



SECRETARÍA DE EDUCACIÓN PÚBLICA
DIRECCIÓN GENERAL DE
EDUCACIÓN SUPERIOR UNIVERSITARIA

ESCUELA NACIONAL DE
BIBLIOTECONOMÍA Y ARCHIVONOMÍA

“EL SOFTWARE DE CÓDIGO ABIERTO: UNA
ALTERNATIVA PARA LA GESTIÓN INTEGRAL DE LA
BIBLIOTECA”

T E S I N A

QUE PARA OBTENER EL TÍTULO DE
LICENCIADO EN BIBLIOTECONOMÍA

P R E S E N T A :

VÍCTOR FERNANDO FLORES VARGAS

ASESORES: Mtro. Oscar Arriola Navarrete
Mtra. Emma Hernández Gómez

MÉXICO, D. F.

2011

TABLA DE CONTENIDO

PREFACIO	I
INTRODUCCIÓN	II
Capítulo 1. La automatización de bibliotecas	1
1.1 Definición	3
1.2 Antecedentes	4
1.3 Objetivos	7
1.4 Razones para automatizar	8
1.5 Ventajas de la automatización de bibliotecas	17
1.6 Planeación de un proyecto de automatización	21
1.7 Requerimientos	29
Capítulo 2. Los Sistemas Integrales de Gestión Bibliotecaria de código fuente abierto	48
2.1 Definición	50
2.2 Antecedentes	51
2.3 Desarrollo de los SIGB	57
2.4 Características	60
2.5 Requerimientos funcionales	61
2.6 La historia del Software Libre (Linux y el proyecto GNU)	69
2.7 Software Libre vs Software de Código Fuente Abierto	73
2.8 Ventajas del uso de Software Libre	79
2.8.1 Ventajas SIGB de Código Fuente Abierto	91
2.9 Tipos de Software y Licencias	93
2.10 Software de Código Fuente Abierto: Una alternativa real de gestión para las bibliotecas	100
Capítulo 3 Koha: Sistema Integrado de Gestión Bibliotecaria Instalación y funcionalidad	115
3.1 ¿Qué es Koha?	117
3.1.1 Historia	117
3.1.2 Características	118
3.1.2.1 Sistema integrado e integrable	119
3.1.2.2 Multiplataforma	119
3.1.2.3 Servicios adicionales a los usuarios	120
3.1.2.4 Desarrollo	120
3.1.2.5 Documentación y soporte	121
3.1.2.6 Requerimientos técnicos	121
3.2 Instalación de Koha en Ubuntu Jaunty Jackalope	122
3.2.1 Instalación de Ubuntu 9.04	123
3.2.2 Línea de comandos para instalación de Koha en Ubuntu	123
3.2.3 Servidor LAMP	125
3.2.4 Configurando Apache Server	127
3.2.5 Webmin	129
3.2.5.1 Instalando Webmin	130
3.2.6 Fuentes, paquetes y librerías	132

3.2.7 Instalando los paquetes necesarios	135
3.2.8 Módulos Perl	138
3.2.9 Koha. Configurando permisos en Linux	140
3.2.10 Configurando MySQL	142
3.2.11 Instalación de Koha	145
3.3 Instalación de Koha V 2.29 en Windows XP SP3 arquitectura X86 .	150
3.3.1 Programas requeridos	151
3.3.2 Instrucciones de instalación	152
3.4 Funcionalidad del sistema	171
3.4.1 Módulo de administración	174
3.4.2 Módulo de circulación	183
3.4.3 Módulo de catalogación	188
3.4.4 Módulo de informes	196
3.4.5 Módulo OPAC (catálogo al público de acceso en línea)	198
3.4.6 Módulo de adquisiciones	205
CONCLUSIONES	208
BIBLIOGRAFÍA	211
ANEXO. Instalación de Ubuntu 9.04 “ Jaunty Jackalope”	216

PREFACIO

La elaboración del presente trabajo se basa de manera personal en dos ejes fundamentales que dieron pauta para llevar a cabo una metodología de investigación y propuesta, enfocada a la promoción de implementación de software libre, concretamente del Sistema Integrado para Bibliotecas llamado Koha, en bibliotecas mexicanas; de primera instancia, como alumno de la licenciatura en Biblioteconomía interesado principalmente en cuestiones relacionadas con las tecnologías de información, con un énfasis principal en el área de automatización de bibliotecas y, de manera secundaria, como usuario de software libre y de código abierto desde el año 2004, surgiendo un interés personal por el tema, al reconocer el desarrollo que han tenido diversas bibliotecas a nivel mundial utilizando distintos sistemas integrales para la automatización de procesos, basados en software libre.

Este trabajo presenta por primera vez en México una propuesta real para la utilización del Sistema Integrado de Bibliotecas “Koha”, fundamentado en una base teórica de acuerdo al área que lo enmarca, para concluir con una secuencia ordenada que describe el procedimiento y las herramientas necesarias para llevar a cabo la implementación y uso basado en la funcionalidad del Sistema Integrado. Esto, sin duda alguna, da seguimiento y representa una contribución a la bibliografía paralela escrita por bibliotecólogos mexicanos, pioneros en el desarrollo, exposición y promoción del tema a distintos niveles en México. Cabe mencionar que lo aquí desarrollado, es sólo una propuesta que sitúa al software libre como alternativa para quien decida llevar a cabo su implementación, tomando responsabilidad de todo el panorama que gira en torno a ella, y a su vez, colaborando en la tarea de fomentar el uso de herramientas libres que contribuyan a la automatización de unidades de información.

INTRODUCCION

Hoy en día muchas bibliotecas tienen automatizadas sus colecciones y cuentan con algún tipo de Sistema Integrado de Gestión Bibliotecaria (SIGB) para la gestión de sus procesos y servicios; dicho sistema implica diversos costos (actualización, soporte, licencias, módulos, etc.) que universidades e instituciones privadas pueden costear fácilmente. Si bien algunos de estos sistemas son relativamente baratos, también existen las bibliotecas que no tienen acceso a los mismos y por consiguiente no están a la vanguardia en cuanto a gestión y a todo lo que representa la utilización de un sistema con ciertas particularidades en la biblioteca. Así mismo, en un marco general se desconocen las características de un sistema de código abierto, que puede funcionar al nivel de un sistema propietario y que podría ser una opción para aquellas bibliotecas que no cuenten con suficiente presupuesto, pero si con muchas ganas de hacer algo positivo por su institución (gestionar servicios de manera computarizada, contar con un catálogo automatizado, bases de datos de usuarios, etc).

Para llevar a cabo este trabajo se trazan tres líneas de investigación divididas en tres capítulos, que tienen como objetivo final, promover el uso de software libre en bibliotecas mexicanas, enfocándonos de manera específica a que cualquier usuario o gestor de la información conozca las características y funcionalidad de un SIGB de código abierto, y los procesos que se involucran dentro del campo de la automatización de bibliotecas, concluyendo con el análisis de uno de los sistemas más prominentes a nivel mundial para la gestión automatizada de las bibliotecas de manera libre, el sistema Koha; todo ello, como lo hemos mencionado anteriormente, con el objetivo de promover su uso en las bibliotecas mexicanas.

El trabajo se enmarca dentro del campo de la automatización de bibliotecas, por ende, en el primer capítulo se revisan los principales conceptos, su aplicación específica en la bibliotecología, abordando de manera general, sus

antecedentes, características y desarrollo, hasta los objetivos que persigue dentro de la biblioteca.

En el segundo capítulo se analizan los conceptos básicos de los SIGB, su origen, características, desarrollo, así como, los conceptos y particularidades del software libre que los interrelacionan, describiendo también de manera breve, las diferentes opciones que existen para llevar a cabo la automatización y gestión de una biblioteca, utilizando software con dichas características.

En México, la investigación referente a software libre para bibliotecas no es muy extensa, pero tampoco es nula, basta con revisar la sucinta bibliografía escrita por bibliotecólogos mexicanos como el Mtro. Oscar Arriola Navarrete, el Mtro. Juan Manuel Zurita Sánchez y recientemente Armando Escobedo, para darnos cuenta que el interés por este tema se está ampliando. El software libre para bibliotecas conforma toda una línea de investigación en sí misma que debe realizarse y requiere de la cooperación mutua de colegas interesados en dicho tema. Como todo proceso exitoso, se parte de un inicio y se traza una línea de investigación en la que alumnos y profesionales del área se involucran para llevar a cabo su desarrollo. Ante esto, para darle seguimiento a lo ya escrito por otros bibliotecarios, el tercer capítulo de este trabajo se basa en llevar a cabo una instalación limpia y exitosa, así como la descripción de cada uno de los módulos que integran el sistema koha en sus dos versiones disponibles, (Linux y Windows), lo cual es de suma importancia, ya que brinda la alternativa de uso a usuarios no familiarizados con entornos de distribución libre, que de alguna manera, por tradición o implantación, tienden al uso común de sistemas operativos propietarios. Todo esto llevado a cabo de la manera más detalladamente posible, ayudados por imágenes que nos guiarán durante todo el proceso que concluye de manera exitosa. De esta forma se complementa la parte teórica descrita en los primeros capítulos y con la cual toda investigación y propuesta debe contar.

Por último, se desarrollan las conclusiones finales, y como anexo a este trabajo se presenta una guía para llevar a cabo la instalación del sistema operativo

“Ubuntu 9.04 Jaunty Jackalope”, sobre el cual instalamos Koha. Cabe mencionar que este sistema, también funciona y es estable en otras distribuciones, como pueden ser “Debian” y “OpenSUSE”, de manera base o en una virtualbox.



CAPÍTULO 1

La automatización de bibliotecas

1. La automatización de bibliotecas

El desarrollo de la sociedad de la información, centrada en el progreso de las telecomunicaciones y de las nuevas tecnologías de la información, ha producido cambios radicales en los modos de producción debido a que los procesos necesarios para la generación de bienes se conciben como procesos de alto consumo de información.¹

Inmersas en estos cambios e incluidas en los mismos, las bibliotecas han emigrado a partir de los años sesenta, de estructuras tradicionales hacia estructuras automatizadas primero, e híbridas después, con el fin de evolucionar hacia la constitución de la biblioteca digital, la cual se constituirá y tendrá como objetivo primario facilitar el acceso universal a los recursos de información.²

Los países desarrollados han generado, en esta dirección, una serie de proyectos que, según observa Mary Bolin³ se concentraron en las décadas de los setenta y de los ochenta, en la conversión manual/automatizada del catálogo de la biblioteca y en la generación de catálogos en línea de acceso público (OPAC). Estas iniciativas fueron factor clave para avanzar hacia una nueva dimensión en el intercambio de información y la conformación de redes, y por ende, en el desarrollo del término “automatización de bibliotecas”.

Los países en desarrollo, por el contrario, afrontan en la actualidad, bajo la influencia de una diversidad de factores externos e internos, el desafío de poner a disposición de la comunidad académica local e internacional los fondos y colecciones de sus bibliotecas mediante un adecuado proceso de conversión de los catálogos, con el objetivo de acompañar los profundos cambios estructurales y funcionales a los que se hallará sometida la biblioteca en los inicios del milenio.

¹ CANO, V. “Information technology and the future of professional Library Practice”. En: H. P. GEH y H., WALCKIERS, M. *Library Networking in Europe*. London: TFPL, 1996. p.51-65.

² LANKES, D. R. “Lesson learned from K-12 digital reference services”. En: *Reference & User Services Quarterly*. v. 38, no. 1, 1998. p.63-71.

³ BOLIN, M. K. “Automating Idaho’s libraries”. En: *Resource Sharing & Information Networks*, v.9, no.1, 1993. p.79-94.

1.1 Definición

El término “automatización”, por sí mismo, se puede englobar en distintos ámbitos y diversas áreas del conocimiento; como concepto general, Parasuraman y Riley lo definen como “la ejecución realizada por un agente computarizado previamente programada por un humano”.⁴

El English Oxford Dictionary lo define automatización como “la aplicación de un control automático a cualquier campo de la industria o ciencia”.⁵

Haciendo un análisis breve de estos conceptos, entendemos que para llevar a cabo la automatización es necesario el uso de computadoras coadyuvadas del factor humano, mismas que tendrán como objetivo la realización de distintas tareas o procesos con la mayor eficacia posible.

Dentro del ámbito bibliotecológico, el término “automatización de bibliotecas” no discierne en lo absoluto; para Voutssás el término “involucra las técnicas, esfuerzos, y dispositivos para mecanizar, soportar o producir en masa los productos y servicios con que cuenta una biblioteca.”⁶ Globalmente, el término “automatización de bibliotecas” ha sido aplicado a las herramientas y actividades que incorporan tecnologías dentro de los procesos y servicios de la misma.⁷ Desde la aparición del término, éste se ha referido al procesamiento de la información que puede ser manipulada, reorganizada, almacenada o comunicada en lo que comúnmente se le ha denominado como “formato legible por máquina”, de gran relevancia en la historia de la automatización y del que se hablará más adelante.

⁴ DEGANI, Asaf. *Taming HAL: Designing interfaces beyond 2001*. USA: Mcmillan, 2003. p.284.

⁵ *Ibid.*

⁶ VOUTSSÁS MÁRQUEZ, Juan. “Historia de la automatización de bibliotecas en México”. En: LICEA DE ARENAS, Judith. *Cuarenta y cinco años de estudios universitarios en bibliotecología*. México: UNAM, Facultad de Filosofía y Letras, 2001. p.56.

⁷ WEDGEWORTH, Robert. *World Encyclopedia of Library and Information Services*. USA: American Library Association, 1993. p.470.

El ODLIS (Online Dictionary for Library and Information Science) define la “automatización de bibliotecas” como el diseño e implementación de sistemas computarizados con el objetivo de reemplazar tareas realizadas originalmente de manera manual en la biblioteca.⁸

Por su parte, Herrera⁹ sugiere que la automatización de bibliotecas consiste en aplicar las tecnologías de información y computación en los procesos primordiales de una biblioteca.

Haciendo un análisis de los conceptos mencionados anteriormente, se podría concluir que el término “Automatización de bibliotecas” implica el uso de tecnologías de la información y la de comunicación, así como sistemas computarizados con el propósito de gestionar cualquier tipo de proceso y servicio en una unidad de información.

1.2 Antecedentes

Para entender los alcances que hoy en día puede lograr un proyecto de automatización, es importante mencionar el inicio y el desarrollo que éstos han tenido. A continuación se proporciona un atisbo de la automatización de bibliotecas.

Después del simple procesamiento de datos aplicado a actividades de circulación, los primeros sistemas automatizados basados únicamente en la catalogación de los registros bibliográficos, fueron introducidos en los Estados Unidos a finales de los años 60¹⁰ pudiendo considerarse a MELVIL (Catálogo automatizado de la Universidad de California) y MSUS/PALS (El sistema de la Universidad Estatal de Minnesota), como los pioneros.¹¹ El elemento clave en

⁸ M. REITZ, Joan. “Library automation”. En: *On-line dictionary of library and information science*. [en línea]. [Consulta: 02 de Agosto de 2009]. Disponible en Internet: <http://lu.com/odlis/search.cfm>.

⁹ HERRERA MORALES, José Román. *Software de Acceso a la Biblioteca Digital*. México: Facultad de Telemática: Universidad de Colima, 2004. p.1.

¹⁰ LASS, Andrew. *Library automation in transitional societies: lessons from Eastern Europe*. USA: Oxford University Press, 2000. p.93.

¹¹ YEE, Martha. “System design and cataloging meet the user: user’s interfaces to online public access catalogs”. En: *Journal of American society for information science* N° 42 (2) p.78.

la automatización de bibliotecas, fue reconocer que se debía contar con un registro legible por máquina, mismo que una vez creado, pudiera ser usado repetidamente y para diferentes propósitos. El estándar para hacer esto posible fue lo que se conoce hasta hoy en día, como Formato MARC, el cual fue creado por la Library of Congress y que marcó la pauta para el inicio de la producción a gran escala de registros bibliográficos, con la ventaja principal que las bibliotecas que lo utilizaban podían intercambiarlos y a su vez crear catálogos compartidos. La segunda generación de sistemas automatizados emerge a mitad de los años 70; fueron hechos en base a un estándar de registros bibliográficos, con la ventaja de múltiples usos en la biblioteca; una vez que el registro había sido ingresado al sistema podía ser utilizado en diferentes módulos del mismo.

A la par en el campo del hardware hubo importantes desarrollos, lo cual amplió las posibilidades para las bibliotecas. La revolución de las microcomputadoras tomo lugar, volviéndose relativamente baratas el costo original, con lo cual las bibliotecas pudieron comprar sus propios equipos y por ende sus propios sistemas de automatización. Pero los sistemas comprados no fueron del todo satisfactorios, trabajaban bajo un entorno “Mainframe” en computadoras “Mainframe”, con lo cual se limitaba a los bibliotecarios el acceso y la intervención en los mismos; de cualquier forma estos sistemas de automatización convencieron a las bibliotecas para automatizar los procesos de catalogación, circulación y adquisición.

La tercera generación de sistemas de automatización nació en los años 80; un dramático y acelerado desarrollo de hardware resultó en poderosas computadoras a bajo costo y puestas en el mercado al público en general; el nacimiento de sistemas abiertos, especialmente UNIX, eliminaron la dependencia a hardware específico, que anteriormente había sido un serio problema. Para ese entonces la base de un sistema de automatización vino a ser principalmente el poder gestionar bases de datos, considerándose clave el soporte de SQL y protocolos como el Z39.50 para el intercambio de información.

Los bibliotecarios entonces se familiarizaron con la estructura cliente-servidor y cambios radicales tuvieron lugar en el desarrollo de interfaces de usuario. El progreso acelerado de las tecnologías de redes ha influido de manera importante en el desarrollo que ha tenido la automatización en los últimos 20 años de manera continua; estas redes, originalmente concebidas como utilidades bibliográficas, nacieron para afrontar el conflicto de que cada biblioteca como entidad aislada no puede tener disponible toda la información existente, ya sea por falta de recursos económicos o de capacidad de proceso, y cuyo fundamento se basa en la compartición de registros bibliográficos, la construcción de catálogos colectivos, servicios de préstamo y acceso al documento; todo esto habilitado mediante políticas de cooperación.¹²

Así mismo, para finales de los 80's ya se podía disfrutar de velocidades altas en tazas de transferencia, con lo que sonidos, textos e imágenes podían ser transferidos sin problema alguno entre redes locales. El surgimiento de Internet llegó a marcar la pauta para un acceso a la información global, lo cual ha sido siempre el objetivo del usuario.

A lo largo de cuatro décadas los sistemas de automatización han experimentado grandes cambios; tendencias actuales en el campo de la automatización incluyen el crecimiento importante de "add ons", mayormente relacionados a la entrega de contenido digital a los usuarios (portales, ligas a contenidos de texto completo, metabuscadores, etc.), y una mejor integración con ambientes Web (uso de plantillas XML, aplicaciones ejecutables vía Web, etc.); y para el uso de bibliotecas académicas ,abarcan la integración de sistemas de aprendizaje para el usuario, otorgando acceso a material en línea relacionado con las búsquedas del autor.¹³ Esto demuestra que ha habido una evolución razonable en los sistemas de automatización de bibliotecas, cada día adaptándose más a las tecnologías de la información y la comunicación y a los retos que éstas presentan.

¹² SAORÍN PÉREZ, Tomas. *Modelo conceptual para la automatización de bibliotecas en el contexto digital*. [en línea]. España: El autor, 2002 [Consulta: 06 de Agosto de 2009]. Disponible en Internet: <http://www.tesisenred.net/TDR-0725106-121514/> p.81.

¹³ M. REITZ, Joan. *Op. cit.*

1.3 Objetivos

La automatización de bibliotecas persigue objetivos específicos que globalmente están relacionados hacia la mejora de los servicios y el tratamiento computarizado de los procesos que se llevan a cabo en una biblioteca, lo que, en consecuencia, genera niveles óptimos de productividad en relación a los mismos. Si bien hoy en día debemos extender el término hacia nuevos enfoques tecnológicos, existen objetivos que se han perseguido y son base en cualquier proyecto de automatización; éstos se mencionan a continuación:

- ☞ Mejora de los procesos.
- ☞ Optimización de tiempos, costos y movimientos.
- ☞ Mejora de los servicios.
- ☞ Compartir recursos de información con bibliotecas afines.
- ☞ Implementar el control que se tiene sobre el material con el que cuenta la biblioteca.
- ☞ Eficacia y eficiencia en las actividades realizadas por la biblioteca.

Para Saorin Pérez, la automatización de bibliotecas debe ser entendida dentro de un contexto más amplio, que se puede denominar tecnologías de información para la documentación, en donde existen numerosas aplicaciones externas al ámbito estrictamente bibliotecario (bases de datos comerciales, digitalización, gestión de documentos corporativos, servicios de información en Internet, edición electrónica), que le afecta y con el que se prevé un intercambio de experiencias y soluciones cada vez más estrecho.¹⁴ A su vez, Lancaster, menciona que el objetivo de la automatización es integral y se enfoca en reducir los procesos manuales para sistematizar la congruencia de procesos automatizados en forma armónica y congruente, evitando duplicaciones, procesos extraordinarios o incompatibilidades.¹⁵

¹⁴ SAORÍN PÉREZ, Tomas. *Op. cit.* p.79.

¹⁵ LANCASTER, F. W. *Evaluación y medición de los servicios bibliotecarios*. México: UNAM, 1983. p.310.

Si bien los objetivos básicos de la automatización son claros, deben estar bien planteados en la fase de planeación del proyecto de automatización.

1.4 Razones para automatizar

El surgimiento de Internet como entorno global de comunicación, la reducción de costos en equipos de cómputo y las oportunidades de desarrollo que implica un sistema automatizado, han favorecido para que cada vez más bibliotecas opten por poner en marcha un sistema de automatización como herramienta base de gestión; éste se ha enfocado tradicionalmente en los procesos técnicos bibliotecarios y al tratamiento de datos: catalogación, adquisiciones, circulación y publicaciones periódicas, Jacquenson cita las razones que orientan a la necesidad de una informatización a gran escala:

- ✓ Optimización de costos.
- ✓ Mejora de las herramientas de gestión, al disponer de las herramientas actualizadas, sobre el uso de recursos como soporte a la decisión.
- ✓ Reorganización de la biblioteca poniendo de manifiesto conflictos internos a resolver.
- ✓ Cooperación.
- ✓ La informatización en sí misma como elemento común a todas las esferas de nuestra sociedad.
- ✓ Normalización de la información.¹⁶

Dichas razones son suficientes y justifican la necesidad de contar con un sistema de automatización que provea a la biblioteca con servicios efectivos, mientras reduce costos y tiempo en las actividades llevadas a cabo. Por otra

¹⁶ Cfr. JACQUESSON, A. *L'informatisation des bibliothèques: historique, stratégie et perspectives*. Paris: Cercle de la Librairie, 1995.

parte, Chacón Alvarado¹⁷ menciona las áreas básicas que se deben automatizar, sus características y las razones primordiales de éstas.

a) Circulación:

Cuando una biblioteca decide automatizar, generalmente inicia por la sección de circulación; la mayoría de personas que asisten a una biblioteca lo hacen con el propósito de solicitar material bibliográfico en préstamo, sea en sala o a domicilio, consecuentemente es la sección que más tiene contacto con el público.

Sin embargo, en entrevista con Oscar Arriola Navarrete, él hace referencia al proceso de adquisiciones como el punto de partida en todo proyecto de automatización de bibliotecas, ya que éste es el paso inicial de todo el proceso en las bibliotecas.¹⁸

Cuando se decide automatizar la sección de circulación, se debe tener un archivo maestro y un archivo de registro de usuarios en forma automatizada. El archivo maestro de circulación contiene un registro por cada documento que se preste, con la siguiente información:

- Número de clasificación.
- Autor del documento.
- Título.
- Fecha de publicación o cualquier otra información bibliográfica adicional.
- Número de identificación del usuario.
- Nombre del usuario.
- Fecha de préstamo.
- Fecha de devolución.

¹⁷ CHACÓN ALVARADO, Lucía. *Automatización de la biblioteca*. Costa Rica: EUNED, 2002. p.72-77.

¹⁸ Entrevista con el Mtro. Oscar Arriola Navarrete, profesor de Tiempo Completo de la Escuela Nacional de Biblioteconomía y Archivonomía, titular del Seminario de Automatización de Bibliotecas. 6 de diciembre de 2009.

El archivo de registro de usuarios incluye un registro por cada usuario con la siguiente información:

- Número de identificación del usuario.
- Nombre del usuario.
- Dirección.
- Teléfono.
- Fecha en que está autorizado para solicitar material bibliográfico en préstamo.

Los vendedores de software ofrecen diversos paquetes que se utilizan para automatizar la circulación, de los cuales destacan dos, conocidos como software preescritos y sistemas de llave en mano.

Software Preescrito: este tipo de software posee programas que ejecutan todas las tareas de circulación en forma automática, como préstamos, devolución, generación de informes, reclamos, estadísticas de libros más solicitados. También poseen terminales que permiten adaptar lectores de códigos de barras.

Software de Llave en Mano: Son conocidos como “Turnkey vendors” y una de las ventajas que poseen es que los vendedores suministran el software y el hardware específico para ese sistema. El hardware normalmente está compuesto de computadoras personales con suficiente capacidad de almacenamiento para realizar las funciones de circulación, así como de dispositivos de entrada para reconocimiento óptico, como el lápiz óptico que interpreta la información contenida en los códigos de barras; para la impresión de reclamos e informes estadísticos, se utilizan una o más impresoras, dependiendo de las necesidades de la biblioteca.

b) Catalogación automatizada

La ventaja que presenta la catalogación automatizada es que permite generar bibliografías, catálogos en formato libro por autor, título y materia. Es posible también la implementación de catálogos en línea, que es un conjunto de registros bibliográficos automatizados que permiten realizar búsquedas de material específico a través de una terminal. La información bibliográfica que se recupera es mucho más amplia y efectiva que en los catálogos formato libro o de fichas, ya que la extracción de la información es más amplia y permite combinar búsquedas.

La catalogación automatizada representa una de las áreas más complejas de automatizar ya que necesita que el hardware tenga una amplia capacidad de almacenamiento y la información bibliográfica de catalogación requiere de muchos campos. Para un sistema automatizado de catalogación necesita de amplios conocimientos sobre descripción bibliográfica, por lo que se requiere de un bibliotecólogo para que ayude a diseñar un sistema específico.

Para Tedd, un sistema automatizado de catalogación debe tener las siguientes características:

- a) Acceso a la base de datos para recuperar aquellos registros bibliográficos que se necesiten.
- b) Alta calidad y consistencia de los registros de la base de datos, siguiendo las últimas normas de catalogación y clasificación.
- c) Control de autoridades en línea.
- d) Posibilidad de realizar catalogación propia en línea cuando sea necesario y de obtener mensajes de ayuda apropiados.
- e) Capacidad de acceso a los registros del catálogo a partir de diversos puntos y en el formato de salida adecuado.¹⁹

¹⁹ TEDD, Lucy A. *An introduction to computer-based library systems*. London: John Wiley, 1985. p.262.

c) Referencia automatizada

La sección de referencia es la que tiene más contacto directo con el público, porque es ahí donde acuden los usuarios a obtener cualquier tipo de información que necesitan. La información que requieren incluye desde preguntas sencillas sobre ubicación de secciones en la biblioteca, hasta preguntas que demandan investigación especializada por parte del bibliotecólogo. Para esto se pueden auxiliar de bases de datos bibliográficas, las cuales contienen citas bibliográficas de artículos de revistas o documentos técnicos y el resumen correspondiente. También están las bases de datos, como enciclopedias, diccionarios, directorios, atlas y periódicos, que son las más utilizadas por el bibliotecario ya que presentan la información en forma completa. A cualquiera de éstas se puede acceder de manera local o en línea, si se ha establecido un contrato con los vendedores de las mismas.

Hoy en día el concepto ha evolucionado a nuevos niveles, llamándose también “**referencia virtual**”, el cual es definido por OCLC y la Library of Congress como: “el uso de computadoras y tecnologías de comunicaciones”, para proporcionar servicios de referencia a usuarios en cualquier momento y en cualquier lugar.

Fernanda Rodríguez opina al respecto, que “... las tecnologías de información han sido claves para el desarrollo de nuevos servicios en las bibliotecas. A la eclosión de documentos en nuevos soportes físicos se le han ido sumando variados canales virtuales de interacción entre especialistas en información y usuarios de ella. Las opciones se vuelven cada vez más complejas y llegan a los servicios de referencia, consultas a través del teléfono, correo electrónico, chat, formularios en línea o de los llamados entornos colaborativos. Estos cambios han obligado a la implementación de nuevos modos de proporcionar información a los usuarios en el momento que la demandan”.²⁰

²⁰ RODRÍGUEZ BRIZ, Fernanda. Los servicios de la referencia virtual : surgimiento, desarrollo y perspectivas a futuro. Buenos Aires: Alfagrama, 2005. Contratapa.

En este sentido, Parker plantea: "En Internet, los usuarios demandan, cada vez más, el contacto directo en línea con las redes de fuentes de información y con otros usuarios, en un contexto dinámico que supere las restricciones relativas a espacio geográfico, tiempo, tamaño y extensión, que han caracterizado el acceso a los productos y servicios de información operados en los límites físicos de las bibliotecas y centros de documentación tradicionales. Los productores e intermediarios de información en el mundo realizan una verdadera carrera para ofrecer opciones más eficientes y atractivas para la diseminación, interacción y navegación en las más diferentes fuentes con vistas a responder a las demandas de información de las más diferentes comunidades de usuarios."²¹

Los tipos de referencia digital son:

- Referencia digital asíncrona; por ejemplo, consultas por correo electrónico y por medio de formularios en el Web, por el cual los usuarios envían una consulta y el bibliotecario la responde después.
- Referencia digital síncrona; por ejemplo, sesiones de chat y voz sobre IP, por la cual el usuario y el bibliotecario se comunican en tiempo real.²²

Para concluir este punto, se puede decir que el servicio de referencia es el responsable de posibilitar el acceso a la información de quien la necesite, en el momento y en la forma precisa, atender las solicitudes de los usuarios, a partir no sólo de los documentos que posee la biblioteca, sino de todos los recursos disponibles también en forma electrónica y/o digital.

d) Adquisiciones automatizadas

La función de un departamento de adquisiciones de una biblioteca es adquirir el material bibliográfico que necesitan los usuarios; sean estos libros, publicaciones periódicas, materiales audiovisuales, fuentes de referencia,

²¹ PARKER, AL. "La Biblioteca Virtual en Salud para América Latina y el Caribe". En: *Liaison*, v.10, no.3, 1999, p. 2-5.

²² *Referencia virtual*. [en línea]. [Consulta: 06 de Agosto de 2009]. Disponible en Internet: <http://www.oclc.org/americalatina/es/questionpoint/about/virtual/default.htm>.

informes técnicos, etc., generalmente en esta sección se llevan dos archivos distintos: uno bibliográfico, que controla todo el proceso de adquisiciones y otro contable, que controla el presupuesto asignado.

Cuando llega una solicitud a la biblioteca para adquirir un material específico, se debe verificar si el mismo ya ha sido adquirido. Esta verificación se puede hacer en forma automatizada consultando los archivos de circulación y catalogación si ya se han automatizado. Los datos bibliográficos del material solicitado se deben introducir al sistema y almacenarlos en un archivo maestro de adquisiciones. Este archivo va a contener un registro por cada material solicitado y generalmente incluye los siguientes campos:

- Número de pedido.
- Fecha de pedido.
- Nombre del proveedor (indicando si es pedido nuevo, pendiente, etc.).
- Precio del material, incluyendo descuentos, impuestos, etc.
- Código contable de la cuenta que corresponde a este rubro.
- Datos bibliográficos del material solicitado.
 - Autor.
 - Título.
 - Pie de imprenta.
 - Número de inscripción.

Además de este archivo se poseen 3 adicionales: el archivo de proveedores, el archivo de solicitudes y el archivo contable que se mencionó anteriormente.

Archivo de proveedores:

El objetivo de éste es tener un control de las casas editoriales con las cuales se mantienen las transacciones de pedido de material bibliográfico. Este archivo tiene los siguientes campos:

- Nombre del proveedor.
- Código del proveedor, si es necesario.
- Dirección.

- Teléfono.
- Número de fax.
- Fecha de inicio de actividades con el proveedor.
- Fecha de la última transacción que se realizó.
- Descuentos que aplican.

Archivo de solicitudes:

Tiene como objetivo llevar un control de las personas u organizaciones que solicitan adquirir material bibliográfico, para comunicarles oportunamente cuándo ingresó el material solicitado.

Los campos que posee este archivo son:

- Nombre de la persona u organización que solicita adquirir material bibliográfico.
- Dirección.
- Teléfono.
- Lugar de trabajo, en el caso de las personas.
- Cualquier otro dato pertinente a las personas u organizaciones.

Archivo contable:

Tiene como objetivo llevar un control del presupuesto designado a adquisiciones, y debe incluir cuentas independientes para cada tipo de adquisición.

El archivo debe incluir los siguientes campos:

- Código del presupuesto.
- Cantidad presupuestaria asignada.
- Gastos totales.
- Saldos.
- Cualquier otro dato adicional importante.

