place des Halles
72500 CHATEAU DU

Votre panier contient 1 notice(s)..

Recherche simple | Recherche multi-critères | Recherche par termes

Vous pouvez lancer une recherche portant sur un ou plusieurs mots (titre, auteur, éditeur, ...).

tout type de document

amour | Rechercher | Aide

☑ Titres
□ Collections
□ Indexations décimales
☑ Tous les champs
☑ Auteurs
□ Sous-collections
□ Mots clés
□ Editeurs
□ Catégories
□ Résumé et notes

résultat de la recherche pour le(s) mot(s) amour

Titres 2 résultat(s) afficher

Tous les champs 4 résultat(s) afficher

Imani, mon amour / Connie Porter
Le temps de l'amour / Colleen McCULLOUGH
Sarah de Cordoue / Rolande Causse
Au sud de la frontiere, a l'ouest du soleil

Printing

Koha's OPAC offers several options for exporting bibliographic data in different formats (text, MARC 21, MARCXML, MODS, Dublin Core or Agris) as well as options to print, save or email. Koha also incorporates the OpenURL protocol COinS used to create contextual links between references in an article and to the resource itself. This protocol allows one click downloading into bibliographic management software such as Zotero.

The PMB OPAC offers opportunities for exporting bibliographic data in the different formats (UNIMARC, xml, html, text, Bibtext and EndNote) and to also print, save or email.

Circulation

The three FOSS ILS products all offer many automated document circulation functions such as real-time verification and validation as well as account expiration data and access to patron disputes within the patron record. These tools help facilitate circulation both at the time of borrowing, renewal and return of materials to the library.

All products include automated email and mail generators producing notices for patrons based on circulation requests (hold notices, late notices, reserves, etc..) or the receipts of circulation transactions (items borrowed, returned, billed, paid, etc..). All also have the capability to determine library closures/holidays as well as tools for generating specific batch mailing or email notices to patrons.

Although PMB does effectively handle most basic circulation functions, certain of these functions are not as well developed as one might hope. The management of circulation fines is very limited (one rule for all documents, customers and branches) and its the same concerning the rules for renewals. There is no possibility to manage reserves lists. The circulation rules are spread out over several tables and settings and not located in one centralized area.

In all of the software products patrons are able to access their personal accounts by providing an identification number and password. Different options are available to patrons who log in to their accounts such as review of materials on loan, their basic patron information and user profile as well a more detailed description documents on loan. When connected, patrons can also reserve library items, renew materials on loan, update their address and password as well as keeping lists of materials of interest to them.

Acquisitions

In Koha and PMB, it is possible to track orders but this function is limited by three assigned commands: “ordered”, “received” and “completed”. Therefore, all orders are firm and it is not possible to make prepaid orders, according to an approval process, donations, subscriptions or follow-up of backlog. The low scores in this area of the evaluation reflect that the order tracking is not highly developed and there is great room for improvement for both applications in this area of automated operations.

Evergreen does not have a module for managing acquisitions.

Periodicals

Koha and PMB are both able to create rules based on the frequencies of all types of periodical documents. The check-in and tracking function easily accounts for all received materials upon arrival and also manages subscription data and claims information. Koha was the only FOSS ILS studied that offered a function for tracking internal routing lists to patrons. Evergreen has no module for serials management.

Results by ILS

Koha

Koha ILS was ranked the most complete FOSS ILS because:

- the number of functions answered by functionality - routing periodicals, inventory control, authorities, generation of notices to customers, order tracking etc. ;
- the quality of the implementation features.

Evergreen

Evergreen stands by the quality of the implementation of its features:

- extensive access management - security;
- robust construction to support a consortium of libraries (275 libraries in the consortium Pines of the state of Georgia, 9.8 million documents, 15 million of loans in 2007), but also a documentation center;
- power tool for generating reports and statistics.

Evergreen has no module for authority control, acquisitions and serials management (modules in development).

PMB

PMB is very functional ILS, with all necessary basic functions of library automation especially with regard to:

- integration of Web 2.0-oriented features and other patron-based Web services: (review of documents, add comments and tags from users);
- federated search connectors for Z39.50 and OAI;
- safeguarding the security of all data and full system restore from the safety backup;
- the creation of item records on the fly for documents with no bar code.

