Technology trends, requirements and models for providing sustainable technological support for libraries in an evolving environment

P.Stathopoulos[1], N.Houssos[1], and G.Stavrou[1]

1 EKT - Hellenic National Documentation Center/N.H.R.F., 48, Vas. Constantinou Av., 116 35 Athens, Greece, phone: +30 210 7273996,, email: pstath@ekt.gr, nhoussos@ekt.gr, gstavrou@ekt.gr

Abstract: In this paper we present requirements, trends, models and proposed methods relating to the issue of providing libraries with sustainable, technological support, both in terms of economic sustainability and in terms of successful adaptation to the continuously evolving user needs and environment. Tangible benefits, but also limitations, regarding established trends such as open source software, appropriate to library needs are highlighted while new application and service delivery models, such as Software as a Service (SaaS), virtualization and /or Cloud Computing are presented, in relationship with practical issues libraries are facing.

Keywords: Cloud Computing, SaaS, Virtualisation, ABEKT, Repositories

1. Introduction

In this paper we present requirements, trends, models and proposed methods relating to the issue of providing libraries with sustainable technological support platforms, both in terms of economic sustainability and in terms of successful adaptation to the continuously evolving user needs and environment. Different scenarios, based on a coarse grained taxonomy of libraries profiles and requirements, highlighting their capacity to access various levels of economic and human resources, are examined. Factors that affect the Total Cost of Ownership (TCO) that libraries' technological support platforms pose are qualitatively examined, in relationship with the requirements and the key technological models employed for the purchase, development and operation of the technology support platform elements.

In greater detail, open access to scientific information has presented libraries with an excellent opportunity to expand their role but also frequently requires a multitude of new technological support systems, i.e. new services, software, applications, and infrastructure, in combination with the need to operate with the more traditional library automation systems, OPACs, union catalogues, which are still of utter importance. Furthermore, these tools, and the end user requirements and trends that drive their development, are dynamic, quickly changing and transforming, thus requiring continuous updating and modernization. Technology support systems that libraries could be required to operate in order to fully exploit the open access environment are expanding in numbers and sometimes also in complexity.

Balancing the distribution of the available resources efficiently, among the operation of such technological support infrastructures and the actual support and delivery of end user services can be quite a challenge and it is tightly dependent on the specific library profile and requirements. The sustainability of this level of technological support poses a significant challenge for libraries

both with regard of the expenses associated with operating it, it but also with the complexity of further developing it in order to address the evolving and non static user requirements in a dynamic environment. Thus, sustainability can be considered as both the capability to keep adapting services and the relevant software tools to the end users needs, and the capability to support the operation of an expanding list of technological infrastructure and software systems. Libraries are faced with the two dimensional problem, with regard to the technology support required, of combining economic sustainability and functional sustainability in order to adapt to long term user needs.

Benefits that can be achieved by key, technology related, decisions such as employing or not open source, cloud computing, software as a service and virtualization are examined. The full technological support lifecycle is taken into account, from development, testing, maintenance, operation and subsequent upgrade and refinement. It will be shown that open source for a wide variety of technological support elements has real tangible effects; however a certain level of expertise and local or contracted support is required. However, open source alone cannot guarantee that the level of technology support a library requires, is cost effective and sustainable in the long run. The variety of systems needed for a library technological support makes the employment of a mix of local virtualization or Software as a Service and/or cloud computing delivery models, appropriate for ensuring long term sustainability of services and depending on the library profile, when combined with open standards, open interfaces and libraries cooperation consortiums.

2. Changing requirements and new applications

The necessary software tools, infrastructures and systems for providing services to users has expanded and now includes a great variety of systems, usually employing different technologies and each one requiring an amount of libraries resources. These systems may include, depending on the library role:

- Integrated Library Systems supporting the core of library activities. Acquisition, Cataloguing, Circulation and OPAC are the most common modules of library process in an ILMS; mature open source and proprietary tools are available in this area internationally.
- Electronic reading rooms, for providing local users with electronic resources on the library.
- Backoffice digitization processes and software tools necessary for supporting that such as OCR, content management and workflow support tools.
- Metasearch engines for providing integrated research results to libraries users among a variety of resources.
- Institutional/Subject repositories. Repositories, like the online library catalogs, contain bibliographic information about items in the library. However, their focus is on the digital material itself; it is worth noting that typically the metadata in repositories is less detailed and rigorously defined than in library systems. Furthermore, a particularly interesting category of repositories is those that maintain collections of cultural heritage material, for example images of works of art, monuments, etc.
- Electronic academic, scientific and educational Journals. A single platform should handle the entire peer-review based publishing procedure comprising, among others, tasks like the uploading of articles by authors, submission of reviews by reviewers, editor decisions, copy-editing and

final layout production by journal personnel, storing the publications full-text and providing it to Internet users as well as indexing and search engines. Recently, there is a trend to build e-publishing platforms for material other that e-journals and in particular e-books/monographs, something that is critically important for subject domains like social sciences and humanities.