Para automatizar las adquisiciones de material bibliográfico, las bibliotecas tienen varias opciones:

- ☞ Diseñar un software específico para adquisiciones, sin embargo, esta es la última alternativa que debe considerar una biblioteca, porque resulta demasiado costosa y no hay garantía de que el sistema tenga éxito ya que no ha sido probado con anterioridad.
- ☞ Se puede adquirir un sistema llave en mano que combine el hardware y el software para automatizar adquisiciones.
- ☞ Utilizar el software de adquisiciones que ofrecen los distribuidores de libros como Brodart o Blackwell/North America.

En cada caso hay ventajas y desventajas que se deben de considerar al automatizar esta sección.

e) Control de publicaciones periódicas

La automatización para controlar las publicaciones periódicas representa una de las áreas más difíciles de automatizar porque un registro para una publicación periódica consta de muchos campos que generalmente no se mantienen estables. Así, tenemos que se dan cambios de título, de editor o de frecuencia de publicación; algunas veces es bimestral y en otras trimestral, la secuencia de la numeración suele cambiar, etc.

Sin embargo, algunas bibliotecas han desarrollado sistemas automatizados que generan listados de publicaciones periódicas en orden alfabético y que pueden incluir los siguientes campos:

- Número de clasificación de la publicación periódica.
- Título.
- Editor.
- Volúmenes y números de cada publicación que posee la biblioteca.

- Fecha de cada volumen.
- Idioma de la publicación.
- Periodicidad.

También es posible que la biblioteca diseñe un sistema automatizado que, además de los listados implemente un archivo maestro de publicaciones periódicas para controlar sus adquisiciones, el recibo de estas y los reclamos.

Los campos que incluyen este fichero son:

- Número de la publicación periódica.
- Título.
- Editor.
- Pie de imprenta.
- Volúmenes y números que posee la biblioteca.
- Periodicidad.
- Fecha de inicio.
- Idioma.
- ISSN (Internacional Standard Serial Number), que es el número normalizado que identifica cada publicación periódica.

La automatización permite mejorar la eficacia de los procesos descritos y la gestión interna, prestar nuevos servicios y facilitar la cooperación con otras bibliotecas. Durante los últimos años se ha incrementado la optativa por la gestión automatizada de las actividades bibliotecarias, favorecida por las indiscutibles aportaciones de las tecnologías de la información y la comunicación a la calidad de los servicios prestados.

1.5 Ventajas de la automatización de bibliotecas

El automatizar una biblioteca siempre será un *Plus*; dotar a la biblioteca con sistemas que permitan la gestión computarizada de sus procesos generará múltiples beneficios y ventajas tanto para el bibliotecario como para el usuario final. Algunas de las ventajas que aporta la automatización son:

- Brinda a los usuarios continuo acceso a los materiales disponibles en la biblioteca.
- Elimina tareas rutinarias y/o las desarrolla más eficientemente.
- Reduce la cantidad de tiempo utilizado para la gestión de nuevas adquisiciones, procesos administrativos, técnicos y mantenimiento de registros bibliográficos.
- Permite estrategias de búsqueda que superan por mucho las utilizadas en un catálogo manual.
- Permite las consultas remotas (consultas en línea del catálogo).
- Motiva a los usuarios y los equipa con herramientas que les ayudarán a resolver sus problemas informativos.²³

De igual manera, Kimber²⁴ analiza cuatro grandes ventajas obtenidas con la automatización:

1. **Prestigio:** Es evidente que la biblioteca que instala un sistema automatizado gana prestigio. No sólo se lo brinda a la institución a la que pertenece, si no también obtiene reconocimiento de otras bibliotecas y de los miembros de la profesión bibliotecaria en general. Para Kimber se trata de un prestigio que no daña en lo absoluto, aunque los logros que se obtengan serán vigilados y evaluados constantemente, ya que por cada entusiasta de la computación hay un grupo de personas que no lo son ni lo serán nunca. El prestigio que se gane no es el único factor determinante en pro de la automatización, pero si es una razón de peso.

2. **Análisis estadístico y dificultad para determinar unidades de medida:** Aunque no podemos fijar un valor a nuestros servicios, sabemos estos son importantes para el desarrollo individual o colectivo de nuestra comunidad. Con la automatización se tendrá la capacidad de mostrar y reportar las operaciones en formas que no habían sido posibles anteriormente, generando, por ejemplo, estadísticas de trabajos

²³ FAISAL, S. y SURENDAN, B. *Report on Automation of Library at Kendriya Vidyalaya Pattom 2008*. [en línea]. [Consulta: 12 de Agosto de 2009]. Disponible en Internet: <http://librarykvpatom.files.wordpress.com/2008/07/library-automation.pdf>. p.14-15.

²⁴ KIMBER, Richard. *Automation in libraries*. USA: Pergamon, c1974. p.19-27.

en progreso mediante el cálculo de medias, porcentajes, coeficientes, índices y parámetros del trabajo en la biblioteca, determinando cifras totales directas como cantidad gastada y en reserva, libros adquiridos, prestados, encuadernados y perdidos, calculando subtotales como dinero gastado vs libros comprados y en préstamo, cantidad gastada por materia, por tipo y por idioma de publicación, libros comprados por materia, por tipo de material, por idioma, etc. precios medios de los libros según las materias, dinero asignado a distintas materias, cantidad de lectores interesados en ellas, tipos de libros sacados de la biblioteca y tipos de libros adquiridos. Los resultados pueden utilizarse para hacer cambios en las políticas y actividades de la biblioteca, efectos que se sufren cuando se realiza algún cambio interno o en el servicio de la biblioteca, etc.

3. Factor económico:

- Los sistemas automatizados de control de circulación tienen menor costo unitario que los sistemas manuales cuando el volumen de préstamos excede los 250 volúmenes diarios.
- La producción automatizada de catálogos cuesta generalmente 50% más que un sistema manual, sin embargo hay que tomar en cuenta que los catálogos por computadora proporcionan mejor servicio al público, y por tanto, no son equiparables.
- Los costos de recursos humanos en general tienden a elevarse, en tanto que los sistemas de cómputo tienden a bajar (independientemente de factores inflacionarios). Estas tendencias favorecen la decisión en pro de la automatización.

4. Servicio a los usuarios: Esta es la razón más importante a favor de la automatización. Si el sistema que se pretende implantar redundara en un beneficio tangible en cuanto al servicio a los usuarios, entonces la influencia de los demás factores será negativa solamente en cuanto que se podrá retrasar el proyecto, pero se tratará de realizar lo antes posible.

Las ventajas que ofrece la automatización son realmente interesantes; no existen muchos informes sobre sistemas que hayan fracasado y las posibles razones de su fracaso, sin embargo, pueden existir problemas nada graves y de fácil solución si son previstos con anticipación. Tedd²⁵ divide los problemas potenciales en la implantación de un sistema automatizado de bibliotecas, en los siguientes puntos:

- a) El equipo.
 - Fallo de proveedores en lo que se refiere a costos y tiempos de entrega.
 - Equipos incapaces de proporcionar el servicio prometido.
- b) El software.
 - Fallas del sistema provocadas por errores de instalación, al no ser verificados y documentados después de la misma.
- c) Personas.
 - Insatisfacción de los usuarios, al no cumplir el sistema con las necesidades reales de los mismos.
 - Fracaso en el proyecto de automatización si no existe una buena comunicación entre la biblioteca y el personal informático implicado en la planificación, implantación, y puesta en marcha del mismo.
- d) Recursos financieros.
 - Recursos financieros insuficientes para adquisición de equipos, actualizaciones del sistema y capacitación del personal.

La automatización de bibliotecas es un proceso que requiere de una planeación adecuada, continua implementación y periódica evaluación.

El bibliotecario y demás personas involucradas en el proceso deben establecer las prioridades a seguir tomando en cuenta los requerimientos actuales y futuros de la biblioteca. La selección de un sistema de automatización integrado, que cumpla con las necesidades de usuarios y biblioteca, la capacitación de personal y la educación de usuarios son claves para el éxito del proceso.

²⁵ TEDD, Lucy A. *Op. cit.* p.12-13.

1.6 Planeación de un proyecto de automatización

La base para el éxito de cualquier proyecto implica que exista una buena planeación. La automatización de bibliotecas no es una excepción, existen factores determinantes a considerar para llevarla a cabo exitosamente, los que en términos globales deben estar enfocados a etapas previas, presentes y posteriores a la automatización.

Es fundamental que exista una congruencia entre la funcionalidad del sistema de automatización que se pretende instalar y las características propias de la biblioteca, esto con el objetivo de lograr un óptimo aprovechamiento del mismo y un éxito continuo en el desarrollo del proyecto. Antes de mencionar las etapas de la planeación conviene mencionar algunos aspectos estrictamente administrativos que Hayes y Becker²⁶ proponen para esta etapa:

- a) Determinar objetivos y definir prioridades. El o los analistas que hayan elaborado el estudio pueden presentar un panorama de las operaciones presentes y las posibilidades futuras, pero la decisión final será del administrador de la biblioteca.
- b) Preparar planes para lograr los objetivos, incluyendo la elaboración del presupuesto, así como fijar fechas límite. Los costos de la automatización son tan altos, que deben planearse con detenimiento las asignaciones presupuestales para cada fase del desarrollo e implementación.
- c) Autorizar y controlar el trabajo requerido. Nuevamente, aunque los analistas determinen fechas para cada fase del desarrollo, el administrador del proyecto decidirá el momento oportuno para realizar cada tarea.

²⁶HAYES, Robert y BECKER, Joseph. *Handbook of Data Processing for Libraries*. USA: Becker and Hayes, 1970. p.64.

- d) Vigilar y evaluar el progreso hacia los objetivos. Dado que el carácter del proyecto es altamente técnico, el administrador debe entender los aspectos técnicos del sistema para poder administrarlo adecuadamente.
- e) Identificar las acciones correctivas al surgir los problemas. El administrador del sistema requiere que se negocien o concilien los requerimientos y capacidades que entren en conflicto.
- f) Informar al personal sobre los efectos que el programa puede tener sobre ellos. La automatización ocasionará cambios importantes entre las relaciones “*biblioteca-personal-usuarios*”. Esto implica el diseño de un programa de capacitación para el personal y otro de relaciones públicas con los usuarios, con el objetivo de lograr que exista un entendimiento adecuado de los efectos de la automatización y su aceptación final.

A su vez Michael Von Cotta sugiere que deben cumplirse varias condiciones para lograr resultados óptimos y anota lo siguiente:

1. Una profunda comprensión de la computadora y de la tecnología de la información.
2. Una visión clara y actualizada de los avances en materia de tratamiento de los recursos de la información.
3. Reestructuración de la biblioteca para disponer de un ambiente adecuado, en el que se explote todo el potencial de la tecnología de la información.²⁷

Etapas de la planeación

1. Evaluación de necesidades

La planeación exige que se haga una evaluación previa de las necesidades existentes en la biblioteca, antes de intentar cualquier procedimiento de automatización. Es necesario hacer un inventario de los servicios que se

²⁷ VON COTTA-SCHONBERG, Michael. “Automation and academic library structure”. *En: Libri*, 1989, 39 (1) p.47-63.

ofrecen actualmente, analizar su efectividad y determinar en qué forma se pueden mejorar.

Punto importante a considerar dentro de la planeación es el referente a realizar un estudio de comunidad, con el objetivo de conocer los niveles de “alfabetización tecnológica” de los usuarios y del personal de la biblioteca. En entrevista con Oscar Arriola Navarrete, menciona que la alfabetización tecnológica consiste en desarrollar los conocimientos y habilidades tanto instrumentales como cognitivas en relación con la información vehiculada a través de las tecnologías de la información y la comunicación (manejar el software, buscar información, enviar y recibir correos electrónicos, utilizar los distintos servicios de www, etc.), además de plantear y desarrollar valores y actitudes de naturaleza social y política con relación a las TIC. Así mismo menciona que existen tres niveles de alfabetización:

1. El nivel básico se define como el conjunto de habilidades necesarias para manejar programas de uso general en su forma más simple, como procesadores de palabras, juegos, chat y acceso a páginas Web con dirección previamente conocida.
2. El nivel medio es el conjunto de habilidades y conocimientos necesarios para manejar programas de uso general en su forma más compleja así como programas más sofisticados en forma elemental, como las planillas de cálculo, procesadores de imágenes (photoshop, etc.).
3. El tercer nivel es el conjunto de habilidades y conocimientos necesarios para usar programas sofisticados en sus formas más avanzadas y software de gestión para uso en actividades productivas.²⁸

²⁸ Entrevista con el Mtro. Oscar Arriola Navarrete, profesor de Tiempo Completo de la Escuela Nacional de Biblioteconomía y Archivonomía, titular del Seminario de Automatización de Bibliotecas. 4 de Agosto de 2009.

2. Análisis de factibilidad

Cualquier proceso administrativo serio requiere que se haga un análisis de sistemas, donde se estudian los siguientes factores:

- Procedimientos que se realizan en la organización sujeto de análisis.
- Organigrama de la institución.
- Cantidades producidas o unidades procesadas por departamento o sección en un periodo definido.
- Tiempos (tiempo que tarda un proceso, por ejemplo, trámite de un préstamo, frecuencia de adquisiciones, frecuencia con que se debe actualizar un listado para manejo público, etc.).
- Costos. Conceptos: trabajo, equipo y materiales (incluyendo también el material bibliográfico).

3. El software ¿desarrollo o aprovechamiento de lo existente?

Existen en el mercado diversos paquetes de programas destinados específicamente a la automatización de actividades bibliotecarias; la experiencia invertida en estos paquetes no ha de despreciarse; muchas bibliotecas los han adoptado ya y por esto ellos se enriquecen tan sólo con su aplicación, así como con la solución que hayan proporcionado a los diversos problemas existentes en cada una de ellas.

Es necesario evaluar dichos paquetes con el fin de determinar si es conveniente conseguirlos y aplicarlos, o si es necesario desarrollar internamente los programas de automatización. Si se opta por un paquete ya desarrollado, se evitará mucho trabajo de planeación, análisis, programación y pruebas, así como problemas inherentes a la implantación del sistema automatizado. Los paquetes ya desarrollados para automatización en bibliotecas ofrecen también, en algunos casos, las siguientes ventajas/servicios adicionales:

- Aplicación y adaptación del paquete a las necesidades específicas de la biblioteca.
- Normalización e intercambiabilidad de los datos con otras bibliotecas.

- Entrenamiento al personal que operara el sistema en la biblioteca.
- Acceso a las modificaciones y mejoras que resulten de cambios o del desarrollo del paquete en sí (por ejemplo, publicación de nuevas versiones de los programas).

Las ventajas que ofrecen los paquetes ya desarrollados son numerosas, sin embargo, es necesario evaluarlos y considerar su aplicabilidad a la solución de los problemas específicos de la biblioteca donde se instalarán. Los aspectos a evaluar son los siguientes:

- Aplicabilidad del sistema ya desarrollado. En algunos casos los paquetes pueden adaptarse según las necesidades, y en otros se ofrecen “tal como son”, sin garantía alguna.
- Recursos financieros disponibles contra dimensiones del problema existente o del proyecto de automatización.
- Costo del desarrollo de programas a nivel interno contra costo de la operación y puesta en operación del paquete ya desarrollado.

En términos generales, debería optarse por un paquete desarrollado si éste resuelve/satisface por lo menos entre un 75 y 80 por ciento de los problemas o necesidades actuales (o previstas) de la biblioteca.

4. Estimación de recursos

Con base en lo anterior se determinarán los recursos necesarios para realizar el proyecto de automatización.

- Recursos tecnológicos y humanos: se contemplan los recursos actuales y los que se necesitarán para completar el proceso.
- Recursos financieros: se estimará el costo del personal de los recursos tecnológicos necesarios para el proyecto de automatización.²⁹

²⁹ CHACÓN ALVARADO, Lucía. *Op. cit.* p.68.

5. Instalación e implantación del sistema

En esta etapa conviene establecer que la implantación del proyecto se realizará en cuatro series paralelas de eventos:

- Selección del equipo, adquisición, instalación y prueba del sistema de cómputo.
- Especificaciones del sistema, especificaciones de archivos, estructuras y diseño detallado del sistema, programación y pronta corrección.
- Edición y conversión de archivos.
- Reclutamiento y entrenamiento del personal.

En esta etapa los requerimientos del sistema se deben convertir detalladamente en las funciones que realiza un sistema de computación de entrada, procesamiento y salida. Es básico presentar una descripción detallada de cual información es la que se va a usar, cómo y cuándo debe ser introducida a la computadora, qué va hacer la computadora con esa información, cómo se va procesar y que resultados se obtendrán. Estos requerimientos deben ser aprobados por el personal de la biblioteca y es en ese momento cuando la biblioteca decide el sistema a usar.

En esta fase es donde se realiza el proceso de selección del hardware y software que respondan a los requerimientos de automatización de la biblioteca. Con respecto al programa que seleccionara, es necesario considerar varias alternativas y entre ellas, se pueden valorar los programas que utilizan otras bibliotecas del país para identificar el que se adapte adecuadamente a las necesidades específicas de la biblioteca. La ventaja es detectar los “pros” y “contras” de un programa que ya ha sido probado y utilizado.³⁰

³⁰ *Ibíd.* p.69.

Estudio de usabilidad

Es importante llevar a cabo también estudios de usabilidad, Jakob Nielsen, considerado el padre de la usabilidad, la definió como el atributo de calidad que mide la facilidad de uso en cualquier software. Es decir, un sistema usable es aquél en el que los usuarios pueden interactuar de la forma más fácil, cómoda, segura e inteligentemente posible.

Un estudio de usabilidad revelará si el sistema en uso cumple con las siguientes características:

- Entendible.
- Novedoso.
- Comprensible.
- Inteligente.
- Atractivo.

En un estudio de usabilidad, se definen cinco componentes de calidad:

Aprendizaje: qué tan fácil es para los usuarios llevar a cabo tareas básicas en su primer contacto con el software.

Eficiencia: una vez que los usuarios han aprendido el uso del sistema, qué tan rápido pueden llevar a cabo tareas específicas.

Memorabilidad: Después de cierto tiempo de no usar el sistema, que tan fácil resulta para los usuarios restablecer su pro eficiencia, una vez que tienen contacto de nuevo con el sistema.

Errores: cuántos errores son cometidos por los usuarios, qué tan severos son estos errores, y qué tan fácil se pueden recobrar de los mismos.

Satisfacción: qué tan satisfecho está el usuario con el uso del sistema.

Algunos puntos de consideración:

- En un estudio de usabilidad el usuario conversa con el entrevistador, entiende qué tiene que hacer, recibe una serie de tareas que debe ejecutar en el sistema, comenta sus impresiones, responde a preguntas verbales y rellena un cuestionario.
- La experiencia de usuario no es más que el conjunto de percepciones, sentimientos y comportamiento de un individuo frente a un sistema determinado. Es decir, todo lo que sucede y le sucede cuando está frente al sistema en cuestión (en este caso el Sistema Integral para Bibliotecas). Esta experiencia es clave para atraer y fidelizar a su público.
- Un estudio de usabilidad puede ser extremadamente útil cuando se quiere mejorar un sistema ya existente. Es decir, hacer un estudio para saber en qué se está fallando y sobre la base de los resultados, planificar un proceso de rediseño.³¹

6. Mantenimiento

Al formular el proyecto de automatización debe asignarse parte del presupuesto a la previsión de fallas (mantenimiento preventivo) y a la corrección de errores en el sistema (mantenimiento correctivo). El mantenimiento es parte importante, y la solución a cualquier falla está en obtener accesibles contratos de mantenimiento de hardware de alguna compañía de cómputo, así como la colaboración y ayuda del analista de sistemas o programador que participó en la elaboración del proyecto, siendo ésta la persona más adecuada para dar el mantenimiento al software debido al conocimiento que éste tiene sobre las estructuras y registros del sistema automatizado, así como las necesidades y requerimientos específicos inherentes al mismo.³²

³¹ *Usability 101: Introduction to Usability*. [en línea]. [Consulta: 14 de Diciembre de 2009]. Disponible en Internet: <http://www.useit.com/alertbox/20030825.html>.

³² SILVA ZAMORA, Oscar. *La automatización de Bibliotecas en México*. Tesis. México: El autor, 1989. p.52-68.

Un proyecto de automatización debe estar bien fundamentado en cada una de las etapas mencionadas, desde la evaluación de necesidades hasta la puesta en marcha del sistema y su constante mantenimiento. Esto llevara a obtener un proyecto exitoso, con resultados satisfactorios que culminen en el cumplimiento de los objetivos preestablecidos.

1.7 Requerimientos

El desarrollo continuo en equipos de cómputo, ha posibilitado la instalación y adecuación de diversos sistemas de automatización con requisitos mínimos y especificaciones detalladas, en equipos que se vuelven cada día más estandarizados. Para lograr un óptimo desempeño en cualquier sistema de automatización es necesario cumplir con ciertos requerimientos, los que Chaín Navarro define como “recursos tecnológicos”: herramientas e instrumentos informáticos con los que se cuenta para gestionar la información.³³ En el caso de automatización de bibliotecas, será primordial contar con los siguientes elementos:

- Hardware.
- Software.
- Dispositivos periféricos.
- Redes de telecomunicación.

Hardware

El hardware es el conjunto de dispositivos y componentes electrónicos de los que consta el ordenador, es decir, es la parte “física” o “mecánica”. Proporciona un marco para el desarrollo de soluciones a problemas concretos.³⁴ El hardware constituye uno de los elementos más importantes, ya que será la infraestructura sobre la que se apoye cualquier sistema informático. El término hace referencia a los elementos físicos que constituyen al ordenador como son: la unidad central de procesamiento y los dispositivos periféricos, así como otros

³³ CHAIN NAVARRO, Celia. *Gestión de Información en las Organizaciones*. España: ICE-Universidad de Murcia, 2000. p.113.

³⁴ IDEAS PROPIAS PUBLICIDAD. *Introducción a la informática y al uso y manejo de aplicaciones comerciales*. España: GESBIBLO, 2006. p.16.

medios físicos gracias a los cuales se vinculan esos dispositivos (tarjetas, buses, cables, etc.).

Software

En sentido estricto se entiende por software al conjunto de rutinas, programas de utilidad general, procedimientos, técnicas, y documentación, que conllevan al funcionamiento de un equipo y a su mejor y más eficiente explotación.³⁵ El término hace referencia a toda la parte inmaterial incorporada al equipo, que permite su funcionamiento ajustado a las necesidades del usuario.

La finalidad del software se centra en facilitar la mejor explotación del equipo, haciendo agradable su uso, por tanto, con este recurso se tiende a conseguir, en mayor o menor medida lo siguiente:

Reducir al máximo la acción manual del usuario, para lo cual la secuencia de ejecución de los diferentes trabajos y su control serán ejecutados por el software.

Proporcionar los medios de comunicación entre el hombre y la máquina. Esto se realizará mediante los diferentes sistemas de programación, que cada vez transfieren a la máquina más trabajos de los que tenía que desarrollar el hombre.

Suministrar rutinas que faciliten la creación, explotación, mayor aprovechamiento de medios y mejoras en tiempos de proceso de archivos.

Suministrar rutinas que realicen automáticamente, y que eviten al usuario el desarrollo de las mismas o los trabajos repetitivos, como pueden ser rutinas matemáticas, programas de servicios, etc.

³⁵ PARDO CLEMENTE, Ezequiel. *Microinformática de Gestión*. España: Servicio de Publicaciones-Universidad de Oviedo, 1993. p.117.

El software se divide en dos clases generales:

Software básico (sistema operativo). Es el conjunto de programas que actúan como intermediario entre el PC y el usuario, para que este último pueda aprovechar todos los recursos físicos del primero de una forma fácil, sin tener que poseer grandes conocimientos sobre el hardware.³⁶

Algunas de las operaciones básicas que realiza un sistema operativo (SO) son:

El control del hardware y su configuración, la gestión de la ejecución de los otros programas, el control de errores y la gestión de los archivos. Actualmente existen 3 familias de SO: Windows, Macintosh y Unix, de entre los cuales tenemos por ejemplo:

Windows

- Windows 95
 - Windows 98
 - Windows ME
 - Windows NT
 - Windows 2000
 - Windows 2000 server
 - Windows XP
 - Windows Server 2003
 - Windows CE
 - Windows Mobile
 - Windows XP 64 bits
 - Windows Vista
 - Windows 7
-

Macintosh

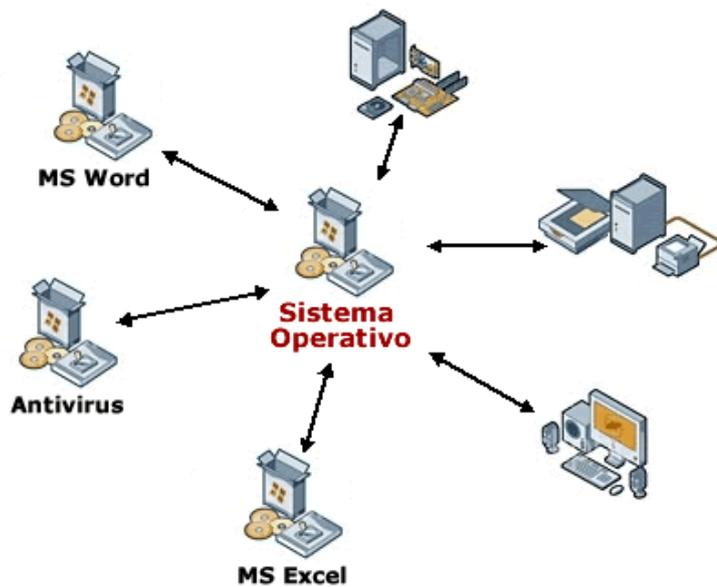
- Mac OS 7
- Mac OS 8
- Mac OS 9
- Mac OS X

³⁶ RODRÍGUEZ DURÁN, Luis. *Sistemas Operativos*. España: Boixareu Editores, 2000. p.11.

Unix

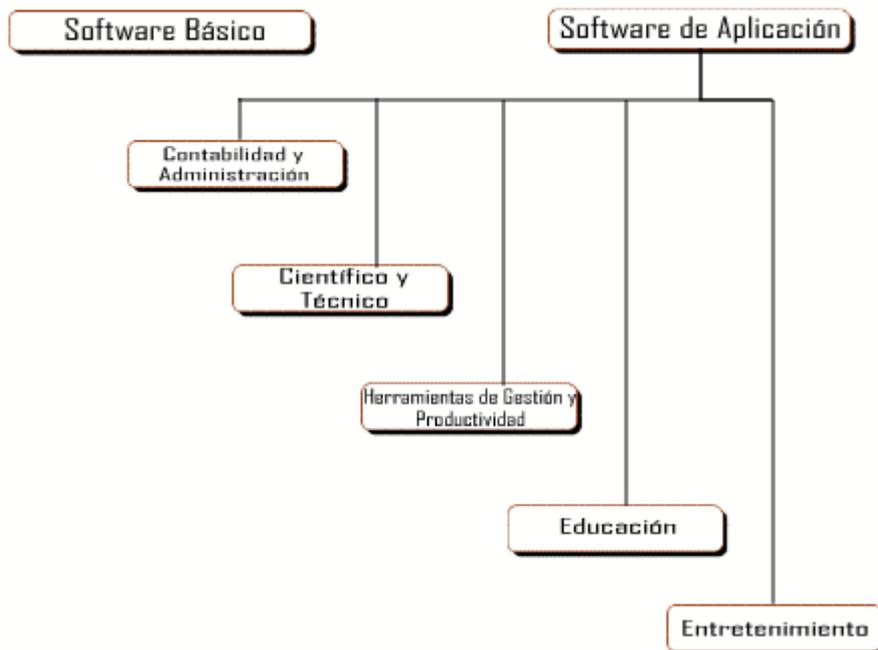
-
- AIX
 - AMIX
 - GNU/Linux (Debian, Ubuntu, etc.)
 - GNU / Hurd
 - HP-UX
 - Irix
 - Minix
 - System V
 - Solaris
 - UnixWare
-

De manera gráfica, un sistema operativo funciona de la siguiente manera:



Software de aplicaciones. Es el conjunto de programas desarrollados por el usuario o por terceras personas para resolver sus problemas concretos.³⁷ Éste se divide en cinco grandes grupos, dentro de los cuales encontramos los sistemas de automatización de bibliotecas como herramientas de gestión y productividad.

³⁷ PARDO CLEMENTE, Ezequiel. *Op. cit.* p.118.



Dispositivos periféricos

La función principal de los dispositivos periféricos es la de comunicar el procesador con el mundo exterior; concretamente conecta al procesador con el usuario y/o con otras computadoras con el objetivo de permitir la entrada, la salida y el almacenamiento auxiliar de información.³⁸

Gracias a los periféricos la computadora puede acceder a información de muy diversos tipos: impresa, en forma de voz, en forma de imágenes, introducida manualmente, etc., igualmente gracias a éstos se pueden transferir los resultados de los procesos realizados por la computadora a diversos formatos.

³⁸ DE PABLOS HEREDERO, Carmen. *Informática y Comunicaciones en la Empresa*. España: Esic, 2004. p.74.

Redes de telecomunicación

La necesidad de contar con redes de comunicación a distintos niveles se vuelve fundamental para la automatización de bibliotecas, por una parte, como elemento básico en el que los sistemas han adoptado estándares y protocolos para la transferencia e intercambio de información y por otras, por las numerosas ventajas que ofrecen las llamadas redes de telecomunicación, ya sean locales, de área extensa, así como el llamado sistema de información global de redes interconectadas, "Internet".

Por definición, una red es un grupo de computadoras interconectadas, cuya función fundamental es permitir comunicarlas entre sí para intercambiar datos e información.³⁹ Esta definición es muy similar a la ofrecida por la bibliotecaria Marlene Clayton quien define a una red como el "conjunto de computadoras interconectadas e independientes, con el objetivo de distribuir servicios de información a los usuarios y/o proporcionar comunicación efectiva entre los mismos".⁴⁰ la función de la red en un sistema de automatización de bibliotecas será sin duda habilitar canales de comunicación entre los equipos interconectados y facilitar el uso de distintos recursos compartidos en un esquema de cooperación (Internet, archivos, dispositivos periféricos etc.).

Tipos de redes

En la actualidad, el diseño y el alcance que puede tener una red es muy diverso; de manera general las redes se clasifican en tres grupos básicos: Redes de área local (LAN), redes de área extensa (WAN) y redes de área metropolitana (MAN).

³⁹ BIRD, Drew. *Network+ Training guide*. USA: QUE Certification, 2002. p.18.

⁴⁰ CLAYTON, Marlene. *Gestión automatizada de bibliotecas*. Madrid: Fundación Germán Sánchez Ruipérez, 1998. p.166.

Una **Red de Área Local** o **LAN** (del inglés **Local Area Network**) es la interconexión de varios ordenadores y periféricos. Su extensión está limitada físicamente a un edificio o a un entorno de 200 metros, aunque con repetidores se puede alcanzar hasta un kilómetro. Su aplicación más extendida es la interconexión de computadoras personales y estaciones de trabajo en oficinas, colegios, instituciones, fábricas, etc., para compartir recursos e intercambiar datos y aplicaciones. En definitiva, permite que dos o más máquinas se comuniquen.

El término red local incluye tanto el hardware como el software necesario para la interconexión de los distintos dispositivos y el tratamiento de la información. Las redes locales pueden ser muy rápidas pero están limitadas en cuanto a cobertura-distancia.

Una **Red de Área Amplia** o **WAN** (del inglés **Wide Area Network**), es un tipo de red de computadoras capaz de cubrir distancias que van de los 100 a los 1000 Kms., aproximadamente, dando el servicio a un país o un continente. Un ejemplo de este tipo de redes lo es Internet así como cualquier red en la cual no estén en un mismo edificio todos sus miembros.

Una Red de Área Amplia o WAN se extiende sobre un área geográfica extensa, a veces un país o un continente, y su función fundamental está orientada a la interconexión de redes o equipos terminales que se encuentran ubicados a grandes distancias entre sí. Una red WAN conecta e interlaza distintas LAN's para crear un ambiente Internetwork desde distintos puntos.

Una **Red de Área Metropolitana** (**Metropolitan Area Network** o **MAN**, en inglés) es una red de alta velocidad (banda ancha) que dando cobertura en un área geográfica extensa, proporciona capacidad de integración de múltiples servicios mediante la transmisión de datos, voz y video, sobre medios de transmisión tales como fibra óptica y cable par trenzado. Su distancia de cobertura es mayor de 4 kms. Técnicamente una MAN es una WAN, razón por

la cual el término se usa con menor frecuencia; si se quiere distinguirlas se puede decir que una red MAN es más pequeña que una WAN.⁴¹

En una biblioteca, el contar con una red ofrecerá múltiples beneficios, como lo son:

- ✓ Compartir impresoras, scanner, CDR-W, y otros dispositivos, ahorrando con esto recursos financieros.
- ✓ Compartir una conexión a Internet de alta velocidad en todos los equipos interconectados.
- ✓ Compartir archivos de cualquier tipo entre los equipos disponibles en la red (esta característica debe ser cuidadosamente controlada, contando con mecanismos de seguridad que limiten el acceso a distintos niveles).
- ✓ Mantener un respaldo de datos (backup) en algún equipo central de la red; esto con el fin de facilitar su recuperación en cualquier momento.
- ✓ Acceso remoto, múltiple y simultáneo de usuarios al OPAC disponible en la biblioteca.