VII. Final Result

In this section, we identified and analyzed 20 FOSS ILS using rigorous selection methods.

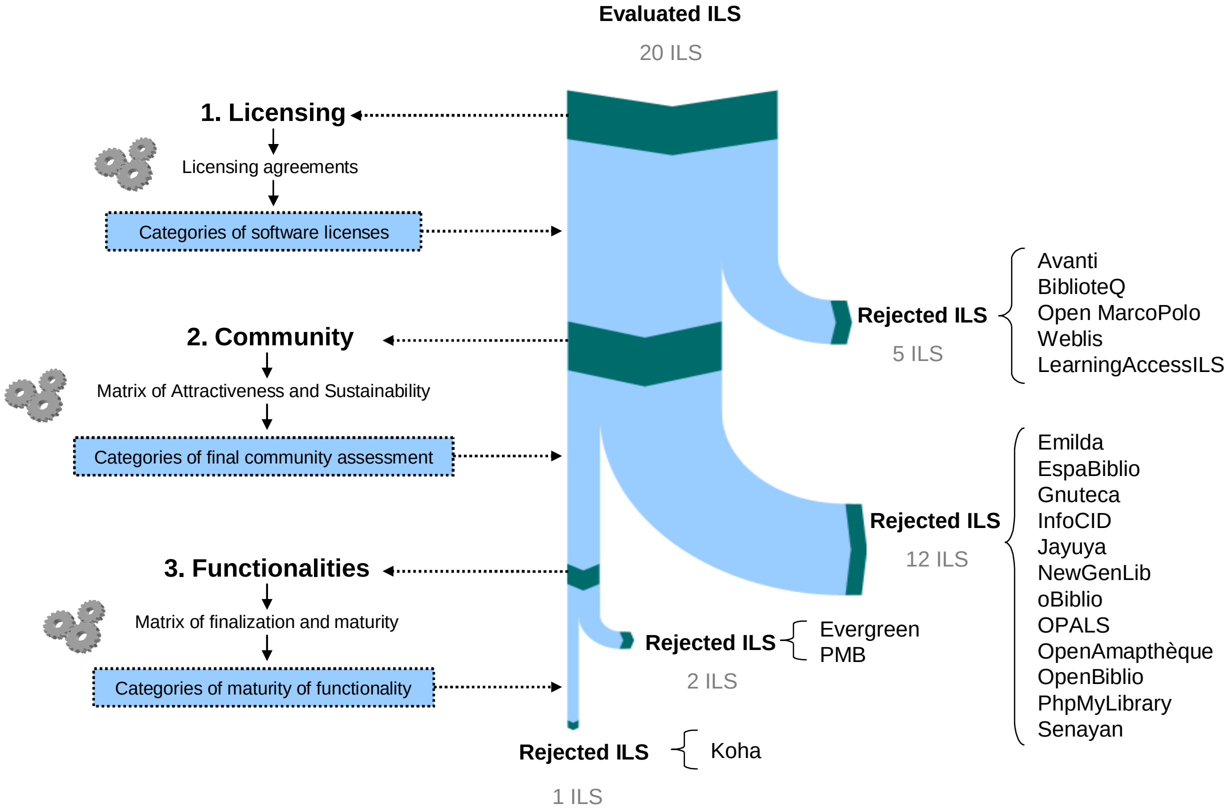
Initially, we analyzed the 20 FOSS ILS in terms of licensing. This was done by studying their licenses and analysing the consistency of product licensing agreements. The result of these observations allowed us to rank each of the ILS in a specific software category. Of these 20 ILS, 5 were rejected as not conforming to a category of free and open source software products.

In a second step, we analyzed the 15 remaining ILS in terms of the affiliated software development communities. This was done by discerning the attractiveness and sustainability of their affiliated communities following 40 established criteria. The result of these observations allowed us to rank each of the ILS into category of communities. Of the 15 ILS, 12 were rejected for lack of established or evolving development communities.