- Interoperation of repositories, especially for academic and research libraries, with systems such as Current Research Information Systems (CRIS), which aim to capture the research context in which scientific publications have been produced. For example, a typical CRIS maintains information about researchers and research organization, projects, funding programmes and research infrastructures, all linked with each other and – most importantly – with research output, for example in the form of publications, patents or data sets. While there is overlap among the information held in CRIS and repositories, the latter concentrate more on storing, indexing and preserving the digital content (e.g., publications full-text) besides the metadata.

The aforementioned list is not in any case exhaustive; it aims to highlight some of the necessary tools and software systems that a library may have to support. For most of these areas a number of mature tools have evolved, both open source and proprietary. Open source software has being a disruptive changing force over the last years, especially in the area of libraries, Cervon F., (2003)., however the sustainability of libraries services, and the corresponding IT systems, cannot be guaranteed only with the open source promise.

In greater detail for systems such as repositories, there are available quite a few popular, robust, open source platforms (e.g., DSpace, Fedora, EPrints) that are being used in thousands of real-life installations worldwide for institutional, thematic or cultural heritage repositories. A number of commercial products also exist, providing an alternative to consider for repository managers. Repositories can significantly vary in size, from local installations with a few hundred items to aggregators providing access to millions of documents through federated repository architectures, needing IT infrastructures from the small to the large scale. Furthermore, electronic journals and publishing can be handled by commercial platforms or services; a few open source solutions are also available, the most popular one being by far Open Journal Systems (OJS) which, unlike other free software alternatives, provides an out-of-the box solution for all major tasks in the operation of a scientific e-journal with a rich set of features. Regarding Integrated Library Systems (ILS) there is a greater maturity, due to the decades old experience in relevant systems, however, an ILS needs to transform in order to support the new library functions. Furthermore lately a number of successful open source ILMS systems have become available such as the Koha (http://www.koha.org), the OpenBiblio (http://obiblio.sourceforge.net/), the PhpMyLibrary (http://sourceforge.net/projects/phpmylibrary/), and the Evergreen system (http://www.open-ils.org/). These systems target from small to large scale libraries and they offer some of the basic to the most advanced features sets. Many of them are built around the open source Linux-Apache-MySQL - Perl/Python (LAMP) software stack, while proprietary systems have required specialized software stacks, including proprietary databases, operating systems, etc.

It is becoming apparent that while there a lot of common building blocks for each one of these tools, each additional software system being added adds also burden to the libraries resources, in terms of in house or contracted software

developers, system administrators, and IT equipment and infrastructure. In the next section we will refer with greater detail to necessary IT resources an information organization, such as the libraries, needs to provide in order to afford a sustainable technological support path.

3. Applications technological support requirements

The aforementioned applications require a wide range of resources for their technological support. These resources and the cost they inhibit are only partially defined if the initial purchase or development costs are taken into account. Every software and IT system cost should be calculated over the system's entire lifecycle, specified as the Total Cost of Ownership (TCO), Ellram L. (1998), of the relevant asset. The total cost of ownership, can be simplified as the total cost for the organization to have the relevant IT system running and providing its services. Libraries when examining IT systems costs for their operation and when assessing their sustainability should always look beyond the initial purchase and installation cost. They should include the whole software's development lifecycle, the iterations needed for coping to the changing library needs, and the accompanying operational costs.

In greater detail, at the development phase, regardless of whether open or closed source software is being used, costs include the design, implementation, testing and delivery of the finished software product. Furthermore, a software system is seldom a finished product by itself; continuous updates are required in order to better cope with changing library requirements and/or bug fixing etc. Agile processes, which are state-of-the-art in software development, require different iterations with each iteration being based on the previous one and produces at least a functional prototype of the system, even a small part of it. This prototype is available for testing to users; their feedback is taken into account in subsequent iterations. The planning of iterations changes after the end of each one, depending on developer experiences, un-anticipated issues encountered and user feedback. Tools are needed like version control, unit testing, continuous integration, issue tracking, that assist in the rapid modification of the code according to changes in requirements– the latter are unavoidable in real-life systems.