Cabe señalar que dentro de una red se cuenta con equipos que realizan funciones a distintos niveles, mejor conocidos como cliente y servidor; como lo menciona Arriola Navarrete⁴² “ambos equipos (cliente y servidor) son computadoras conectadas en red, el cliente está orientado a operaciones de escritorio y el servidor está dedicado a funciones de proveedor de información y servicios”. También se puede usar un cliente para que realice funciones de servidor o mixtas. Es importante mencionar lo anterior ya que muchos sistemas de automatización incluyen en los paquetes de instalación dos archivos ejecutables, uno para ser instalado en la máquina que actuará como cliente y otro para el servidor.

⁴¹ BIRD, Drew. *Op. cit.* p.27-28.

⁴² ARRIOLA NAVARRETE, Oscar. *Creación de un portal: el caso de la biblioteca Daniel Cosío Villegas de El Colegio de México*. Tesis. México: El autor, 2002. p.22.

Basado en un breve análisis de las diferentes especificaciones propuestas por sistemas propietarios, y la flexibilidad y adaptación que pueden tener sistemas operativos libres como Debian y Ubuntu (GNU/LINUX), así como la necesidad de contar con equipos capaces de satisfacer las demandas de informatización, se presentan a continuación una serie de especificaciones a nivel cliente, servidor y de infraestructura, con las que se debe contar para lograr el óptimo funcionamiento de cualquier sistema de automatización de bibliotecas.

Requerimientos Cliente:

- Procesador Pentium IV o superior a 1.5 GHz
- Disco duro 40 GB.
- Memoria RAM 512 MB o superior para mejor desempeño de los procesos.
- Tarjeta de red tipo Ethernet 10/100 Mbps (ahora incluidas en la Motherboard del equipo).
- Unidad lectora CD/DVD ROM.
- No-Break (recomendable).
- Navegador de Internet (Internet Explorer 7.0 o Mozilla Firefox 3.0).
- Sistema operativo (Windows XP SP3 o superior y en el caso de Linux, distribuciones como Debian Etch y Ubuntu consideradas de las mas estables).

Requerimientos Servidor:

- Procesador Pentium IV o superior a 1.5 GHz.
- Disco duro 40 GB para uso e instalación de la aplicación “servidor”.
- Disco duro 80 GB para respaldo de datos.
- Unidad lectora/escritora de CD/DVD.
- Tarjeta de red tipo Ethernet 10/100 Mbps.
- No-Break (básico).
- Sistema operativo para servidores (recomendable, aunque no es necesario; Windows Server 2006 o superior y en el caso de Linux distribuciones como Debian Server o Ubuntu Server Edition).

- Navegador de Internet (Internet Explorer 7.0 o Mozilla Firefox 3.0).
- Manejador de Base de Datos (DBMS) con soporte SQL/ANSI. Se recomienda el uso de SQL Server o Visual Fox Pro en su versión 7.0 o superior.

Requerimientos de infraestructura:

- Soporte de redes.
- Servidor Web.
- Soporte de DBMS.
- Dispositivos periféricos de entrada y salida (lectores de códigos de barra, impresoras).
- Equipos personales.

Partiendo de los factores que se deben tomar en cuenta para automatizar una biblioteca, los sistemas integrales engloban ciertos requerimientos, como los requerimientos generales, que, como lo señala Oscar Arriola Navarrete.⁴³ no son más que “las operaciones que se refieren específicamente a las aplicaciones de ejecución de un programa y por ende de los procedimientos y expectativas de cada sección o departamento a la cual se opere un sistema de automatización”. Es decir, que se refiere a lo que se debe hacer, pero sobre todo al tipo de elementos que cada módulo necesita para ser funcional dentro de un Sistema Integral de Automatización de Bibliotecas -tomando en cuenta la posibilidad de conectarse con otras redes y sistemas de información, como se muestra a continuación:

1. Debe ser un sistema integral de automatización de bibliotecas, relacional, con diferentes salidas.

⁴³ARRIOLA NAVARRETE, Oscar y GARMENDIA BONILLA, Lovania. “Evaluación de software para bibliotecas: requerimientos técnicos” [en línea]. En: *Bibliotecas y archivos: órgano de la Escuela Nacional de Biblioteconomía y Archivonomía*. 1(4): p. 23-31. [Consulta: 04 de Agosto de 2009]. Disponible en Internet: http://eprints.rclis.org/archive/00012967/01/Evaluaci%C3%B3n_software.pdf.

2. Que el sistema permita el control de usuarios, acervos, OPAC, préstamo, catalogación, autoridades, existencias y adquisiciones de las unidades de información que forman la red de bibliotecas públicas del estado.
3. Que el sistema sea robusto, fácil de usar y sumamente flexible, que aproveche a fondo las últimas tecnologías de bases de datos relacionales, aplicaciones Web y aplicaciones cliente-servidor.
4. Operar como un sistema integral que permita definir los privilegios para que el personal tenga acceso sólo a las aplicaciones y funcionalidades que se le autoricen.
5. Operar un sistema integral con subsistemas para: proceso gerencial, reportes y estadísticas, adquisiciones, administración de publicaciones periódicas, catalogación, autoridades, ítems, catálogo público en línea (OPAC), circulación, módulo de disseminación selectiva de información, préstamo interbibliotecario, módulo Z 39.50.
6. Facilitar la parametrización del sistema a través de herramientas gráficas propias, perfectamente documentadas y comprensibles por personal bibliotecario. Y que sea flexible a fin de adaptarlo a las peculiaridades de estructura, funcionamiento, métodos y usos de la institución.
7. Tener toda la documentación del software, manuales, ayudas en línea e impreso en español.
8. Contar con una interfaz de usuario amigable, intuitiva y que contemple un número ilimitado de registros.
9. Ser un sistema escalable, desde uno hasta miles de usuarios concurrentes.

10. Funcionar sobre entornos ajustados a los sistemas abiertos: UNIX, TCP/IP, entre otros, así como el manejador de bases de datos ORACLE, SQL o POSTGRESS.
11. Cumplir con la normalización del formato MARC 21 para la descripción de registros bibliográficos, autoridades y existencias.
12. Diseñar plantillas en formato MARC 21 para la catalogación de documentos, registro de autoridades y existencias, a partir de herramientas gráficas que faciliten su diseño al bibliotecario.
13. Generar e imprimir tarjetas, etiquetas de clasificación, códigos de barras y credenciales.
14. Desarrollar plataforma del sistema, manejador de bases de datos e interfases gráficas.

Otros requerimientos son los requisitos funcionales, que se refieren específicamente a las aplicaciones del software en cada módulo y depende ampliamente de las necesidades de cada biblioteca; es decir, lo que el sistema tiene que hacer. Por ello los requerimientos básicos que debe tener cada módulo son:

1. Seguridad

Contar con tres niveles de seguridad:

- 1.1. Utilizar conexiones seguras mediante el estándar SSLv3 para el acceso del personal a los módulos de operación del sistema que trabajen en arquitectura Web, para lo cual todo el acceso a dichos módulos deberá hacerse utilizando el protocolo HTTPS.
- 1.2. Utilizar en todos los módulos conexiones seguras a la base de datos, usando el estándar SSLv3 o encriptamiento a 128 bits.

1.3. Controlar el acceso de los usuarios del sistema por medio de una cuenta y una clave secreta personalizada.

2. *Proceso Gerencial*

Contar con ambiente gráfico para el proceso gerencial que, permita realizar cada una de las actividades que a continuación se describen:

2.1. Facilitar el alta y configuración de la biblioteca o una red de ellas, incluyendo la especificación para cada biblioteca de su horario de atención por cada día de la semana, días laborales, días con horario especial, días festivos y periodos vacacionales.

2.2. Permitir la administración de usuarios y categorías de usuarios por biblioteca.

2.3. Administrar de manera ágil y sencilla las categorías y privilegios del personal de staff por biblioteca.

2.4. Proporcionar un mecanismo ágil y sencillo para la especificación de las políticas de circulación, relacionando las categorías de usuario, la colección y estatus del material por biblioteca.

2.5. Facilitar la obtención de manera ágil y sencilla de la información de títulos y volúmenes para el control de inventarios por biblioteca.

2.6. Facilitar el monitoreo de licencias staff y OPAC concurrentes en el sistema.

3. *Reportes y estadísticas*

Este módulo deberá proporcionar al menos los siguientes reportes:

3.1. Acervo (títulos, volúmenes) por biblioteca y colección.

3.2. Adquisiciones (títulos, volúmenes) por biblioteca y colección.

- 3.3. Adquisiciones (títulos, volúmenes) por proveedor.
- 3.4. Reporte de presupuesto general y por biblioteca.
- 3.5. Productividad del personal.
- 3.6. Préstamo a domicilio, en sala e interbibliotecario, por periodo.
- 3.7. Préstamos por tipo de usuario.
- 3.8. Préstamos por colección.
- 3.9. Materiales más prestados.
- 3.10. Incidencias de adeudo de material.

4. *Organización bibliográfica (módulos de catalogación, autoridades e ítem)*

- 4.1. Contar con las ayudas contextuales en todas las plantillas de captura de los módulos de ítem, catalogación y autoridades.
- 4.2. Permitir la definición de campos obligatorios.
- 4.3. Contar con mecanismos que eviten la duplicación de registros bibliográficos.
- 4.4. Realizar cambios globales en la base bibliográfica, a partir de la validación de registros en la base de autoridades.
- 4.5. Permitir el desarrollo para definir el privilegio de registros visibles y no visibles.

- 4.6. Proporcionar una opción para indizar el texto en la base de datos de forma tal que se puedan hacer búsquedas eficientes en texto completo de documentos electrónicos almacenados en un servidor local.
- 4.7. Permitir la importación y exportación de registros bibliográficos y de autoridad.
- 4.8. Validar los registros a almacenar con respecto al formato MARC 21 (campos, indicadores y subcampos), impidiendo el almacenamiento del registro y/o alertando sobre errores de codificación.

5. *Adquisiciones*

Este módulo deberá proporcionar las herramientas necesarias para la administración y control del presupuesto, mediante las siguientes funciones:

- 5.1 Administración de proveedores.
- 5.2 Configuración de partidas presupuestales.
- 5.3 Uso de cuentas contables para el presupuesto (global, comprometido y ejercido).
- 5.4 Asignación, transferencia y recorte de presupuesto.
- 5.5 Adquisiciones en diferentes monedas, conversión monetaria y cargo al presupuesto.
- 5.6 Control de pagos y cargo al presupuesto.
- 5.7 Control de facturas.
- 5.8 Administración y control de solicitudes de adquisición.
- 5.9 Administración y control de órdenes de compra.

- 5.10 Administración y control, donaciones y canje.
- 5.11 Recepción de materiales adquiridos.
- 5.12 Control de cancelaciones.
- 5.13 Control de devoluciones.
- 5.14 Administración de suscripciones a publicaciones periódicas.
- 5.15 Generación de calendario de publicaciones con base en la definición de los patrones de regularidad.
- 5.16 Catálogo de listas de circulación.
- 5.17 Control del arribo de ítems.
- 5.18 Distribución de ítems.
- 5.19 Circulación de ítems.
- 5.20 Control de reclamaciones y envío automático de reclamaciones por e-mail.

6. *Publicaciones Periódicas*

Este módulo deberá proporcionar al menos las siguientes funcionalidades:

- 6.1. Registro de fascículos.
- 6.2. Control de llegadas (calendarización).
- 6.3. Reclamación automática de ejemplares.

7. *Circulación*

Contar con un módulo de circulación que proporcione las siguientes funcionalidades:

7.1. Préstamo a domicilio, apartado, renovación y devolución.

7.2. Préstamo interbibliotecario, renovación y devolución.

7.3. Impresión de recibos de préstamo y devolución.

7.4. Aplicación de multas y suspensión con bloqueo automático del préstamo en caso de retrasos en la devolución.

7.5. Uso del código de barras para cada usuario y cada ítem.

7.6. Emisión automática de aviso de multa por correo electrónico para usuarios con retraso.

7.7. Control de préstamo en sala.

7.8. Consulta de materiales en préstamo

8. *Catálogo Público en Línea (OPAC)*

Ofrecer un OPAC que opere en ambiente Web con las siguientes características:

- ∞ Búsqueda por autor, título, tema, editorial y serie.
- ∞ Búsqueda libre en todos los campos.
- ∞ Búsqueda con operadores booleanos (Y, O, NO).
- ∞ Búsqueda por clasificación, código de barras, ISBN y número de registros.
- ∞ Búsqueda en texto completo.
- ∞ Búsqueda a través de un registro de resultado.
- ∞ Búsqueda refinada.
- ∞ Búsqueda truncada.

- ∞ Búsqueda por índices (autor, título, tema).
- ∞ Búsqueda por código de barra, clasificación, ISBN, número de registro.
- ∞ Generación de bibliografías.
- ∞ Diseminación selectiva de la información.
- ∞ Renovación y reserva de materiales.
- ∞ Solicitudes de adquisición por usuarios autorizados.
- ∞ Tutorial.
- ∞ Buzón de sugerencias.
- ∞ Historial de búsquedas.
- ∞ Formatos de despliegue (MARC, bibliográfico, catalográfico).
- ∞ Guardar e imprimir.
- ∞ Envío por correo electrónico.
- ∞ Generación de bibliografías a partir de registros marcados.⁴⁴

En conclusión, los requerimientos necesarios para instalar e implementar cualquier sistema de automatización en la biblioteca dependen mucho de las características del software (nivel, capacidad, plataforma, etc.), mismo que determinará las especificaciones del hardware a utilizar. Cabe señalar que en los sistemas operativos libres (código abierto) los requisitos de hardware son mínimos para lograr un óptimo funcionamiento de los sistemas de automatización de bibliotecas compatibles, ofreciendo por parte de los desarrolladores una serie de recomendaciones en cuanto a hardware e infraestructura tecnológica se refiere, con el objetivo de garantizar un óptimo desempeño en los mismos; estas recomendaciones también son propuestas por los proveedores de sistemas propietarios. De cualquier modo, el bibliotecario profesional debe contar con nociones sobre informática para poder entender los distintos requerimientos y especificaciones necesarios para instalar, mantener e implementar un sistema de automatización de bibliotecas.

⁴⁴ ARRIOLA NAVARRETE, Oscar. *Proyecto de desarrollo bibliotecario: anexo técnico*. Guanajuato: Instituto Estatal de la Cultura, Coordinación Estatal de Bibliotecas Públicas, 2007. 10 h.

Es importante hacer mención que hoy en día la mayoría de las bibliotecas optan por adquirir un sistema de gestión integral en lugar de desarrollar sistemas propios para cada proceso. Dichos sistemas integran todos los módulos, mismos que se detallarán en el capítulo siguiente.

La automatización de bibliotecas ha sido una realidad a distintos niveles y en constante desarrollo, sin embargo, hay instituciones que por diversas causas no han optado por poner en marcha un sistema, que como hemos visto genera múltiples ventajas en distintos ámbitos; la automatización invita a las bibliotecas a liberarse de la visión tradicional, inseguridades financieras y miedo de tomar decisiones adecuadas para apostar por un camino hacia la excelencia, con las distintas herramientas que ésta nos ofrece. Recientemente, los Sistemas Integrales de Gestión Bibliotecaria de distribución libre y/o código fuente abierto, han impactado de manera positiva y creciente en el ámbito bibliotecario. Por consiguiente en el siguiente capítulo se analiza a fondo su definición, antecedentes e impacto, con el objetivo de promover su uso y aplicación en bibliotecas mexicanas.



CAPÍTULO 2

Los Sistemas Integrales de Gestión Bibliotecaria de código fuente abierto

2. Los Sistemas Integrales de Gestión Bibliotecaria (Integrated Library Management Systems, en inglés) de código fuente abierto (Open Source)

Es bien sabido que la elección de un Sistema de Automatización para Bibliotecas siempre estará condicionada por factores de diversa índole (política, económica y/o social), resultando en la adquisición final de un software que puede no ser del todo funcional en relación a las necesidades de informatización de la biblioteca. Cuando estos sistemas son propietarios se habrán desperdiciado recursos financieros, humanos y/o tecnológicos (ya que todo implica costos); irrecuperables para la unidad de información.

Globalmente el uso del llamado software de código abierto (OSS Open Source Software) está impactando positivamente en distintos ámbitos y áreas, no siendo las bibliotecas una excepción. Hoy en día, muchas bibliotecas están apostando al uso de Sistemas Integrados de Gestión Bibliotecaria de código abierto, justificándose en las ventajas que los mismos ofrecen (distribución, modificación, copia, cero restricciones, así como la capacidad de interoperabilidad con otras aplicaciones), y que se traducen en reducción de costos a distintos niveles, funcionalidad óptima y adaptación basada en las necesidades reales que presente determinada biblioteca.

Ante esto, en este capítulo se revisan los conceptos de “Sistema Integrado de Gestión Bibliotecaria”, (Integrated Library Management System, del inglés) sus antecedentes y avances tecnológicos, la funcionalidad con la que operan, así como un panorama general sobre el término “Software Libre”, sus características, terminología, desarrollo e impacto en las bibliotecas, para finalizar con un análisis breve de diversas herramientas de código abierto, auxiliares en el tratamiento de la información en las bibliotecas a distintas escalas.

2.1 Definición

Existen distintas definiciones para el término “Sistema Integrado de Gestión Bibliotecaria” (ILMS, ILS o LMS por sus siglas en inglés), que en términos globales está enfocado a la gestión automatizada de los procesos que se llevan a cabo en la biblioteca.

Moya conceptualiza los SIGB como los “Sistemas para el proceso automatizado o informático, de información estructurada y no estructurada sobre actividades y documentos, adaptable a la estructura organizativa de la biblioteca”.⁴⁵

Por su parte, Wilson los define como “El software capaz de gestionar operaciones bibliotecarias, mediante módulos separados que interactúan e intercambian un registro central de datos”.⁴⁶

Podemos entonces definir a un SIGB como un grupo de programas informáticos (módulos) inter-relacionados, que automatizan múltiples operaciones y funciones bibliotecarias basados en datos centralizados e intercambiables; esto con el objetivo de facilitar la gestión de las actividades llevadas a cabo en la biblioteca.

Un SIGB es aplicable prácticamente a todos los procesos llevados a cabo dentro de la biblioteca, desde los procesos administrativos, técnicos y referenciales, hasta los que involucran el desarrollo de colecciones, resultando en beneficios directos tanto para la biblioteca como para el usuario.

⁴⁵ MOYA ANEGÓN, F. *Los sistemas integrados de gestión bibliotecaria*. Madrid: ANABAD, 1995. p.25.

⁴⁶ WILSON, Katie. *Computers in libraries: an introduction for library technicians*. USA: The Haworth information Press, 2006. p.27.

2.2 Antecedentes

La década de los años 80 marcó el despegue para los sistemas Multi-tarea, que permitían la integración de distintas funciones dentro una sola aplicación; los bibliotecarios ahora contaban con una herramienta “modular” basada en distintos componentes llamados “módulos”, estableciendo con esto las bases para el concepto Sistema Integrado. Dichos sistemas trabajaban bajo un mismo núcleo, que era un pequeño dispositivo, el cual podía ser comprado, o licenciado para su uso por el cliente.⁴⁷

Los años 80 son clave para el desarrollo de sistemas integrados, que surgen como una evolución de los sistemas monofuncionales que se emplearon hasta finales de los años 70, mismos que tenían por objetivo resolver el problema de la gestión mecánica de funciones que suponían un mayor costo de recursos humanos a las grandes bibliotecas (Library of Congress y The British Library).⁴⁸

Durante esta década se desarrollaron sistemas como ADVANCE, CLSI, y ADLIB, que trabajaban bajo un esquema de multifuncionalidad permitiendo la integración de distintos módulos necesarios para la gestión de la biblioteca; así mismo Ex libris (una de las compañías con mas éxito en la provisión de soluciones comerciales para automatización de bibliotecas) desarrolló su primer SIGB, el exitoso ALEPH 500 para la Universidad Hebrea de Jerusalén, el cual prontamente llegaría a ser el sistema más popular en Europa.⁴⁹

Para entender un poco más sobre el origen de dichos sistemas es necesario remontarnos a la década de los años 60's, caracterizada sin duda por el surgimiento del formato MARC, que permitiría una normalización en los formatos bibliográficos legibles por máquina y que sería adoptado internacionalmente, creando una amplia gama de variaciones hasta hoy en día: UKMARC, UNIMARC, USMARC, OCLC MARC, MARC21, etc.; esto,

⁴⁷ DRAKE, Miriam. *Encyclopedia of Library and Information Science*. USA: Taylor & Francis Group, 2003. p.250.

⁴⁸ JACQUESSON, A. *Op. cit.*

⁴⁹ BOWMAN, J.H. *British Librarianship and Information Work 1991-2000*. USA: Ashgate, 2006. p.454.

aunado al desarrollo y rápida expansión de las computadoras, marcó la pauta para la creación de los primeros sistemas monofuncionales, auxiliares en tareas y ámbitos bibliotecarios, entre los cuales destaca un software desarrollado por Peter Luhn (un empleado de IBM en 1961), llamado “*Palabra-clave en contexto*” (KWIC, por sus siglas en inglés), el cual se encargaba de reproducir palabras clave e indexar los títulos aparecidos en el “*Chemical Abstract*”.⁵⁰ Luhn también participó en actividades de automatización con la National Library of Medicine bajo el proyecto MEDLARS (Medical Literature Analysis and Retrieval System), y es considerado pionero en tecnologías enfocadas a las ciencias de la información en Estados Unidos. Tras el devenir de la década otras instituciones incursionaron en el campo de la automatización, la University of California at San Diego, con el control automatizado para publicaciones periódicas; Southern Illinois University, Carbondale, con el sistema de circulación automatizada, y la Ontario New University Library, con el catálogo de libros en computadora.

Las instituciones europeas no se quedaron atrás. En Inglaterra, la New Castle University Library introdujo el llamado “New Castle file-handling system” (NFHS), una suite de programas desarrollados por el English Electric KDF9, que fueron utilizados para la adquisición de materiales; y en Alemania dos bibliotecas universitarias: Bochum y la Universidad Técnica de Berlín implementaron un sistema automatizado para el proceso de circulación utilizando tarjetas perforadas.⁵¹ El desarrollo que se le daba a los Sistemas de Automatización en Estados Unidos y Europa nos hace notar el interés que se tenía por los mismos como herramienta de apoyo en los quehaceres bibliotecarios.

Ubicándose en la misma década y principios de la subsiguiente se mencionarán algunos sistemas relevantes en cuanto a funcionalidad y uso se

⁵⁰ *Class Lecture Notes: H. P. Luhn and Automatic Indexing; References to the Early Years of Automatic Indexing and Information Retrieval.* [en línea]. [Consulta: 30 de Noviembre de 2009]. Disponible en Internet: <http://www.ischool.utexas.edu/~ssoy/organizing/luhn.htm>.

⁵¹ PANDEY, S.K. *Encyclopaedia of Library Automation Systems and Networks.* India: ANMOL Publications, 1999. p.10-11.

refiere, que marcaron un hito en el desarrollo de los Sistemas Integrados de Gestión Bibliotecaria.

Primeramente se da pie al sistema creado por la Organización Internacional del Trabajo (OIT) en 1965, llamado ISIS (Integrated Set of Information Systems, Conjunto Integrado de Sistemas de Información), creado con el objetivo de automatizar sus actividades bibliotecarias y el cual fue desarrollado y a su vez operado en computadores IBM 360. Una vez implementado, la organización inició su distribución a nivel internacional cubriendo con esto el vacío prevaleciente de sistemas auxiliares en el tratamiento de la información documental. Para 1975 y ante la necesidad de adaptar ISIS a los nuevos equipos desarrollados por la industria de la computación, el Internacional Development Research Center (IDRS) desarrolló la versión "A" de un sistema compatible a la que denominaron como MINISIS para liberar posteriormente en 1978 la versión "F", misma que sería distribuida hasta los albores de la década de los 90. Para 1980 la administración del ISIS es transferida de la OIT a la UNESCO y el sistema pasa a ser denominado CDS/ISIS.⁵² La distribución comienza a darse de manera gratuita hacia las bibliotecas de países miembros interesados en su adquisición.

En segundo lugar se encuentra la iniciativa creada a principios de los años 70 por dos organizaciones sin fines de lucro, las cuales tenían como objetivo principal compartir recursos y reducir incrementos en los costos de las bibliotecas que las conformaban. Por una parte, el OCLC (Online Computer Library Center) con 50 bibliotecas del estado de Ohio afiliadas, y por otra el RLG (Research Libraries Group) fundado por las bibliotecas universitarias de Harvard, Columbia y Yale, así como por la Biblioteca Pública de Nueva York. Recientemente, ambas organizaciones responsables de proveer información a sus miembros se han fusionado y RLG pasó en 2006 a ser parte de OCLC bajo un esquema de cooperación, integrando todos sus programas y bases de datos como "Eureka" a los servicios ofrecidos por OCLC.

⁵² *La historia de Labordoc*. [en línea]. [Consulta: 10 de Diciembre de 2009]. Disponible en Internet: <http://www.oit.org/public/spanish/support/lib/labordoc/history.htm>

Durante la misma década, el sistema NOTIS de la North Western University Library, inició operaciones, únicamente como un sistema de circulación interactivo; fue desarrollado por miembros de la biblioteca universitaria y fue a principios de los años 80 cuando se convirtió en el primer sistema real puesto en línea para uso único de los clientes del sistema, implementándose también un módulo de catalogación

A la par de este sistema nació también CLSI (Computer Library Services Inc.), originalmente desarrollado para bibliotecas públicas y posteriormente implementado en todo tipo de bibliotecas. Este sistema incorporaba una tecnología realmente innovadora para la época, conocida hasta hoy en día como Touch-Screen, que brindaba una interfaz amigable y de fácil acceso a los usuarios del sistema, quienes por primera vez se animaron a utilizar estos sistemas por cuenta propia.⁵³

Para finales de los 70's estaba claro que las bibliotecas que no habían sido partícipes en el desarrollo de sus propios sistemas, estaban listas para adquirir sistemas comercialmente disponibles. Como se ha mencionado, durante esta década los primeros sistemas fueron monofuncionales y muchas bibliotecas comenzaron a enfocar sus necesidades eventualmente en evolucionar hacia un sistema en línea, interactivo e integrado.

Ante esto, para finales de los años 60 y principios de los 70 IBM desarrolló varios sistemas para bibliotecas que jamás estuvieron disponibles comercialmente; éstos fueron experimentales y se instalaron en la biblioteca corporativa de la empresa ubicada en "Los Gatos", California.

IBM entonces firmó un contrato para desarrollar un sistema integrado para bibliotecas con la universidad de Dortmund, en lo que entonces era la República Federal de Alemania. El sistema llegó a ser conocido como DOBIS (Dortmunder Bibliotheks Informatisches Systeme) y consistía de dos módulos, uno para catalogación en línea y otro para búsqueda. IBM estableció entonces

⁵³ DRAKE, Miriam. *Op. cit.*

más contratos para desarrollar sistemas integrados para bibliotecas; para la Universidad Católica de Leuven en Bélgica se desarrolló un sistema llamado LIBIS (Leuvens Integraal Bibliotheek Systeem), dotado de módulos como circulación, adquisiciones, así como de procesamientos multitarea también llamados “batch”. Este sistema se basó esencialmente en una adaptación hecha del formato legible por máquina llamada BNB MARC. Ambos sistemas tuvieron éxito e IBM los fusionó para formar el sistema DOBIS/LIBIS; para finales de la década la Universidad de Pretoria, en Sudáfrica, la Universidad de Perugia, en Italia, y el Colegio Bibliocentre, en Scarborough, Canadá, ya tenían licenciado e instalado el software.

A principios de los años 80 surgieron algunas modificaciones al sistema, de las cuales la más estable fue adoptada por la Biblioteca Nacional de Canadá (NLC National Library of Canada), la cual se encargó de crear la compatibilidad del sistema con el formato MARC, agregar una base de datos bilingüe al mismo, así como definir especificaciones para la búsqueda mediante operadores booleanos. Con este plus IBM comenzó activamente a promover su sistema en los Estados Unidos.

Por último, se comentará acerca de uno de los sistemas más significativos en la historia de los SIGB, que hasta la fecha sigue en uso y constante desarrollo; se trata del Sistema Integrado de Bibliotecas de la Universidad de Chicago.

Financiado con fondos estatales, el sistema fue diseñado para optimizar el control de adquisiciones, pedidos, entregas, publicaciones periódicas, catalogación y etiquetado de material, de igual forma se desarrolló una compatibilidad total con registros MARC, que soportaba el tratamiento de los mismos. Un módulo de circulación también fue desarrollado usando una terminal especialmente diseñada llamada JRL 1000; un archivo de recursos compuesto de registros MARC en cinta pertenecientes a la Library of Congress fueron utilizados para copiar los registros ya catalogados a la base de datos. El Sistema de la Universidad de Chicago también fue pionero en el tratamiento de técnicas para el manejo de grandes bases bibliográficas; crearon la llamada estructura de datos Quadraplanar la cual permitía a las instituciones

intercambiar entre sí sus bases datos, conservando el contenido de su propio registro, en una estructura efectivamente Quadraplanar.⁵⁴

Éste sin duda, marcó el inicio de la comercialización de la información. Durante esta década comenzaron a promoverse servicios de información por medio de la creación y acceso a bases de datos bibliográficas como: Abi/Inform, creada en 1971 con 186,000 registros y Compendex, con 122,000 registros, entre otras bases de datos para bibliotecas universitarias y especializadas de Estados Unidos.⁵⁵

Con el advenimiento y la entrada de las microcomputadoras al terreno comercial, las bibliotecas con recursos financieros estables contaban con mayores posibilidades de crecimiento tecnológico. Muchas bibliotecas reconocieron el incremento en la complejidad para desarrollar sus propios sistemas por lo que, consideraron en muchos casos la adquisición de un sistema integrado para bibliotecas, mediada a través de un vendedor externo.

El trabajo conjunto entre instituciones de educación y compañías privadas impulsó el desarrollo de los Sistemas Integrados para Bibliotecas de carácter comercial a gran escala; durante los años 80 el escenario fue dominado por los llamados “sistemas llave en mano” utilizados en conjunto para gestionar los distintos procesos llevados a cabo dentro de la biblioteca. Consecuentemente, podemos decir que esta época marcó sin duda hitos de relevancia en el desarrollo de los sistemas integrados para la gestión bibliotecaria.

⁵⁴ GROSCH, Audrey. *Library Information Technology and Networks*. USA: Marcel Dekker, 1995. p.27-29.

⁵⁵ CARRILLO LÓPEZ, María Eugenia. *Automatización de bibliotecas de la Unidad de Torreón de la Universidad Autónoma de Coahuila*. Monterrey: El autor, 1992. p. 28.

2.3 Desarrollo de los SIGB

El desarrollo de las nuevas tecnologías, gestado en la década de los 80 con gran impacto en la subsiguiente y centrado en las telecomunicaciones, la informática y los medios audiovisuales, influyeron de manera considerable en las formas de almacenamiento y acceso a la información. Durante esta década se desarrollaron nuevos soportes ópticos como el CD-ROM, ofreciendo grandes ventajas frente a otros soportes electrónicos como los disquetes, especialmente por su alta capacidad de memoria (20 veces más que un disco duro de la época), su rapidez de recuperación de datos y la posibilidad de integrar diferentes medios junto a elementos textuales. Dicho soporte implicó innovaciones para la biblioteca y sus sistemas integrados; en 1984 algunos vendedores de sistemas realizaron pruebas del disco óptico con la intención de incorporarlo a corto plazo en las rutinas de los sistemas de servicios.⁵⁶

En el ámbito de las telecomunicaciones aparecieron la fibra óptica y las microondas, que sustituyeron paulatinamente al hilo de cobre que predominaba entre redes centrales. Ésta tenía como ventajas indiscutibles, la alta velocidad en transferencia de datos, fácil instalación, conectividad permanente, así como la inmunidad ante ruidos e interferencias. El impacto de esta tecnología se tradujo en la reducción de costos y en la optimización de la operabilidad de los sistemas de comunicación de las organizaciones, instituciones y empresas privadas.

En esta misma década de los 80 Jean Cloutier concretó el desarrollo de Internet como una plataforma comunicacional y multimedial, complementando la estructura de la naciente sociedad del conocimiento.⁵⁷

Todo esto influyó notablemente en el desarrollo de los Sistemas Integrados de Gestión Bibliotecaria de carácter comercial, abriendo la posibilidad de ofrecer

⁵⁶ REYNOLDS, Dennis. *Automatización de bibliotecas*. Madrid: Fundación Germán Sánchez Ruipérez, 1998. p. 283.

