

# Software propietario vs software libre: una evaluación de sistemas integrales para la automatización de bibliotecas

Oscar Arriola Navarrete  
Graciela Tecuatl Quechol  
Guadalupe González Herrera \*

*Artículo recibido:*  
9 de marzo de 2011  
*Artículo aceptado:*  
15 de junio de 2011.

## RESUMEN

En este documento se realiza una evaluación de dos Sistemas Integrales de Automatización de Bibliotecas (SIAB) con el fin de identificar ventajas y desventajas de los sistemas propietarios y los sistemas libres (SPX vs. Koha). Con el objetivo de proporcionar información al bibliotecario para que conozca mayores alternativas que le permitan elegir el SIAB que más convenga. De manera específica se evaluarán los sistemas SPX y KOHA, describiendo los requerimientos y servicios que ofrece cada uno de ellos; y poder dar elementos para la toma de decisión en el momento de adquirir un sistema.

\* Los tres autores pertenecen a la ENBA, México. (Oscar:oscar@sep.gob.mx); (Graciela: gtecuatl@sep.gob.mx); (Guadalupe: gglez@sep.gob.mx)

**Palabras claves:** Sistemas Integrales de Automatización, Software libre, Software propietario, Automatización de bibliotecas.

## ABSTRACT

### **Free software vs proprietary software: an assessment of integrated systems for library automation**

*Oscar Arriola Navarrete; Graciela Tecuatl Quechol and Guadalupe González Herrera*

This paper assesses two Integrated Library Systems (ILS) to identify advantages and disadvantages of proprietary and free software (SPX vs. Koha). With the aim of providing information to the librarian to expand alternatives to choose the comprehensive of ILS is most appropriate. Specifically be evaluated SPX and KOHA systems, describing the requirements and services offered by each one of them, and to give elements for decision making when purchasing a system.

**Keywords:** Integrated Library Systems, Free software, Proprietary software, Library automation

## INTRODUCCIÓN

Las bibliotecas están en continua adaptación debido a las nuevas necesidades que plantean los entornos y los cambios en las Tecnologías de Información y Comunicación (TIC). Esto hace posible que exista en el mercado una gran diversidad de Sistemas Integrales de Automatización de Bibliotecas (SIAB). ¿Cómo saber cuál elegir? ¿Cuáles criterios considerar para hacer una selección adecuada? ¿Elegir uno propietario o uno libre?

Se debe estar consciente que no es mejor SIAB el más costoso o el más sofisticado, sino el que mejor permita satisfacer las necesidades de la biblioteca, el que consienta habilitar cambios para brindar servicios de calidad y, sobre todo, el que permita la oportunidad de crecer y de estar a la vanguardia sin modificar totalmente el programa. Aquel que con sólo adaptarlo se obtengan los mejores procesos y servicios automatizados en la unidad de información.

## ¿QUÉ ES LA AUTOMATIZACIÓN DE BIBLIOTECAS?

Existen varios autores (López Yepes, Magan Wals) que han definido de diferentes formas la automatización de bibliotecas, pero manteniendo en general un punto en común: el uso y la aplicación de computadoras en los procesos y servicios de las unidades de información. Actualmente se considera que estas visiones han sido superadas con mucho y se debe entender que el enfoque es mucho más amplio y que además de los SIAB, se debe explotar toda la gama de posibilidades que ofrecen las TIC (Redes sociales, Administradores de contenidos, Administradores de recursos, Gestores bibliográficos, etcétera).

Se podría decir entonces que la automatización de bibliotecas es el uso de las TIC en la administración, en los procesos y en los servicios que ayudan a optimizar tiempos, costos y movimientos.

Actualmente los SIAB se ocupan de la organización, la catalogación, la adquisición y los servicios de una biblioteca, y que constituyen un elemento importante para el desarrollo de ésta.

La aplicación de un SIAB sirve para optimizar los procesos y agilizar las labores de la biblioteca incluyendo costos, tipología, proveedores y la toma de decisiones.

Hoy en día los sistemas integrales de automatización de bibliotecas cuentan con una gran oferta, encontramos en la literatura especializada que existen posibilidades de adquirir un SIAB para cualquier tipo de unidad de información, además de todos aquellos que existen en la red y que son ofrecidos de manera gratuita y con la ventaja adicional de proporcionar el código de fuente abierto.<sup>1</sup>

La aplicación de estos SIAB mejora el funcionamiento de las bibliotecas y permite controles más adecuados de los materiales, rapidez en las consultas, precisión en el registro de información, mayores beneficios para los usuarios, y una mejora en el acceso a los recursos de información.

En nuestros días se encuentra una extensa oferta de SIAB tanto propietarios como libres, pero parece ignorarse cuál puede ser la diferencia principal entre ambos, es por ello de suma importancia conocer con precisión este tipo de información para que se convierta en un elemento clave que ayude a la comunidad bibliotecaria nacional en su proceso de selección de software.

Las preguntas que surgieron al elaborar el presente trabajo fueron las de ¿saber realmente si el software propietario y el software libre, eran iguales?, ¿hasta qué punto son comparables? y si la única diferencia que existe entre el software libre y software propietario, es el código abierto y el costo.

1 El código fuente son los elementos de programación del sistema, cuando se dice que es abierto se permite que cualquier persona analice, cambie, mejore, redistribuya, modifique y desarrolle sobre la base del mismo.

Hasta ahora en nuestro país existe sólo una escasa cultura de la evaluación que aclare y resalte las ventajas y desventajas del software libre y del software propietario, y que le permita a uno formarse un juicio acerca de si lo que ofrecen es lo mismo.

## METODOLOGÍA DE LA EVALUACIÓN

Se realiza este análisis bajo un estudio descriptivo en el que se contemplan las características a partir de una matriz de comparación entre un software propietario desarrollado en México, y que se encuentra diseñado para automatizar las funciones de archivos, bibliotecas convencionales y digitales, bosquejado para instituciones medianas y grandes que requieren de sistemas apegados a estándares internacionales, y un software libre llamado Koha, el cual es de los más citados e instalados en Estados Unidos, según lo reportado en el artículo "Automation System Marketplace" 2010, de la revista *Library Journal* de abril de 2010. La intención es presentar las ventajas y desventajas de ambos.

A los resultados obtenidos en el estudio descriptivo se les aplicará tanto el método analítico, que permitirá delimitar si estos SIAB cumplen con los requerimientos que se prevén en nuestro caso de estudio, así como un método comparativo que permitirá confrontar los contrastes entre las similitudes y las diferencias establecidas por cada uno de los sistemas.

Para obtener la información de ambos sistemas se llevó a cabo una investigación documental y algunas entrevistas con usuarios del software propietario en la ciudad de México.

### *Hipótesis*

El software libre Koha cuenta con una estructura tecnológica semejante y realiza las mismas funciones que el software propietario que denominamos como SPX.

## SOFTWARE PROPIETARIO VS. SOFTWARE LIBRE

En México, por lo general, las bibliotecas cuentan con partidas presupuestales que tienen entre sus objetivos el agilizar y optimizar las actividades de desarrollo de colecciones, la organización bibliográfica y los servicios; para todo lo cual se emplean SIAB que cubren las necesidades básicas, además de contar con el personal adecuado para su manipulación, mantenimiento y posibles mejoras.

Optar por un software amigable, que aporte soluciones prácticas para todas las necesidades y políticas de organización de la biblioteca no es tarea fácil; existen en el mercado un número importante de software propietarios que se ofrecen a las bibliotecas con el fin de resolver de forma ágil esta tarea; por esta razón deviene importante cuestionarse si un software libre puede satisfacer tales necesidades.

Pero no se trata de confrontar al software propietario con el software libre para demostrar qué tipo de software tiene mayor valía en el uso, manejo e implementación; tampoco se pretende apostar a uno u otro como una panacea que resuelva de manera radical la situación sobre la gestión de las bibliotecas; la intención es informar sobre las ventajas y desventajas que se pueden encontrar en ambos, sin olvidar que la parte más importante al momento de resolver cuál software es el más adecuado para los propósitos institucionales es la capacidad profesional del personal para manejo y aprovechar totalmente sus recursos y características.