Agile development process suit open source software customization projects, due to the low initial cost for trying / using a new tool – not only in terms of money for paid licenses but in terms of acquisition time. The aforementioned description of tasks, design, implementation, testing, are even more complex in other s/w development processes. Furthermore for development and operation purposes, there should be a robust staging environment, with at least three separated installations of the system: **development** (where newly added code is tested by developers), **quality assurance** (where end users do their testing) and **production**.

In the operational phase of the technological system supporting the library a number of factors contribute to the TCO of each software application and service. These factors include:

1. The initial development and customization costs such as the **design**, **implementation**, testing of the application.
2. **Hardware** equipment and support needs, including the **initial purchase cost**, but also the **depreciation**, **maintenance** and **support** costs, which are different based on the level of availability required.
3. **Software maintenance and support**. As aforementioned in many ways after the initial rollout only a part of the final functionality

required by the library eventually has being fulfilled. In order to better provide services, especially in emerging areas, a flexible work plan is needed in order to provide the features requested by the library users. Thus even after the software roll out a significant amount of personnel, in house or contracted resources, should be employed.

4. **System platform support**, personnel or external contractors for Operating Systems updates, security patches and fixes, applied to the whole range of the software stack, from the application to the operating system, or the storage/networking firmware.

5. **Monitor and control loop**, which includes the aforementioned platform support, costs. At the time of the single ILMS application monitoring and estimating system health and performance was a simple task, nowadays that a multitude of systems are required in order for the library to provide its services a strong real time services and performance monitoring system, such as the open source nagios and cacti systems, enterprise functions such as local and remote backup, require a complex enterprise –level IT support infrastructure able to implement aspects that are described with the IT infrastructure Library best practice as the "control and monitor loop".

6. **Datacenter/computer room infrastructure**, electricity, facilities and infrastructure cost.

Each of the aforementioned resources usually requires a different type of internal or external personnel capabilities and profiles, even if it is required as a percentage of equivalent full time staff. In conclusion, IT systems development and operation is a process, not a goal with visible end. There are future costs that one must pay in order to keep the system viable and sustainable. This is common, although with varying weights, for both open source and proprietary software, internal or external development teams. Nevertheless, open source has truly been a disruptive force in library information systems and in information systems in general. It can provide libraries with lower initial cost, having in principle no licence fees attached, greater control over their computer environment, no vendor lock-in, better support for open standards, and more predictable ownership costs. However, factors such as software development processes and operation costs, which require a multitude of resources and technological capabilities, are still present.

4. Technology trends: the emerging landscape

Over the last years emerging technology trends have arisen that change significantly the landscape for the whole range of library information systems. These technology advances can be highlighted as the open source development model, virtualization and cloud computing paradigms such as Software as a Service (SaaS), and Infrastructure/Platform as a Service (IaaS/PaaS) systems, Geng L, (2009). The benefits of open source software have been thoroughly described and libraries have exploited them. However, as described in the previous section, the software license is only a part of the libraries IT systems sustainability factors. Developments and trends such as virtualization, SaaS and IaaS/PaaS, can provide a much broader area of choices that could improve IT systems sustainability and possible lower IT costs. In greater detail the emerging trends are:

1. Virtualisation is a mature, nowdays, technology, that can provide a mean for virtualising system hardware, in order to improve utilization of idle IT infrastructure equipment. It can reduce operational costs

related to hardware systems maintenance and support, and can significantly increase flexibility and availability. However, due to the specialized technologies being used it requires personnel with a high degree of technical competence that ranges from system administration to networks and storage systems.

2. Cloud computing in the forms of IaaS/PaaS and SaaS. Using IaaS and PaaS models enables the organization to have access to hardware computing storage and networking resources, over a virtualized infrastructure that a Cloud Computing provider offers. In this case libraries can eliminate the need and hidden costs of hardware and Datacenter operations costs, while still having full control to their software and capability to adapt to new requirements. These can include system level support relating to a hosted system service, however with much greater flexibility. Examples include the Amazon Elastic Compute Cloud, the Amazon Simple Storage Service and the Rackspace Cloud. A standardized API and interfaces however do not yet exist.

3. Cloud Computing in the form of Software as a Service. As referred at Boss R., (2010), SaaS resembles functionality provided by Application Service Providers. Using SaaS an organization can outsource the whole operational lifecycle of an IT system. It is worth noting that major ILS have being offered using a SaaS model, Hadro J., (2009), Kimpton M, (2010). Using the SaaS model libraries do not need in-house support while SaaS providers can achieve significant economies of scale.