⁵⁷ MURCIA FLORIAN, Jorge. *Redes del saber, investigación virtual, proceso educativo y autoformación*. Colombia: Cooperativa Editorial Magisterio, 2004. p.89.

nuevos servicios a menores costos y con mayor alcance, como lo fue la expansión a Internet de los llamados OPAC's, los cuales ahora fueron llamados "web-based" (basados en web), y tenían el potencial de proveer acceso remoto a los recursos informativos disponibles en la biblioteca, mediante un entorno grafico mas agradable.⁵⁸

El desarrollo continuo que han tenido los Sistemas Integrados de Gestión Bibliotecaria hasta nuestros días es notable; Wals hace referencia al aumento en la competitividad entre las empresas dedicadas al desarrollo de Sistemas Integrados de Bibliotecas, las cuales han realizado un conjunto de mejoras en sus sistemas, entre las cuales destacan las siguientes:

- El desarrollo de módulos de aplicación que disminuyen la necesidad de operadores humanos por ser tareas esencialmente repetitivas: módulos de autopréstamo, empleo de EDIFACT, para la adquisición de recursos de información, etc.
- La principal línea de desarrollo tiene que ver con el incremento de funcionalidades del OPAC, que permite acceder a un número mayor de recursos de información: bases de datos, revistas electrónicas, websites (catálogos de otras bibliotecas). Así a mediados de los años 80 los sistemas han permitido la consulta de catálogos de otras bibliotecas que disponían del mismo sistema. Actualmente, esta interconectividad ha aumentado gracias al uso de la norma Z39.50, que permite la consulta distribuida del catálogo entre diferentes sistemas conformes a la norma.
- Asimismo, se desarrollan herramientas que permiten instalar bases de datos en soportes ópticos conformes a Z39.50 en computadoras diferentes a los que albergan los sistemas bibliotecarios y que pueden consultarse a través del OPAC.
- Cada vez más, los proveedores de sistemas ofrecen módulos de biblioteca digital que permiten la digitalización de documentos y su consulta a través de la red (libros, artículos de revista, fotos, etc.). Ejemplos de ello son: el módulo ENCompass de Endeavor Information

⁵⁸ STOVER, Mark. *Theological Librarians and the Internet: Implications for practice*. USA: The Haworth Press Inc, 2001. p.18-20

Systems, el ImageManager de VTLS, el Hyperion Digital Archive System de SIRSI. Con todo, es necesario desarrollar un módulo de archivo electrónico o de biblioteca digital, del que carecen los actuales sistemas, que permita la gestión y archivo de la documentación electrónica propia de la biblioteca, así como de los recursos de información de naturaleza digital, que configuran lo que en la actualidad se denomina Biblioteca Digital.

- Por último, en un entorno donde la gestión se afronta desde una perspectiva global, es preciso que los sistemas de gestión bibliotecaria sean capaces de llevar a cabo una gestión consorciada, es decir, de administrar tanto los recursos de información propios como aquéllos a los que una biblioteca puede acceder en virtud de los acuerdos de consorcio que mantenga con otras instituciones bibliotecarias.⁵⁹

Estos son algunos elementos que se han incorporado como parte del desarrollo de los Sistemas Integrados de Bibliotecas; si bien éstos surgieron, como ya se ha mencionado, con base en sistemas monofuncionales, hoy en día es posible integrar la multifuncionalidad que poseen para llevar a cabo distintos quehaceres bibliotecarios de la mejor manera posible, basados en el tratamiento de los procesos con la ayuda de sistemas computarizados. De igual manera, el gran desarrollo de recursos de información ha influido de manera directa en las bibliotecas y sus sistemas; el reto es reconfigurar los esquemas de procedimientos y abrirse a nuevas tecnologías en un ambiente de red compartido, que reúne a usuarios, recursos de información y proveedores de servicios.

A continuación se revisan algunas características de los ILMS, su funcionalidad, así como las nuevas tendencias que proponen diversos autores para el desarrollo de estos sistemas, enfocados a modelos que amplíen funciones y servicios en relación a las nuevas tecnologías de la información y la comunicación.

⁵⁹ MAGAL WALSH, José. *Temas de biblioteconomía universitaria y general*. España: Editorial Complutense, 2001. p.259.

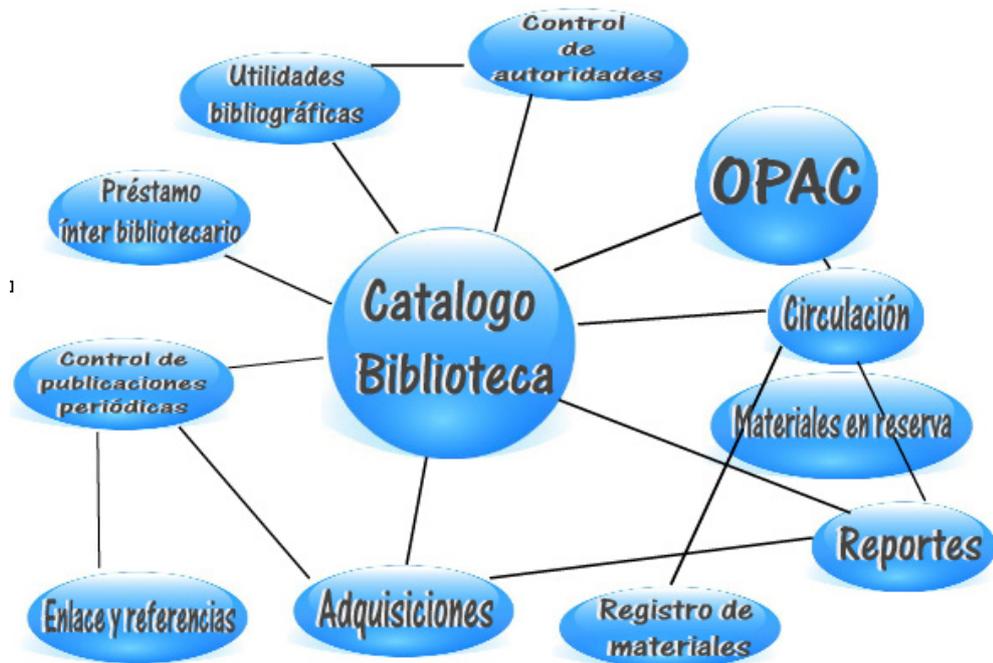
2.4 Características

Hoy en día existe una oferta amplia de Sistemas Integrales de Gestión Bibliotecaria, los hay de carácter comercial así como libres. El creciente desarrollo en las tecnologías de la información (Internet) y en los nuevos soportes de información (documentos digitales) han influido en las características que estos poseen, añadiéndose nuevos módulos y nuevos modelos que permitan de una manera sencilla potenciar y facilitar su uso dentro de la biblioteca, adaptándose a las necesidades de la misma.

Ante todo, podemos mencionar que de una manera estándar los SIGB reúnen las siguientes características:

- Trabajan bajo arquitectura cliente-servidor.
- Se ajustan a normas y estándares oficiales para el tratamiento de información (MARC, ISBD, AACR2, Z39.50).
- Comparten la misma información de una base de datos (optimizada por los llamados DBMS, "Database Management System", que muchas veces se incluye en el sistema, o se instalado como software adicional).
- Generalmente incluyen dos o mas interfaces (una para el administrador del sistema, otra para el equipo de bibliotecarios y una más para los usuarios).
- Suelen ser sistemas abiertos, refiriéndonos con esto a que pueden ser instalados en un amplia variedad de marcas y modelos de equipos de cómputo.

Éstas se enfocan a las características tecnológicas de un SIGB; por otro lado las características de funcionalidad hacen mención a los módulos que integra el sistema con el objetivo de optimizar tareas bibliotecarias específicas; de manera gráfica, un SIGB está integrado y se interrelaciona de la siguiente forma:



2.5 Requerimientos funcionales

Para que un sistema integrado para bibliotecas pueda ser funcional, debe incluir ciertas características y elementos tanto a nivel general como por cada módulo que lo conforma.

De manera general, debe tomarse en cuenta lo siguiente:

- a) Actualización inmediata y automática de la información.
- b) Formato MARC para el almacenamiento de los registros bibliográficos.
- c) Facilidad para la importación y exportación de registros.
- d) Escalabilidad.
- e) Portabilidad.
- f) Idioma.

- g) Integración.
- h) Amplitud.
- i) Despliegue, controles y estadísticas.
- j) Capacitación.
- k) Respaldo y continuidad.
- l) Costo.

Arriola Navarrete menciona que los requerimientos funcionales “se refieren específicamente a las aplicaciones del programa y dependen ampliamente de las necesidades propias de cada biblioteca y por ende de los procedimientos y expectativas de cada sección o departamento”,⁶⁰ e indica de forma general los requerimientos básicos con los que debe contar cada módulo.

Adquisiciones. Los requerimientos contemplados dentro del módulo de adquisiciones son:

a) Permitir el manejo de fondos en orden jerárquico y total, calcular el monto a gastar contra el presupuesto total, facilitar la conversión de divisas, permitir el control de presupuestos, gastos, compromisos, ajustes, transferencias y cambios de moneda. De igual manera, es fundamental que incluya una opción para el manejo de distintos tipos de adquisición, de publicaciones seriadas y no seriadas, sin ignorar el material especial y audiovisual.

b) Consulta e ingreso. Facilitará el acceso y la consulta sobre la situación de cada ítem por diferentes llaves, con un mínimo de caracteres de digitación, y a su vez contará con los índices requeridos por la biblioteca (autor, título, ISBN, ISSN, No. de pedido, No. de orden y proveedor, código de barras, entre otros). El ingreso parcial o total de los registros debe ser único, llámese material o proveedores; ofreciendo la posibilidad de transferir la información por medio de correo electrónico.

⁶⁰ ARRIOLA NAVARRETE, Oscar y GARMENDIA BONILLA, Lovania. *Op. cit.*

c) Órdenes. Referente a las órdenes de compra es necesario que el software suministre las opciones de: ingreso de órdenes, recepción parcial o total, cálculos de costos y renovación automática de suscripciones. También debe puntualizar períodos de reclamo y éste conservará también un archivo de materiales no adquiridos para futuras cotizaciones, así como del historial de reclamos. Es de suma importancia disponer de la opción de envío de solicitudes pro forma a diversos proveedores.

d) Proveedores. El programa producirá un análisis por proveedor de acuerdo a: solicitudes, entregas contra solicitudes, transacciones satisfechas e inconclusas, gastos por proveedor, descuentos y otros.

e) Reportes. Entre los reportes generales requeridos en el módulo de adquisiciones encontramos: proformas, notificaciones, órdenes de compra, impresión de recibo total o parcial, modificación de órdenes, órdenes por pagar, reportes de recursos, notificaciones de reclamo de material no recibido por proveedor, facturas, etiquetas autoadhesivas, formularios para canje o donación y etiquetas de código de barras, entre otros, siempre con un mínimo de digitación. Éste debe producir además estadísticas por diferentes campos según el tipo de adquisición, área temática, tipo de material, etc.

Catalogación. En el módulo de catalogación observamos los siguientes requisitos:

a) Registros. El módulo nos posibilitará crear, modificar y consultar cualquier registro con un mínimo de digitación. Este, a su vez, habilitará la importación directa de registros bibliográficos de otras bases de datos y soportes, brindando la opción de crear o modificar el control de autoridad, verificando los campos con contenido normalizado, admitiendo cambios globales mediante el uso de referencias cruzadas y relaciones semánticas.

b) Control. Es de suma importancia tomar en cuenta la supervisión y el control de calidad. Éste permitirá la validación de los campos y la producción

automática de una lista que contenga los registros creados y/o modificados en determinada fecha y por catalogador específico.

c) Reportes. En el módulo de catalogación el programa producirá los materiales impresos que determine la biblioteca, mismos que pueden ser primarios o secundarios, así como estadísticas e informes por diferentes campos y en diferentes formatos de salida: marbetes, lista de nuevas adquisiciones procesadas, estadísticas por catalogador, por fechas, áreas, etc.

Catálogo público en línea (OPAC). En este módulo el software debe incluir una compatibilidad con el protocolo Z39.50. Habilitar la búsqueda por diferentes puntos de acceso, desplegar la información en varios formatos, así como permitir la grabación de registros en medios magnéticos y/o impresión.

a) Búsqueda. Deberá cubrir los campos de: autor, título, materia, clasificación y serie (o las variantes que defina la biblioteca, por ejemplo, tablas de contenido); esto permitirá la recuperación de información, ya sea por medio del campo completo o por subcampos, omitiendo las palabras no significativas mediante el uso de delimitadores. El mismo facilitará la búsqueda por operador boleano, identificando el término en minúsculas o mayúsculas indistintamente, remitiendo a sinónimos en caso que sea necesario (interactuando con los archivos de control de autoridad) y conservando un archivo de consultas no resueltas, con términos utilizados como llave de búsqueda.

b) Despliegue. Se hará por registro de autor y/o título y en orden alfabético o con formato de índice. La selección del tipo de despliegue será establecido por la biblioteca, permitiendo señalar los ítems de interés para consulta en línea, grabación y/o impresión.

c) Grabación y/o impresión. Se realizará de acuerdo a los lineamientos establecidos por la biblioteca, coexistiendo ambos o únicamente uno de ellos.

d) Reportes. Los reportes de información serán generados por: índices, tipos de búsqueda, número de búsquedas por equipo, área, consultas resueltas y otras estadísticas.

Circulación. Este módulo estará enfocado a gestionar en línea la información sobre la colección y el registro de usuarios. Generará de manera automática, un archivo a manera de historial por cada usuario, incluso por categorías de usuarios y tipos de material consultado (libros, recursos continuos y audiovisuales, entre otros).

a) Préstamo y devolución. Mediante la lectura de códigos o digitación del código del usuario y/o del material, éste generará el tipo de transacción a realizar, brindando la posibilidad de asignar distintos tipos y periodos de préstamo, categoría de usuario, estado del material, número de copias de un título, control de renovación, apartados y días no laborales, esto con el objetivo de habilitar o no el préstamo al usuario.

b) Control de morosidad. Éste se encargará de registrar para la biblioteca, los datos del usuario, tipo de morosidad (multa o material), tipo de material, diferentes montos de multa según la categoría del usuario, así como el estado del material. De igual manera, permitirá el pago parcial o total y emitirá comprobante de pago.

c) Inventario. Producirá un inventario parcial o total por equipo fijo o portátil, tomando en cuenta préstamos, material en encuadernación y otros.

d) Archivo de usuarios. Ofrecerá la opción de ingresar, excluir, agregar, o reemplazar registros de manera ágil y eficaz.

e) Reportes. Este módulo emitirá reportes sobre las transacciones realizadas por la sección o departamento de circulación, tomando en cuenta: préstamos y devoluciones, usuarios atendidos por categoría, inventario en estantería, en préstamo, en encuadernación, por áreas temáticas, por autor y por título. Es decir, mediante la relación de parámetros emitirá formularios para cada

sección, según la periodicidad requerida por la biblioteca. Igualmente, se encargará de producir notas de recordatorio o reclamo por usuario, así como cartas de “no adeudo de material”.

Publicaciones seriadas. La función de este módulo será registrar el ingreso de material, reclamo de ítems, inventario, reportes, producción de materiales y registro para la Diseminación Selectiva de Información.

a) Registro. Este módulo facilitará el ingreso ágil de información con un mínimo de digitación, considerando cambios de título, periodicidad, etc., de las publicaciones. Registrará los fascículos recibidos e informará si una publicación es parte del servicio de alerta o de otra modalidad, controlará la producción de hojas de ruta (esto para los títulos que circulan entre un grupo de personas) e identificará la localización del ítem.

b) Ingreso y solicitud rápida. Permitirá renovar suscripciones, revisar faltantes, así como preparar registros para encuadernación. De igual manera, permitirá generar de manera automática: fecha, número de orden y otros datos, permitiendo su modificación.

c) Consulta. La recuperación de registros se realizará a través del OPAC, en el cual se desplegará tanto la información bibliográfica como la de los acervos, manteniendo los perfiles de interés de los usuarios para la Diseminación Selectiva de la Información.

d) Reportes. Será capaz de generar reportes referentes a: notas de reclamo, etiquetas para Diseminación Selectiva de la Información y estadísticas.

Préstamo interbibliotecario. Este módulo puede ser independiente o ligado al módulo de circulación.

Incluye la capacidad para crear solicitudes en línea con un mínimo de digitación, la importación de datos, la selección y formateo de solicitudes. A su vez, el mismo tramita la devolución ágil de material, reportes por fechas,

renovaciones, devoluciones, por institución solicitante, por institución solicitada, áreas temáticas y categorías de la solicitud, generando reportes de ítems no recibidos, búsqueda de consultas resueltas, notificaciones sobre el proceso de la solicitud, notas de atraso, recordatorios y otros, previamente especificados por la biblioteca.⁶¹

Estos son algunos de los módulos y funciones básicas con las que debe contar un sistema integrado para bibliotecas, brindando, tanto al bibliotecario como al usuario, la posibilidad de contar con diversas herramientas, por una parte para la gestión y organización en la biblioteca, y por otra para acceder de manera ágil a la información pertinente en el momento requerido. De acuerdo a los puntos anteriores, es recomendable seleccionar un software capaz de cubrir el mayor número de requerimientos propios de la biblioteca; basándonos en las características propias de cada sistema, su adaptabilidad y el alcance que se puede lograr con el mismo.

Se han revisado algunas características básicas y de funcionalidad con las que debe contar un sistema integrado para bibliotecas; sin embargo, el desarrollo constante en tecnologías Web, los nuevos soportes documentales, las redes en el entorno de la información y la expansión a Internet del mercado informacional, han influido para que autores como Saorin Pérez, propongan nuevos modelos de SIGB (de manera conceptual), que respondan a la transformación constante observada en el amplio campo de la automatización de bibliotecas; proponiendo para cada característica básica de un sistema integrado para bibliotecas, una nueva tendencia a desarrollar; mismas que conviene analizar en la siguiente tabla:

⁶¹ *Ibíd.*

Principales transformaciones en marcha

Sistema de información general		Sistema de información personalizado
SIGB individual		SIGB distribuido
Sistema de información referencial		Sistema de información a texto completo
Sistema de información cerrado		Sistema de información abierto (Bibliotecas, Productores y Agregadores)
Sistema de información Bibliográfico		Sistema de información multipropósito
Documentos locales papel		Colección digital distribuida
OPAC en sala		Puntos Servicio Interactivo
Sala de consulta		Sala de consulta digital
Biblioteca electrónica		Biblioteca digital / virtual
Sistemas de Información Público = OPAC		Sistema de Información Público = WEB
Recuperación de información		Descubrimiento de información
Sistema para la Consulta		Sistema Difusión – Comunidad Virtual
Biblioteca-Unidad productos		Biblioteca-Red. Participación en y servicios cooperativos en línea
Gestión préstamo		Gestión Servicios Biblioteca

Esta propuesta es fundamental para los SIGB, y por ende, para la automatización de bibliotecas, siendo capaces de responder y adaptarse a las nuevas demandas y requerimientos solicitados; a su vez, el entorno global de la biblioteca también se ve beneficiado en la medida en que la herramienta de gestión básica permita más y mejores servicios a los usuarios, que siempre serán el objetivo final de toda biblioteca.

Llevar a cabo un proyecto de automatización en la biblioteca con, software propietario, requiere de una inversión cuantiosa en cuanto a tiempo, equipo, mobiliario, personal, capacitación, etc. Por esto, la elección debe ser realizada cuidadosamente y contemplando el hecho de que el sistema vendrá a resolver situaciones a corto, mediano y largo plazo, para que se pueda identificar claramente la recuperación de la inversión realizada.⁶²

Pero existen otras alternativas, las cuales están impactando positivamente en el ámbito de los SIGB y la automatización de bibliotecas, las llamadas “de carácter libre”, que se basa en la libertad otorgada al usuario final para realizar cualquier operación con el código fuente del programa en cuestión, liberándolo para su uso mediante distintas licencias y permitiendo una adaptación del sistema a las necesidades reales de la biblioteca, generalmente con un costo mucho menor y con mucho mayor capacidad de expansión en la misma, que con un software propietario. Esto sin duda es un atractivo para las bibliotecas, las cuales se han visto beneficiadas por la comunidad del software libre, donde desarrolladores de todo el mundo están programando, utilizando, y evaluando software libre enfocado a la Gestión Integral de las Bibliotecas, bajo el entorno de sistemas operativos libres (GNU/LINUX) y propietarios como el ya popular Windows, reduciendo principal y notablemente los costos en la adquisición por licencias de software propietario y contando con la posibilidad de adquirir un software sin restricciones, capaz de gestionar en un nivel óptimo las tareas bibliotecarias.

2.6 La historia del Software Libre (Linux y el proyecto GNU)

Durante los años 60, cuando IBM distribuía los primeros equipos de cómputo comerciales, el software incluido era libre, en el sentido que el mismo podía ser libremente intercambiado y distribuido entre usuarios. Éste venía con el código fuente del programa y podía ser implementado y modificado para su mejora: el crear y compartir software era algo normal y habitual. Sin embargo, la situación cambió tras el devenir de los años 80 cuando las computadoras más modernas

⁶² ARRIOLA NAVARRETE, Oscar y GARMENDIA BONILLA, Lovania. *Op. cit.* p.30

comenzaron a utilizar sistemas operativos propietarios bajo los cuales existía una licencia de uso que restringía a los usuarios a modificar cualquier parte del software y que garantizaba a las empresas proveedoras un control único sobre el mismo.

Para finales de los años 70 y principios de los 80, dos grupos completamente distintos establecieron las raíces de lo que hoy en día se conoce como Software Libre / Open Source.

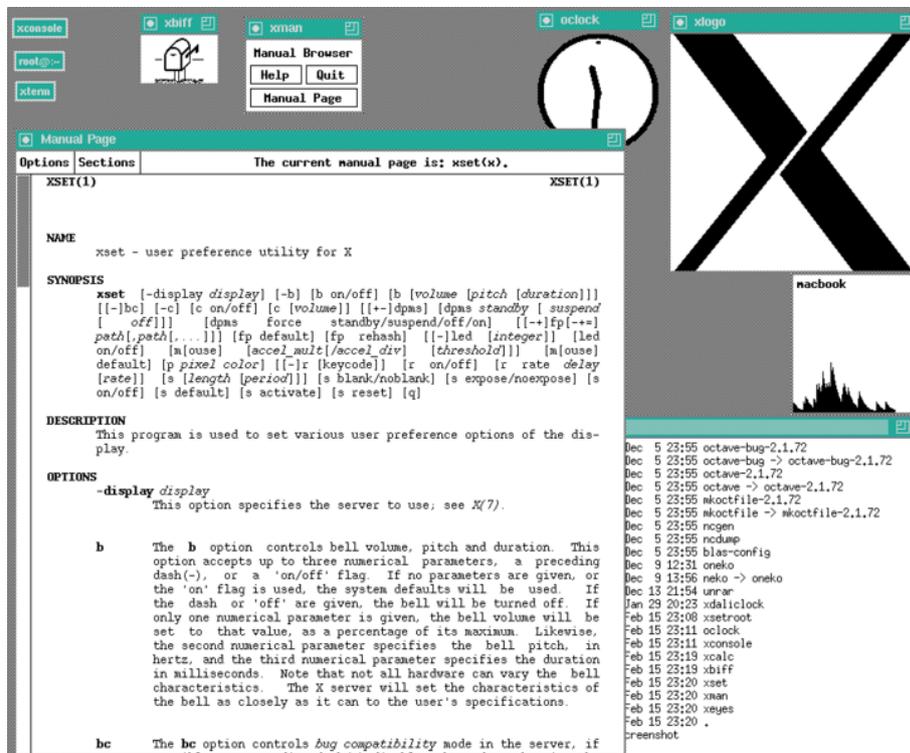
En la costa Este de los Estados Unidos, Richard Stallman, originalmente programador en el laboratorio de IA del MIT (Massachusetts Institute of Technology), comenzó a darse cuenta de que las cosas estaban cambiando e inició un proyecto con el objetivo de formar una comunidad de personas en las que compartir el código volviese a ser algo natural, y que las libertades del software prevalecieran como fue en un principio. Fue entonces cuando creó el proyecto GNU y la Free Software Foundation; la meta del proyecto GNU era diseñar un sistema operativo libre y robusto, similar a UNIX, que pudiera ser distribuido libremente, a su vez como herramienta legal la licencia GPL (General Public License) fue diseñada no sólo para asegurar que el software desarrollado bajo GNU permaneciera libre sino también para promover la producción de más y más software, con la posibilidad de incluir mejoras en cada versión.

En su parte más filosófica, Stallman redactó el manifiesto GNU, fundamentado en la disponibilidad del código fuente y la libertad de modificar y redistribuir el software, como derechos fundamentales.

A la par, en el lado Oeste de los Estados Unidos, el CSRG (Computer Science Research Group) de la Universidad de California implementaba el sistema UNIX, desarrollando un sin número de aplicaciones las cuales rápidamente se convirtieron en BSD UNIX. Estos esfuerzos fueron principalmente financiados por DARPA y a lo largo del mundo un sin fin de hackers ayudaron y complementaron a depurar el sistema para su óptima funcionalidad, manteniéndolo estable e implementando nuevas mejoras. Durante mucho

tiempo el software no fue redistribuido más allá de los poseedores de licencias AT&T UNIX, pero para finales de los 80 finalmente se distribuyó bajo la licencia BSD, una de las primeras licencias libres de código fuente abierto. Desafortunadamente todo usuario tenía que contar con una licencia AT&T UNIX, ya que algunas partes del módulo kernel y ciertas utilidades, las cuales eran necesarias para un óptimo funcionamiento, seguían siendo propietarias.

Durante 1980 y principios de los años 90, el software libre bajo el proyecto GNU continuó su desarrollo, inicialmente en grupos aislados para posteriormente, durante la segunda década mencionada, verse beneficiado por el Internet, sumando esfuerzos transnacionales que ayudarían a construir comunidades sólidas, las cuales aportaron y desarrollaron software de relevancia (utilidades UNIX, compiladores, etc.) que en muchas ocasiones sustituyeron a las herramientas estándar con las que se contaba. Especialmente interesante es el caso del sistema Window X, uno de los primeros casos de software de código fuente abierto financiado por un consorcio de compañías.



Durante 1991-1992 el panorama del software libre y del software en general estaba listo para cambiar.⁶³

En Finlandia, Linus Torvalds, un estudiante de sistemas computacionales, construyó a principios de los años 90, con ayuda de cientos de programadores de todo el mundo y partiendo desde cero, un kernel de sistema operativo similar a Unix, al portar a ese kernel muchas aplicaciones libres ya disponibles en esa época, en especial producidas por el proyecto GNU. Así es como vio la luz el sistema operativo LINUX o como es conocido popularmente, GNU/LINUX. Hoy en día, el proyecto sigue siendo coordinado por Linus Torvalds, incorporándose desarrolladores destacados como Alan Cox,⁶⁴ hasta el punto de ser considerado como uno de los sistemas operativos más avanzados del mundo, especialmente para el procesamiento de información basado en Internet.⁶⁵

Desde entonces, el proyecto GNU no ha dejado de producir virtualmente cientos de programas libres, actualmente se ha desarrollado también un kernel (HURD) que ya es muy utilizable y es la base, por ejemplo, para una distribución Debian (de las más estables). Muchos de los programas están entre los mejores en su campo y GNU se ha convertido en garantía de calidad. El proyecto más innovador desarrollado dentro del marco del proyecto GNU es GNOME, que tiene tanto impacto como para ser considerado un proyecto por sí mismo. La modernización en la licencia GPL incluye nuevas modalidades de uso de software, como servidores de aplicaciones en Internet, por ejemplificar alguno; ésta fue publicada como GPL 3.0.⁶⁶

⁶³ WORKING GROUP ON LIBRE SOFTWARE. *Free software history*. [en línea]. [Consulta: 15 de Diciembre de 2009]. Disponible en Internet: <http://eu.conecta.it/paper.pdf>.

⁶⁴ GONZALES, Jesús. "Como van los proyectos de software libre". En: GRUPO DE SISTEMAS Y COMUNICACIONES. *Compilación de ensayos sobre software libre*. [en línea]. [Consulta: 17 de Diciembre de 2009]. Disponible en Internet: <http://gsyc.escet.urjc.es/~grex/sobre-libre/libro-libre.pdf> p.111.

⁶⁵ Castells, Manuel. *La Galaxia Internet*. España: Plaza and Janes, 2001. p.28

⁶⁶ GONZALES, Jesús. *Op. cit.*

De manera general:

- Principios de 1980. Se especula sobre la creación de comunidades cooperativas que permitan la distribución libre de software, en las que compartir el código volviese a ser algo natural (Richard Stallman).
- Finales de 1980. GNU (GPL, licencia que respalda y promueve), BSD (*Berkeley Software Distribution*), sistema operativo derivado de UNIX. Primeros resultados.
- Finales de 1990. Arquitectura LAMP, (Linux, Apache, Mysql, Perl, PHP, o Python), combinación de tecnologías las cuales de manera combinada representan un conjunto de soluciones que soportan servidores de aplicaciones.
- Principios de 2000. GNOME, KDE; especialmente importantes, ya que se enfocan en la usabilidad por usuarios no avanzados.
- Primera década 2000-2010. Software libre, nueva tendencia en organizaciones, empresas e instituciones.(Ubuntu, SO de fácil uso, se convierte cada vez más en la distribución por excelencia).

El software libre ofrece libertad total a sus usuarios, no sólo para ejecutarlo y utilizarlo sino también para estudiarlo, modificarlo, copiarlo, y distribuirlo con nuevas mejoras (esto implica tener siempre acceso al código fuente). De esta forma, cualquier usuario puede mejorar el software en cuestión y puede liberar públicamente dichas mejoras (con el código fuente propio), beneficiando consecuentemente a un sin fin de usuarios.

2.7 Software Libre vs Software de Código Fuente Abierto

Nos referiremos a software de código fuente abierto como OSS (Open Source Software, por sus siglas en inglés). OSS es tanto, una filosofía como un proceso. Como filosofía describe, el uso destinado que se le dará al software y los métodos para su distribución. Dependiendo de la perspectiva con la que se mire, el concepto de OSS es relativamente nuevo, gestándose alrededor de 11 años atrás, a finales de la década de los años 90, pero fundamentado tras la

historia de UNIX, el Software Libre, y la cultura Hacker. El movimiento OSS es liderado por la OSI (Open Source Initiative), la cual es fundada bajo un esquema educacional, administrativo y de promoción, en el momento cúspide de esta cultura. La cadena inmediata de eventos que dirigió la formación de la OSI, comenzó con la publicación de “La Catedral y el Bazar”, un ensayo a favor del software libre escrito por el hacker Eric S. Raymond, en 1997, el cual analiza porqué el software de código fuente abierto trabaja tan bien,⁶⁷ comprobando las diferencias teóricas entre dos modelos de desarrollo: “catedral” (software no libre) y “bazar” (código abierto). El término Open Source va más allá del acceso al código fuente; éste describe la forma en que el software es licenciado y los términos en los cuales será distribuido siguiendo los siguientes criterios:

1. Redistribución libre

La licencia no debe restringir la venta o colocación del software como componente de una distribución de software agregado que generalmente contiene programas estructurados de diversos códigos fuente. La licencia no debe requerir el pago de regalías u honorarios.

2. Código fuente

El programa debe incluir el código fuente y debe permitir su distribución en código fuente abierto de forma compilada. Si por alguna razón un producto no se distribuye en código fuente, deberá haber una buena publicidad que se refiera a cómo obtener el código fuente pagando no más de un costo simbólico por reproducción, descargándose vía Internet sin cargo alguno. Generalmente se prefiere el código fuente porque así los programadores tienen posibilidad de modificar el programa.

⁶⁷ *History of the OSI*. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.opensource.org/history>.

3. Trabajos derivados

La licencia debe autorizar modificaciones y trabajos derivados, así como su distribución, bajo los mismos términos que la licencia del software original establece.

4. Integridad del código fuente del autor

La licencia tiene autoridad para impedir la distribución de versiones modificadas del código fuente. Este tipo de comercialización sólo es posible si la licencia autoriza la distribución de “archivos parches” adjuntos al código fuente, a fin de que el programa pueda ser modificado durante el tiempo de construcción. La licencia debe avalar explícitamente la distribución del software construido a partir del código fuente modificado y puede ordenar que el nombre o número de versión de los trabajos derivados sea distinto al “software original”.

5. No a la discriminación de personas o grupos

La licencia no debe discriminar a ninguna persona o grupo de personas.

6. No a la discriminación de campos laborales

La licencia no debe restringir el uso del programa a particulares ni a un campo laboral específico. Por ejemplo, no puede impedir que el software sea utilizado en un negocio o que se utilice en una investigación genética.

7. Distribución de la licencia

Los derechos adjuntos al programa son aplicables a todos los usuarios que lo reciben, sin necesidad de tramitar una licencia adicional para estas partes.

8. La licencia no tiene que ser específica de un producto

El programa no puede licenciarse sólo como parte de una distribución mayor. Si el programa se extrae de esa distribución y es usado o distribuido dentro de los términos de la licencia del programa, todas las partes a las que el programa se redistribuya deberán tener los mismos derechos que los concedidos en conjunción con la distribución de software original.

9. La licencia no debe restringir a otro software

La licencia no debe insistir que todos los programas distribuidos bajo el mismo medio proporcionen su código de manera abierta.

10. La licencia debe ser tecnológicamente neutral

No se proveerá de la licencia mediante tecnologías individuales o estilos de interfaz.⁶⁸

Por su parte, Richard Stallman, cabeza en el movimiento del Software Libre y creador del proyecto GNU, la Licencia General Pública (GPL) y la Free Software Foundation (FSF), denota que la intención del Software Libre es ofrecer una serie de libertades a los usuarios, a distintos niveles. El término free (libre) se enfatiza dentro de libertad y no de cero costos. La FSF, al igual que la OSI, se enfocan a diferentes aspectos sobre el uso y distribución del software, disponibilidad y responsabilidades que competen al usuario tomar en cuenta.