## DIFERENCIAS ENTRE SOFTWARE PROPIETARIO Y SOFTWARE LIBRE

### *Software propietario*

Culebro Juárez, Gómez Herrera y Torres Sánchez afirman que el software no libre, también llamado software privativo, se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones), o cuyo código no esté disponible o el acceso a éste se encuentre restringido.<sup>2</sup>

La Free Software Foundation (FSF Fundación de Software Libre), fundada en 1985, define al software propietario como aquel que no es libre. Su uso, redistribución o modificación está prohibida, o requiere una autorización; está tan restringida que no puede ser libre de un modo efectivo. Es de dominio privado, porque una determinada persona tiene la titularidad de los derechos de autor y goza de un derecho exclusivo respecto de su utilización. Le niega a otras personas el acceso al código fuente del software y el derecho a copiar, modificar y estudiar el software.<sup>3</sup>

2 Montserrat Culebro Juárez, Gómez Herrera, Wendy Guadalupe y Torres Sánchez, Susana, *Software libre vs software propietario: ventajas y desventajas*, [en línea], México: Creative Commons, 2006, [Consulta: 29 de Junio de 2010], disponible en Internet: <http://www.softwarelibre.cl/drupal//files/32693.pdf>.

3 Martín Carranza Torres, *Problemática jurídica del software libre*, Buenos Aires, Lexis Nexis, 2004, p. 103.

O bien es aquel software que está siendo desarrollado por una entidad que tiene la intención de hacer dinero del uso del software, como lo señala Stella Rodríguez.<sup>4</sup> El software propietario es aquel que es imposible de utilizar en otro hardware o terminal, modificar y transferir sin pagar derechos a su creador o desarrollador.

En general se puede decir que al adquirir un software propietario se dependerá totalmente de la empresa desarrolladora, se firmarán contratos de mantenimiento anuales, y si se requiere crecer en cuanto a licencias se incrementarán los costos iniciales. El software propietario se encuentra protegido por el sistema *copyright*, el cual consiste en asignar y concederle derechos al autor o creador.

### *Software libre*

la FSF estipula que el software libre hace referencia a la capacidad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software que se ha puesto en sus manos; esto implica una gran ventaja para los usuarios de él porque el código fuente es colocado a disposición del usuario, es factible adaptarlo a los cambios del entorno en donde se esté utilizando este tipo de recursos, y esto da pie a la satisfacción de las necesidades particulares.<sup>5</sup>

La primera característica y tal vez la más llamativa es que para utilizar o descargar esta clase de software no es necesario realizar ningún pago, pero lo más importante de este tipo de software es que se proporciona el código del programa; es decir que cualquier usuario puede realizar las adecuaciones y modificaciones que requiere para mejorar sus prestaciones a la luz de ciertos requerimientos particulares o generales. El Software libre incluye la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el propio software. De modo más preciso, se refiere a cuatro libertades de los usuarios del software:

4 Gladys Stella Rodríguez, "El software libre y sus implicaciones jurídicas", en *Revista de derecho*, Núm. 30, Barranquilla, Universidad del Norte, 2008, p. 168.

5 Oscar Arriola Navarrete y Avila González, Armando, "El software libre y la enseñanza de la catalogación: una relación amistosa", en *Revista Codice*. Vol. 4, núm. 2 (Julio-Diciembre 2008), pp. 21-32.

1. **Libertad de ejecutar** el programa sea cual sea el propósito (libertad 0)\*.
2. **Libertad de estudiar** cómo funciona el programa para ajustarlo a tus necesidades (libertad 1). (Es indispensable tener acceso al código fuente).
3. **Libertad de redistribuir** copias, colaborando con otras personas (libertad 2).
4. **Libertad de modificar**, de tal forma que la comunidad pueda aprovechar las mejoras (libertad 3). (Es indispensable tener acceso al código fuente).

Stallman numeró las libertades empezando por el cero porque así era como lo hacían los informáticos. Alguien calculó que era más sencillo empezar a numerar las bases de datos con el cero porque no tienes que restar 1 tan a menudo.<sup>6</sup>

Las libertades del software libre hacen partícipe al usuario de las ventajas que se obtienen a partir del empleo de éste; en resumen, estas cuatro libertades, le permiten al usuario hacerse propietario y colaborador, pues recibe y otorga al mismo tiempo un producto que le es útil y en el que, siguiendo el principio de colaboración, se hace participar a otros de la oportunidad de usar y posiblemente mejorar ese software. Es importante conocer también los derechos y responsabilidades que el consumidor debe tomar en cuenta, por eso, Tennant publica en el 2007, el *Library Software Manifesto* (Manifiesto del Software Libre)<sup>7</sup>, y señala los

- derechos como consumidor:
  - Tengo derecho a saber lo que existe ahora y cuál es su potencial funcionalidad futura.
  - Tengo derecho a usar lo que compro.
  - Tengo derecho al API (*Application Programming Interface* Interfaz de Programación para la Aplicación) si he comprado el producto.
  - Tengo derecho a contar con documentación completa y actualizada.

6 Cfr. Richard M. Stallman, *Software libre para una sociedad libre*, [en línea], GNU Press, 2002, [consulta: 30 de Junio de 2010], disponible en Internet: <http://biblioweb.sindominio.net/pensamiento/softlibre/softlibre.pdf>.

7 Roy Tennant, *Library Software Manifesto*, [en línea], TechEssence, 2007, [consulta: 30 de Junio de 2010], disponible en Internet: <http://techessence.info/manifesto/>.

- Tengo derecho a mis datos.
- Tengo derecho a tener acceso de sólo lectura a la base de datos.
- Tengo derecho a no hacer las cosas sencillas innecesariamente complicadas.
- Tengo derecho a conocer las líneas de desarrollo y la estimación de tiempo de desarrollo del producto que he comprado.
- Tengo derecho a hacerle preguntas técnicas a un equipo capaz de comprenderlas y responderlas.
- Tengo derecho a no ser un probador involuntario.
- Tengo derecho a que se conserven mis personalizaciones y configuraciones en futuras actualizaciones.
- Responsabilidades como consumidor
  - Tengo la responsabilidad de conocer las necesidades de mis usuarios.
  - Tengo la responsabilidad de poner las necesidades de mis usuarios por delante de las mías.
  - Tengo la responsabilidad de comunicar mis necesidades clara y específicamente.
  - Tengo la responsabilidad de que las mejoras que pido sean realmente lo que quiero.
  - Tengo la responsabilidad de asignar honestamente las prioridades de las mejoras.
  - Tengo la responsabilidad de darme cuenta de que no soy especial.
  - Tengo la responsabilidad de elegir software usando un procedimiento limpio y razonable.
  - Tengo la responsabilidad de informar de los errores reproducibles de forma que puedan reproducirse.
  - Tengo la responsabilidad de informar de los errores irreproducibles con todos los detalles que pueda.
  - Tengo la responsabilidad de ver críticamente cualquier ajuste a las
    - configuraciones predefinidas.
- Responsabilidades compartidas
  - Tenemos la responsabilidad de comenzar desde una posición de respeto mutuo.
  - Tenemos la responsabilidad de comunicarnos correctamente.
  - Tenemos la responsabilidad de establecer y mantener un proceso de mejora racional.
  - Tenemos la responsabilidad de mantener las necesidades del



- usuario final como primordiales.
- Tenemos la responsabilidad de relajarnos y divertirnos

Si un programa dice que es software libre pero los usuarios no tienen todas estas libertades, entonces no lo es. Para que lo sea hay que tener la libertad de distribuir copias, sea con o sin modificaciones, sea gratis o cobrando una cantidad por la distribución a cualquier persona o institución. El ser libre de hacer esto significa, entre otras cosas, que no se tienen que pedir o pagar permisos. También se debe tener la libertad de hacer modificaciones y utilizarlas de manera privada en el trabajo u ocio, sin tener que anunciar que dichas modificaciones existen.

## DOS TENDENCIAS QUE HAY QUE TENER EN CUENTA

En el movimiento que promovió el software libre participaron dos tendencias:

- El movimiento de software libre, liderado por *Free Software Foundation (FSF)*
- El movimiento de código fuente abierto, liderado por *Open Source Initiative (OSI)*

El movimiento de software libre tiene como máximo representante a Richard Stallman, creador de GNU,<sup>8</sup> de FSF y de la Licencia General Pública (GPL General Public License).

Mientras que por otra parte Eric Raymond se instituyó como líder de la OSI, personaje que promovió entre su movimiento una posición más flexible, y que aceptó que el código fuente sea copiado, modificado y distribuido sin restricciones de ningún tipo.