Despite its advantages the aforementioned Cloud Computing approaches also have disadvantages such as cost in the long run, questionable migration paths to alternative systems and eventually no capability of internal support for areas considered as critical. Furthermore, data migration and security issues or concerns still exist, both of them are of crucial importance for libraries. It should be noted that both open source and virtualization has being a key technology enabler of Cloud Computing. Recently open source projects started offering SaaS options; there is a concern about possible conflict of interest, Hadro J., (2009). In conclusion while a SaaS system can be a value for some of the libraries systems some key prerequisites should be taken into account when selecting a Cloud Computing solution:

- Ensuring data portability and preservation both in IaaS/PaaS and in SaaS solutions. It is crucial to be able to migrate data from particular systems and vendors. In IaaS/PaaS services standardization on some key areas has begun. Regarding libraries, it can be considered that they are on an advantageous position due to the inherent focus on data compatibility and portability/preservation issues. While for bibliographic data mature other aspects should be taken into account, especially for ILS systems, repositories and journals.
- Readily available open source alternatives. It is crucial that apart from data/metadata migration path also an open source alternatively exists. Especially to the SaaS case there should be an open source alternative that one could migrate easily on a local solution with small burden. While for library related systems this seems to be the case it is not a widespread rule of all SaaS offered IT systems.
- Capability to manage and monitor the Service Level Agreements the service provider and the library have agreed. It is necessary to both

formulate the SLA and to have the necessary tools in order to enforce and monitor it. Furthermore, as in the case of outsourced solutions there should a mechanism for monitoring and enforcing the appropriate quality levels and checks.

- Another important factor is the trust issue at a private organisation, etc. In this case library consortia and alliances can find in SaaS and Cloud Computing technologies an enabler for achieving economies of scale.

Cloud computing infrastructures can assist libraries to better manage their costs, since library IT systems are thoroughly being built around standards and migration strategies, while most widespread cloud solutions have also an open source alternatively. Regarding work done at the Hellenic National Documentation Center (EKT), these developments have being taken into account. EKT operates since 2007 a large scale virtualization platform, Stathopoulos P., (2009), has used extensively open source software and has included cloud related approaches to the design of new tools and services. EKT is considering to offer a DSpace based repository installation as a virtual appliance, ready to be used in virtual infrastructures while the new version of ABEKT, which is under planning, could include features such as being available as open source software, design from the beginning with the option to run as a standalone and as a Software as a Service application, delivery of the application as a lightweight virtual appliance. Furthermore ABEKT's planned version will include integration of all available bibliographic record formats (MARC21, UNIMARC, MODS, DC), new web design using open source development tools and frameworks, authority database development supporting bibliographic catalog functions via web services and linking of digital content to bibliographic records and publishing on the Internet through an integrated search and retrieve environment, among others.

5. Conclusions and future work

New technologically driven capabilities are available today that promise to reduce libraries IT operation costs while increasing the services they offer. As mentioned previously, the open source approach can provide a lot of benefits but only if the limitations and the advantages are well understood. Cloud computing offers an opportunity for small and medium size organizations to reduce operating costs in areas that are not considered a key strategy asset, however different scenarios can be employed based on each library profile. Crucial to the adoption of cloud computing approaches is the availability and widespread adoption of standards both in the infrastructure and in the service area, the capability of migration to alternatives and the trust to operators of the Cloud infrastructures. The library software ecosystem features a number of well established open source tools, that are used as cloud computing infrastructures building blocks, widespread standards adoption and seems a proper candidate, based always on the libraries profiles and needs, and to the operation of Library consortiums and cooperation's for adopting Cloud computing approaches in order to further enhance the services offered to the final user.

References

Cervon F., (2003). The open source option, The Library Journal NetConnect Summer, pp. 8–12.

Ellram L. (1998) Total cost of ownership: a key concept in strategic cost management decisions - *Journal of Business Logistics*, Vol 19. No.1, 1998

Geng L, (2009). Cloud Computing: IT as a Service, *IEEE IT Professional*, pp. 10-13, March/April, 2009

Hadro J., (2009). LibLime's "Enterprise Koha" Prompts Frank Conversations on Open Source Issues, *libraryjournal.com website* available at http://www.libraryjournal.com/lj/technology/productsvendors/855795-296/story.csp,

Boss R., (2010), Software as a Service, *PLA technote* available at http://www.ala.org/ala/mgrps/divs/pla/plapublications/platechnotes/

Kimpton M, (2010), Strategic overview DuraSpace Roadmap DSpace and Fedora, *Open Repositories 2010, Madrid,* 2010

Stathopoulos P.,(2009),The case study of an F/OSS virtualization platform deployment and quantitative results, *Proceedings, 5th International Conference on Open Source Systems*, Skövde, Sweden 2009