Para que un software se defina como software libre, de código abierto o en su caso ambos, debe cumplir con ciertas reglas o normas que lo enmarcarán en la denominación correspondiente. A manera de comparativo y resumiendo las 10 premisas mencionadas anteriormente, tenemos lo siguiente:

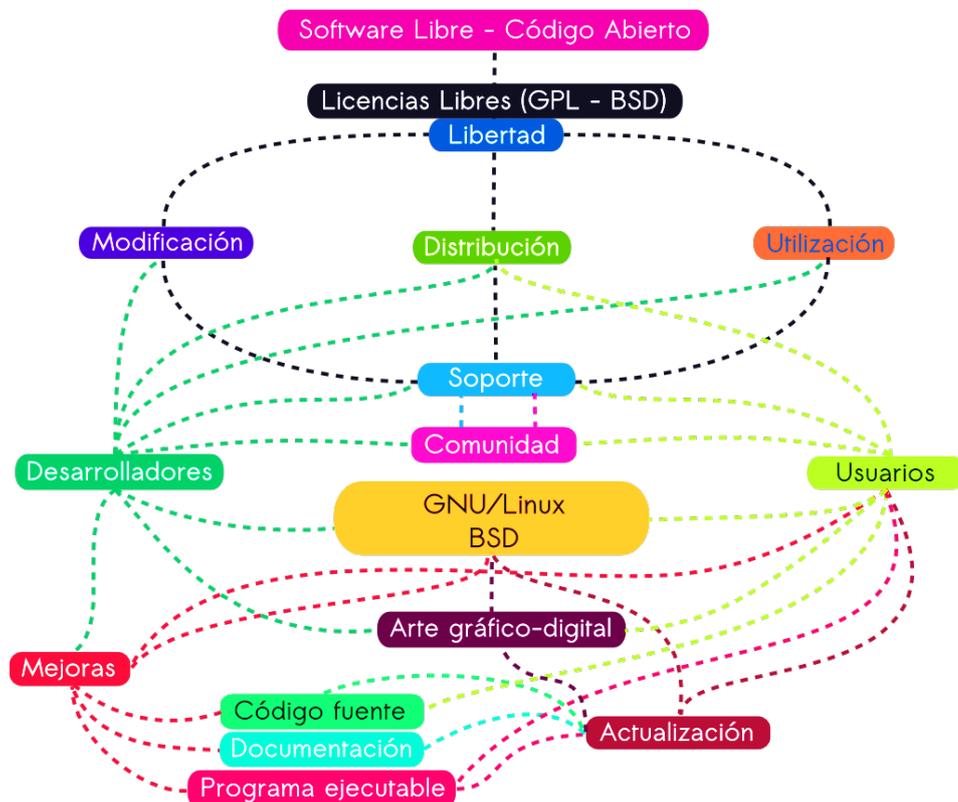
⁶⁸ *The Open Source Definition*. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.opensource.org/docs/osd>.

SOFTWARE LIBRE	CÓDIGO ABIERTO
Las 4 Libertades del Software Libre	Los 10 criterios de Software de Código Abierto
<p>Ejecutar el programa con cualquier propósito (Libertad 0)</p> <p>(Privado, educativo, público, comercial, militar, etc.)</p>	<p>Libre redistribución: el software debe poder ser arreglado o vendido libremente.</p> <p>Código fuente: el código fuente debe estar incluido u obtenerse libremente.</p> <p>Trabajos derivados: la redistribución de modificaciones debe estar permitida.</p>
<p>Estudiar y modificar el programa (Libertad 1)</p> <p>(Para lo cual es necesario poder acceder al código fuente)</p>	<p>Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas sólo como parches.</p> <p>Sin discriminación de personas o grupos: a nadie puede dejársele fuera.</p> <p>Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos.</p>
<p>Distribuir el programa de manera que se beneficie directamente a terceros. (Libertad 2)</p>	<p>Distribución de la licencia: deben aplicarse los mismos derechos a todo el que reciba el programa.</p> <p>La licencia no debe ser específica de un producto: el programa no puede licenciarse sólo o como parte de una distribución mayor.</p> <p>La licencia no registrará otro software: la licencia no obliga a que algún otro software que sea distribuido con el software abierto deba también ser código abierto.</p>
<p>Distribuir las versiones modificadas propias (Libertad 3)</p>	<p>La licencia debe ser tecnológicamente neutral: no se aceptará la licencia por medio de un clic de ratón o de otra forma específica del medio de soporte del software.</p>

Es importante entonces, tener bien definidas estas dos tendencias:

- ☞ Software libre: Es encabezado por la Free Software Foundation (FSF) y tiene como máximo representante a Richard Stallman, creador de la licencia GPL, (General Public License), del proyecto GNU y fundador de la FSF. El movimiento enfoca el término “libre” como libertad de manipulación de software y no de eliminación de costos.
- ☞ Código fuente abierto: Aparece en 1998 con un grupo de personas, entre los que destacan Eric S. Raymond y Bruce Perens, como líderes del movimiento y fundadores de la OSI. (Open Source Initiative). Este pretende darle mayor relevancia a los beneficios prácticos de compartir el código fuente de determinado software, reconociendo el beneficio cualitativo cuando se puede usar, modificar y redistribuir el código fuente de un programa.

En el siguiente mapa conceptual se establece una interrelación en un mismo esquema, entre los dos movimientos, sus características, y el enlace dado entre los elementos que lo conforman.



El movimiento por el software libre y el movimiento por el código abierto son movimientos separados con visiones y metas distintas, pero fundamentados y encaminados hacia el beneficio directo al usuario, mediante la cooperatividad y experiencia de programadores, traductores, “testeadores”, diseñadores, etc.

Se puede decir que la diferencia fundamental entre los dos movimientos está en sus valores, las formas en las que cada uno visualiza al mundo. Para el movimiento Open Source, el asunto sobre si el software debiera ser de código abierto es una cuestión práctica, no ética. Se entiende esto de la siguiente manera: el código abierto es una metodología de desarrollo; el software libre es un movimiento social. Para el Open Source, el software que no sea libre es una solución ineficiente. Para el movimiento por el software libre, el software que no es libre es un problema social y el software libre viene a ser la solución.⁶⁹

2.8 Ventajas del uso de Software Libre

El software libre presenta una serie de ventajas sobre el software propietario por los derechos que otorga a sus usuarios. Algunas de estas ventajas pueden ser más apreciadas por los usuarios particulares, otras por las empresas, y otras por las administraciones públicas. En el caso de bibliotecas, muchas instituciones en países europeos (Nueva Zelanda, Francia, Inglaterra), Norte y América Latina (Colombia, Argentina, Chile, Perú), han puesto en marcha proyectos de automatización basados en software libre, con resultados positivos fortuitamente.

Desafortunadamente, en México el panorama de automatización de bibliotecas con software libre es aún escaso (se sabe únicamente que la biblioteca de la Fundación Javier Barros Sierra usa un software libre como el OpenBiblio). Las ventajas que nos ofrece este tipo de software son amplias, por lo tanto debe tenerse en cuenta por parte de la comunidad bibliotecaria que se encuentre en posibilidades de llevar a cabo un proyecto de automatización en las bibliotecas correspondientes. Las ventajas y desventajas son mencionadas a continuación:

⁶⁹ *Philosophy of the GNU Project*. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.gnu.org/philosophy/philosophy.html>.

Principales ventajas:

Bajo costo de adquisición y libre uso

El software con fines de lucro por lo general no está a la venta. Lo que el usuario adquiere a través de una erogación monetaria o sin ella, es una licencia respecto de los usos que puede darle.

El software no sólo cuesta un precio de adquisición de licencia, también cuesta mantenerlo, operarlo, ajustarlo. Es importante para el usuario el poder mantener estos costos bajo control, pues de lo contrario puede llegar a verse impedido de llevar a cabo sus metas, a fuerza de erogaciones no planificadas. 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. Cualquier persona con una computadora y una conexión a Internet puede utilizar un software libre. Para la mayoría de usuarios individuales el software libre es una opción atractiva por las libertades que garantiza sin necesidad de verse agobiados por el precio. Sin embargo, en el caso de empresas y la Administración Pública, el costo del software es un factor importante y a veces determinante en la elección de nuevos sistemas informáticos. Cuando se analiza el precio de una solución tecnológica, se suele hablar del TCO (Total Cost of Ownership), es decir, del costo total de la propiedad que tiene una determinada solución de software.

Este concepto fue inventado por el Gartner Group en 1987, como herramienta de análisis exhaustiva de los costos de una solución de mercado, y se convirtió en un estándar. El análisis refleja el costo del programa, la ayuda, y el mantenimiento tecnológico de la solución, si partimos de la base que el software libre prácticamente carece 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. La gran mayoría del software libre tiene como postulado la determinación de que el usuario no es un iletrado computacional. Parte del hecho de que confía en los conocimientos y habilidad del usuario para lograr su objetivo. Esta situación se observa desde el diseño del sistema operativo: decenas de pequeños

comandos que hacen una tarea específica y se pueden conectar entre sí para realizar tareas más complejas. Este punto de partida otorga gran poder y flexibilidad para los usuarios capaces y con ánimo de aprender, pero limita al resto de las personas que sólo quieren facilitarse una tarea de manera inmediata.

El software libre tiene las siguientes características:

- Todo el mundo tiene derecho a usarlo sin costo alguno.
- Todo el mundo tiene derecho a acceder a su diseño y aprender de él.
- Todo el mundo tiene derecho a modificarlo: si el software tiene limitaciones o no es adecuado para una tarea, es posible adaptarlo a necesidades específicas y redistribuirlo libremente.
- No tiene un costo asociado (gratuito).
- Es de libre distribución (cualquier persona puede regalarlo, venderlo o prestarlo).

Estos derechos típicamente no están disponibles con el software propietario. Usualmente hay que pagar una “licencia de uso” al creador (como el pago de derechos por el uso de una patente) y se está sujeto a las condiciones del fabricante. Normalmente estas condiciones no otorgan ningún derecho al usuario final. Los derechos mencionados anteriormente tienen una serie de efectos colaterales sobre el software:

- Tiende a ser muy eficiente: porque mucha gente lo optimiza, lo mejora
- Tiende a ser muy robusto: mucha gente “puede” arreglarlo, no solamente el creador o la compañía que lo produce
- Tiende a ser muy diverso: la gente que contribuye tiene necesidades diferentes y esto hace que el software esté adaptado, a una cantidad más grande de problemas.

El éxito del software libre se debe en gran parte a Internet, ya que éste ha permitido que las personas interesadas en los varios componentes del software

libre se pongan fácilmente en contacto con otras. De esta manera, internet actúa como un catalizador que acelera el desarrollo y sintetiza el conocimiento en áreas muy específicas. Hay diferentes motivaciones que impulsan a los contribuidores y desarrolladores a trabajar en el software libre; las más importantes son:

- El deseo de crear nuevo software.
- El deseo de crear software más robusto.
- La posibilidad de estar en control del software (se establece un control total sobre diversos problemas, en cualquier punto).
- Crear aplicaciones de bajo costo.
- Reutilizar el conocimiento: esto permite que la gente reutilice el conocimiento que se ha sintetizado en el software. En vez de empezar desde cero (que es el caso de la industria de software actual) siempre se puede empezar un proyecto desde un fundamento establecido.
- La posibilidad de adaptar el software a sus necesidades.
- Aprender alguna técnica de programación.
- Obtener el reconocimiento de colegas.

Lo mencionado ha dado pie a que se desarrollen sistemas de cómputo que compiten en casi todos los niveles con los sistemas propietarios, pero no contemplan sistemas de marketing y son tradicionalmente esfuerzos que no son conocidos por el público en general.⁷⁰

Innovación tecnológica.

El software libre tiene como objetivo principal compartir la información trabajando de manera cooperativa. Este es principalmente el modelo sobre el que la humanidad ha innovado y avanzado. La ideología de los defensores del software libre es que el conocimiento le pertenece a la humanidad, sin hacer distinciones. Por lo tanto, los usuarios tienen un destacado papel al influir decisivamente en la dirección hacia donde evolucionan los programas: votando

⁷⁰ DE ICAZA, Miguel. *El impacto al sistema e-México de la incorporación de software libre*. [en línea]. [Consulta: 19 de Diciembre de 2009]. Disponible en Internet: <http://www.emexico.gob.mx/>.

los errores que quieren que sean corregidos, proponiendo nueva funcionalidad al programa o contribuyendo ellos mismos en el desarrollo del software (a finales del año 2004 se publicó una lista de las innovaciones más importantes en software de ese año. Se consideró como innovación número uno el navegador libre FireFox, y de los diez programas mencionados también se encontraba OpenOffice disponible en: <http://www.openoffice.org/>).

Requisitos de hardware menores y durabilidad de las soluciones

Aunque resulta imposible generalizar, sí existen casos documentados que demuestran que las soluciones de software libre tienen unos requisitos de hardware menor, y por lo tanto, son más baratas de implementar. Por ejemplo, los sistemas Linux que actúan de servidores pueden ser utilizados sin la interfaz gráfica, con la consecuente reducción de requisitos de hardware necesarios. También es importante destacar que en el software propietario el autor puede decidir en un momento dado no continuar el proyecto para una cierta plataforma, para un hardware que considera antiguo, o discontinuar el soporte para una versión de su software. En las aplicaciones de software libre, estas decisiones no pueden ser tomadas por una empresa o individuo sino por toda una comunidad, con diferentes intereses. Ello se traduce en un mejor soporte -de manera general- para las versiones antiguas de software y de plataformas de hardware o software más minoritarias.

Escrutinio público

El modelo de desarrollo de software libre sigue un método a través del cual de forma cooperativa los programadores que en gran parte son voluntarios, trabajan coordinadamente en Internet. Lógicamente, 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 desarrollo del software libre imprime un gran dinamismo al proceso de corrección de errores. Los usuarios del programa de todo el mundo, gracias a que disponen del código fuente de dicho programa, pueden detectar sus posibles errores, corregirlos y

contribuir a su desarrollo con sus mejoras. Son comunes los casos en que un error de seguridad en Linux y la solución del mismo se hacen públicos. Con el software propietario, la solución de los errores no llega hasta que el fabricante del programa puede asignar los recursos necesarios para solventar el problema y publicar la solución.

Independencia del proveedor

El software libre garantiza una independencia con respecto al proveedor gracias a la disponibilidad del código fuente. Cualquier empresa o profesional, con los conocimientos adecuados, puede seguir ofreciendo desarrollo o servicios para nuestra aplicación. En el mundo del software propietario, sólo el desarrollador de la aplicación puede ofrecer todos los servicios; con el software libre, como su denominación lo indica, su uso es libre: todo aquel que lo tiene en su poder puede usarlo cuantas veces quiera, en cuantas máquinas quiera, para los fines que quiera. 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. Uno de los grandes problemas en la industria del software propietario es la dependencia que se crea entre el fabricante y el cliente.

Este hecho se acentúa con especial gravedad cuando el fabricante no entrega el código fuente, ya que inevitablemente el cliente queda atado a él para nuevas versiones y, en general, para cualquier mejora que necesite. Si el programa usa un formato de almacenamiento estándar, entonces el usuario puede estar seguro de que en el futuro podrá seguir descifrando la información. Si, por el contrario, los datos se almacenan en un formato secreto, el usuario queda atrapado en un determinado proveedor, que es el único que puede ofrecer alguna garantía de acceso a ellos. A menudo los proveedores de software propietario se ven obligados a dejar de fabricar un producto por un cambio drástico de las condiciones del mercado, o simplemente porque consideran que ya no podrán rentabilizar la inversión. Disponiendo del código fuente, cualquier programador puede continuar su desarrollo y sus actualizaciones hasta que el cliente decida que es el momento adecuado para migrar a un nuevo sistema informático.

Industria local

Si el usuario está habilitado para ejecutar un programa, pero no para inspeccionarlo ni modificarlo, entonces no puede aprender de él, se vuelve dependiente de una tecnología que no sólo no comprende sino que le está expresamente vedada. Los profesionales de su entorno, que podrían ayudarlo a alcanzar sus metas, están igualmente limitados: como el funcionamiento del programa es secreto y su inspección está prohibida, no es posible arreglarlo. De esa manera, los profesionales locales ven sus posibilidades de ofrecer valor agregado cada vez más limitadas, y sus horizontes laborales se estrechan junto con sus oportunidades de aprender más. Si lo que necesita es un campo fértil de profesionales competentes, que conozcan a fondo los productos que soportan, y que estén en condiciones de ofrecer soluciones definitivas a los problemas que surjan, y sólo parches, estaría en mucho mejores manos adoptando software libre para su operación.

Si bien es cierto que no existen aún soluciones libres para todas las necesidades de los usuarios, tampoco hay soluciones propietarias para todas las necesidades evidentes. En aquellos casos en que la solución libre no existe, hay que desarrollarla, lo que significa esperar a que alguien más tropiece con la necesidad y lo desarrolle, hacerlo uno mismo, o pagar para que alguien lo haga. La diferencia está en que en aquellos casos en que sí hay una solución libre disponible, el usuario puede utilizarla inmediatamente y sin reparos de ningún tipo, mientras que con las soluciones propietarias siempre tiene que pagar, y lo que obtiene a cambio es una “solución” cerrada y secreta, en vez de una herramienta que le permita crecer y operar con seguridad y libertad.

En México, es casi nula la industria de software y las aplicaciones de consumo masivo se desarrollan en otros países. Un gran porcentaje de la industria se basa en distribuir y dar apoyo e información de productos realizados fuera de nuestras fronteras, por lo tanto, la parte de creación y desarrollo de software es realmente la parte de la industria que requiere de excelentes ingenieros y programadores -que sin duda los hay en México-, lo que generaría que nuestra

industria local creciera generando valor y conocimiento, y trascendiera tecnológicamente.

Respecto al ámbito de la administración pública, una parte importante de la inversión en software se realiza a través de licencias de sistemas operativos, servidores y paquetes de ofimática, producidos totalmente en el extranjero y que sólo tienen repercusión económica en México, con respecto a su distribución. Debido a que en el software libre no hay costo de licencia en atención al derecho a copia, y al disponer del código fuente de la aplicación, es posible desarrollar internamente las mejoras o las modificaciones necesarias, en vez de encargarlas 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. En el software propietario, estas mejoras o no se pueden llevar a cabo o quedan en manos de la empresa creadora, que normalmente se reserva los derechos de uso y propiedad intelectual y establece en qué condiciones las comercializará.

Datos personales, privacidad y seguridad. Seguridad nacional

Para cumplir con sus funciones, el Estado debe almacenar y procesar información relativa a los ciudadanos.

La relación entre el individuo y el Estado depende de la privacidad e integridad de estos datos, que por consiguiente deben ser adecuadamente resguardados contra tres riesgos específicos:

- ✘ Riesgo de filtración: los datos confidenciales deben ser tratados de tal manera que el acceso a ellos sea posible, exclusivamente para las personas e instituciones autorizadas.
- ✘ Riesgo de imposibilidad de acceso: los datos deben ser almacenados de tal forma que el acceso a ellos por parte de las personas e instituciones autorizadas, éste garantizado durante toda la vida útil de la información.

- ✘ Riesgo de manipulación: la modificación de los datos debe estar restringida, nuevamente a las personas e instituciones autorizadas.

La concreción de cualquiera de estas tres amenazas puede tener consecuencias graves tanto para el Estado como para el individuo. Algunos fabricantes de software propietario han colaborado con agencias gubernamentales para incluir accesos secretos al software para así poder visualizar datos confidenciales, de este modo, se comprometen aspectos de la seguridad nacional cuando estos sistemas se utilizan para almacenar datos críticos de gobiernos; debido a esto la Secretaría de la Defensa Nacional en México utiliza software libre.

Cuando los datos son procesados electrónicamente, su vulnerabilidad a estos riesgos está dada por el software que lo procesa. El software libre, al disponer del código fuente, mejora diversos aspectos relacionados con la perennidad de los datos y su seguridad. Para empezar, los sistemas de almacenamiento y recuperación de la información del software son públicos y cualquier programador puede ver y entender cómo se almacenan los datos en un determinado formato o sistema, lo que garantiza la durabilidad de la información y su posterior migración. El software propietario trabaja habitualmente con formatos propios, cuyos mecanismos de almacenamiento no siempre se han hecho públicos, por lo que quizá no sería posible, en caso de que se precisara migrar el sistema, recuperar el contenido de éstos. El software libre sienta las bases para un desarrollo sólido y autónomo de los profesionales locales que ofrecen soluciones, privacidad y seguridad a los usuarios. Si le añadimos a todo lo anteriormente expuesto, que en México no existe una ley federal de protección de datos personales, la situación se vuelve preocupante y nos deja en estado de indefensión si se introdujera código malicioso, espía o de control remoto, es decir, las llamadas “puertas traseras”, debido a que se le puede dar un tratamiento a los datos personales con fines publicitarios, comerciales, policíacos e incluso pudiendo llegar a ser utilizados como medios para realizar opresión (como sobornos o más grave aún al ser un instrumento para identificación y localización para cometer ilícitos) o ser tratados como derechos de propiedad, mercantilizándose hasta convertirse en una forma de

espionaje, sin el conocimiento del titular de los datos, violando su 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 software propietario nunca podremos saber si los programadores originales introdujeron a título personal, por encargo de la empresa o por descuido “puertas traseras” que pongan en peligro la seguridad del sistema o la privacidad de los datos.

En el mundo del software libre, cualquier programador puede realizar una auditoria 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. Respecto a esta ventaja, podemos comentar que resulta favorable seguir incrementando el uso del software libre, por la posibilidad que ofrece de solucionar errores de seguridad de forma inmediata sin, depender de un proveedor externo.

Adaptación del software

El software propietario habitualmente se vende en forma de paquete estándar, que muchas veces no se adapta a las necesidades específicas de empresas y administraciones. Una gran parte de la industria del software se basa en desarrollar proyectos donde se requiere software personalizado. El software libre permite personalizar, los programas, tanto como sea necesario hasta que cubran exactamente nuestra necesidad, gracias al hecho de que disponemos del código fuente. La personalización es un área muy importante ya que el software libre puede responder mucho mejor que el software de propiedad, a unos costos mucho más razonables. Un gran porcentaje de uso de software es de uso interno para las empresas y para las dependencias de la administración pública, lo que requiere un alto grado de personalización y donde el software puede proporcionar desarrollos más económicos.

La licencia de uso de un programa en particular regula las maneras en las que el usuario puede utilizarlo. La variedad de tipos de licencia abarca todo el rango de posibilidades, desde las condiciones más leoninas hasta las más liberales,

pero en general se las puede clasificar en dos grandes categorías: por un lado están las licencias conocidas como “libres”, y por otro las “propietarias”. La gran diferencia entre estos tipos de licencia consiste en que un software licenciado de modo propietario, por lo general otorga al usuario solamente el derecho de ejecutar el programa “tal como es” (es decir, con errores incluidos) en determinada computadora, prohibiendo expresamente todo otro uso, mientras que el software gobernado por una licencia libre permite al usuario no sólo ejecutar el programa en tantas computadoras como desee, sino también copiarlo, inspeccionarlo, modificarlo, mejorarlo, corregir errores y distribuirlo, o contratar a alguien para que lo haga por él.

Lenguas minoritarias, traducción, uso e impulso de difusión

Las lenguas minoritarias existentes en México, como el náhuatl, zapoteca, mixteco, purépecha, entre otros, de nuestras comunidades indígenas, tienen pocas posibilidades de desarrollarse en el mundo del software propietario, y aquellas personas que no dominan el castellano y sólo la lengua original de la comunidad no tendrían acceso al uso y manejo de las computadoras, además de que se irían perdiendo cada vez más estos idiomas, aunque muchos quizá opinen que esto no es una ventaja importante o un tema relevante, pero pensamos que podría servir como un medio para impulsar la difusión de estas lenguas a fin de y que no queden en el olvido y se pierda parte de esta cultura y por lo tanto de nuestro patrimonio nacional.

En el mundo del software libre, estas lenguas y muchas otras, pueden gozar de desarrollo y propagación, gracias a que no precisan autorización de ningún propietario y cualquier persona o institución interesada puede realizar las traducciones. En cambio, en el software propietario sólo la empresa productora posee los derechos para realizar la traducción. 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. Por último, cabe destacar que cada vez que se crea un nuevo recurso lingüístico en el ámbito del software libre (una traducción, un diccionario, un glosario, etc.), éste, al quedar a la disposición de todo el mundo, puede ser reutilizado en futuras aplicaciones. En el mundo del software

propietario cada traducción y recurso lingüístico creado está ligado al fabricante y a sus restricciones de uso.

Principales desventajas:

1. La curva de aprendizaje es mayor. Si ponemos a dos usuarios que nunca han tocado una computadora, probablemente tardarán lo mismo en aprender a usar software propietario, por ejemplo, de Microsoft, que software libre como Gnome o KDE; pero si antes los usuarios ya usaron software propietario generalmente tardan 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 por el uso de sus programas. El software libre se adquiere y se vende "AS IS" (tal cual), sin garantías explícitas del fabricante, sin embargo, puede haber garantías específicas para situaciones muy particulares.

4. Se necesita dedicar recursos a la reparación de errores. Sin embargo, en el software propietario es imposible reparar errores, hay que esperar a que salga a la venta otra versión.

5. No existen compañías únicas que respalden toda la tecnología.

6. Las interfaces gráficas de usuario (GUI) y la multimedia apenas se están estabilizando, aunque hay un número cada vez mayor de usuarios que aseguran que las interfaces gráficas más populares en el software libre (KDE, GNOME y el manejador de ventanas WindowMaker), son ya lo suficientemente estables para el uso cotidiano y lo suficientemente amigables para los neófitos de la informática.

7. La mayoría de la configuración de hardware no es intuitiva. Se requieren conocimientos previos acerca del funcionamiento del sistema operativo y

fundamentos del equipo a conectar para lograr un funcionamiento adecuado. Sin embargo, la documentación referente a la configuración del hardware es tan explícita y detallada que permite al usuario neófito profundizar en el conocimiento de su hardware en muy pocas horas, y una vez teniendo ese conocimiento la configuración se vuelve trivial.

8. Únicamente los proyectos importantes y de trayectoria tienen buen soporte, tanto de los desarrolladores como de los usuarios. Sin embargo, existen muchos proyectos más pequeños y recientes que carecen del compromiso necesario por parte de sus usuarios o desarrolladores para que sean implementados de manera confiable. Estos proyectos importantes que tienen un excelente soporte cubren más del 90% de las necesidades de cómputo del usuario promedio.

9. El usuario debe tener nociones de programación. 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.). Sin embargo, existen en la actualidad muchas herramientas visuales que permiten al usuario no técnico llevar a cabo tareas de configuración del sistema de una manera gráfica muy sencilla, sin la necesidad de conocimientos de programación.⁷¹

2.8.1 Ventajas SIGB de Código Fuente Abierto

Utilizar un Sistema Integral para la Gestión Bibliotecaria, basado en Open Source/Software Libre, generara múltiples ventajas que favorecerán a la biblioteca en distintos aspectos. Uno de ellos y con mucha razón de peso, como se ha mencionado anteriormente, es el presupuesto; las bibliotecas que no cuenten con el presupuesto suficiente para adquirir un software propietario, (que aunque los hay de diversos precios, los mas completos son excesivamente costosos) tendrán la opción de optar por un sistema libre, que generalmente puede ser descargado gratuitamente de Internet.

⁷¹ CULEBRO JUAREZ, Monserrat. [et al.] *Software libre VS software propietario*. [en línea]. [Consulta: 21 de Diciembre de 2009]. Disponible en Internet: www.softwarelibre.cl/drupal//files/32693.pdf.

Utilizar software libre en la biblioteca, invita al bibliotecario a conocer el software en cuestión, su funcionamiento, operatividad, algunas veces incluso, lo lleva a modificarlo y adaptarlo a sus propias necesidades e involucrarse más en el movimiento, promoviéndolo y compartiendo conocimientos y experiencias con otras personas y/o colegas, con absoluta libertad. A su vez, el sentido de cooperatividad entre comunidades de usuarios específicos de cualquier SIBG, se ve reflejado en una dinámica de aprendizaje y de colaboración, contribuyendo notablemente al desarrollo de la profesión y actividad bibliotecaria. Con esto existe la posibilidad de que un bibliotecario se especialice en ofrecer soluciones encaminadas a los servicios de información, utilizando software libre y ampliando con esto su campo laboral y las posibilidades de desarrollo personal.

En resumen, se puede afirmar que el software libre aplicado en bibliotecas favorece en lo siguiente:

- Permite tomar un mayor control sobre los servicios de información que brinda la biblioteca.
- Ayuda a crear soluciones personalizadas para aquellos problemas que se presentan en el uso de herramientas informáticas.
- Ofrece la oportunidad de compartir e intercambiar experiencias sobre varios recursos y conocimientos generados a través de su utilización.
- Proporciona los medios y mecanismos de comunicación que posibilitan devolver a la comunidad usuaria de software libre bibliotecario, las ideas tomadas previamente de ella, con el fin de enriquecerlas y desarrollar nuevas aplicaciones.
- Permite generar conocimiento de manera colectiva y con alto grado de sentimiento de trabajo en grupo y en comunidad.⁷²

⁷² ZURITA SÁNCHEZ, Juan Manuel. "Software libre: una alternativa para la gestión de recursos de información en bibliotecas". En: *Memorias de la Séptima Conferencia Iberoamericana en Sistemas, Cibernética e Informática*. Orlando, FL: International Institute of Informatics and Systemics, 2008. p.10-15.

Algunas de las ventajas notables a tomar en cuenta para adquirir un SIGB de carácter libre, son:

- ✓ Costos bajos de adquisición (muchas veces gratis), y libre uso del software.
- ✓ Requisitos de hardware menores y expansión de las soluciones.
- ✓ Independencia del proveedor.
- ✓ Innovación tecnológica.
- ✓ Modificación/adaptación del software.
- ✓ Multiplataforma, multilinguaje.
- ✓ Libre de virus.
- ✓ Alta seguridad.

Analizando los puntos anteriores, denotamos que el desarrollo y aplicación de sistemas para la gestión de las bibliotecas, basados en una infraestructura de carácter libre y abierto en cuanto al software se refiere, representan hoy en día una alternativa real para las bibliotecas interesadas en llevar a cabo, proyectos de automatización comprometidos con las libertades de acceso y democratización de la información, bajo un esquema de cooperación entre distintas áreas y colegas, mismos que se traducirán, primeramente, en una mejora enfocada al tratamiento de los procesos llevados a cabo en la biblioteca, y por consiguiente, en la mejora en el entorno de los servicios ofrecidos al usuario.

2.9 Tipos de Software y Licencias

Es importante conocer el tipo de licencia bajo el cual es liberado un programa, ya que ésta es un contrato entre el licenciante (programador/autor/titular de los derechos de explotación/distribuidor) y el licenciario del programa informático (usuario consumidor /usuario profesional o empresa), para utilizar el software cumpliendo una serie de términos y condiciones establecidas dentro de sus cláusulas.

Las licencias de software pueden establecer, entre otras cosas: la cesión de determinados derechos del propietario al usuario final sobre una o varias copias del programa informático, los límites en la responsabilidad por fallos, el plazo de cesión de los derechos, el ámbito geográfico de validez del contrato e incluso pueden establecer determinados compromisos del usuario final hacia el propietario, tales como la no cesión del programa a terceros o la no reinstalación del programa en equipos distintos al que se instaló originalmente.

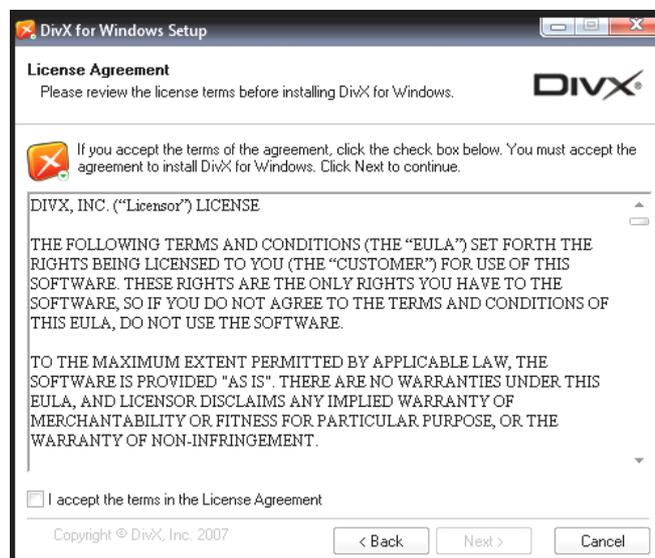
Básicamente se clasifican en:

Licencias Propietarias

Licencias Libres

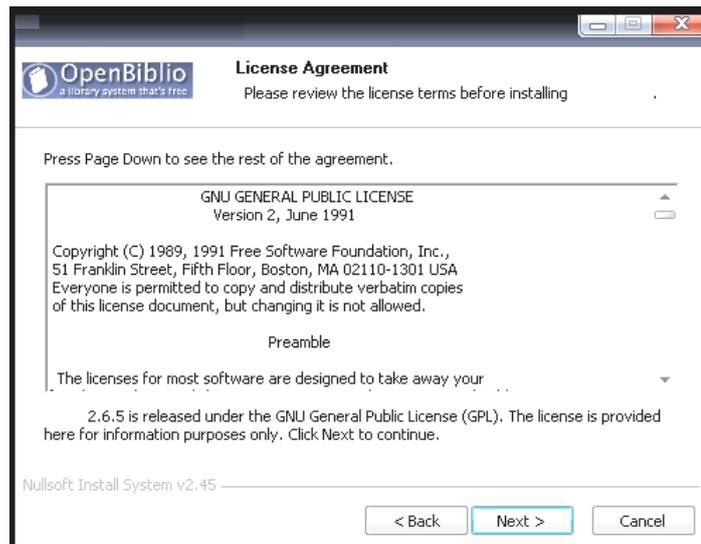
Licencias Permisivas

Licencias Propietarias. Este tipo de licencias, por lo general, no permiten que el software sea modificado, desensamblado, copiado o distribuido de formas no especificadas en la propia licencia; regulan el número de copias que pueden ser instaladas e incluso los fines concretos para los cuales puede ser utilizado. La mayoría de estas licencias limitan fuertemente la responsabilidad derivada de fallos en el programa.