## TIPOS DE SOFTWARE

Existen diferentes categorías de software y es necesario mencionar esto con la finalidad de presentar las diferencias que existen entre un software y otro:

8 Fundación Software Libre América Latina, *Celebrando el 25 aniversario del proyecto GNU*, [en línea], 2008, [consulta: 13 junio 2011], disponible en Internet: <http://www.fsfla.org/svnwiki/anuncio/2008-09-gnu-25.es>

- Software libre (*Free Software*).  
El software libre posee una autorización para que cualquiera pueda usarlo, copiarlo y distribuirlo gratis o mediante una gratificación. El código fuente puede estar disponible pero no es una condición.
- Software de código fuente abierto (*Open Source*).  
El término software de “código fuente abierto” además de poder ser usado, copiado y distribuido, siempre ofrece el código fuente abierto como una característica.
- Software de dominio público.  
El software de dominio público es aquel que no está protegido con *copyright*. Dominio público es un término legal que quiere decir precisamente sin *copyright*.
- Software con *copyleft*.  
El software protegido con *copyleft* es un software libre cuyos términos de distribución no les permite a los redistribuidores agregar ninguna restricción adicional cuando éstos redistribuyen o modifican el software. Esto significa que cada copia del software, aun si se ha modificado, debe ser software libre.
- Software libre no protegido con *copyleft*  
El software libre no protegido con *copyleft* viene desde el autor con autorización para redistribuir y modificar, así como para añadirle restricciones adicionales. Si un programa es libre pero no está protegido con su *copyleft*, entonces algunas copias o versiones modificadas pueden no ser libres completamente. Una compañía de software puede compilar el programa, con o sin modificaciones, y distribuir el archivo ejecutable como un producto propietario de software.
- Software cubierto por la GPL  
La GNU GPL (Licencia Pública General) es un conjunto específico de términos de distribución para proteger con *copyleft* a un programa. El Proyecto GNU la utiliza como los términos de distribución para la mayoría del software GNU.
- El sistema GNU  
El sistema GNU es un sistema operativo libre completo similar a Unix. Debido a que el propósito de GNU es ser libre, cada componente individual en el sistema GNU tiene que ser software libre. No todos los componentes tienen que estar protegidos con *copyleft*, sin embargo; cualquier tipo de software libre es legalmente apto para incluirse si ayuda a alcanzar metas técnicas.
- Software GNU  
Software GNU es un software que se libera bajo el auspicio del Proyecto

GNU. La mayoría del software GNU está protegido con *copyleft*, pero no todos; sin embargo, todo el software GNU debe ser software libre.

- Software semi-libre

El software semilibre es software que no es libre, pero viene con autorización para que los particulares usen, copien, distribuyan y modifiquen --incluye la distribución de versiones modificadas-- sin fines de lucro, pero también tiene otras restricciones.

- Freeware

El término “freeware” no tiene una definición clara aceptada, pero se utiliza frecuentemente para paquetes que permiten la redistribución pero no la modificación, y su código fuente no está disponible. Estos paquetes no son software libre.

- *Shareware*

El *shareware* es software que viene con autorización para redistribuir copias, pero establece que quien continúe el uso de una copia deberá pagar un cargo por licencia. El *shareware* no es software libre, ni siquiera semilibre. Existen dos razones por las que no lo es:

1. Para la mayoría del *shareware*, el código fuente no está disponible; de esta manera, no puede modificarse el programa en absoluto.
2. El *shareware* no viene con autorización para hacer una copia e instalarlo sin pagar una cantidad por la licencia, ni aun para particulares involucrados en actividades no lucrativas. En la práctica, la gente a menudo hace caso omiso de los términos de distribución y los copia o instala y de todas formas, pero los términos no lo permiten.

La aplicación de uno u otro tipo de software estará sujeta a las particularidades de la unidad de información donde se aplique y dependerá del o los procesos que se deseen automatizar.

A continuación se presenta un cuadro elaborado por Casanova Huerta<sup>9</sup> que identifica y concentra las principales diferencias entre este tipo de software:

<sup>9</sup> Araceli Casanova Huerta, *Software libre una opción viable en la organización bibliográfica de fotografías: caso Biblioteca de las Artes*, México, La autora, 2010, Tesis de Licenciatura en biblioteconomía, p.125.

SOFTWARE LIBRE	SOFTWARE PROPIETARIO
No está limitado a los usuarios, y garantiza las libertades de los usuarios de usar, modificar, copiar y distribuir el software.	Tiene licencias, las cuales están limitadas a los usuarios y en algunos casos son costosas. Estas licencias restringen las libertades de los usuarios a usar, modificar, copiar y distribuir el software.
En el desarrollo de este, pueden intervenir cualquier persona, empresa u organización del mundo. Si bien se genera una gran cantidad de ideas innovadoras, también se posibilita la oportunidad de adecuación de avances tecnológicos en estos productos.	El desarrollo, programación y actualización de este software solo lo hace la empresa que tiene los derechos.
Los avances, modificaciones y descubrimientos tecnológicos son constantes, y se encuentran en Internet de forma gratuita. La principal meta del software libre es compartir los avances tecnológicos con los demás.	El futuro del software que adquirió el usuario solo depende de una empresa comercial.
El usuario no depende de una sola empresa, ya que el software que implementó puede ser mantenido y modificado, ajustándolo a sus necesidades.	En ocasiones las estrategias comerciales suele hacer que los usuarios actualicen su software comercial, sin que exista una necesidad verdadera de ello, consiguiendo que el usuario invierta en nuevas licencias, la mayoría de las veces innecesarias.
Sigue siendo compatible tanto en software como en hardware, no obliga al usuario a cambiar de sistema operativo o equipo, ya que no persigue los mismos fines económicos que el software comercial.	Siendo creado con fines de lucro, las actualizaciones o mejoras al software quedan limitadas bajo las estipulaciones de contrato.

### DESCRIPCIÓN DE LOS SISTEMAS

En este punto se proporcionan las características principales de los dos sistemas que se evalúan, por cuestiones éticas se decidió no proporcionar el nombre del software propietario y denominarlo Software Propietario “X” y que en adelante se manejará como SPX.

### Características de SPX

- *Amplio alcance:* su arquitectura está basada en Web por lo que su alcance geográfico es ilimitado.
- *Adaptable:* puede ser configurado de acuerdo con las necesidades de cada institución.
- *Versátil:* opera en diferentes ambientes de cómputo como son: Windows XP, Windows NT/2000, Mac OS X, FreeBSD, Linux, Solaris y Unix. Además trabaja con bases de datos relacionales (RDBSD) de distintos proveedores como PostgreSQL, SQLserver, Oracle, Sybase, Informix y DB2.
- *Elevado estándar:* ofrece a los usuarios servicios avanzados como los de las mejores bibliotecas del mundo y su sistema de búsqueda es fácil de usar, rápido y poderoso.
- *Compatibilidad:* puede contener en su base de datos registros de distintos formatos de manera simultánea, como MARC 21, Dublin Core, ISIS y CEPAL; además es posible vincular objetos digitales en diferentes formatos (JPG, PDF, TXT, MPGE, WAV).
- *Flexible:* es posible cargar datos de formatos personales o libres de otros sistemas; también pueden convertirse a formato MARC.
- *Seguro:* su diseño permite mantener la base de datos sin riesgos de modificaciones por parte de personas ajenas o no autorizadas. Cada registro y campo puede ocultarse a los usuarios que no tengan los privilegios requeridos para verlos.
- *Ilimitado:* el número de títulos que puede manejar es prácticamente infinito, depende únicamente de la capacidad del servidor utilizado.

Ante tales características el sistema SPX presenta ciertos beneficios como son:

- Arquitectura de cómputo en tres capas (base de datos, aplicación – servidor Web y clientes delgados). Todo el sistema trabaja en ambiente Web.
- Integra servicio Web (SOAP, HTTP, etcétera).
- Las estaciones de trabajo funcionan en cualquier ambiente de cómputo: Windows, Macintosh, Linux, etc. conectadas en una Intranet o propiamente en Internet.
- No requiere software propietario en cada estación de trabajo.
- Puede instalarse en diferentes navegadores Web: Explorer, Safari, Firefox, Opera.

- Gracias a su administración, simple, no se requiere invertir en especialistas en sistemas.
- Base de datos en SQL con un diseño integrado para brindar consistencia, velocidad y seguridad en la información.
- Índices configurables y con excelente tiempo de respuesta, incluyen los de texto completo en metadatos y en objetos digitales.
- Intercambio de información con XML o ISO 2709
- Manejo de multi-bibliotecas con un número ilimitado de títulos.

### *Arquitectura del sistema*

El SPX cuenta con una arquitectura en tres capas basada en la Web, utiliza un navegador en la estación de trabajo, en lugar de la interfaz de usuario, lo que hace que la aplicación pueda instalarse en una gran variedad de equipos y configuraciones por esta razón resulta ser un sistema altamente configurable, al mismo tiempo que simplifica el soporte y el mantenimiento, este sistema utiliza la base relacional de Oracle, ya que hace uso de los estándares internacionales como el Open Database Connectivity (ODBC).