Thirdly, we analyzed the 3 remaining ILS in terms of functions. This was done by observing the maturity and performance of their processes as defined by 799 established criteria. The result of these observations allowed us to rank each of the ILS in a category of maturity of functionality. From these 3 ILS, 2 were rejected for lack of mature functionality.

Finally, one ILS has passed through three elimination rounds: the Koha ILS software product. Koha consistently received the best results at all stages of the study.

Figure 15 : Results of the application of methodological steps for selecting a free ILS



However, although the analysis and the results show a clear winner, Koha is not the only ILS to consider when selecting a FOSS ILS. Both the Evergreen and PMB ILS products should be taken into account and very seriously considered because they both obtained very good results both in the evaluation of communities that function.

The observations and results of Koha and Evergreen PMB indicate that these three ILS can not only meet the needs of libraries, but they remain alternatives that library decision makers must seriously consider.

VIII. Conclusion

This article aims to serve as a reference tool for the selection and acquisition of FOSS ILS but also to correct persisting prejudices against FOSS ILS, including the preconceived risks associated with the implementation and sustainability of ILS FOSS.

According to Tim O'Reilly, founder and CEO of O'Reilly Media and champion of the free and open source software movement there are three important trends in FOSS development: the democratization of access to software, collaborative networking and personalization of software, and an increasing shift toward seeing FOSS as services. FOSS ILS fit perfectly into these three trends. Many are offered free of charge based upon the principles of free licensing and open source software solutions, they are developed through the collaborative efforts of established development communities they often offer robust features, functional, quality and innovation.

It also clear that the spirit of free and open source software development is very similar to the principles of librarianship, as noted by Eric Lease Morgan, a staunch proponent of FOSS in libraries. He explained that libraries encourage intellectual freedom and free access to library resources while FOSS encourage respect for intellectual property and the free distribution of its source code and documents. The two movements maintain a mission of open access and the sharing of knowledge to enrich and advance society.

When a library selects free software, it chooses not only a powerful means of reducing its spending, but also opportunities to become more independent in terms of its choices of business and software vendors, more importantly, it reinforces its primary mission of disseminating information while further justifying its choice of technology in a context of public service.

IX. Bibliography

Licences

Free Software Foundation, available at: http://www.fsf.org/ (accessed 22 March 2008).

Open Source Initiative, available at: http://www.opensource.org/ (accessed 22 March 2008).

St. Laurent, A.M. (2004), *Understanding Open Source and Free Software Licensing*, O'Reilly, Cambridge, available at: http://www.oreilly.com/catalog/osfreesoft/book/ (accessed 25 March 2008).

Free and Open source communities

van den Berg, K. (2005), *Finding Open options. An Open Source software evaluation model with a case study on Course management Systems*, Thesis, Tilburg University, Tilburg, Netherlands, available at: http://www.karinvandenberg.nl/Thesis.pdf (accessed 18 March 2008).

Cornu, J-M. (2004), “La coopération, nouvelles approches”, available at: http://www.cornu.eu.org/files/cooperation1_2.pdf (accessed 26 December 2008).

Druel, F. (2007), *Évaluation de la valeur à l'ère du Web : Proposition de modèle de valorisation des projets non marchands*, Thesis, Institut des Sciences et Techniques de l'Ingénieur d'Angers, Angers, France, available at: http://www.druel.com/francois/docs/Memoire_These_Druel.pdf (accessed 19 April 2009).

English, R. and Schweik, C.M. (2007), “Identifying Success and Tragedy of FLOSS Commons: A Preliminary Classification of Sourceforge.net Projects”, paper presented at the First International Workshop on Emerging Trends in FLOSS Research and Development in conjunction with 29th International Conference on Software Engineering, May 20-26, Minneapolis, MN, available at: http://www.upgrade-cepis.org/issues/2007/6/upg8-6English_Schweik_v2.pdf (accessed 24 March 2008).

Fogel, K. (2005), *Producing Open Source Software: How to Run a Successful Free Software Project*, O'Reilly, Cambridge, available at: http://downloads-guests.open.collab.net/files/documents/61/48/producing_oss_book.pdf (accessed 24 March 2008).