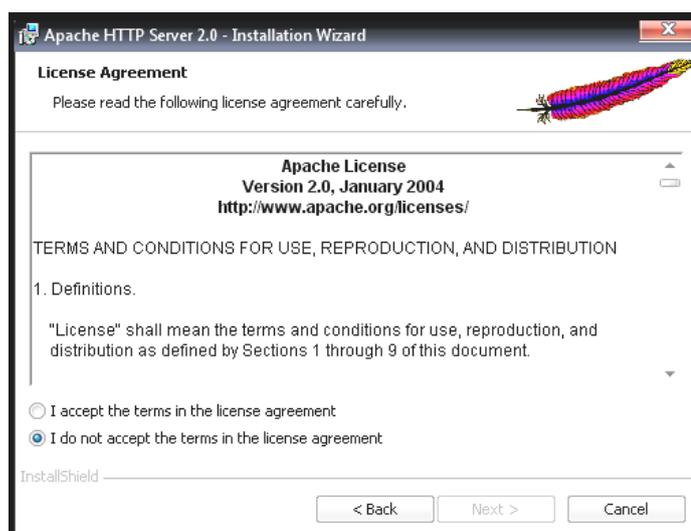


Algunos ejemplos de este tipo de licencias son las llamadas CLUFs (Contrato de Licencia para Usuario Final) o EULAs (End User License Agreement, por sus siglas en inglés).

Licencias Libres. Contienen una cláusula que obliga a que las obras derivadas o modificaciones que se realicen al software original se deban licenciar bajo los mismos términos y condiciones de la licencia original; se pueden adaptar hasta hacer la licencia a la medida conveniente para el autor del programa. Es por esto que existen muchas licencias libres: licencia apache, bzip2, php, postgresql, mozilla, sendmail, etc. Normalmente se suele bautizar la licencia con el nombre del programa liberado.



Licencias Permisivas. Este tipo de licencias habilitan el crear una obra derivada sin que ésta tenga obligación de protección alguna.



A continuación se citan algunas de las más conocidas, así como algunas de sus características distintivas. De igual manera, se indican cuales son aprobadas por la OSI, es decir, aquellas que siguen la filosofía del código abierto. El término “vírica” hace referencia a las licencias que obligan a extender la licencia a todos sus derivados y compuestos.

GNU GPL: Filosofía del software libre, asegura las cuatro libertades y por eso es muy vírica. La idea es que la persona que recibe un programa tiene los mismos derechos que tenía el distribuidor cuando se obtuvo el programa. (Aprobada por la OSI).

GNU LGPL: (Lesser General Public License) Sigue la filosofía GNU GPL, pero no es tan vírica porque está pensada para bibliotecas de funciones de software. La idea que subyace tras esta licencia es facilitar el acceso al código libre al gran público, para hacerlo, permiten que ciertas bibliotecas de software puedan utilizarse desde programas no libres: es menos restrictiva que la GPL. Antes se denominaba GNU Library General Public License; más tarde cambió su nombre por el actual. (Aprobada por la OSI).

Licencia X11 y Xfree86: Filosofía GNU GPL pero no es vírica. Permite hacer modificaciones en el software y comercializarlo pero sin publicar los cambios.

BSD modificada: Versión modificada de la BSD (Berkeley System Distribution) con el fin de hacerla compatible con la GPL, es muy similar a X11, es decir, una licencia no vírica para software libre. (Aprobada por la OSI).

Artistic License: Utilizada por Perl, parecida a GPL, pero no es vírica. Tiene una redacción bastante laxa, y por tanto, los defensores de GNU no la aconsejan y prefieren que si alguien desea una licencia no vírica utilice Xfree86 o similares. (Aprobada por la OSI).

A partir de éstas, todas las licencias siguen parte de la filosofía del software libre, no son compatibles con la GNU GPL, y por tanto, se enumeran separadamente:

Apache: Más permisiva y no vírica.

IBM Public License Versión 1.0: Muy parecida a la GNU/GPL, pero tiene diferencias en el tratamiento de las patentes. (Aprobada por la OSI).

Mozilla Public License (MPL): Es muy parecida a la X11, es decir, no es vírica, pero no permite enlazar códigos bajo la licencia GNU/GPL. (Aprobada por la OSI).

Netscape Public License (NPL): Casi idéntica a la Mozilla, pero con el inconveniente añadido que permite a Netscape utilizar un código creado bajo esta licencia para las versiones propietarias del programa.

QPL: No vírica y obliga a que las modificaciones se distribuyan como correcciones añadidas al original. (Aprobada por la OSI).

BSD: Es una licencia libre pero con la característica especial de que cualquier persona que tenga el programa, puede realizar modificaciones y ofrecerlo de forma cerrada cambiando la licencia. Con esto, los desarrolladores se reservan la posibilidad de volver a cerrarlo. Es una licencia bastante habitual.⁷³

⁷³ PORCEL ITURRALDE, María Laura y RODRÍGUEZ MEDEROS, Mabel. "Software libre: una alternativa para las bibliotecas". [en línea]. En: ACIMED. 2005,13 (6). [Consulta: 25 de Diciembre de 2009]. Disponible en Internet: http://www.bvs.sld.cu/revistas/aci/vol13_6_05/aci090605.htm.

Tipos de software:

Desde una perspectiva legal, existen dos categorías de software: de dominio público y propietario.

El software de dominio público no está protegido por una reserva de derechos de autor, porque ha expirado o el programador ha puesto el software en el dominio público para que todos lo usen sin restricciones. El software de dominio público se puede copiar, distribuir, e incluso revender sin problemas. La restricción principal con el software de dominio público es que no pueden reservarse los derechos de autor sobre el mismo. Dentro de éste ubicamos al software libre.

- **Software libre (Free Software).** Es un software que posee una autorización para que cualquier persona u organización pueda usarlo, copiarlo y distribuirlo, ya sea en forma íntegra o con modificaciones, de manera gratuita o mediante un pago significativo. En particular, esto significa que el código fuente debe estar disponible.

El software propietario tiene restricciones en su uso, señaladas por la reserva de los derechos de autor, las patentes o los acuerdos de licencia. Cierta software propietario se distribuye de manera comercial, mientras que otro es gratuito. Con base en los derechos de las licencias, el software propietario se distribuye como software comercial, demoware, shareware, freeware y software de código fuente abierto (que en base a la licencia bajo la cual fue liberado puede convertirse en propietario).

- **Software comercial.** Es un software que se desarrolla por una entidad que tiene la intención de obtener utilidades con su uso. Debemos entender que "comercial" y "propietario" no es lo mismo. La mayoría del software comercial es propietario, pero existe software libre comercial y software no libre no comercial.

- **Demoware.** Cierta software comercial está disponible en versiones de prueba que suelen denominarse “demoware”. El demoware se distribuye en forma gratuita y viene preinstalado en las computadoras nuevas, pero está limitado de algún modo, hasta que es adquirido por el usuario. El software es limitado, funciona una cantidad precisa de días y bajo un número limitado de ejecuciones antes de expirar y solicitar su pago; en este software están inhabilitadas funciones importantes del software en sí.

- **Shareware.** El shareware es un software con una reserva de derechos de autor que se comercializa bajo una política de “testeo anticipado a la compra”. Suele incluir una licencia que permite emplear el software durante un periodo de prueba. Para utilizarlo después de este periodo se debe pagar una tarifa por licencia de registro; a diferencia del demoware, el shareware es un software con todas las funciones.

- **Freeware.** El freeware es un software con una reserva de derechos de autor que se ofrece de manera gratuita. Tiene funciones completas y no requiere un pago para utilizarlo. Debido a que el software esta protegido por derechos de autor, está permitido usarlo, copiarlo y distribuirlo, pero no lo autoriza a alterarlo o venderlo.

- **Software de código fuente abierto (Open Source).** Este tipo de software proporciona a los programadores las instrucciones no compiladas de un programa -código fuente- para que las modifiquen y mejoren. El software de código fuente abierto puede venderse o distribuirse gratuitamente en forma compilada pero, en todos los casos, también debe incluir el código fuente. LINUX es un ejemplo de software con código fuente abierto, igual que FreeBSD, una versión de UNIX diseñada para computadoras personales. Otro ejemplo muy conocido de software con código fuente abierto es OpenOffice.org, una suite con todas sus funciones. A pesar de la falta de restricciones en la distribución y el uso, el software de código fuente abierto tiene derechos de autor y no es de dominio público.

2.10 Software de Código Fuente Abierto: Una alternativa real de gestión para las bibliotecas

Hemos analizado en puntos anteriores las distintas clases de software disponible, la tipología correspondiente al licenciamiento del mismo, las ventajas y bondades ofrecidas por el llamado software libre y open source, las características peculiares de cada uno, y concretamente de los SIGB. A continuación se presenta una lista de soluciones enfocadas a la gestión integral de las bibliotecas y al tratamiento de información digital basadas en software libre / open source. La mayoría del software descrito a continuación se basa en la más reciente tecnología informática, desarrollada sobre interfaces gráficas. Gran parte de éstos implementan la arquitectura cliente-servidor como base del sistema, permitiendo el acceso mediante un Web browser, y están desarrollados en PHP, lenguaje particularmente adaptado para la gestión de aplicaciones que interactúan con base de datos. Otros lenguajes utilizados son el Java, PERL, etc. Una característica importante es que estos programas utilizan otros sistemas de gestión de bases de datos externos al SIGB, como MySQL, Postgres, etc.

Es importante denotar que el gran interés suscitado por el software de código abierto ha dado vida a iniciativas colaterales, sobre todo como promoción de esta tecnología. Muchos de los proyectos citados en este trabajo son alojados en Sourceforge (<http://www.sourceforge.net>), uno de los más grandes "repositorios" de proyectos de software de código abierto en Internet. Sourceforge se consagra como uno de los sitios de alojamiento de proyectos libres, encargándose de alojar, informar, difundir y sensibilizar sobre el software de código abierto. Existen algunos otros sitios más específicos al ámbito bibliotecario, como por ejemplo el portal OSS4LIB (<http://www.oss4lib.org>), que promueve el estudio, uso y aplicación de software libre y de código abierto como alternativa para las bibliotecas.

Emilda (<http://www.emilda.org>)

Es uno de los programas más importantes que han aparecido en los últimos tiempos; se trata de la reedición del software eLib. La iniciativa fue llevada a termino por una empresa de software finlandesa, La Realnode Ltd., especializada en el desarrollo de ICT (Information and Communication Technology); el contacto para Emilda es Mattias Nordstrom. El proyecto tiene soporte financiado por la administración pública de la ciudad de Esbo, Finlandia. Respecto a su predecesor, eLib, Emilda soporta plenamente el MARC (más concretamente el USMARC) en la exportación de datos, así como el Standard Z39.50. El programa consiste en una serie de módulos base (OPAC, módulo de catalogación, módulo de préstamo, módulo de administración del sistema); la interfaz gráfica ha sido concebida por el uso de instituciones educativas.

En diciembre de 2003 apareció la versión 1.0 del programa, a la cual le sigue con poca distancia la 1.1. El programa y la documentación se pueden descargar del sitio oficial de la iniciativa. Está disponible una demostración en línea en la URL <http://demo.emilda.org>. El programa se libera bajo licencia GNU GPL.

Emilda ha sido adaptado actualmente para un grupo de bibliotecarios escolares de la ciudad de Esbo, y parece un producto estable, pero todavía de escasa documentación para el usuario, como lo son manuales, guías, etc.

Koha (<http://www.koha.org>)

Koha constituye una de las primeras iniciativas y uno de los pilares en el campo de los sistemas integrales de gestión bibliotecaria de código libre (ILS *open source*).

La iniciativa empezó en 1999 para hacer frente a la exigencia del Horowhenua Library Trust (HLT), una biblioteca pública de Nueva Zelanda. La realización del programa viene financiada por una firma de informática local, "Katipo Communications Ltd". La política de Katipo sugiere a la comunidad el cierre del

programa como open source: la primera liberación pública del programa con licencia GNU GPL apareció a principios del 2000.

En ese tiempo y todavía la esencia rigidez del soporte MARC (el programa viene concebido por la exigencia del HLT, que no está interesado en el estándar internacional de intercambio bibliotecario) lleva a reconsiderar el código fuente, sobre todo para garantizar el pleno soporte de este estándar. Esto es señal del inicio de una nueva fase para el proyecto Koha, caracterizada por la participación de una gran cola de colaboradores de todas las partes del mundo.

En marzo del 2004 se da otra etapa histórica en el desarrollo del programa con la aparición de la versión 2.0, esta implementa el MARC21 en vez del UNIMARC. El estado actual de Koha es un programa rico en funcionalidad: contiene los módulos clásicos (OPAC, catalogación, préstamo de documentos, administración del sistema), y la aplicación gestiona también las adquisiciones, el control de autoridades, un tesauro y permite la captura de registros bibliográficos transmitidos de clientes Z39.50 (que es el protocolo para transferencia de información). Para la futura versión está previsto también un módulo para la gestión de los fascículos de revistas y la gestión de autoridades en MARC. Koha es un programa multilingüe con traducciones disponibles en inglés, francés, polaco y chino. En el caso de México, podemos descargar la versión UNLP que es una traducción del programa hecha por la Universidad Nacional de la Plata en Argentina.

El programa se distribuye en dos variantes, la nativa para el entorno Linux y la de Microsoft Windows. A pesar de todo Koha, es un programa maduro, con una pequeña falta de homogeneidad en la interfase debido a las sucesivas modificaciones de la primera versión del software.

El proyecto se aloja a la vez en la página Web de Katipo (<http://www.koha.org>), y en Sourceforge (<http://sourceforge.net/projects/koha>). Hay disponible una demostración en línea del programa en <http://koha.org/drive/>.

Como se ha dicho, el programa está hecho con licencia GNU GPL. A pesar del gran esfuerzo realizado, Koha resiente todavía de la falta de un exhaustivo manual para el usuario aplazado por la nueva versión 3.0; la versión 1.x 2 y consecutivas, ya se han cerrado

GNUTeca (<http://gnuteca.codigolivre.org.br>)

GNUTeca es una iniciativa de un equipo de programadores y bibliotecarios de la Universidad UNIVATES, Porto Alegre, Brasil, en el ámbito de un amplio proyecto para el desarrollo de un sistema de gestión integrado de la ciudad.

El módulo GNUTeca se concibió después de un exhaustivo análisis de las exigencias de las bibliotecas de UNIVATES (4 sedes, aproximadamente 30.000 volúmenes). Después de la primera versión pública (versión 1.0), acabada a finales del 2001, se han finalizado otras más, hasta la actual versión 1.4.

GNUTeca comprende los siguientes módulos: WEBOPAC con la posibilidad de reservar y prorrogar préstamos de parte de clientes, catalogación, préstamo de documentos, administración del sistema; la versión 1.4 ya soporta un script multi-idioma. GNUTeca adopta MARC, concretamente MARC21, el estándar bibliográfico que se utiliza en UNIVATES. Una característica del producto, es la posible integración al sistema SAGU, que es una interfase uniforme de cliente. GNUTeca, como el resto de los módulos de SAGU, viene desarrollada por un entorno informático llamado MIOLO, creado originalmente en UNIVATES mismo. Todo el sistema SAGU, GNUTeca comprimido, nace para el entorno Linux y usa PHP como lenguaje de programación, y Postgres o MySQL como motor para la base de datos.

Son diversas las bibliotecas que se han adaptado al sistema, en particular en Brasil. Es rica la documentación que acompaña al programa, sobre todo, aquella destinada para el bibliotecario. GNUTeca se distribuye con licencia GNU-GPL y se puede descargar en la página donde se aloja la iniciativa SGU.

Learning Access (<http://www.learningaccess.org/website/techdev/ils.php>)

Learning Access ILS es un proyecto promovido por alguna institución americana no gubernamental y consiste en un producto derivado (en lenguaje técnico llamado *fork* del ya citado programa Koha, durante la época en que este último no soportaba todavía MARC.) El proyecto se inicia en el año 2000, con la finalidad de proporcionar a las bibliotecas de la comunidad rural de EUA un instrumento compatible con el estándar bibliográfico y catalográfico nacional (MARC21). El proyecto goza desde el comienzo del soporte financiero del Estado de Washington, testeando el software como un proyecto piloto. La realización del programa y su soporte están actualmente bajo el desarrollo del Learning Access Institute.

El programa se caracteriza por los módulos (WEBPAC, catalogación, gestión de autoridades, gestión de préstamo, administración del sistema); en el futuro podría ser desarrollado también el módulo para adquisiciones y para la gestión de publicaciones en serie. El sistema es plenamente compatible con MARC21: soporta este estándar no sólo para los registros bibliográficos sino también para las autoridades, el registro de *holding* (relativo a ejemplares), y el registro *community* (una tipología particular de autoridades de interés para la comunidad local). El software soporta Z39.50, ya sea a nivel de cliente o de servidor, y nace para la gestión de sistema bibliotecario.

El ILS implementa el UNICODE a nivel de menú y de datos, es por esto uno de los pocos ILS de código abierto potencialmente utilizado de forma libre internacionalmente.

La adaptación del programa original al MARC ha ido acompañada de una modificación parcial del código del lenguaje PERL en PHP. El Learning Access ILS usa MySQL como motor para la base de datos. El sistema viene testeado en entorno Linux, pero el entorno en que ha sido escrito tendría que permitir su funcionamiento también en entorno Microsoft Windows.

El programa está en fase de prueba y será distribuido públicamente con licencia GNU GPL sólo después de haber demostrado una suficiente estabilidad. Esta disponible una demostración en línea de cómo funciona con el OPAC en <http://ils.learningaccess.org/>.

OpenBiblio (<http://sourceforge.net/projects/obiblio>)

OpenBiblio fue concebido el primer mes del 2002 y fue registrado oficialmente en el sitio SourceForge desde marzo del mismo año. El creador e iniciador del proyecto es Dave Stevens, *un programador Web o Webmaster* como se le conoce en México. A los pocos meses después del lanzamiento de la iniciativa, el proyecto fue considerado en su totalidad de tipo "*bazaar style*".

El objetivo de OpenBiblio es realizar un sistema para la gestión de bibliotecas que sea fácil de usar, bien documentado, fácil de instalar, concebido para poseer las funcionalidades típicas que se piden a la mayor parte de las bibliotecas escolares y de las bibliotecas públicas, que constituyen el punto privilegiado de la iniciativa.

El programa está todavía en fase de desarrollo, la última versión finalizada, la 0.4 beta, contiene ya todos los módulos base previstos en el proyecto: el OPAC, el módulo para la catalogación, el de préstamo y el de configuración del sistema. La futura versión 1.0 tendrá que implementar: la visualización del estado de los documentos en préstamo mediante el OPAC, el préstamo interbibliotecario, el soporte de USMARC todavía en fase de *exportación* de datos, el *control de autoridades* y la internacionalización del paquete; para la sucesiva versión 2.x se pondrá a prueba incluso el soporte Z39.50. OpenBiblio es un programa multiplataforma escrito en PHP, utilizado tanto en entorno Linux como en Microsoft Windows.

La atractiva interfaz gráfica que usa, su simplicidad y el uso intuitivo han hecho que cuente con un elevado *feedback*, o excelente reputación y demanda como lo podríamos decir en México. Por otra parte, parece ser que no obstante su estado inicial de desarrollo (en el estado de programa "beta"), ya se ha estado adaptando por un cierto número de bibliotecas. Hay disponible una demo en

línea en: <http://obiblio.sourceforge.net/demo/openbiblio/home/index.php>; se puede descargar en el sitio Sourceforge, <http://obiblio.sourceforge.net/>, y está disponible con licencia GNU GPL.

PhpMyBibli (<http://www.pizz.net>)

PhpMyBibli es una iniciativa muy reciente, si consideramos de 2000 a la fecha. Se creó en octubre de 2002 con la versión 1.0 y poco más de un año después apareció la versión 1.2, la cual surgió en mayo de 2004. Tal hecho parece ser debido a la deliberada elección por parte de sus desarrolladores de implementar solo el UNIMARC como formato de datos, con un vacío de potencia de uso y sobre la capacidad de su creador Francois Lemarchand, en el trato con otras colaboraciones.

Es uno de los ILS de código abierto mas completos desde el punto de vista de los módulos que comprende un OPAC: catalogación, préstamo de documentos, gestión de las publicaciones en serie, registro de autoridades, gestión de tesauros y administración del sistema. Recientemente se ha integrado al programa el protocolo Z39.50. El software tiene una interfase de usuario muy detallada y agradable. PhpMyBibli es un programa multilingüe que soporta UNICODE, también a nivel de datos; en la actualidad existen diversas traducciones -algunas todavía parciales- del mismo en lenguas francesa, inglesa, española e italiana. La documentación del producto es muy buena, tanto la del administrador del sistema, o la destinada al encargado de la biblioteca, a pesar que de momento sólo está redactada en lengua francesa.

Hay disponible una demo en línea en: <http://www.pizz.net/PhpMyBibli/>.

Una página informativa sobre el programa también se puede encontrar en el sitio de Sourceforge, <http://sourceforge.net/projects/phpmybibli>. El programa y la relativa documentación están disponibles en el mismo sitio y están distribuidos con licencia GNU GPL. PHPMyBibli es un programa de instalación relativamente fácil y se distribuye en versión para Linux, Microsoft Windows y MacOS. PhpMyBibli ha tenido inmediatamente (y merecidamente) una notable

respuesta en la comunidad bibliotecaria internacional y ha sido adaptado oficialmente por un número amplio de bibliotecas.

PHL (<http://www.elysio.com.br>)

Este software es en cierta forma libre y en cierta forma no, pero se incluyó debido a que reúne las características GPL a nivel 1, como lo explicare a continuación.

Se trata de un programa que entra sólo parcialmente en las licencias previstas de la comunidad código abierto. Se distribuye con una GNU GPL sólo si se usa en una computadora que une el servidor y el cliente al mismo tiempo, mientras que requiere una licencia específica de uso, bajo pago, si se usa como servidor Intranet/Internet. Este tipo de licencia y su relativa limitación la acercan notablemente al programa propietario o de pago como común término. El programa se basa en un motor de gestión de bases de datos llamado WXISIS, que tiene una licencia análoga a ésta, de la aplicación ILS. Este motor parte de la más amplia familia del programa CDS/ISIS, desarrollado a partir de los años ochenta por la UNESCO y del que hablamos en puntos anteriores.

WXISIS en particular ha sido desarrollado por Bireme, una afiliación latinoamericana de la ONU en el campo sanitario. El programa PHL comprende los siguientes módulos, todos basados en tecnología WEB: OPAC, catalogación, préstamo con reserva *en línea* y administración del sistema. PHL parte, como el resto de las aplicaciones de la familia ISIS, de la arquitectura inicial de este programa, concebido como recuperador de información, y presenta, por tanto, las limitaciones en cuanto al uso en el entorno de producción de datos; tiene buena capacidad de gestionar gran cantidad de datos y de obtener excelente tiempo de respuesta en la interrogación.

PHPMYLibrary (<http://sourceforge.net/projects/phpmylibrary>)

Se trata de uno de los primeros programas ILS de código abierto que se publicó libremente. La iniciativa fue registrada en el sitio Web Sourceforge, en noviembre de 2001, cuando fue liberada públicamente la primera versión del producto (versión 1.0); en 2002 el programa entró en la versión 2.x. A partir de

2002 el producto se alarga a otras colaboraciones, mostrando un salto de calidad de la iniciativa. A diferencia de los otros programas de código abierto, PHPMyLibrary adopta la política de sólo liberar versiones estables. En la versión 2.x están presentes los siguientes módulos: WEBPAC con reserva *en línea* de documentos, catalogación, préstamo de documentos, estadísticas, administración del sistema, carga de datos en formato MARC, importación directa de datos producidos con programas CCDS/ISIS. No existe actualmente una versión de demostración en línea del programa, pero es posible visitar la URL <http://phpmylibrary.sourceforge.net/libraries.php>, un catálogo de bibliotecas que han adaptado el software. El entorno informático sobre el cual se basa este programa "*WEB-based*" comprende el lenguaje PHP y el "*Postgres Database Management System*". El software ha sido probado en el entorno Linux y Microsoft Windows.

PhpMyLibrary es históricamente uno de los primeros programas de código abierto estables que soporta MARC, concretamente el USMARC. Esta característica ha favorecido su adaptación respecto a los otros programas de código abierto: en el sitio del proyecto hay un enlace a una lista amplia de bibliotecas, algunas muy prestigiosas, que han implementado el sistema.

El software es liberado con licencia GNU GPL y se puede descargar desde el sitio del proyecto.

Avanti (<http://home.earthlink.net/~schlumpf/avanti>)

Avanti representa sin duda un proyecto único en el ámbito de los programas que trata. Lo caracterizan una serie de aspectos.

Avanti es, en primer lugar, un programa que no usa componentes externos. Mientras las otras aplicaciones descritas aquí se enfocan en la gestión de los datos a base de administrarlas (almacenaje, indización, recuperación de la información, gestión de los procesos, etc.) en programas producidos por otros, Avanti incorpora un *gestor de base de datos* en el objeto desarrollador *ad hoc* y estrechamente compacto. El sistema es comprensivo y muy versátil en cuanto a que ha sido pensado así desde su inicio como programa multiplataforma,

basado enteramente en entorno Java, y puede, por tanto, "girarse" como programa autónomo o como aplicación (*applet*) en el interior de un común *buscador* Web. Avanti es todavía un programa que presta una particular atención al usuario final, sea en el sentido que ha sido concebido como un producto fácil de instalar, sea en el sentido que se distribuye con una rica y exhaustiva documentación, y porque no requiere conocimientos informáticos específicos para el que lo tiene que gestionar. Otros elementos que caracterizan al programa Avanti es el ser desarrollado por una sola persona, Peter Schlumpf, *administrador de sistemas* de la North Suburban Library System de Wheeling, Illinois, EUA. Pero el aspecto más interesante de Avanti está en su arquitectura biblioteconómica, basada en un modelo puramente abstracto de biblioteca. Avanti no implementa reglas o estándares preferidos; éstos están si se configura el programa, pero de forma externa.

La versión 1.0 final del programa tendrá que incluir: un OPAC, un módulo para la catalogación, otro para la gestión del préstamo, otro más para la gestión del sistema; el soporte para MARC y la eventual gestión del Z39.50 en un futuro cercano. Avanti es un programa escrito íntegramente en Java.

MyPHPLibrarian (<http://sourceforge.net/projects/phpmylibrarian>)

El objetivo del proyecto es el desarrollo de un sistema integrado para la gestión de bibliotecas o sistemas bibliotecarios de grandes dimensiones. La iniciativa nace de tres desarrolladores: Jamie Jameson, John Murtha y Mike Wattier, según el registro del proyecto en el sitio Sourceforge. Además de los módulos clásicos (OPAC, catalogación, préstamo de documentos, soporte de MARC), el sistema prevé también funcionalidad ligada al "*e-learning*". Están previstas dos interfaces de usuario: la clásica Web y, para algunos módulos, la GTK. El entorno de desarrollo de la aplicación consiste en PHP y PEARL. La licencia prevista para la distribución del paquete es la Mozilla 1.1.

OSLDS (*Phyteas*)

OSLDS (Open Source Digital Library System) es uno de los primeros y uno de los más ambiciosos proyectos para la realización de un ILS de código abierto. El proyecto nace en 1999 con Jeremy Frumkin, bibliotecario de la University of Arizona, EUA, en la cual coordina el Art Rhyno, sistema bibliotecario de la Leddy Library de la Universidad de Windsor, Canadá, como referencia por la parte informática. El proyecto se configura desde el inicio por la "*bazaar style*".

Inicialmente se estableció crear un sistema mínimo, creado con el menor esfuerzo posible y utilizando un código ya desarrollado preferentemente de alguna de sus colaboraciones. Después de la primera tentativa que ven parecida a un sistema coherente, basado en código escrito en diversos lenguajes de programación, un punto debatido en la lista de discusión del proyecto lleva a definir los objetivos y los métodos de la iniciativa: se decide por un sistema de alto aprovechamiento, rico en funcionalidad biblioteconómica, como, por ejemplo, la gestión de los registros de autoridades, que soportan la más avanzada tecnología de la información (XML, web semántica, etc.) y la relación de un sistema modular, basado en *plug-in* (módulo externo que interactúa con un programa "central" que constituye el corazón del sistema) y que la comunidad de código abierto, con su complejidad, la ha hecho disponible.

La definición de los objetivos viene acompañada de una definición de sus múltiples colaboradores que no se reconocen y no se ponen de acuerdo sobre el proyecto. Esto lleva, sobre todo entre el 2000 y el 2001, a un fuerte retraso de la iniciativa para la distribución de algunas versiones de la edición del código fuente. En abril de 2002 Rhyno distribuye la última versión del software pero sobre todo, pretende librarse de la iniciativa, despidiéndose temporalmente de ella. El despido de Rhyno parece una señal del final del proyecto.

La experiencia OSDL es importante sobre todo para la elaboración teórica y la solución tecnológica individual en el desarrollo de un ILS de *código abierto*, cabe decir que muchas de las soluciones propuestas por Phyteas han sido

adaptadas por los proyectos sucesivos aparecidos en el campo de los ILS ", una fuente abierta" o mejor dicho programas de código abierto.

Projecte Free Library (<http://sourceforge.net/projects/freelibrary>)

El proyecto prevé el desarrollo de un sistema dotado de los módulos OPAC, catalogación, gestión de registro de autoridades, préstamo de documentos con gestión de usuarios, reservas y reclamaciones; está previsto también el soporte MARC, tanto en fase de *importación* como en *exportación* de datos. Hecho por cuatro voluntarios (Alexis Castillo, Álvaro H. Buitrago Giraldo, Ander Said Bazurto y Gustavo A. Barretto), el proyecto ha sido registrado en Sourceforge en diciembre de 2000. No parece que haya sido aplazado del repositorio después de la fecha indicada. Es muy probable que la iniciativa no haya tenido código fuente hasta ahora; la licencia prevista para la distribución del programa es la GNU GPL.⁷⁴

Aplicaciones para Servicios de Información Digital

En el mundo de la publicación académica, hoy en día existe un movimiento conocido como Open Access, el cual busca crear las condiciones para que cualquier usuario pueda leer, descargar, buscar, distribuir e imprimir textos completos de artículos académicos y e-prints, sin barreras financieras, legales o tecnológicas, más las que supone Internet en sí misma.

Para cumplir con los objetivos de este movimiento se han desarrollado una serie de estándares y herramientas basadas en software libre que permiten levantar Repositorios de Recursos Digitales (Open Archives) de manera rápida y con pocos medios. Este tipo de aplicaciones son numerosas y se encuentran gratuitamente en Internet.

⁷⁴ STURMAN, Robi. "Software open source per la gestione integrata delle biblioteche: una nuova risorsa?". En: *Bollettino AIB* N° 3 p.257-270.

Greenstone (<http://www.greenstone.org/>). Es una aplicación que permite la creación y utilización de una biblioteca digital con sus respectivas colecciones. Fue desarrollado por la Universidad de Waikato de Nueva Zelanda, y se distribuye bajo una licencia GNU. Sus requerimientos son los siguientes:

- Se ejecuta en plataformas Windows, Linux y Macintosh OS X, así como con un servidor Apache.
- Utiliza GDBM como manejador de bases de datos, además de Java Runtime Environment para su interfaz gráfica.
- Es compatible con varios estándares de metadatos, entre ellos MARC y Dublin Core, y con el protocolo Z39.50.

Dspace (<http://www.dspace.org/>). Es una aplicación orientada a la gestión de repositorios digitales, creada por el MIT y Hewlett-Packard en el año 2002. Se distribuye bajo una licencia Open Source (BSD Berkeley).

Esta herramienta permite la gestión de diferentes tipos de materiales digitales, al tiempo que organiza la producción científica de una institución. Sus características son:

- Trabaja sobre plataformas Unix, así como con un servidor Web Apache, Tomcat y PostgreSQL.
- Los registros sobre los contenidos son decodificados en el esquema de metadatos Dublin Core y MARC.
- Es capaz de interactuar con cualquier otro sistema que maneje el protocolo OAI-PMH.