### *Requerimientos del sistema*

- Requerimientos de equipo para terminales:
  - Soporte de redes
  - Servidor Web bajo plataforma Windows NT 4.0 ó 2000
  - Soporte de manejador de bases de datos
  - Dispositivos lectores de códigos de barras
  - Dispositivos de impresión
  - Navegador de Internet
  - Equipos personales (PC) que soporten Windows 95 o superior
- Requerimientos de hardware para el servidor:
  - Procesador Pentium III a 800 MHZ
  - Memoria RAM 512 MB
  - Disco duro de 30 GB para el uso de aplicación
  - Disco duro de 20 GB para respaldos (opcional)
  - Unidad de lectura y escritura de discos compactos (CD-ROM)
  - Tarjeta de red tipo Ethernet 10/100
  - Unidad de disco flexible de 3.5 "HD"
  - No-brake (recomendable)
- Requerimientos de hardware para los clientes:
  - Procesador Pentium I a 300 MHZ

- Memoria RAM 128 MB
- Disco duro de 10 GB para uso e instalación de la aplicación
- Tarjeta de red tipo Ethernet 10/100
- Unidad de disco flexible de 3.5 “HD”
- No break
- Navegador de Internet (Internet Explorer 4.0 o superior)
- Requerimientos de software para el servidor:
  - Windows XP, NT, 2000, Mac OSX, Free BSD, Linux, Solaris, y UNIX
  - Navegador de Internet, recomendable Explorer 4.0 ó superior
  - Manejador de base de datos con soporte de SQL/ANSI. Se recomienda SQL Server 7 o superior.
- Software requerido para los clientes:
  - Windows 95 o superior a 800 MHZ
  - Navegador de Internet (Internet Explorer 4.0 ó superior)

### *Descripción del sistema*

SPX está integrado por 6 módulos: control bibliográfico, circulación, control de suscripciones, adquisiciones, reportes y catálogo al público (OPAC):

- Control bibliográfico (catalogación): dedicado a la creación y mantenimiento de los registros de los materiales en diferentes formatos: MARC 21, Dublin Core, ISIS, ISAD (G), etc.; además cuenta con herramientas para la importación en línea vía Z39.50 y la exportación de datos en ISO2709. En los registros es posible vincular objetos digitales en diferentes formatos (JPG, PDF, TXT, MPEG y WAV, por ejemplo). Además, el sistema incluye el formato MARC de Autoridades y Existencias
 

*Control de autoridades:* con esta aplicación se crean y mantienen los registros de autoridades MARC 21 y NO-MARC.

*Impresión de etiquetas:* permite la impresión de etiquetas de lomo, bolsillo y código de barras, en lote o para un ejemplar en específico; es posible utilizar cualquier tipo o tamaño de etiquetas e impresoras.
- Circulación: registra las operaciones de préstamo, devolución, creación de multas monetarias y administrativas, suspensiones, apartados, registros de usuarios, envío de notificaciones vía correo electrónico o impresa. Cada unidad de información dentro de un sistema *multi-bibliotecas* puede tener políticas de circulación distintas.
- Control de suscripciones: cuenta con las funciones de administración

y control de suscripciones, que incluye la predicción, recepción y reclamación de fascículos. Los operadores crean los patrones de predicción de diferentes publicaciones para controlar las suscripciones, aun las que tienen un patrón irregular. Una vez creado el patrón, basta recibir fascículos para que sean creados los registros de existencias (ítems y existencias MARC).

- **Adquisiciones:** mantiene el control exhaustivo de las adquisiciones por centro de costos o presupuesto. Puede controlar lo adquirido a través del proceso normal de adquisiciones o por otros medios como canje o donación. Le permite saber cuánto dinero se ha comprometido o se ha pagado, maneja múltiples monedas y realiza conversiones a la moneda local.
- **Reportes:** genera listados y conteos de los diferentes registros (bibliográficos, autoridades, usuarios, uso del material, etc.) con base en filtros específicos, de acuerdo con las necesidades de la institución. Permite exportar la información de los reportes a hojas de cálculo o archivos de texto para manipularlos en otras aplicaciones.
- **Catálogo al público (OPAC):** cuenta con las funciones de búsqueda rápida y avanzada, ya sean en la base local o en otras vía Z3950. Permite el uso de operadores booleanos y de truncaciones a la derecha y a la izquierda, así como herramientas de limitación (formatos, tipo de material y/o colecciones).

Incluye la función *Referencista electrónico*: controla las búsquedas predefinidas en el catálogo, representadas por íconos o imágenes; al hacer clic sobre un icono en particular se ejecuta automáticamente la búsqueda. Mantiene una estructura jerárquica de imágenes que guía al usuario en búsquedas sobre ciertos temas, autores, bibliografías, etcétera.

Adicionalmente cuenta con la opción de *Servicios al usuario* que permite realizar apartados de material prestado, enviar solicitudes a la institución, cambiar su contraseña o los datos de su domicilio y revisar la información y el estado en que se encuentra.

Además de buscar en la base de datos local, es posible hacerlo en textos completos de archivos que estén en cualquier sitio, ya sea remoto o interno. El archivo es independiente del registro descriptivo y puede estar en cualquier formato que el cliente sea capaz de abrir. Las búsquedas pueden hacerse sobre el texto o combinando índices del registro descriptivo con el índice de dicho texto. Tiene la capacidad de indizar múltiples formatos de archivo: texto, PDF, DOC, RTF, HTML (y, por lo tanto, sitios Web), XML, Excel, Postscript, etcétera.



### *Ventajas y desventajas del sistema SPX*

Como representante de un sistema propietario o comercial el sistema SPX posee las siguientes ventajas, como señala Montserrat Culebro:<sup>10</sup>

1. Control de calidad.- Las compañías productoras de software propietario por lo general tienen departamentos de control de calidad que llevan a cabo muchas pruebas sobre el software que producen.
2. Recursos para la investigación.- Se destina una parte importante de los recursos a la investigación sobre los usos del producto.
3. Personal altamente capacitado.- Se cuenta con programadores muy capaces y con mucha experiencia.
4. Uso común por los usuarios.- Por ser un software usado por muchas personas es relativamente fácil encontrar a alguien que lo sepa usar.
5. Software para aplicaciones muy específicas.- Existe software propietario diseñado para aplicaciones muy específicas que sólo lo posee la compañía que lo produce.
6. Difusión de publicaciones acerca del uso y aplicación del software.- Existe una gran cantidad de publicaciones ampliamente difundidas, que documentan y facilitan el uso de las tecnologías provistas por compañías de software propietario, aunque el número de publicaciones orientadas al software libre va en aumento.

Y por tanto las siguientes desventajas:<sup>11</sup>

1. Cursos de aprendizaje costosos.- Para aprender a utilizar eficientemente el software propietario se debe asistir a cursos de capacitación.
2. Secreto del código fuente.- El funcionamiento del software propietario es un secreto que guarda celosamente la compañía que lo produce.
3. Soporte técnico ineficiente.- En la mayoría de los casos el soporte técnico es insuficiente o tarda demasiado tiempo en ofrecer una respuesta satisfactoria.
4. Ilegal o costosa la adaptación de un módulo del software a necesidades particulares.- Es ilegal extender una pieza de software propietario, en caso de que sea vitalmente necesaria tal modificación, es necesario pagar a la compañía fabricante, para que sea ésta quien lleve a cabo la modificación a su propio ritmo de trabajo y sujeto a su calendario de proyectos.

<sup>10</sup> Cfr. Montserrat Culebro Juárez, *Op. Cit.*

<sup>11</sup> *Ibid.*

5. Derecho exclusivo de innovación.- Si alguien tiene una idea innovadora con respecto a una aplicación propietaria, tiene que elegir entre venderle la idea a la compañía dueña de la aplicación o escribir desde cero su propia versión de una aplicación equivalente, para una vez logrado esto poder aplicar su idea innovadora.
6. Ilegalidad de copias sin licencia para el efecto.- Es necesario contratar licencias para poder hacer copias del software propietario.
7. Imposibilidad de compartir. – Al igual que el punto antes mencionado, nuevamente es necesario contratar las licencias necesarias para poder utilizar el sistema.
8. Quedar sin soporte técnico.- Si la compañía fabricante del software propietario se declara en banca rota el soporte técnico desaparece, la posibilidad de tener versiones mejoradas de dicho software desaparece al igual que la posibilidad de corregir los errores de dicho software también. Los clientes que contrataron licencias para el uso de ese software quedan completamente abandonados a su propia suerte.
9. Descontinuación de una línea de software.- Como consecuencia del punto anterior, si una compañía fabricante de software es comprada por otra más poderosa, es probable que esa línea de software quede descontinuada y nunca más en la vida vuelva a tener una modificación.
10. Dependencia a proveedores.- Al no poder realizar modificaciones libremente al sistema propietario, por no contar con el código fuente, el cliente se vuelve dependiente de los proveedores quienes se harán cargo de los cambios que se requieran. Dichos cambios requieren de una inversión monetaria.