Golden, B. (2004), *Succeeding with Open Source: An Overview*, Addison-Wesley, Reading, MA.

Gottlieb, S. (2006), “Evaluating a Community”, available at: http://contenthere.blogspot.com/2006/12/evaluating-community.html (accessed 18 March 2008).

Halloran, T.J. and Scherlis, W.L. (2003), High Quality and Open Source Software Practices, available at: http://opensource.ucc.ie/icse2002/HalloranScherlis.pdf (accessed 24 March 2008).

Kraft, B. and Hinderink, D. (2006), “Evaluating Open Source Communities. Why Open Source is more than Code - and why you should care”, paper presented at Transformation and Innovation Conference 2006, May 22-24, Washington DC, available at: http://news.typo3.org/fileadmin/user_upload/evaluating_OS_communities.pdf (accessed 18 March 2008).

Mockus, A., Fielding, R.T. and Herbsleb, J. (2002), “Two case studies of open source software development: Apache and Mozilla”, *ACM Transactions on Software Engineering and Methodology*, Vol. 11, No. 3, pp. 309-346.

Stürmer, M. (2005), *Open Source Community Building*, Thesis, University of Bern, Bern, Switzerland, available at: http://opensource.mit.edu/papers/sturmer.pdf (accessed 17 March 2008).

Waugh Partners et al. (2008), “The Foundations of Openness: Evaluating aspects of openness in software projects”, available at: http://pipka.org/blog/2008/07/23/the-foundations-of-openness/ (accessed 27 March 2009).

Usability

Kaiser, S., “WebsiteTips.com”, available at: http://websitetips.com/usability/ (accessed 28 April 2008).

National Institute of Standards and Technology (2007), Common Industry Specification for Usability – Requirements (NISTIR 7432), available at: http://zing.ncsl.nist.gov/iusr/documents/CISU-R-IR7432.pdf (accessed 3 May 2008).

Nielsen, J. and Mack, R. (1994), *Usability inspection methods*, John Wiley and Sons, New York, NY.

Nielson, J., “useit.com: Jakob Nielsen on Usability and Web Design”, available at: http://www.useit.com/ (accessed 22 April 2008).

Request for Proposal

Library of Congress (1997), “Request for Proposal for an Integrated Library System”, available at: http://lcweb.loc.gov/ils/rfp97-12.wpd (accessed 4 February 2007).

Minnesota Library Information Network (1997), “Request for Proposal. Components Relating to an Integrated Library Management System”, available at: http://www.fcla.edu/FCLAinfo/lms/minnesot.pdf (accessed 4 February 2007).

State University of New York (1999), “Integrated Library Management System. Request for Proposals”, available at: http://www.sunyconnect.suny.edu/slam/SLAMrfp.doc (accessed 4 February 2007).

State University System of Florida (2000), “Integrated Library Management System. Invitation to Negotiate”, available at: http://www.fcla.edu/csul/pspc/illsub/SUSILL.doc (accessed 4 February 2007).

Free and Open source ILS comparaisons

Riewe, L.M. (2008), *Survey of Open Source Integrated Library Systems*, Thesis, San José State University, San José, CA, available at: http://users.sfo.com/~lmr/ils-survey/080831-paper-Riewe.pdf (accessed 1 October 2008).

Müller, T. (2009), “Étude comparative des principaux SIGB libres”, in Savard R. and Gazo D. (Ed.), *Actes du premier congrès de l'Association Internationale Francophone des Bibliothécaires et Documentalistes et satellite IFLA Montréal, 3 - 6 août 2008*, AIFBD, Montréal, pp. 77-91.

Free and Open source ILS

Emilda, available at: http://www.emilda.org/ (accessed 15 May 2009).

EspaBiblio, available at: http://www.desem.cl/espabiblio/index.php (accessed 10 January 2010).

GNUTECA, available at: http://www.gnuteca.org.br/ (accessed 19 April 2008).

INFOCID, available at: http://cidtel.inictel.gob.pe/Infocid/index.php?name=1 (accessed 22 June 2009).