E-Prints (<http://www.eprints.org/>). Se trata de un sistema para la creación de repositorios institucionales de producción científica. Fue creado por la Universidad de Southampton en el año 2000. Es capaz de manejar diversos tipos de documentos y formatos de datos, al tiempo que proporciona acceso abierto a sus contenidos. Sus características son:

- Trabaja sobre plataformas Linux y Macintosh OS X, así como con un servidor Web Apache y MySQL.
- Utiliza el formato de metadatos Dublin Core.
- Es capaz de interactuar con cualquier sistema de información que utilice el protocolo OAI-PMH.

Fedora (<http://www.fedora-commons.org/>). Es un sistema flexible para la gestión de repositorios digitales, capaz de administrar de forma eficiente hasta un millón de objetos.

Cuenta con tres interfaces: una para administrar el repositorio, acceso a la información en Web. Fue desarrollado por la Universidad de Cornell en el año 2001 y actualmente es utilizado en proyecto de repositorios de 3R de la UNAM (<http://www.3r.unam.mx/>). Sus características son las siguientes:

- Trabaja sobre cualquier plataforma Unix, así como con cualquier servidor Web que cuente con TomCat, McKoiSQL y Java SE Development Kit.
- Utiliza XML para decodificar los metadatos, mismos que pueden estar asentados bajo cualquier esquema.
- Interactúa con cualquier otro sistema que utilice el protocolo OAI-PMH.⁷⁵

⁷⁵ ZURITA SÁNCHEZ, Juan Manuel. *Op. cit.*

Son numerosas las aplicaciones disponibles que se desarrollan constantemente bajo el esquema de cooperatividad entre usuarios, para la mejora constante de cualquier software respaldado en las libertades ofrecidas por el software libre, en este caso aplicado a las bibliotecas. El software libre está impactando positivamente en el ámbito bibliotecario, partiendo de la reducción significativa de costos e infraestructura que conlleva un proyecto de automatización para bibliotecas; a esto le sumamos la disponibilidad del código fuente que permite la modificación del programa en términos de adaptación a las necesidades locales de la biblioteca, de la interfaz que estará disponible tanto a nuestros usuarios, como para el personal encargado de los procesos llevados a cabo, así como la posibilidad de integración con otras aplicaciones de gestión y/o bases de datos, entre muchas otras ventajas, que enmarcan al software libre y de código abierto como una alternativa sólida y viable para la informatización de las bibliotecas.

Es necesario ubicar este tipo de software más allá de un entorno meramente tecnológico, que si bien es importante y fundamental, prescinde de un entorno cultural basado en valores ético-filosóficos, que promueven el desarrollo libre de estas alternativas, configurando instrumentos óptimos que contribuyen notablemente a la puesta en marcha de la misión social, que la biblioteca es convocada a desarrollar.

En el siguiente capítulo se determinarán las características y funcionalidades del sistema integral de gestión bibliotecaria de tipo libre (código abierto), Koha, con el objetivo de crear una alternativa al software propietario, para quien así lo necesite y/o requiera.



CAPÍTULO 3

Koha: Sistema Integrado de Gestión Bibliotecaria Instalación y funcionalidad

3. Koha: Sistema Integrado de Gestión Bibliotecaria, instalación y funcionalidad

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 4 años, para darnos cuenta 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, enfocadas al desarrollo y mejora de Sistemas Integrados para la Gestión en Bibliotecas, que se traduzcan en beneficios directos para las mismas bajo un ambiente de libertades donde usuarios e instituciones sean los únicos beneficiados.

Uno de los sistemas más robustos creados bajo este esquema, es el SIGB Neo Zelandés llamado “Koha”, que descansa su estructura 100% sobre software libre, y el cual, a través de los últimos años, se ha convertido en el Sistema Integrado para Bibliotecas de carácter abierto, más completo, ofreciendo un paquete de calidad para cualquier institución que desee implementarlo.

Ante esto, se intentará profundizar en la funcionalidad del sistema, comenzado con un manual de instalación detallado para cada una de las distintas plataformas compatibles con el mismo, así como un análisis de los distintos módulos que lo integran, no sin antes realizar una introducción donde se revisan las características generales del software, desde su historia, hasta los requerimientos técnicos necesarios para llevar a cabo su implementación. Todo con el objetivo de promover este sistema y crear una alternativa al software propietario, para quien así lo necesite y/o requiera.

3.1 ¿Qué es Koha?

Koha es un sistema integrado de gestión bibliotecaria multiplataforma, basado en tecnologías Web. Es el primer sistema para la gestión en bibliotecas de código fuente abierto, liberado bajo la licencia GNU/GPL. La palabra “Koha” proviene del maorí, que se interpreta como obsequio o donación.

3.1.1 Historia

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. Nace 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 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 de 2000 ya estaba lista la primera versión. Ante esto, Katipo y la HTL determinaron liberar el sistema bajo la licencia GNU/GPL, abriendo posibilidades de beneficio y desarrollo para otras bibliotecas.



En enero de 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, contratando a Katipo para el soporte. En 2002 Paul Polain (Marsella, Francia), patrocinado por la Nelsonville Public Library de Ohio, comenzó a 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 llamada Liblime, con base en Ohio, 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.

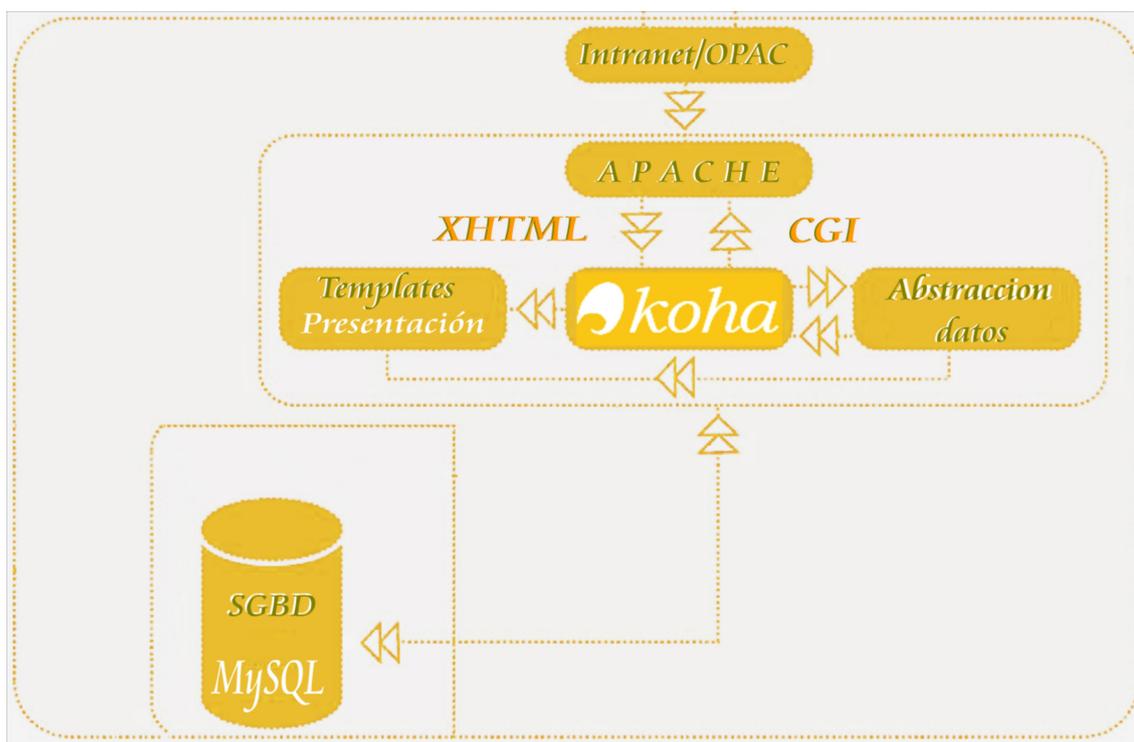
3.1.2 Características

Koha incluye todas las características previstas para un sistema integrado de bibliotecas; el sistema basa su funcionamiento en una estructura cliente-servidor (Web) soportada mediante una arquitectura LAMP (Linux, Apache, MySQL, PHP y/o Perl), ofreciendo 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.

El sistema comprende una serie de módulos, como son: administración, adquisiciones, catalogación, circulación, catálogo público en línea, autoridades, publicaciones periódicas, usuarios, e informes.

El sistema se conecta a un servidor Web local para mostrar las dos interfaces disponibles y utiliza MySQL para gestionar las distintas bases de datos que contiene.

De manera gráfica, el sistema funciona de la siguiente manera:



3.1.2.1 Sistema integrado e integrable. Koha es un sistema integrado, que se traduce como multifuncional, incluyendo diversas funciones de gestión, dentro de un mismo software. El sistema es altamente adaptable y configurable para cualquier tipo de biblioteca, permitiendo incluso la comunicación con otros sistemas informáticos; actualmente Koha para Linux permite la búsqueda e importación de registros bibliográficos de diversos catálogos de bibliotecas a través del protocolo Z39.50, sin que éstas cuenten necesariamente con el mismo sistema para gestionar sus procesos. Además, permite la integración con el sistema informático del sitio AMAZON, para visualizar imágenes de libros y es posible también crear nuevas interfaces entre Koha y cualquier sistema de información como sitios Web, bases de datos, etc.

3.1.2.2 Multiplataforma. Koha puede instalarse en diversos sistemas operativos más allá de plataformas Linux, como MS Windows y Solaris. No es necesario contar con equipos de cómputo de alta gama, e incluso se puede ejecutar el sistema desde máquinas virtuales (virtualbox). El sistema corre a través de cualquier navegador gráfico, por lo cual la biblioteca puede acceder al

mismo utilizando distintas computadoras, cada una de ellas con distintos sistemas operativos, sin ningún problema.

3.1.2.3 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 a los que el usuario puede definir 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 a 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 avisa al bibliotecario cuando el mismo es devuelto y quién lo ha reservado para un siguiente préstamo.

3.1.2.4 Desarrollo

Koha es desarrollado por una comunidad internacional que cada día se extiende más y en la cual convergen informáticos, bibliotecarios, traductores, usuarios comunes, desarrolladores clave del sistema y colaboradores casuales, todos bajo un entorno de cooperatividad que se centra en mejorar día con día las capacidades de este sistema, sin intenciones comerciales o corporativas.

Como proyecto GNU, la toma de decisiones está organizada por medio de GNU Savannah. Como administradores de dicho proyecto se encuentran Chris Cormack, Paul Poulain y Joshua Ferraro, quienes colaboran en el desarrollo además de coordinar a los participantes y/o miembros para asignarles sus tareas. Las versiones actuales y estables de Koha son la 2.2.9 y 3.0.0.

3.1.2.5 Documentación y soporte

Koha cuenta con diversos instrumentos que tienen como objetivo ofrecer soporte a los usuarios del sistema, así como una serie de documentos en distintos idiomas (inglés, francés, español, chino, japonés e italiano), y en diversos formatos (PDF, XML, HTML), que abarcan desde guías básicas de instalación para diversos sistemas operativos, mantenimiento continuo, migración de bases de datos, hasta personalización del sistema, solución de problemas específicos, ayuda en línea, traducciones, etc. Dicha documentación se puede encontrar en los sitios: <http://wiki.koha.org/> y <http://www.kohadocs.org>.

Una parte importante para establecer contacto entre la comunidad, son los canales de comunicación que Koha ofrece, como las listas de correo y chat en:

- koha@lists.katipo.co.nz
- Koha-devel@lists.koha.org
- Koha-win32@lists.koha.org
- Koha-es@yahoogroups.com
- [Irc.katipo.co.nz: 6667#koha](irc://irc.katipo.co.nz:6667/#koha)

El portal oficial también ofrece fechas de eventos relacionados con el desarrollo del sistema, y dos herramientas alternas para usuarios: una GitWeb para el control de versiones y un sistema de seguimiento de errores basado en Web, llamado “Bugzilla”.

3.1.2.6 Requerimientos técnicos

Requerimientos de Software:

- Servidor Web Apache.
- MySQL Server 4.1 o superior.
- Interprete Perl.
- Koha Tarball o aplicación MSI según convenga.

- Módulos exclusivos de Perl para Linux o para el sistema operativo base, según sea el caso.
- Navegador Web (recomendable Mozilla Firefox).
- Distribución Linux (Debian, Ubuntu, SUSE, Fedora).
- Windows XP SP2-3.

Requerimientos de Hardware

Para un óptimo desempeño del sistema se recomienda:

- Procesador a 1.5 GHz o superior.
- Disco duro de 40 GB.
- Resolución mínima de pantalla 1024x768 pixeles a 16 bits de profundidad en colores.

Koha puede ser descargado desde el sitio oficial en <http://koha.org> contando con la posibilidad de probar versiones demostrativas a través de Liblime en: <http://www.liblime.com/demos>

3.2 Instalación de Koha 3.0 en Ubuntu Jaunty Jackalope 9.04

Ubuntu es una distribución GNU/LINUX que ofrece un sistema operativo predominantemente enfocado a computadoras de escritorio, aunque también proporciona soporte para servidores.

Basada en Debian GNU/Linux, Ubuntu concentra su objetivo en la facilidad y libertad de uso, así como en las actualizaciones constantes (cada 6 meses) y la facilidad en la instalación. Ubuntu es patrocinado por Canonical Ltd., una empresa privada fundada y financiada por el empresario sudafricano Mark Shuttleworth.

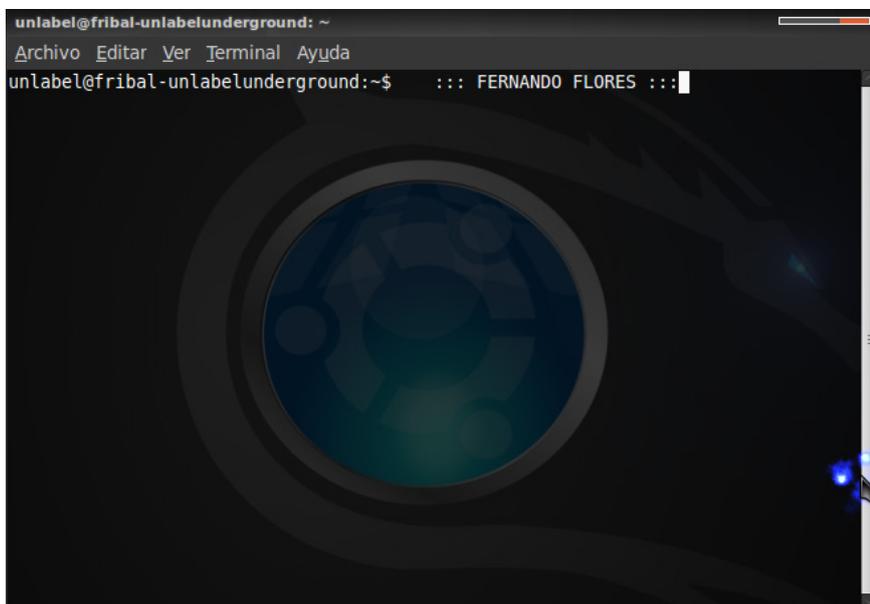
El eslogan de Ubuntu – “Linux para seres humanos” (en inglés "Linux for Human Beings") – resume una de sus metas principales: hacer de Linux un sistema operativo más accesible y fácil de usar.

3.2.1 Instalación de Ubuntu 9.04

La siguiente guía para la instalación de Koha 3.0 es aplicable únicamente utilizando la versión 9.04 de Ubuntu, el cual puede ser descargado gratuitamente del sitio oficial de la distribución (<http://www.ubuntu.com/getubuntu/download>). Una vez descargado el CD/DVD, necesitaremos crear una imagen .ISO al disco correspondiente y proceder con la instalación. Como anexo a esta tesina, se adjunta una guía para llevar a cabo una instalación limpia de Ubuntu, con la capacidad de crear un boteo dual si se trabaja con cualquier versión de Windows, y así tener la posibilidad de utilizar los dos sistemas operativos en un mismo equipo de escritorio. Esto tendrá como ventaja principal para los usuarios que así lo deseen, la posibilidad de utilizar Ubuntu sin perder sus datos valiosos almacenados en la partición de Windows (x versión), e irse adaptando a un nuevo sistema operativo, gratuito y altamente funcional.

3.2.2 Línea de comandos para instalación de Koha en Ubuntu

Para llevar a cabo la serie de pasos incluidos en esta guía de instalación, será necesario acceder a la línea de comandos del sistema. En Gnome (escritorio por defecto en Ubuntu) podemos acceder a la consola desde *Aplicaciones>Accesorios>Terminal*.



Es recomendable crear un ícono de acceso directo a la terminal; podemos hacerlo de dos formas:

- a) *Aplicaciones>Accesorios>Terminal* donde arrastraremos el ícono a cualquier panel disponible en nuestro escritorio.



- b) Crear un acceso directo en el Dockbar, simplemente arrastrando el ícono a la barra.

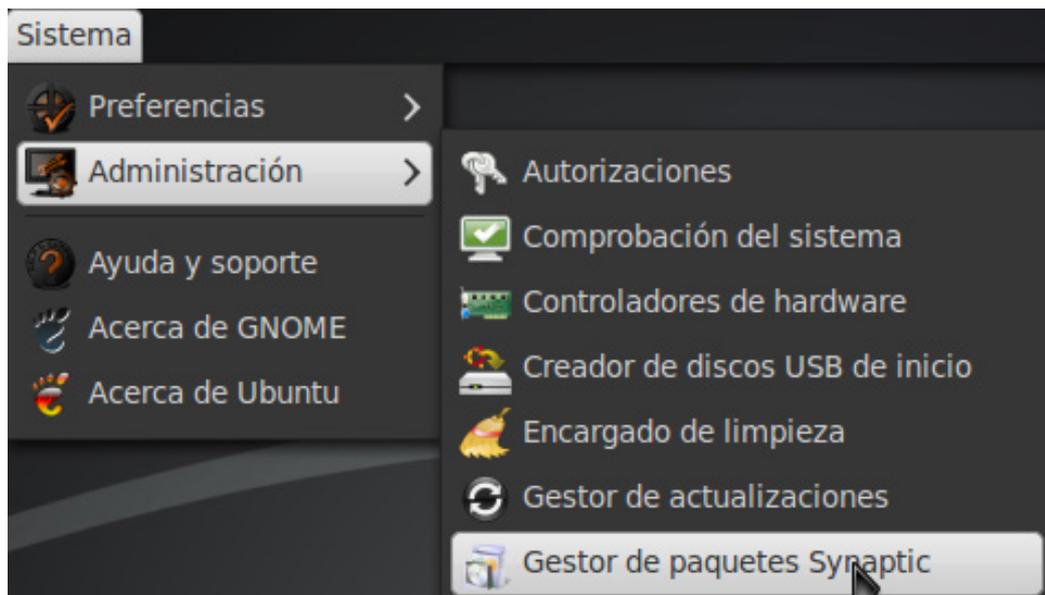


3.2.3 Servidor LAMP

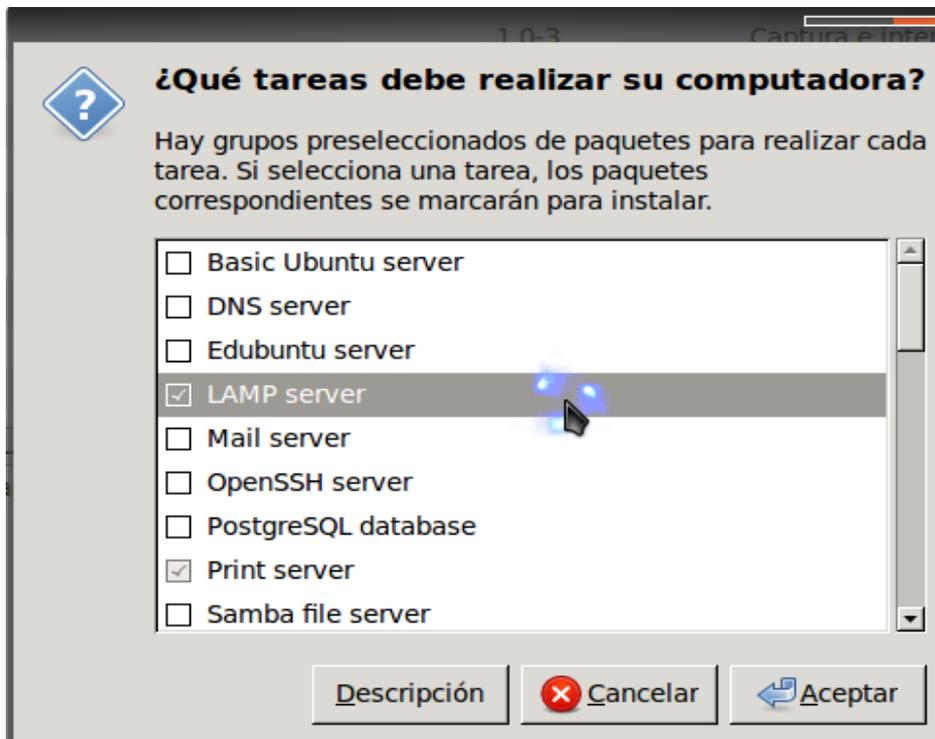
La base para el correcto funcionamiento de Koha se centra en su arquitectura LAMP. Un servidor LAMP es aquel en el que se combinan los siguientes elementos: **L**inux + **A**pache + **M**ySQL + **P**HP (Se puede extender a Perl y/o Python) Es en términos simples; un servidor Web con soporte para páginas dinámicas y bases de datos relacionales, montado sobre una plataforma Linux.

Apache es el servidor Web HTTP más usado en el mundo, es de código abierto y viene por defecto en los repositorios de Ubuntu. PHP es uno de los lenguajes de programación más usados para el desarrollo de páginas Web, y MySQL es un gestor de bases de datos relacionales muy completo y que se encuentra bajo licencia GPL.

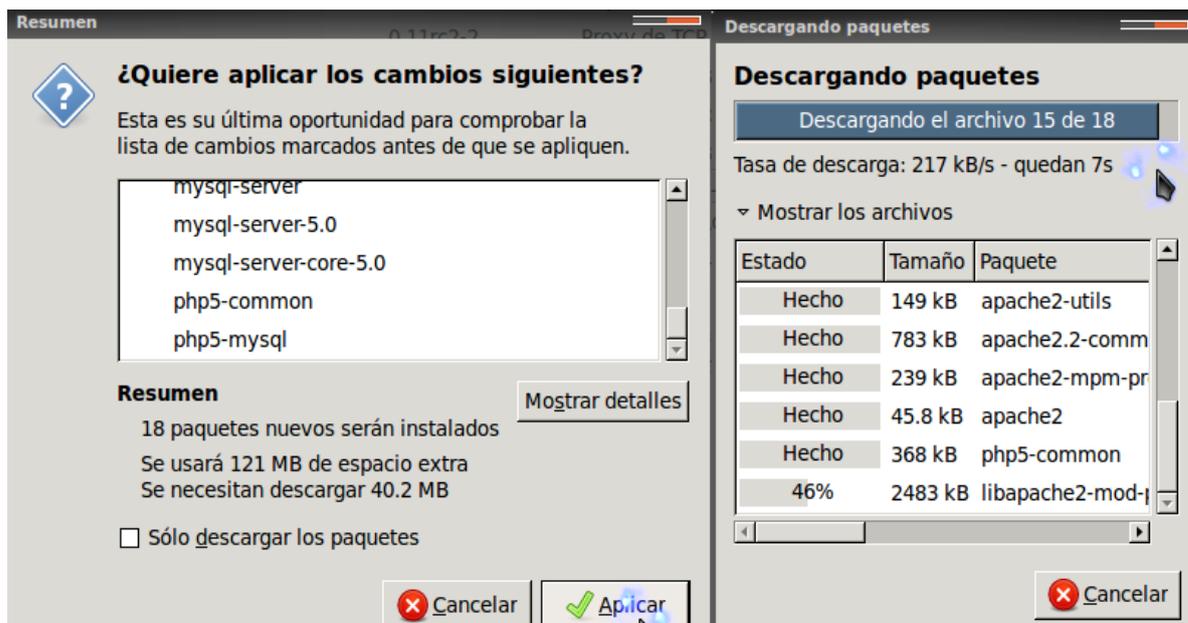
Existen diversas formas de montar un servidor LAMP en nuestra distribución, ya sea a través de la consola o de manera más gráfica mediante el gestor de paquetes Synaptic, sobre el cual nos basaremos. Accedamos desde *Sistema>Administración>Gestor de paquetes Synaptic*.



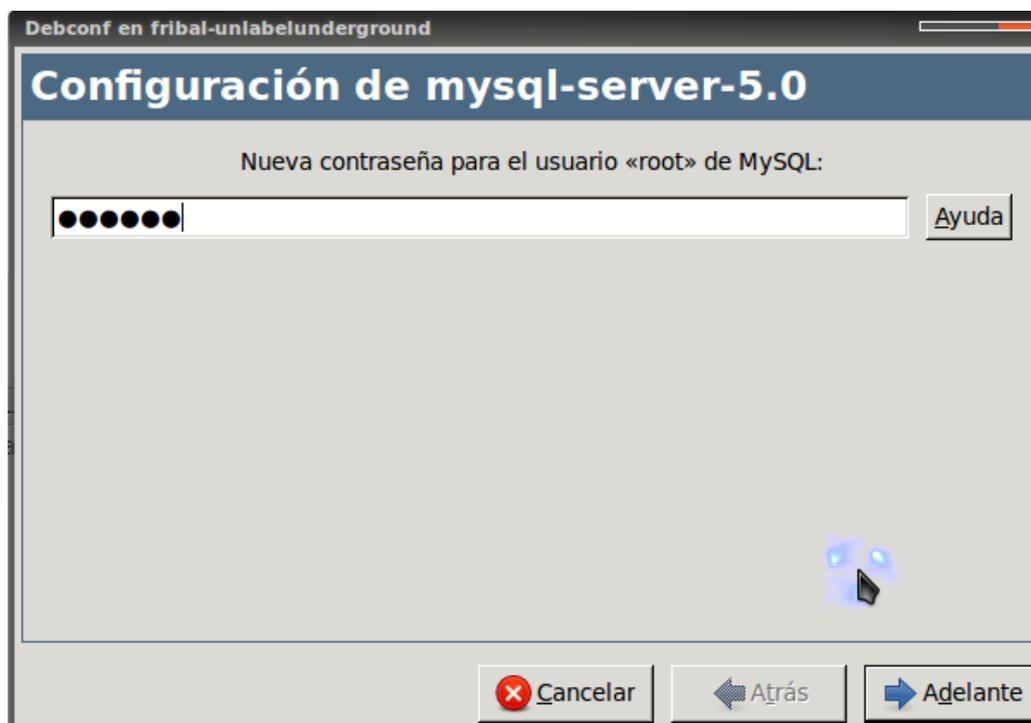
Dentro de Synaptic, en el menú editar, seleccionamos “marcar paquetes por tarea”, seleccionamos LAMP Server en el cuadro de opciones.



Hacemos clic en *aceptar* y después en *aplicar*. El sistema comenzará la descarga e instalación del soporte LAMP.



Como parte del proceso de instalación, Ubuntu te pedirá que asignes una contraseña para la cuenta de administrador (root) de MySQL, es importante que recuerdes esta contraseña ya que posteriormente será necesaria para configurar los permisos necesarios a usuarios y bases de datos en MySQL.



Una vez finalizada la descarga e instalación Ubuntu tendrá el soporte requerido por Koha para funcionar como servidor Web mediante Apache, y gestionar bases de datos relaciones mediante MySQL, soportando PHP.

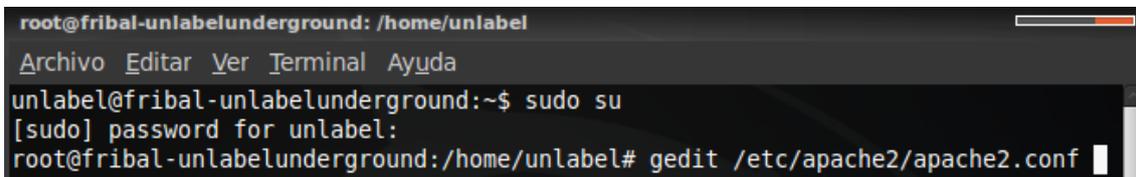
3.2.4 Configurando Apache Server

Es necesario realizar algunos cambios a la configuración del servidor Web. Para esto abrimos una terminal e ingresamos como “super usuario”, también conocido como “root” o administrador del sistema, con lo cual tenemos todos los privilegios para realizar cambios y configurar distintas aplicaciones dentro de Ubuntu. Lo hacemos con el siguiente comando y la contraseña correspondiente:

```
sudo su
```

Lo que haremos es dar un nombre al servidor Web, editando el archivo de configuración de Apache con “gedit”.

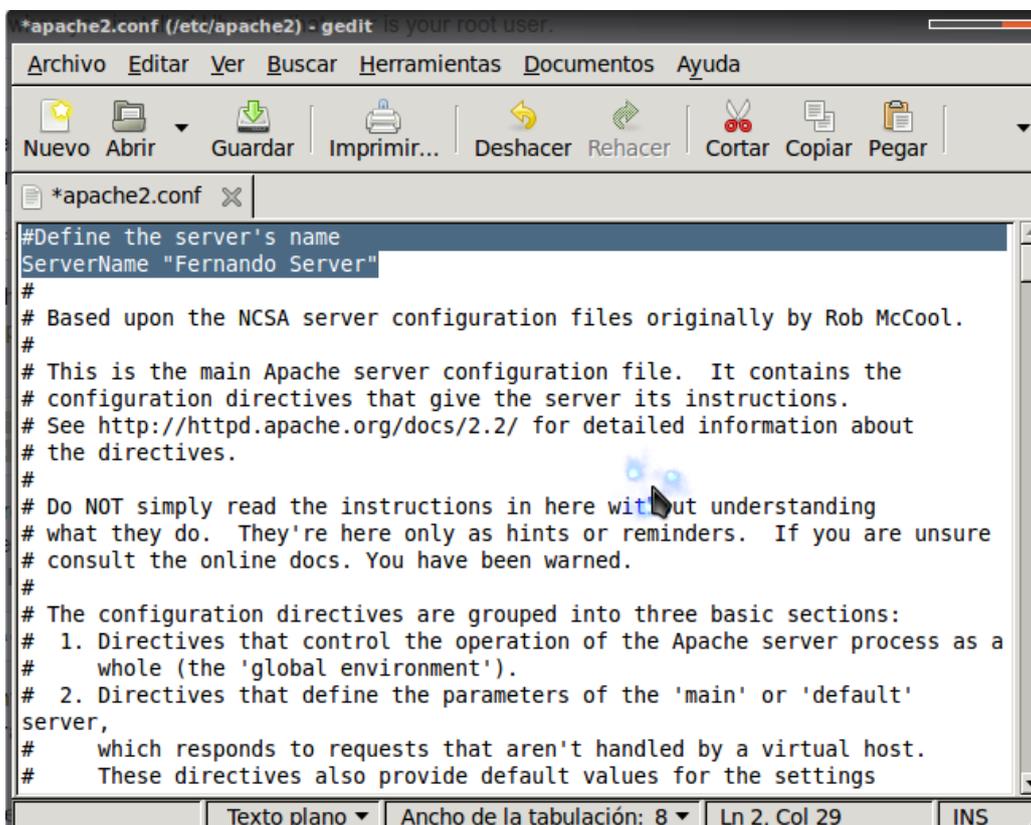
```
sudo gedit /etc/apache2/apache2.conf
```



```
root@fribal-unlabelunderground: /home/unlabel
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ sudo su
[sudo] password for unlabel:
root@fribal-unlabelunderground: /home/unlabel# gedit /etc/apache2/apache2.conf
```

Con esto abriremos el archivo de configuración de Apache en un editor de texto. Agregamos las siguientes líneas al inicio del archivo, lo guardamos y cerramos. Reemplazaremos la parte de “YourSiteName” con el nombre que le daremos a nuestro servidor Koha; en mi caso lo llamaré “Fernando Server”.