#### CARACTERÍSTICAS DE KOHA

Koha fue creado en 1999 por Katipo Communication –especialistas en el desarrollo de proyectos de código fuente abierto– para la Horowhenua Library Trust en Nueva Zelanda. Nació a partir de la necesidad de esta biblioteca de contar con un sistema nuevo, capaz de afrontar los retos y cambios que proponía la llegada del año 2000, y para dejar atrás un sistema de 12 años de antigüedad, el cual no seguía más en desarrollo. La biblioteca tenía conocimiento de los altos costos y recursos necesarios para obtener un nuevo sistema, para lo cual requerirían de mejoras capitales que no podían financiarse. Dichos factores fueron clave para que la biblioteca optara por crear un sistema propio, bajo el desarrollo de Katipo. El proyecto se desarrolló durante los

meses finales de 1999 y para enero del 2000 ya estaba lista la primera versión. Ante esto Katipo y la HTL determinaron liberar el sistema bajo la licencia GNU/GPL, para abrir posibilidades de beneficio y desarrollo para otras bibliotecas.

En enero del año 2000 se logró con éxito la primera instalación, y Koha se hizo acreedor a dos reconocimientos: el 3M a la innovación en biblioteca y el interactivo ANZ (Categoría comunitaria / No lucrativa). Koha fue entonces adoptado por otras bibliotecas en Nueva Zelanda, que contrata a Katipo para el soporte. En 2002, Paul Polain (Marsella, Francia) patrocinado por la Nelsonville Public Library de Ohio, comenzó agregar nuevas características de importante magnitud al sistema, especialmente soporte multilenguaje y protocolos estándar internacionales como MARC y Z39.50 para el intercambio de información. Actualmente otra compañía con base en Ohio llamada Liblime ha estado agregando nuevas características al software como el soporte para Zebra, una base de datos contextual de alta velocidad, característica que fue patrocinada por el Crawford County Federated Library System. En 2009, ByWater Solutions y PTF comenzaron a ofrecer servicios de apoyo a Koha en Estados Unidos. En febrero de 2010, Equinox anuncio su expansión para soportar Koha y servicios de *hosting*.

### *Características del sistema*

Koha incluye todas las características previstas para un SIAB, el sistema basa su funcionamiento en una estructura cliente-servidor (Web) soportada mediante una arquitectura LAMP, (Linux, Apache, MySQL, PHP y/o Perl) y ofrece dos interfaces disponibles para cada tipo de usuario, local para bibliotecarios y OPAC para usuarios comunes. El sistema utiliza normas y estándares bibliotecarios internacionales como el formato MARC, en sus versiones MARC 21 y UNIMARC, y ofrece soporte (actualmente únicamente para Linux) para el intercambio de información mediante el protocolo Z39.50; utiliza la norma ISO-2709 y permite el uso de estándares almacenados en SGML enfocados a ambientes Web como, XML, Dublin Core, RSS, MARCXML y Endnote.

En Koha se encuentran la mayoría de las características requeridas para un SIAB, incluyendo:

- Interfaz simple y clara para bibliotecarios y usuarios.
- Búsqueda adecuada a las necesidades de la biblioteca.
- Corre bajo plataformas Linux y Windows, y en cualquier servidor Web, por lo que puede trabajar con terminales sin disco duro ni hardware especializado preferentemente Apache, PHP y MySQL.

- Es compatible con los estándares de MARC21 y UNIMARC.
- Cuenta con módulos de catalogación, circulación, administración, seriadas y adquisición, además de que incluye una ejecución para el OPAC.
- Nuevas plantillas: interfaz mucho más agradable y 100% XHTML y CSS válido.
- Más módulos: lector de noticias, creador de etiquetas, calendario, comentarios en el OPAC, noticias, registros de transacciones, reportes guiados, programador de tareas, incluye nuevas normas, generación de informes a través de base de datos, etcétera.
- API estándar: interface de programa para utilizar herramientas como SRU/W, Z39.50, UnAPI y COINS/OpenURL. También incluye el estándar popular OpenSearch creado por Amazon.
- Soporte multilingüe: lenguas occidentales, orientales y escritura de derecha a izquierda como árabe y hebreo. Contiene traductor de idiomas.
- Multiplataforma: incluye un multi-motor de base de datos, independiente del servidor web. Es decir que la base de datos que contiene Koha en esta versión puede ser instalada en cualquier plataforma: Linux, Mac OSX, FreeBSD, Solaris, sin mayores problemas. Incluye soporte para multi-RDBMS (5,0 MySQL y PostgreSQL), y corre en Apache2, IIS, o en el servidor web de su elección.
- Cuenta con Plugin Zebra: es un motor de base de datos contextual de alta velocidad que permite manejar grandes cantidades de datos estructurado en una variedad de formatos de entrada (Correo electrónico, XML, MARC, etc.), y puede ser utilizado por medio de operadores booleanos.
- Nuevo instalador Web: ayuda a instalar de manera sencilla Koha, fue creado en el módulo ExtUtils en Perl para ser instalado en cualquier plataforma.

Por tanto algunos de los beneficios que presenta el sistema Koha, son:

- Envía nuevas funciones y versiones periódicas.
- Proporciona ayuda en línea.
- Está basado en estándares Internacionales.
- Cuenta con opciones para que el usuario haga apartados del material prestado, envíe solicitudes a la institución, cambie su contraseña o datos de su domicilio, y revise la información y estado en el que se encuentra.

- Su diseño permite mantener la base de datos sin riesgos de modificaciones por parte de personas ajenas o no autorizadas.
- Ofrece soporte continuo mediante portal Web, correo electrónico, mensajero instantáneo, listas de interés.
- Es un sistema con diseño basado en Web y sus componentes son reconocidos como vanguardistas.

### *Arquitectura del sistema*

Koha es ideal para cualquier tipo de biblioteca y es considerado uno de los Sistemas Integrados para bibliotecas de Open Source mejor desarrollado en su categoría, es robusto, altamente personalizable con interfaces claras y simples, descansa 100% sobre herramientas de licencias libres, opera en diversas plataformas, goza de una arquitectura flexible, y es multilíngüe además de ser 100% Web.

### *Requerimientos del sistema*<sup>12</sup>

- Recomendación de equipo servidor:  
El tipo de hardware a utilizar debe evaluarse de acuerdo con la cantidad de clientes que van a usar el sistema y la cantidad de registros que éste va a manejar.

Koha tiene una arquitectura cliente servidor. Por eso se separarán los requerimientos mínimos en 2 partes: los clientes y el servidor.

#### *Servidor:*

- Servidor Web Apache.
- Base de datos MySQL.
- Perl.
- Instalador Koha W32.

La recomendación mínima es:

- Procesador, basado en alguna de las siguientes familias de procesadores:

12 *Especificaciones servidor Koha*, [en línea], [consulta: 30 de Junio de 2010], disponible en Internet: <http://koha.unlp.edu.ar/presentaciones/Especificacion%20Servidor%20KOHA.pdf>

- “INTEL Xeon” o “AMD Opteron” o rendimiento superior compatible con arquitectura X86.
  - Del tipo “Dual core” (2 núcleos).
  - Del tipo “Quad core” (4 núcleos).
- Cantidad de procesadores:
  - 1 procesador.
  - 2 procesadores.
- Memoria RAM a proveer y su escalabilidad
  - Tipo de memoria: DDR2 con una frecuencia de reloj de 667Mhz mínima.
  - Capacidad: la capacidad a proveer inicialmente 4 ó más GB y la capacidad máxima instalable en el equipo al menos a 8 GB.

### *Clientes:*

Cualquier PC con capacidad para ejecutar un explorador WEB, recomendable con la capacidad para ejecutar firefox 2.0 ó superior. La misma necesidad para un usuario del OPAC o para un usuario interno de la Biblioteca.