Jayuya-ils.org, available at: http://jayuyails.free.fr/ (accessed 19 September 2009).

Koha, available at: http://www.koha.org/ (accessed 21 March 2008).

NewGenLib, available at: http://www.verussolutions.biz/web/ (accessed 1 April 2008).

OPALS, available at: http://opals-na.org/ (accessed 27 September 2009).

OpenAmapthèque, available at: http://umramap.cirad.fr/amap2/logiciels_amap/index.php?page=openamaptheque (accessed 2 September 2009).

OpenBiblio, available at: http://obiblio.sourceforge.net/ (accessed 27 September 2009).

OpenBiblio, available at: http://openbiblio.sourceforge.net/ (accessed 27 September 2009).

Evergreen ILS, available at: http://www.open-ils.org/ (accessed 9 March 2008).

phpMyLibrary, available at: http://sourceforge.net/projects/phpmylibrary/ (accessed 27 September 2009).

PMB, available at: http://www.sigb.net/ (accessed 21 March 2008).

Senayan, available at: http://senayan.diknas.go.id/web/ (accessed 16 July 2009).

Weblis, available at: http://portal.unesco.org/ci/en/ev.php-URL_ID=16841&URL_DO=DO_TOPIC&URL_SECTION=201.html (accessed 21 March 2008).

Decision Matrix

Borysowich, C. (2006), “Constructing a Weighted Matrix”, available at: http://it.toolbox.com/blogs/enterprise-solutions/constructing-a-weighted-matrix-13125 (accessed 21 September 2008).

Brown, R. (2005), *Rational choice and judgment: decision analysis for the decider*, Wiley-Interscience, Hoboken, NJ.

International Society on Multiple Criteria Decision Making, available at: http://www.mcdmsociety.org/ (accessed 1 July 2009).

Rahman, M.F. (2006), “What is a Decision Matrix?”, available at: http://risal.wordpress.com/2006/12/15/what-is-a-decision-matrix/ (accessed 21 September 2008).

Tague, N.R. (2004), *The Quality Toolbox*, ASQ Quality Press, Milwaukee, WI.

Triantaphyllou, E. (2000), *Multi-criteria decision making methods: a comparative study*, Kluwer Academic Publishers, Norwell, MA.