### *Descripción del sistema*

Koha fue el primer sistema de biblioteca que se liberó al mundo, es un pack completo que comprende una serie de módulos como: Administrador, Adquisiciones, Catalogación, Autoridades, Circulación, Publicaciones Periódicas, Catálogo Público en Línea, (OPAC). Ofrece normas y estándares bibliotecarios internacionales habituales y de nueva generación: MARC, el cual soporta UNIMARC, MARC21 y otras versiones MARC, totalmente parametrizables; utiliza la norma ISO 2709 para la importación y exportación de documentos; presenta interfaces de programa para utilizar herramientas como SRU/ W, Z39.50, UnAPI y COinS/OpenURL, y maneja estándares almacenados en SGML en los diferentes formatos y ambientes Web, como MARCXML, Dublin Core, MODS, RSS, Atom, RDF-DC, SRW-DC, OAI-DC, EndNote, y crea referencias bibliográficas en APA, ISO, MLA, también utiliza el estándar popular OpenSearch creado por Amazon, entre otros.

El sistema Koha se integra por 6 módulos: administración, catalogación, adquisiciones, seriadas, circulación y catálogo al público (OPAC):

- Administración:
  - a. Administración de preferencias del sistema.
  - b. Manejo de preferencias como en el formato MARC, fechas, e-mail,

- plantillas.
- c. Parámetros básicos:
- d. Definición de bibliotecas, sucursales y grupos.
- e. Nos permitirá realizar presupuestos y tipos de cambio.
- f. Tipos de ítems y códigos de circulación para préstamo.
- Catalogación:
  - a. Control de campos obligatorios.
  - b. Control de formato de los datos (texto, números, tesauro).
  - c. Administración de tesauros y listas de autoridad.
  - d. 2 tipos de vista:
    - completa (MARC) para bibliotecarios.
    - simple para usuarios del OPAC.
  - e. Búsqueda por cualquier campo MARC.
  - f. Sistema de catalogación rápida a través de Z39.50.
- Adquisiciones:
  - a. Simple.
    - Sólo permite adquirir obras.
  - b. Completo.
    - Administración de proveedores y presupuestos.
    - Manejo de 3 tipos de presupuestos:
      - Disponible
      - Comprometido
      - Utilizado
    - Trabaja con distintos tipos de divisas.
- Seriadadas:
  - a. Permite manejar la llegada de suscripciones, programar su llegada y reclamar los números faltantes.
  - b. La calendarización de los números de una seriada permite frecuencias diarias, semanales, cada 2 y 3 semanas, mensuales, cada 2, 3, 4 y 6 meses, anual y bianual.
  - c. Permite exportar los datos de los ítems reclamados en formato CSV para que sea exportado en una planilla de cálculo.
  - d. Permite enlazar a una plantilla de “Carta de Reclamación” de un procesador de textos.
  - e. Posibilita el registro de ingreso de material, el reclamo de ítems, inventario, reportes, producción de materiales y registro para Diseminación Selectiva de Información.
- Circulación:
  - a. Funciones de préstamo, devolución y reserva.
  - b. Registro de usuarios y manejo de categorías de usuarios, con

- permisos dados a dichas categorías
- c. Alerta en caso de mora del usuario.
- d. Fecha de devolución automatizada por el tipo de material o forzada por el bibliotecario.
- e. Devolución de un documento con sólo escanear el código de barras.
  - Si hay una reserva sobre éste documento, el Bibliotecario es notificado y puede validar la reserva, y deja el documento no disponible para préstamo.
- OPAC:
  - a. Interfaz simple, limpia y clara.
  - b. Diseño personalizable y adaptable a un sitio institucional.
  - c. Sistema de búsqueda simple y avanzada.
    - Envío de resultados al mail.
    - Impresión de los resultados de búsqueda.
    - Descarga del resultado de búsqueda en PDF.
    - Guarda las búsquedas como favoritos en el sistema.
  - d. Sistema de solicitud de reservas y renovaciones.

Servicios adicionales a los usuarios. Más allá de la consulta a la que todo usuario tiene acceso desde el OPAC de Koha, el sistema incluye tres funcionalidades extendidas:

- Estanterías virtuales: son un conjunto de registros seleccionados post-búsqueda, que cumplen la función de bibliografías, y los cuales pueden ser definidos por el usuario como privados, públicos, o libres.
- Canasta de libros: conocido virtualmente como “carrito de compra”, esta herramienta permite organizar materiales durante una sesión de búsqueda en el catálogo, de esta manera al finalizar la misma, el usuario puede seleccionar qué materiales solicitará la biblioteca, de acuerdo con las políticas de préstamo de la institución.
- Reservas: las reservas se realizan a través del OPAC, el usuario puede colocar materiales en reserva y el sistema le avisa al bibliotecario cuando el material es devuelto y quién lo ha reservado para un siguiente préstamo.



### *Ventajas y desventajas del sistema Koha*

Koha es el sistema que representa al software de código abierto o libre, por lo que posee las siguientes ventajas, señaladas por Montserrat Culebro:<sup>13</sup>

- Ventajas:
  1. Bajo costo de adquisición y libre uso.- El usuario que adquiere software libre lo hace sin ninguna erogación monetaria o a muy bajo costo y ofrece un conjunto de recursos muy amplios. Para la mayoría de usuarios individuales el software libre es una opción atractiva por las libertades que garantiza, y no existe la necesidad de verse agobiado por el precio.  
Se parte de la base de que el software libre carece prácticamente de costo de licencia, y por lo tanto esta parte del presupuesto se puede invertir para mejores fines: mejorar la adaptación de los programas y el desarrollo tecnológico del software.  
El éxito del software libre se debe en su mayor parte a Internet, ya que ésta ha permitido que las personas interesadas en los varios componentes del software libre se pongan fácilmente en contacto con otras.
  2. Innovación tecnológica.- El software libre tiene como objetivo principal compartir la información trabajando de manera cooperativa. Por lo tanto, los usuarios tienen un destacado papel al influir decisivamente en la dirección hacia la que evolucionarán los programas: votando los errores que quieren que sean corregidos, proponiendo nuevas funcionalidades al programa, o contribuyendo ellos mismos en el desarrollo del software.
  3. Requisitos de hardware menores y durabilidad de las soluciones.- Aunque resulta imposible generalizar, existen casos documentados que demuestran que las soluciones de software libre tienen menos requisitos de hardware, y por lo tanto son más baratos de implementar. También tienen un mejor soporte -de manera general- para las versiones antiguas de software y de plataformas de hardware o software minoritarias.
  4. Escrutinio público.- El código fuente del programa está a la vista de todo el mundo, y son frecuentes los casos en que se reportan errores que alguien ha descubierto leyendo o trabajando con ese código. El proceso de revisión pública al que está sometido el

13 Montserrat Culebro Juárez, *op. cit.* p. 48.

desarrollo del software libre le imprime un gran dinamismo al proceso de corrección de errores.

5. Independencia del proveedor.- El software libre garantiza una independencia con respecto al proveedor, gracias a la disponibilidad del código fuente. De esta manera, utilizándolo, el usuario se libera de toda dependencia de un proveedor único, y puede administrar su crecimiento y operación con total autonomía, sin temor de costos ocultos ni extorsiones.
6. Industria local.- El usuario está habilitado para ejecutar un programa, pero no para inspeccionarlo ni modificarlo, por lo que no puede aprender de él; se vuelve dependiente de una tecnología que no sólo no comprende sino que le está expresamente limitada. En México es casi nula la industria de software libre y las aplicaciones de consumo masivo se desarrollan en otros países.

Debido a que el software libre no tiene costo el derecho a copia y, a que dispone del código fuente de la aplicación, es posible desarrollar internamente las mejoras o las modificaciones necesarias, en lugar de encargárselas a empresas de otros países que trabajan con sistemas propietarios. De este modo se contribuye a la formación de profesionales en nuevas tecnologías y al desarrollo local. Por otro lado, todas las mejoras que se realicen no tienen restricciones y se pueden compartir con cualquier otra administración, empresa, institución u organismo que las necesite.

7. Datos personales, privacidad y seguridad.- El software libre, por su carácter abierto, dificulta la introducción de código malicioso, espía o de control remoto, en razón de que el código es revisado por infinidad de usuarios y desarrolladores que pueden detectar posibles puertas traseras. En el mundo del software libre, cualquier programador puede realizar una auditoría para comprobar que no se ha introducido ningún código malicioso, y, a su vez, cualquier entidad puede añadir libremente encriptación adicional a la aplicación que utilice para proteger sus datos.
8. Adaptación del software.- Gracias al hecho de que disponemos del código fuente el software libre permite personalizar los programas tanto como sea necesario hasta que resuelvan exactamente nuestra necesidad. La personalización es un área muy importante en la que el software libre puede responder mucho

mejor que el software de propiedad a unos costos mucho más razonables.