X. Appendices

Criteria of attractiveness and sustainability of community

Axes and categories	Weights	Emilda	EspaBiblio	Evergreen	Gnuteca	InfoCID	Jayuya	KOHA	NewGenLib	oBiblio	OPALS	OpenAmapthèque	Open Biblio	PhpMyLibrary	PMB	Senayan
Attractiveness		2.0	1.4	6.1	2.5	1.8	1.2	6.4	3.5	3.0	3.1	0.7	1.4	1.4	4.8	2.0
Multilingual Project	2.0	3.0	1.0	3.0	1.0	1.0	1.0	7.0	2.0	4.0	3.0	1.0	2.0	3.0	3.0	2.0
Regular versioning	2.0	2.5	1.8	6.5	5.3	1.8	2.5	6.5	6.0	6.0	4.0	0.0	4.0	2.5	6.3	7.0
Renamed or referent users	3.0	0.5	1.5	6.0	2.5	2.5	0.5	6.0	3.0	0.5	5.5	0.5	0.5	0.5	5.0	2.5
Developments by third parties	3.0	3.0	1.0	7.0	3.0	2.0	0.0	7.0	4.0	7.0	0.0	2.0	0.0	0.0	7.0	2.0
Project age	2.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	1.0	7.0	5.0	7.0	3.0
Project Sponsors	1.0	1.0	0.5	6.0	3.0	1.0	0.0	7.0	5.0	0.5	4.0	2.0	0.0	0.0	5.0	1.0
Presence of a guru or an initiator	1.0	0.0	0.0	6.0	0.0	0.0	0.0	5.0	2.0	0.0	0.0	0.0	0.0	0.0	2.0	0.0
Unique or differentiating features	1.0	0.0	0.0	5.0	0.0	0.0	0.0	5.0	3.0	0.0	4.0	0.0	0.0	0.0	3.0	0.0
Project talked about	3.0	0.0	0.0	7.0	0.0	0.0	0.0	6.0	1.0	0.0	1.0	0.0	0.0	1.0	3.0	0.0
Project legend	1.0	1.0	0.0	6.0	2.0	0.0	1.0	6.0	3.0	0.0	4.0	0.0	0.0	1.0	4.0	0.0
Sustainability		1.7	1.4	4.8	1.6	1.3	1.1	5.7	2.5	1.8	2.7	1.0	1.3	1.4	2.4	1.6
Active community of developers	3.0	0.9	0.9	3.7	0.9	1.3	1.6	5.0	1.8	1.2	2.0	0.9	0.9	0.9	2.4	2.1
Active user community	3.0	1.0	3.6	2.8	1.0	1.8	1.0	4.6	1.4	1.0	2.2	1.0	1.0	1.0	2.4	1.8
Fundraising success	1.0	0.0	0.0	7.0	3.0	1.0	0.0	7.0	4.0	0.0	5.0	1.0	0.0	0.0	3.0	1.0
Management of system project	2.0	4.0	0.3	5.7	0.9	0.6	0.3	6.7	3.7	3.3	0.9	0.6	3.0	3.7	1.6	1.0
Renewal of project sponsors	2.0	0.0	0.0	4.0	1.0	1.0	0.0	6.0	2.0	1.0	5.0	0.0	0.0	0.0	0.0	1.0
Knowledge management and knowledge base	2.0	1.1	0.8	5.6	2.4	0.9	1.8	5.5	1.6	1.1	1.8	0.6	2.0	1.0	4.0	1.9
Community Governance	3.0	1.8	0.5	5.7	0.8	0.0	0.0	6.2	2.5	2.2	1.8	0.2	0.0	0.5	1.7	0.0
Compliance of standards in the field	1.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0	7.0
Total		1.8	1.4	5.5	2.1	1.6	1.2	6.1	3.0	2.4	2.9	0.8	1.4	1.4	3.7	1.8

Criteria of maturity and finalization functions

Modules and categories	Weights	Evergreen		KOHA		PMB	
		Mt+	Fn+	Mt+	Fn+	Mt+	Fn+
General essential specifications	3	4.3	68%	4.9	79%	4.2	72%
General	3	4.9	84%	6.0	100%	4.8	91%
Technology platform	3	3.7	52%	3.8	57%	3.6	52%
Administration	2	5.8	95%	5.3	95%	4.8	90%
General	2	5.8	94%	6.0	100%	5.4	92%
Software Considerations	3	6.8	100%	6.8	100%	6.8	100%
Patron Initiated Transactions	3	6.8	100%	6.5	100%	4.0	75%
Report Generation	2	5.0	96%	2.6	86%	1.9	94%
Security	2	4.7	89%	3.8	89%	4.1	89%
Backup	1	3.1	72%	3.6	91%	7.0	100%
Cataloging	3	5.0	87%	4.8	87%	3.8	72%
Generalities	3	4.9	86%	4.6	81%	3.9	74%
Security	3	5.3	100%	4.7	100%	5.0	100%