9. **Lenguas minoritarias, traducción, uso e impulso de difusión.**- Las lenguas minoritarias existentes en México, como el náhuatl, zapoteco, mixteco, purépecha, y de nuestras comunidades indígenas, tienen pocas posibilidades de desarrollarse en el mundo del software; por lo que las posibilidades de estas lenguas para acceder al uso y manejo de las computadoras serían casi nulas. En el mundo del software libre estas lenguas, y muchas otras, pueden gozar de desarrollo y propagación, gracias a que no se precisa autorización de ningún propietario y cualquier persona o institución interesada puede realizar las traducciones. Además, si el programa que traducimos no dispone de corrector ortográfico en nuestra lengua podemos desarrollar nuestro propio corrector o adaptar alguno de los existentes en el mundo del software libre.

Al analizar las ventajas que tiene el uso del software libre, se deben tomar en cuenta también sus desventajas, las cuales son:<sup>14</sup>

1. La curva de aprendizaje es mayor.- Si un usuario antes ya usó software propietario generalmente tarda más en aprender a usar un software libre.
2. El software libre no tiene garantía proveniente del autor.
3. Los contratos de software propietario no se hacen responsables por daños económicos, y de otros tipos cuando se usan sus programas.
4. Aunque se necesitan recursos para reparar los errores en el software libre; en el software propietario es imposible reparar errores y hay que esperar a que salga a la venta otra versión.
5. No existen compañías únicas que respalden toda la tecnología para ambos softwares.
6. La mayor parte de la configuración de hardware no es intuitiva.- Se requieren conocimientos previos acerca del funcionamiento del sistema operativo y fundamentos del equipo. Sin embargo la documentación referente a la configuración del hardware es tan explícita y detallada que le permite al usuario inexperto profundizar en el conocimiento de su hardware.
7. Con este software únicamente los proyectos importantes y de trayectoria

14 *Ibid.* p. 55.

- tienen buen soporte, tanto de los desarrolladores como de los usuarios.
8. El usuario debe tener nociones de programación al usar este software. La administración del sistema recae mucho en la automatización de tareas y esto se logra utilizando, en muchas ocasiones, lenguajes de guiones (perl, python, shell, etc.). pero existen en la actualidad muchas herramientas visuales que le permiten al usuario no técnico llevar a cabo tareas de configuración del sistema de una manera gráfica muy sencilla sin la necesidad de contar con conocimientos de programación.
  9. En sistemas con acceso a Internet, se debe monitorear constantemente la corrección de errores en todos los programas que contengan dichos sistemas porque son fuentes potenciales de intrusión.

### *Forma de obtenerse*

Para una mayor apreciación del sistema Koha, existe una demostración en línea del programa en la siguiente dirección: <http://koha.liblime.com/>

Por ser un software de código abierto no se adquiere, se descarga a través de su sitio oficial disponible en la siguiente dirección: <http://www.koha.org/>

## ANÁLISIS COMPARATIVO

En la siguiente tabla se muestra una comparación, en la cual se observan las generalidades y la gestión de estándares de cada sistema, así como sus características funcionales incluyendo de manera cuantitativa los módulos con los que cuenta cada SIAB descrito anteriormente:

Generalidades		
Nombre del sistema	SPX	Koha
País de origen	México	Nueva Zelanda
Idioma	Español e Inglés	Multilingüe
Creador	SPX S. A. de C. V.	Katipo Communications Ltd., Liblime, ByWater Solutions y PTF
Licencia	Licencia de uso (Staff y Web)	GNU GPL
Documentación	Sí	Sí
Arquitectura	Arquitectura en 3 capas basadas en Web	Arquitectura flexible basada en Web
Modificación al código fuente	No	Sí
Costo	Sí	No
Mantenimiento anual	Sí	No

Actualización	Sí	Sí
Linux	Sí	Sí
Unix	Sí	Sí
Bases de datos	Oracle, Db2, Informix y SQL Server	MySQL
Lenguaje(s) de programación	Perl	PHP, Perl
Sistema Operativo	Windows, Mac, Unix y Linux / FreeBSD Solaris AIX HP-UX Mac OS X	Multiplataforma (Windows Linux, Unix)
<b>Características Funcionales</b>		
Administrador	Sí	Sí
Adquisiciones	Sí	Sí
Catalogación	Sí	Sí
Gestión de Autoridades	Sí	Sí
Circulación	Sí	Sí
Gestión de Publicaciones Periódicas	Sí	Sí
Catálogo Público en Línea (OPAC)	Sí, interno y en la Web	Sí, interno y en la Web
Estadísticas	Sí	No
Gestión de Tesauro	Sí	Sí
Módulo de seguridad y control de acceso	Sí	Sí
Inventario	Sí	No
<b>Gestión De Estándares</b>		
ISO 2709	Sí	Sí
Ciiente Z39.50	Sí	Sí
MARC21	Sí	Sí
MARCXML	Sí	Sí
Estándar(es) SRU, XML	Sí	Sí

Al comparar la estructura de ambos sistemas se puede observar que tanto SPX como Koha cuentan con los mismos módulos. Al igual que ambos ofrecen estándares bibliotecarios internacionales.

En cuanto a los requerimientos tecnológicos que solicitan ambos sistemas se puede observar que los requerimientos de Koha son similares a los de SPX, a diferencia de que en el primero los requerimientos informáticos son mínimos como sucede con el sistema operacional en que debe correr el software, y por lo tanto, Koha es más fácil de implementar.

Tanto SPX como Koha poseen la documentación pertinente para entender y usar el software, por lo que cuentan con documentos en distintos idiomas que permiten generar un amplio acceso a los manuales de usuarios. Los documentos se encuentran en formato XML, HTML, y en PDF. Dichos documentos

presentan desde instalaciones parciales por módulos hasta instalaciones completas.

El costo es uno de los puntos que hace notable la diferencia entre el software propietario y el libre; en el primero existen gastos añadidos, que no sólo son un precio de adquisición de licencia. También cuesta mantenerlo, operarlo, parametrizarlo y personalizar su sistema, y tomar cursos de capacitación no contemplados en el contrato inicial. Sin dejar de mencionar la dependencia al proveedor del sistema. Por ejemplo, en el caso de que se necesite realizar una modificación al sistema, es necesario pagar una elevada suma de dinero a la compañía desarrolladora, para que sea ésta quien lleve a cabo la modificación, lo que no permite llegar a la innovación.

En cuanto a las licencias de uso, hay que manejar un mínimo de ellas, siempre y cuando sean concurrentes (liberadas), puesto que éstas tienen vigencia y por tanto, si no se siguen pagando no hay actualizaciones.

Por lo anterior es importante controlar estos costos, ya que si no se tiene una buena planificación, adquirir un sistema propietario puede llegar a ser un fracaso o bien resultar más costoso de lo previsto.

Un gran porcentaje de los usuarios de SPX manifestó que los servicios de mantenimiento no eran precisamente buenos, que a veces el sistema presentaba ciertas fallas y que el tiempo de respuesta por parte de la compañía no era el adecuado.

En cambio, la implementación del software libre es más barata por los requerimientos tecnológicos mínimos que necesita; en cuanto al costo es muy bajo (si es que existe) y ofrece un conjunto de recursos muy amplios. El software libre garantiza una independencia con respecto al proveedor gracias a la disponibilidad del código fuente, lo que permite modificar y realizar las adecuaciones necesarias al sistema para cubrir las necesidades tanto de la biblioteca como del usuario.

Existen valores intangibles en la descarga e instalación de un software libre, prueba de ello son las comunidades, blogs, foros y/o listas de discusión, grupos de apoyo *online*, etcétera; lo que da lugar a una colaboración horizontal donde se atienden cualquier tipo de dudas acerca del funcionamiento, los requerimientos de sistema, etcétera; y además la respuesta es casi inmediata, puntual, con ejemplos resueltos en la misma comunidad, con sugerencias para el aprovechamiento del recurso y con la misma filosofía de libertad.

Koha es un sistema robusto que está ganando adeptos entre la comunidad bibliotecaria internacional que requiere de un SIAB totalmente libre y adaptable, pero sobre todo funcional y en constante desarrollo, factor que ha influido de manera determinante para que al ser comparado con sistemas propietarios excesivamente costosos, demuestre de manera óptima que el

software libre en distintas áreas, en este caso las bibliotecas, es una opción real que no podemos ignorar. El software libre también crea una sostenibilidad económica, y expande el mercado tradicional basado en mercado, oferta y demanda, y elimina la idea de que lo más caro siempre será lo mejor, lo que fuerza a los proveedores de software propietario a reducir costos y mejorar sus servicios ante la alternativa de creciente competencia y éxito que los sistemas libres están generando, como la posibilidad de migrar hacia cualesquiera de ellos, con todas las ventajas que esto representa.

En Estados Unidos y Canadá el interés por el uso de SIAB basados en software libre está creciendo, aunque todavía de una manera incipiente, pero ya se ve reflejado junto a los grandes SIAB propietarios, como lo podemos observar en el último estudio publicado por *Library Journal*:<sup>15</sup>

COMPANY	SYSTEM NAME	NEW CUSTOMERS			TOTAL SALES			2009		
		2007	2008	2009	2007	2008	2009	U.S. SALES	NON-U.S. SALES	TOTAL INSTALLED
<b>Full function multiuser systems for Public, Academic &amp; Consortia</b>										
<b>Auto-Graphics, Inc.</b>	AGent VERSO	44	23	16	47	24	19	18	1	311
<b>BiblioMondo</b>	Concerto						2			470
<b>BiblioMondo</b>	PortFolio						4			930
<b>Biblionix</b>	Apollo		49	55		49	55	55	0	180
<b>ByWater Solutions</b>	Koha			7			7	7	0	105
<b>Civica</b>	Spydus 8 / MS	25	25	15	51	30	25	2	23	263
<b>Equinox Software</b>	Evergreen		6	15		6	18	0	0	205
<b>Ex Libris</b>	Voyager	4	5	1	4	5	2	2	0	1,245
<b>Ex Libris</b>	ALEPH 500	26	23	38	29	26	53	2	51	2,252
<b>Infor</b>	Vubis Smart	9	6	6	40	46	12	0	10	373
<b>Infor</b>	V-Smart		1	6		11	12	1	11	36
<b>Innovative Interfaces, Inc.</b>	Millennium	79	61	34	95	64	45	26	19	1,377
<b>LibLime</b>	Koha	56	40		57	40				
<b>The Library Corporation</b>	CarlX / Carl.Solution	0	0	0	0	0	0			22
<b>The Library Corporation</b>	Library.Solution	35	32	30	35	32	30	26	4	719
<b>OCLC</b>	AmLib Library Management System			5			5		5	522
<b>Polaris Library Systems</b>	Polaris Integrated Library System	22	51	33	32	56	33	30	3	348
<b>PTFS</b>	Koha Support			11			12	12		12

15 Marshall Breeding, "Automation System Marketplace 2010: New Models, Core Systems. Discovery interfaces add a new facet to the marketplace", [en línea], en *Library Journal*, Núm. 6, April 1, 2010, [consulta: 29 de Junio de 2010], disponible en Internet: <http://www.libraryjournal.com/article/CA6723662.html>

Koha es un sistema adaptable que posee normas, protocolos vigentes y de nueva generación consolidados y aprobados por la comunidad bibliotecaria internacional. Además de poseer un constante desarrollo en su sistema, por parte de sus creadores y de la comunidad mundial.

El software libre Koha cumple sin duda alguna con las necesidades que la biblioteca mexicana demanda, por lo que se puede decir que la principal diferencia con el software propietario SPX se enmarca en lo relativo a la dependencia del proveedor y a los costos añadidos por mantenimiento anual. Razón por la cual la hipótesis que dio pie a este trabajo fue verdadera.

## CONCLUSIONES

Hoy en día, cada vez más bibliotecas en el mundo optan por el uso de software libre para gestionar las distintas actividades que llevan a cabo. Basta con hacer un recuento de la literatura que se ha escrito sobre el tema durante los últimos cuatro años, para darse cuenta de que el uso de este tipo de software está creciendo de manera favorable en todo tipo de bibliotecas. Bibliotecarios, informáticos y promotores del software libre trabajan en conjunto para crear alternativas de gestión a los sistemas propietarios, y a buscar que el desarrollo y mejora de SIAB se traduzcan en beneficios directos, bajo un ambiente de libertades donde usuarios e instituciones sean los únicos beneficiados.

Hay que romper muchos miedos y desconocimiento, ya que el software libre no es sinónimo de software pirata, desgraciadamente ésta idea está muy generalizada en el ámbito profesional mexicano. Por esto es importante revisar lo referente a las libertades que el software libre brinda, para saber que la implementación de este tipo de sistemas es completamente legal y se encuentra respaldado por licencias legales que evitarán cualquier tipo de problema, ya sea como usuario particular, biblioteca, institución u organización.

Es necesario que los bibliotecarios mexicanos tomen conciencia de las virtudes que ofrece el software libre y todo el panorama que se desarrolla alrededor de él. Esto implica que se adquieran nuevos conocimientos y nuevas perspectivas, de manera que amplíen su visión más allá del enfoque tradicionalista de trabajo en lo ya establecido y/o comercial. Koha, al igual que muchos otros sistemas libres, es una opción real para las bibliotecas mexicanas y del mundo, la decisión de avanzar está en cada uno de nosotros como bibliotecarios.

El incremento en el uso del software libre en las bibliotecas dependerá en gran medida de que los bibliotecarios ahonden en el conocimiento sobre el software libre y participen con otras comunidades multidisciplinarias que



usan y desarrollan este tipo de software, además de analizar los desarrollos para bibliotecas y percibir que para el desarrollo de SIAB sustentables los recursos financieros no son un obstáculo para que las bibliotecas se automatizen.

Actualmente es posible automatizar cualquier biblioteca con un mínimo de inversión y sin costos ocultos, ni contratos amañados que comprometan el uso de un software por tiempo indeterminado, algo que se promete pero no siempre se cumple. Asimismo, existe la posibilidad de renegociar con los proveedores comerciales los contratos bajo el *ultimatum* de dejarlos y cambiarse a un software libre. Se puede afirmar que el software libre es hoy día una alternativa de competencia real al software propietario.

## BIBLIOGRAFÍA

- Angulo Marcial, Noel, *Terminología de automatización documental*, México, SEP, Coordinación Nacional para la Planeación de la Educación Superior, 1988.
- Arriola Navarrete, Oscar y Avila González, Armando, “El software libre y la enseñanza de la catalogación: una relación amistosa”, en *Revista Códice*, Vol. 4, núm. 2 (Julio-Diciembre 2008). pp. 21-32.
- Arriola Navarrete, Oscar y Butron Yañez, Katya, “Sistemas integrales para la automatización de bibliotecas basados en software libre”, [en línea], en *ACIMED*. V.18, núm.6, diciembre 2008, [consulta: 28 de Junio de 2010], disponible en Internet: [http://bvs.sld.cu/revistas/aci/vol18\\_6\\_08/aci091208.html](http://bvs.sld.cu/revistas/aci/vol18_6_08/aci091208.html)
- Breeding, Marshall. “Automation System Marketplace 2010: New Models, Core Systems. Discovery interfaces add a new facet to the marketplace”, [en línea], en *Library Journal*, Núm. 6, April 1, 2010, [consulta: 29 de Junio de 2010], disponible en Internet: <http://www.libraryjournal.com/article/CA6723662.html>
- Carranza Torres, Martín, *Problemática jurídica del software libre*, Buenos Aires, Lexis Nexis, 2004.
- Casanova Huerta, Araceli, *Software libre una opción viable en la organización bibliográfica de fotografías: caso Biblioteca de las Artes*, México, La autora, 2010.
- Culebro Juárez, Montserrat, Gómez Herrera, Wendy Guadalupe y Torres Sánchez, Susana, *Software libre vs software propietario: ventajas y desventajas*, [en línea], México, Creative Commons, 2006, [consulta: 29 de Junio de 2010], disponible en Internet: <http://www.softwarelibre.cl/drupal/files/32693.pdf>
- Especificaciones servidor Koha*, [en línea], [consulta: 30 de Junio de 2010], disponible en Internet: <http://koha.unlp.edu.ar/presentaciones/Especificacion%20Servidor%20KOHA.pdf>

- Magal Wals, José, *Temas de biblioteconomía universitaria y general*, España, Editorial complutense, 2001.
- Rodríguez, Gladys Stella, El software libre y sus implicaciones jurídicas, en *Revista de derecho*. Núm. 30. Barranquilla, Universidad del Norte, 2008.
- Stallman, Richard M., *Software libre para una sociedad libre*, [en línea], GNU Press, 2002, [Consulta: 30 de Junio de 2010], disponible en Internet: <http://biblioweb.sindominio.net/pensamiento/softlibre/softlibre.pdf>
- Tennant, Roy, *Library Software Manifesto*, [en línea], TechEssence, 2007, [consulta: 30 de Junio de 2010], disponible en Internet: <http://techessence.info/manifiesto/>