Standards	3	4.3	89%	4.6	89%	1.3	44%
Import and Export	2	2.8	67%	4.6	94%	3.8	78%
Indexing	3	6.0	90%	6.3	98%	5.8	95%
Call Numbers	1	6.0	100%	3.0	75%	0.0	0%
Printing and Data Entry	2	5.5	94%	6.0	94%	3.3	78%
Item Records	3	4.6	73%	5.2	85%	5.1	83%
Record created on the fly	1	5.3	81%	0.0	0%	5.3	88%
Temporary Relocation	1	6.2	100%	5.6	100%	0.0	0%
Authority control	1	0.0	0%	5.5	91%	4.0	75%
Authorities	1	0.0	0%	5.5	91%	4.0	75%
OPAC	3	4.3	68%	5.0	80%	3.8	70%
Generalities	3	4.9	77%	6.1	97%	6.0	100%
Search Interface Functionalities	3	4.6	71%	4.6	74%	3.9	71%
Faceted search	2	3.4	48%	3.4	48%	0.0	0%
Navigation	3	5.4	87%	6.0	98%	3.8	81%
Help and “Artificial Intelligence” Enhancements	2	3.6	59%	3.6	59%	2.1	38%
Content Display	3	5.6	88%	6.5	98%	5.4	96%
Web 2.0 Features	1	0.8	16%	1.2	26%	2.7	68%
Printing and Downloading	1	1.5	27%	5.4	86%	4.0	73%
Circulation	3	4.8	84%	4.5	83%	3.2	64%
Circulation Parameters	3	4.2	95%	4.9	96%	3.1	77%
Circulation Control	3	6.3	100%	7.0	100%	7.0	100%
Check out	3	5.8	94%	5.2	94%	4.3	88%
Renewals	3	5.0	100%	4.0	83%	2.7	62%
Checks in	3	6.1	91%	3.6	68%	3.2	59%
Holds	3	3.7	77%	3.4	62%	1.6	35%
Fines	3	3.5	63%	2.2	71%	1.2	37%
Patron file	3	6.1	95%	6.2	98%	3.5	74%
Notices	2	5.2	89%	5.7	100%	3.9	65%
Inventory Control	1	0.8	40%	2.9	73%	0.8	40%
Reserve	1	0.0	0%	1.3	33%	1.2	33%
Acquisitions	2	0.0	0%	3.0	71%	1.7	41%
General	3	0.0	0%	1.7	58%	1.2	48%
Pre-order Searching and Verification	2	0.0	0%	3.0	100%	0.0	0%
Ordering and Order Maintenance	2	0.0	0%	2.0	65%	0.8	39%
Receipts	3	0.0	0%	4.3	87%	1.0	26%
Claiming	2	0.0	0%	1.7	57%	0.1	9%
Payment and Fund Accounting	2	0.0	0%	1.0	26%	2.7	64%
Vendor File	3	0.0	0%	4.2	76%	3.6	57%
Holdings	3	0.0	0%	4.8	84%	3.0	71%
Serials	2	0.0	0%	4.5	88%	3.0	75%
Generalities	3	0.0	0%	6.0	100%	4.0	100%
Serials processing	3	0.0	0%	4.2	100%	4.0	100%
Routing	1	0.0	0%	5.0	100%	0.0	0%
Binding	1	0.0	0%	0.0	0%	0.0	0%
TOTAL		3.5	59%	4.7	83%	3.6	70%

Author

Tristan Müller
tristan.muller@bibliothequeglobale.org
Bibliothécaire, spécialiste des technologies libres (Librarian, Free and Open source ILS specialist)
Observatoire des technologies libres (Open source technologies Observatory)

Fondation pour une bibliothèque globale (Global Library Foundation)
465, rue Saint-Jean, bureau 900
Montréal (Québec)
Canada H2Y 2R6

Phone: +1 514 802-5642
Fax: +1 514 288-4880

Bio

Tristan Müller is a professional librarian and a member of the board of directors of the Global Library Foundation. He holds an MLIS from Université de Montréal. He is a librarian at Bibliothèque et Archives nationales du Québec (BAnQ) since 2004 where he held several positions at the Grande Bibliothèque. He is now Head of the Circulation department where he coordinates a team of more than fifty employees to ensure circulation services to the 260 000 subscribers of the library. Before joining BAnQ, he worked as a consultant librarian for 2 years in governmental libraries in Burkina Faso, for a project funded by the World Bank. He was also a librarian at the Montreal Botanical Garden. Since 1999 he has conducted volunteer librarianship projects of international cooperation in Burkina Faso, Niger and Haiti. Through this consultation and cooperation experience as well as through direct observation, he found that the tools available to libraries do not meet their needs. Given the limited or non-existent financial resources of these libraries, he wanted to give them the means to meet their institutional mission. In that sense, he started focusing on free and open source software and has served, since 2004 as the technology coordinator for the "Open source technologies Observatory" program of the Global Library Foundation.

Translator

Matthew W. Loving
mwloving@ad.ufl.edu
Romance Languages Librarian

George A. Smathers Libraries
University of Florida

Phone: +1 (352) 273.2635