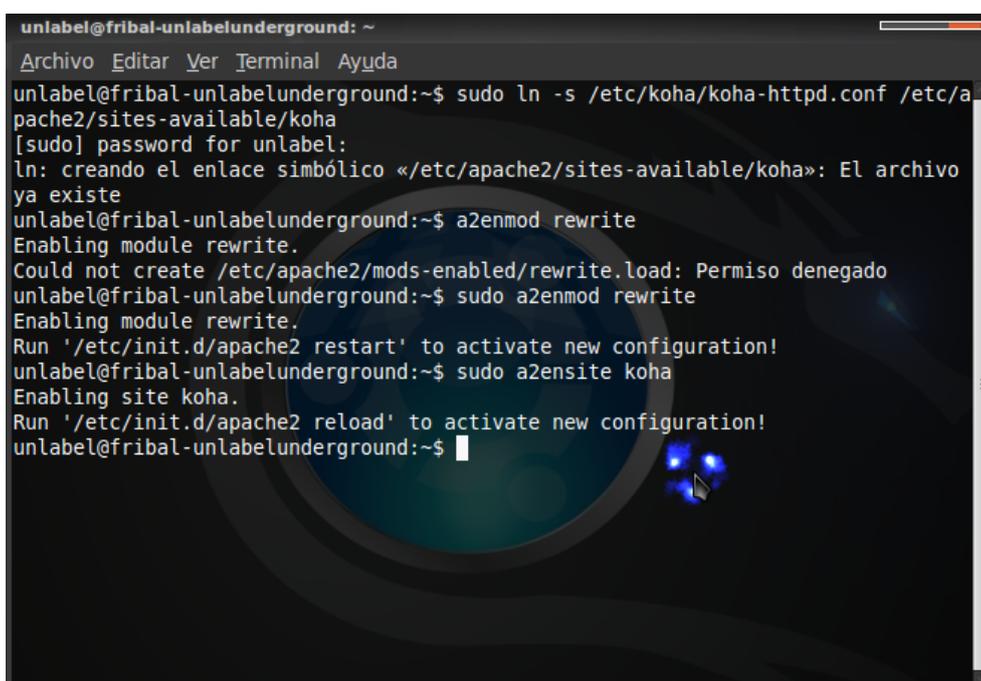
```
#Define the server's name
ServerName "YourSiteName"
```



```
*apache2.conf (/etc/apache2) - gedit is your root user.
Archivo Editar Ver Buscar Herramientas Documentos Ayuda
Nuevo Abrir Guardar Imprimir... Deshacer Rehacer Cortar Copiar Pegar
*apache2.conf x
#Define the server's name
ServerName "Fernando Server"
#
# Based upon the NCSA server configuration files originally by Rob McCool.
#
# This is the main Apache server configuration file. It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.2/ for detailed information about
# the directives.
#
# Do NOT simply read the instructions in here without understanding
# what they do. They're here only as hints or reminders. If you are unsure
# consult the online docs. You have been warned.
#
# The configuration directives are grouped into three basic sections:
# 1. Directives that control the operation of the Apache server process as a
#    whole (the 'global environment').
# 2. Directives that define the parameters of the 'main' or 'default'
#    server,
#    which responds to requests that aren't handled by a virtual host.
#    These directives also provide default values for the settings
```

Lo siguiente será reiniciar Apache; si lo hemos hecho bien no tendremos mensajes de error. Lo hacemos con los siguientes comandos:

```
sudo /etc/init.d/apache2 restart
sudo /etc/init.d/apache2 reload
```



```
unlabel@fribal-unlabelunderground: ~
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ sudo ln -s /etc/koha/koha-httpd.conf /etc/a
pache2/sites-available/koha
[sudo] password for unlabel:
ln: creando el enlace simbólico «/etc/apache2/sites-available/koha»: El archivo
ya existe
unlabel@fribal-unlabelunderground:~$ a2enmod rewrite
Enabling module rewrite.
Could not create /etc/apache2/mods-enabled/rewrite.load: Permiso denegado
unlabel@fribal-unlabelunderground:~$ sudo a2enmod rewrite
Enabling module rewrite.
Run '/etc/init.d/apache2 restart' to activate new configuration!
unlabel@fribal-unlabelunderground:~$ sudo a2ensite koha
Enabling site koha.
Run '/etc/init.d/apache2 reload' to activate new configuration!
unlabel@fribal-unlabelunderground:~$
```

3.2.5 Webmin

Webmin es una herramienta de configuración de sistemas, accesible vía Web, y compatible con GNU/Linux. Con esta aplicación es posible configurar aspectos internos del sistema, como usuarios, cuotas de espacio, servicios, archivos de configuración, apagado del equipo, etcétera, así como modificar y controlar muchas aplicaciones libres, como el servidor Web Apache, PHP, MySQL DNS, entre otros. Webmin está escrito en Perl, versión 5, ejecutándose como su propio proceso y servidor Web. Por defecto, se comunica a través del puerto TCP 10000 y puede ser configurado para usar SSL si OpenSSL está instalado con módulos de Perl adicionales requeridos.

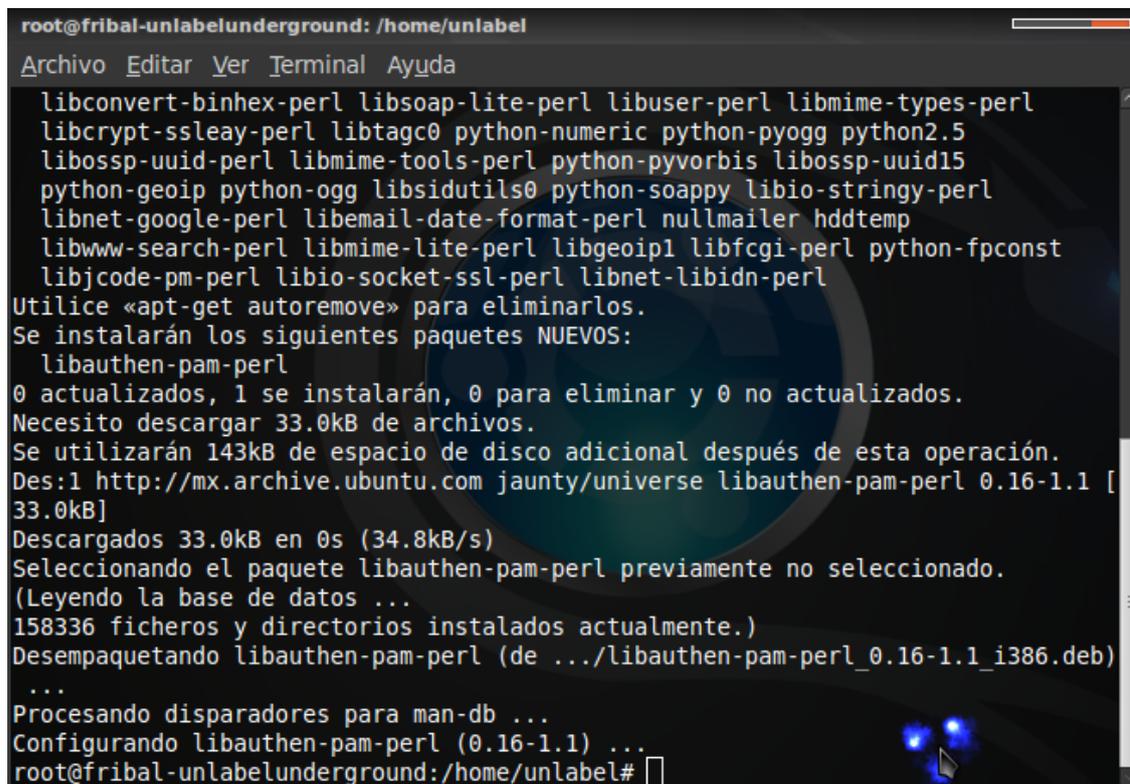
Utilizaremos Webmin para asignar permisos hacia las bases de datos y usuarios que utilizara Koha en MySQL.

3.2.5.1 Instalando Webmin

Lo primero será instalar las librerías Perl necesarias. Para esto ejecutamos los siguientes comandos en la consola:

```
sudo su
apt-get install libnet-ssleay-perl
apt-get install libauthen-pam-perl
apt-get install libio-pty-perl
apt-get install libmd5-perl
apt-get install openssl
```

El sistema comenzará con la descarga e instalación de las librerías necesarias desde los repositorios de Ubuntu.



```
root@fribal-unlabelunderground: /home/unlabel
Archivo Editar Ver Terminal Ayuda
libconvert-binhex-perl libsoap-lite-perl libuser-perl libmime-types-perl
libcrypt-ssleay-perl libtagc0 python-numeric python-pyogg python2.5
libossp-uuid-perl libmime-tools-perl python-pyvorbis libossp-uuid15
python-geoip python-ogg libsidutils0 python-soappy libio-stringy-perl
libnet-google-perl libemail-date-format-perl nullmailer hddtemp
libwww-search-perl libmime-lite-perl libgeoip1 libfcgi-perl python-fpconst
libjcode-pm-perl libio-socket-ssl-perl libnet-libidn-perl
Utilice «apt-get autoremove» para eliminarlos.
Se instalarán los siguientes paquetes NUEVOS:
 libauthen-pam-perl
0 actualizados, 1 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 33.0kB de archivos.
Se utilizarán 143kB de espacio de disco adicional después de esta operación.
Des:1 http://mx.archive.ubuntu.com jaunty/universe libauthen-pam-perl 0.16-1.1 [
33.0kB]
Descargados 33.0kB en 0s (34.8kB/s)
Seleccionando el paquete libauthen-pam-perl previamente no seleccionado.
(Leyendo la base de datos ...
158336 ficheros y directorios instalados actualmente.)
Desempaquetando libauthen-pam-perl (de ../libauthen-pam-perl_0.16-1.1_i386.deb)
...
Procesando disparadores para man-db ...
Configurando libauthen-pam-perl (0.16-1.1) ...
root@fribal-unlabelunderground:/home/unlabel#
```

Lo siguiente será descargar e instalar Webmin 1.441 desde la consola. Para esto ejecutaremos cada línea del siguiente comando:

```
cd /
mkdir build
cd build
wget http://downloads.sourceforge.net/webadmin/webmin_1.441_all.deb
dpkg -i webmin_1.441_all.deb
```

Si todo ha salido bien, tendremos el siguiente mensaje en la terminal.

```
Webmin install complete. You can now login to https://fribal-unlabelunderground:10000/
as root with your root password, or as any user who can use sudo
to run commands as root.

root@fribal-unlabelunderground:/build#
```

Una vez terminada la instalación accederemos a Webmin mediante el puerto 10000 en Firefox; la URL estará determinada por el nombre de nuestro equipo (computadora) y/o también podremos usar la siguiente dirección: `https://localhost:10000`. La primera vez que accedamos a Webmin, el navegador mostrará un mensaje de alerta, lo cual es completamente normal y hace referencia al hecho de que el mismo establecerá conexión con un equipo sin certificado digital o “untrusted”. En este caso, el mensaje no es de gran ayuda ya que el equipo es el propio, y para resolverlo agregaremos una excepción.



Fallo en conexión segura

Un error ocurrió durante una conexión a `fribal-unlabelunderground:10000` porque usa un certificado de seguridad no válido.

El certificado no es confiable porque es auto firmado.
El certificado solamente es válido para *.

(Código de error: `sec_error_untrusted_issuer`)

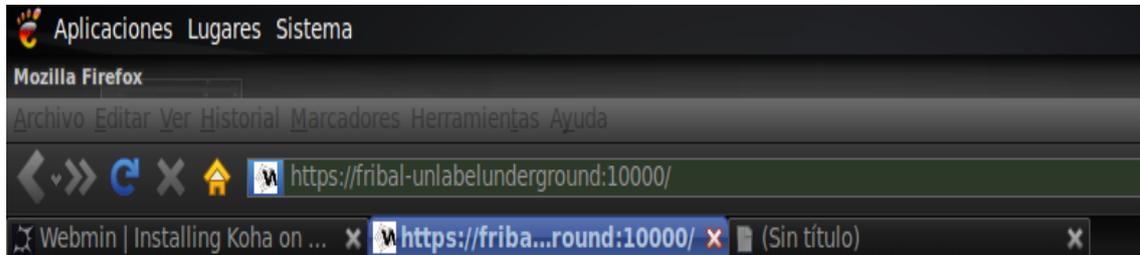
- Ésto podría ser un problema con la configuración del servidor o podría ser alguien tratando de hacerse pasar por el servidor.
- Si se ha conectado a este servidor exitosamente en el pasado, el error puede ser temporal y puede intentarlo nuevamente más tarde.

No debería agregar una excepción si está usando una conexión a internet en la que no confía por completo o si no está acostumbrado a ver mensajes de advertencia de este servidor.

Sáquenme de aquí

Agregar excepción

Una vez hecha la excepción, podremos acceder a la interfaz con privilegios de administrador (explicaremos las opciones de configuración más adelante).

A screenshot of the Webmin login form. The form is titled 'Login to Webmin' and contains the following elements:

- A message: 'You must enter a username and password to login to the Webmin server on fribal-unlabelunderground.'
- A 'Username' field containing the text 'FERNANDO'.
- A 'Password' field with masked characters (dots).
- 'Login' and 'Clear' buttons.
- A checkbox labeled 'Remember login permanently?' which is currently unchecked.

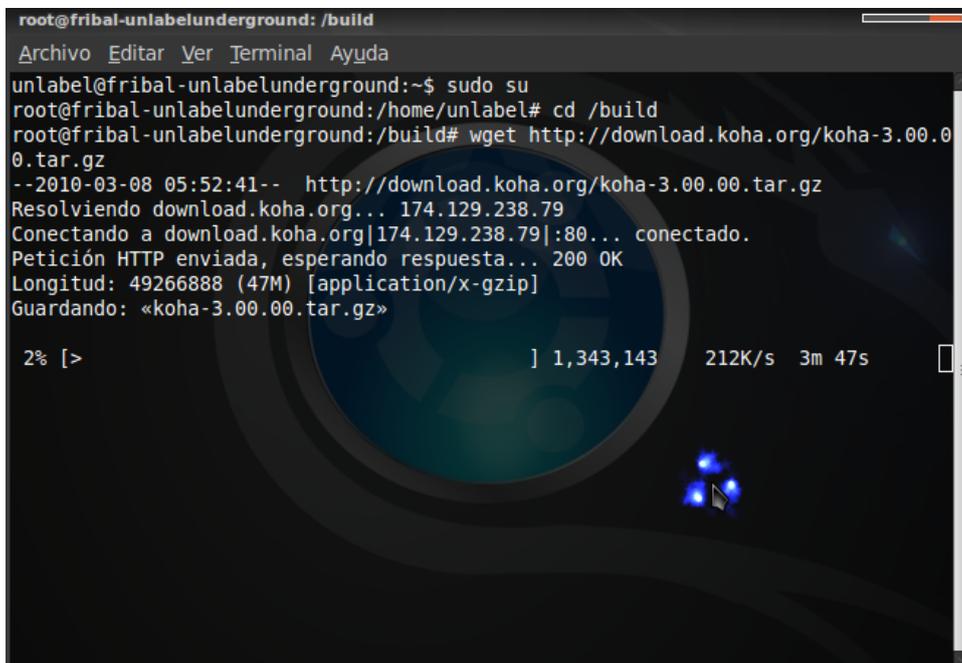
3.2.6 Fuentes, paquetes y librerías

Koha requiere ciertas librerías escritas en Perl para poder funcionar correctamente. Las mismas las obtendremos vía CPAN, que es el acrónimo de Comprehensive Perl Archive Network. Es un enorme archivo de software escrito en Perl, así como de documentación sobre el mismo. Tiene presencia en la Web a través de su sitio www.cpan.org y sus 236 mirrors distribuidos por todo el mundo. De igual manera, es necesario descargar y desempaquetar los archivos fuentes de Koha, mismos que situaremos dentro del directorio “build” que hemos creado previamente, así como las librerías Debian requeridas por Koha.

Para comenzar, vamos a la terminal e iniciamos como administrador, descargaremos la versión 3.00.00. Ejecutamos los siguientes comandos:

```
cd /build
sudo wget http://download.koha.org/koha-3.00.00.tar.gz
sudo tar -xzvf koha-3.00.00.tar.gz
cd koha-3.00.00
```

El sistema comenzará con la descarga en línea del SIGB.



```
root@fribal-unlabelunderground: /build
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ sudo su
root@fribal-unlabelunderground:/home/unlabel# cd /build
root@fribal-unlabelunderground:/build# wget http://download.koha.org/koha-3.00.00.tar.gz
--2010-03-08 05:52:41-- http://download.koha.org/koha-3.00.00.tar.gz
Resolviendo download.koha.org... 174.129.238.79
Conectando a download.koha.org[174.129.238.79]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 49266888 (47M) [application/x-gzip]
Guardando: «koha-3.00.00.tar.gz»

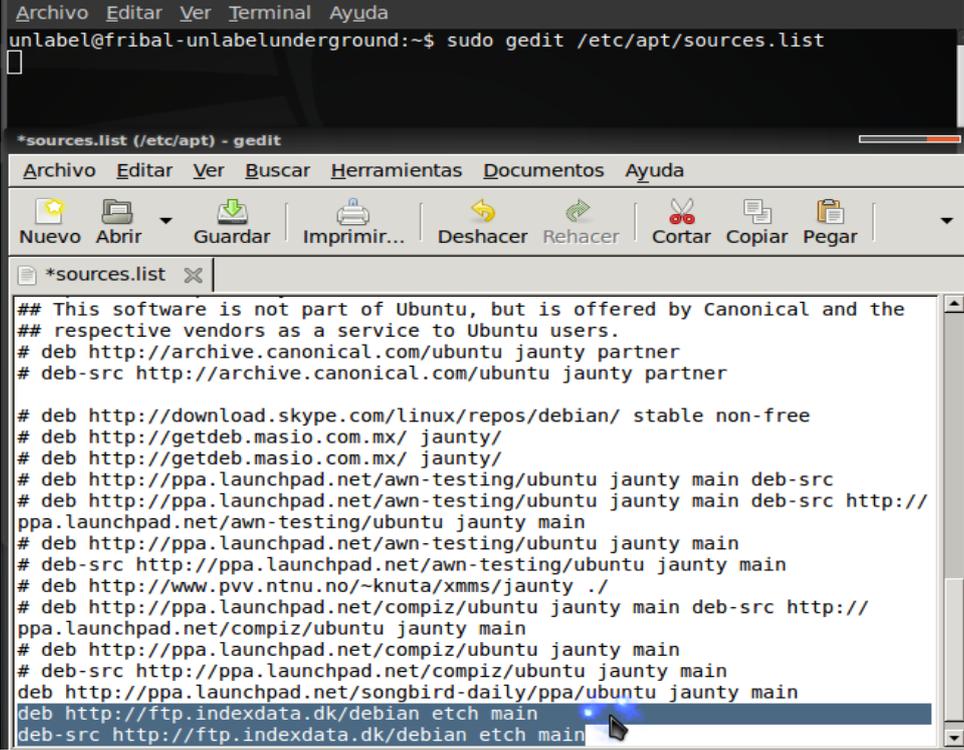
2% [>] 1,343,143 212K/s 3m 47s
```

Para instalar el número de dependencias requeridas, será necesario editar nuestro archivo de fuentes con “gedit”. Esto, con el objetivo de indicarle al sistema los sitios disponibles en línea desde los cuales se realizarán las descargas, para lo cual utilizaremos el comando **apt-get**. Editaremos el archivo de fuentes, con el siguiente comando:

```
sudo gedit /etc/apt/sources.list
```

Agregamos las siguientes líneas al archivo de fuentes, lo salvamos y cerramos.

```
deb http://ftp.indexdata.dk/debian etch main
deb-src http://ftp.indexdata.dk/debian etch main
```



```
unlabel@fribal-unlabelunderground:~$ sudo gedit /etc/apt/sources.list

*sources.list (/etc/apt) - gedit
Archivo Editar Ver Herramientas Documentos Ayuda
Nuevo Abrir Guardar Imprimir... Deshacer Rehacer Cortar Copiar Pegar
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
# deb http://archive.canonical.com/ubuntu jaunty partner
# deb-src http://archive.canonical.com/ubuntu jaunty partner

# deb http://download.skype.com/linux/repos/debian/ stable non-free
# deb http://getdeb.masio.com.mx/ jaunty/
# deb http://getdeb.masio.com.mx/ jaunty/
# deb http://ppa.launchpad.net/awn-testing/ubuntu jaunty main deb-src
# deb http://ppa.launchpad.net/awn-testing/ubuntu jaunty main deb-src http://
ppa.launchpad.net/awn-testing/ubuntu jaunty main
# deb http://ppa.launchpad.net/awn-testing/ubuntu jaunty main
# deb-src http://ppa.launchpad.net/awn-testing/ubuntu jaunty main
# deb http://www.pvv.ntnu.no/~knuta/xmms/jaunty ./
# deb http://ppa.launchpad.net/compiz/ubuntu jaunty main deb-src http://
ppa.launchpad.net/compiz/ubuntu jaunty main
# deb http://ppa.launchpad.net/compiz/ubuntu jaunty main
# deb-src http://ppa.launchpad.net/compiz/ubuntu jaunty main
deb http://ppa.launchpad.net/songbird-daily/ppa/ubuntu jaunty main
deb http://ftp.indexdata.dk/debian etch main
deb-src http://ftp.indexdata.dk/debian etch main
```

Será necesario descargar una llave digital que le permitirá al comando apt-get usar estas fuentes. Lo hacemos con el siguiente comando:

```
cd /build
sudo wget http://ftp.indexdata.dk/debian/indexdata.asc
sudo apt-key add indexdata.asc
```

Y finalizamos con una actualización de las fuentes agregadas al sistema.

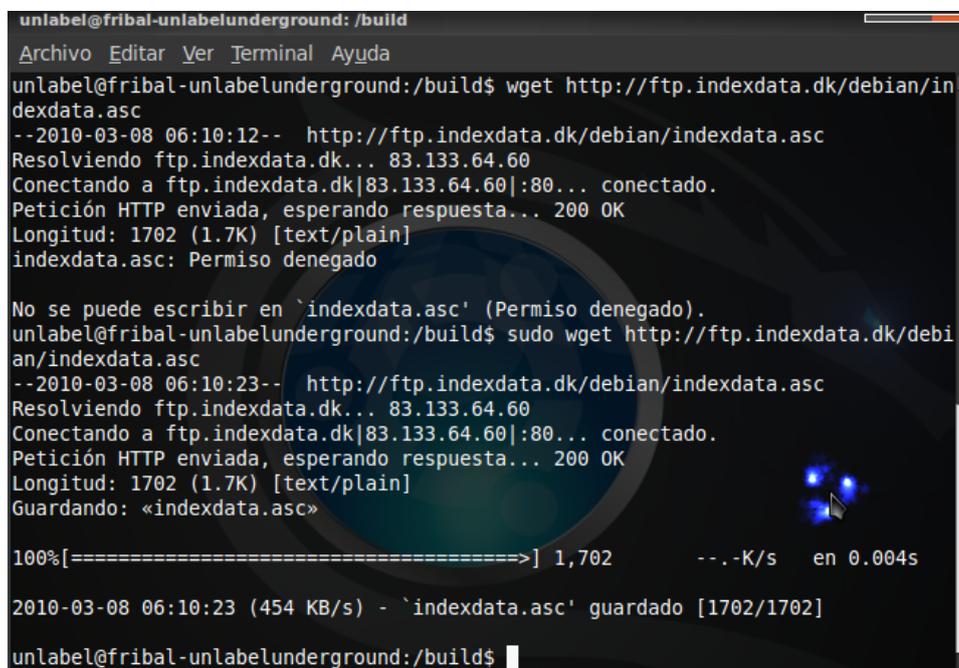
```
sudo apt-get update
```

3.2.7 Instalando los paquetes necesarios

Afortunadamente Koha incluye una lista de paquetes que permiten de manera fácil y rápida instalar la mayoría de las librerías requeridas sin necesidad de verificar detalles individuales por cada una. Utilizaremos el siguiente comando:

```
cd koha-3.00.00
sudo dpkg --set-selections < install_misc/debian.packages
```

El sistema comenzará con la descarga e instalación de los paquetes Debian requeridos por Koha.



```
unlabel@fribal-unlabelunderground: /build
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:/build$ wget http://ftp.indexdata.dk/debian/indexdata.asc
--2010-03-08 06:10:12-- http://ftp.indexdata.dk/debian/indexdata.asc
Resolviendo ftp.indexdata.dk... 83.133.64.60
Conectando a ftp.indexdata.dk[83.133.64.60]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 1702 (1.7K) [text/plain]
indexdata.asc: Permiso denegado

No se puede escribir en `indexdata.asc' (Permiso denegado).
unlabel@fribal-unlabelunderground:/build$ sudo wget http://ftp.indexdata.dk/debian/indexdata.asc
--2010-03-08 06:10:23-- http://ftp.indexdata.dk/debian/indexdata.asc
Resolviendo ftp.indexdata.dk... 83.133.64.60
Conectando a ftp.indexdata.dk[83.133.64.60]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 1702 (1.7K) [text/plain]
Guardando: `indexdata.asc'

100%[=====] 1,702 --.-K/s en 0.004s

2010-03-08 06:10:23 (454 KB/s) - `indexdata.asc' guardado [1702/1702]

unlabel@fribal-unlabelunderground:/build$
```

Como parte de este proceso, instalamos el módulo “dselect” y lo ejecutamos.

```
sudo apt-get install dselect
dselect
```

Elegimos la opción [I] Instalar y actualizar los paquetes deseados.

```
unlabel@fribal-unlabelunderground: /build/koha-3.00.00
Archivo Editar Ver Terminal Ayuda
Interfaz de manejo de paquetes dselect de Debian `1.14.24ubuntu1 (i386)'.

0. [M]étodo Escoger el método de acceso que se usará.
1. [A]ctualiza Actualizar la lista de paquetes disponibles, si se puede.
2. [S]elección Solicitar los paquetes que desea en su sistema.
* 3. [I]nstalar Instalar y actualizar los paquetes deseados.
4. [C]onfigura Configurar los paquetes que no estén configurados.
5. [D]esinstal Desinstalar los paquetes no deseados.
6. sa[L]ir Salir de dselect.

Utilice ^P y ^N, las teclas del cursor, letras iniciales, o dígitos;
Pulse <intro> para confirmar la selección. ^L redibuja la pantalla.

Copyright (C) 1994-1996 Ian Jackson.
Copyright (C) 2000,2001 Wichert Akkerman.
Esto es software libre; vea la Licencia Pública General de GNU versión 2 o
posterior para las condiciones de copia. No hay NINGUNA garantía.
Véase dselect --license para más detalles.

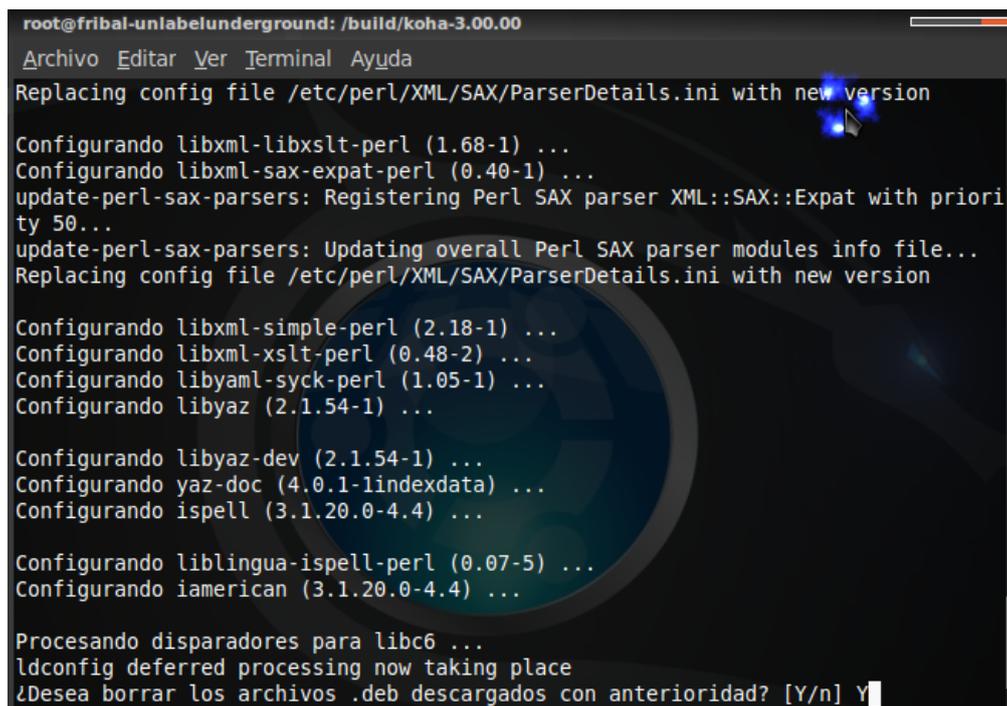
Acceso de sólo lectura: solamente se pueden ver las selecciones!
```

La instalación de estos prerequisites tomará un tiempo, dependiendo de la conexión a Internet con la que se cuente. En ocasiones, el sistema mostrará preguntas relacionadas con el espacio en disco que se utilizará; es necesario estar al tanto y aceptar los términos para continuar con la instalación.

```
root@fribal-unlabelunderground: /build/koha-3.00.00
Archivo Editar Ver Terminal Ayuda
libdate-calc-perl libdate-ical-perl libdate-leapyear-perl libdate-manip-perl
libdatetime-format-mail-perl libdatetime-format-strptime-perl
libdatetime-format-w3cdfp-perl libdatetime-locale-perl libdatetime-perl
libdatetime-timezone-perl libfilter-perl libgrypt11-dev libgpg-error-dev
libhtml-clean-perl libidzebra-2.0 libidzebra-2.0-dev
libidzebra-2.0-mod-alvis libidzebra-2.0-mod-dom libidzebra-2.0-mod-grs-marc
libidzebra-2.0-mod-grs-regex libidzebra-2.0-mod-grs-xml
libidzebra-2.0-mod-text libidzebra-2.0-modules liblingua-ispell-perl
liblingua-pt-stemmer-perl liblingua-stem-perl
liblingua-stem-snowball-da-perl liblist-moreutils-perl liblocale-po-perl
libmail-sendmail-perl libmysqlclient15-dev libnet-ldap-perl
libparams-validate-perl libpdf-api2-perl libpoe-perl
libsnowball-norwegian-perl libsnowball-swedish-perl
libsys-hostname-long-perl libtext-csv-perl libtext-csv-xs-perl
libtext-german-perl libtime-duration-perl libtime-format-perl
libunix-syslog-perl libwrap0-dev libxml-dom-perl libxml-dumper-perl
libxml-libxml-common-perl libxml-libxml-perl libxml-libxslt-perl
libxml-namespacesupport-perl libxml-perl libxml-regexp-perl
libxml-sax-expat-perl libxml-sax-perl libxml-simple-perl libxml-xslt-perl
libyaml-syck-perl libyaz libyaz-dev tcl8.4 yaz-doc
0 actualizados, 75 se instalarán, 0 para eliminar y 0 no actualizados.
Necesito descargar 24.7MB de archivos.
Se utilizarán 88.3MB de espacio de disco adicional después de esta operación.
¿Desea continuar [S/n]? █
```

Para concluir con el módulo “dselect” el sistema te preguntará si deseas borrar los paquetes instalados con anterioridad, aceptamos la opción predeterminada [Y] y elegimos lo siguiente, en orden consecutivo:

1. [C] configurar
2. [B] borrar
3. [S] salir

A terminal window screenshot showing the installation of Perl SAX parser modules. The prompt is root@fribal-unlabelunderground: /build/koha-3.00.00. The terminal output includes: Replacing config file /etc/perl/XML/SAX/ParserDetails.ini with new version; Configurando libxml-libxslt-perl (1.68-1) ...; Configurando libxml-sax-expat-perl (0.40-1) ...; update-perl-sax-parsers: Registering Perl SAX parser XML::SAX::Expat with priority 50...; update-perl-sax-parsers: Updating overall Perl SAX parser modules info file...; Replacing config file /etc/perl/XML/SAX/ParserDetails.ini with new version; Configurando libxml-simple-perl (2.18-1) ...; Configurando libxml-xslt-perl (0.48-2) ...; Configurando libyaml-syck-perl (1.05-1) ...; Configurando libyaz (2.1.54-1) ...; Configurando libyaz-dev (2.1.54-1) ...; Configurando yaz-doc (4.0.1-lindexdata) ...; Configurando ispell (3.1.20.0-4.4) ...; Configurando liblingua-ispell-perl (0.07-5) ...; Configurando iamerican (3.1.20.0-4.4) ...; Procesando disparadores para libc6 ...; ldconfig deferred processing now taking place; ¿Desea borrar los archivos .deb descargados con anterioridad? [Y/n] Y. The terminal has a menu bar with Archivo, Editar, Ver, Terminal, and Ayuda. A mouse cursor is visible over the terminal text.

```
root@fribal-unlabelunderground: /build/koha-3.00.00
Archivo Editar Ver Terminal Ayuda
Replacing config file /etc/perl/XML/SAX/ParserDetails.ini with new version
Configurando libxml-libxslt-perl (1.68-1) ...
Configurando libxml-sax-expat-perl (0.40-1) ...
update-perl-sax-parsers: Registering Perl SAX parser XML::SAX::Expat with priority 50...
update-perl-sax-parsers: Updating overall Perl SAX parser modules info file...
Replacing config file /etc/perl/XML/SAX/ParserDetails.ini with new version
Configurando libxml-simple-perl (2.18-1) ...
Configurando libxml-xslt-perl (0.48-2) ...
Configurando libyaml-syck-perl (1.05-1) ...
Configurando libyaz (2.1.54-1) ...
Configurando libyaz-dev (2.1.54-1) ...
Configurando yaz-doc (4.0.1-lindexdata) ...
Configurando ispell (3.1.20.0-4.4) ...
Configurando liblingua-ispell-perl (0.07-5) ...
Configurando iamerican (3.1.20.0-4.4) ...
Procesando disparadores para libc6 ...
ldconfig deferred processing now taking place
¿Desea borrar los archivos .deb descargados con anterioridad? [Y/n] Y
```

Adicionalmente al módulo “dselect”, necesitaremos instalar algunos paquetes necesarios para el funcionamiento de Koha, de manera individual. Ejecutamos el siguiente comando en la terminal:

```
apt-get install libexpat1-dev
apt-get install idzebra
apt-get install libgd2-xpm-dev
apt-get install libgdbm-dev
```

El sistema comenzará con la descarga e instalación de las librerías solicitadas.

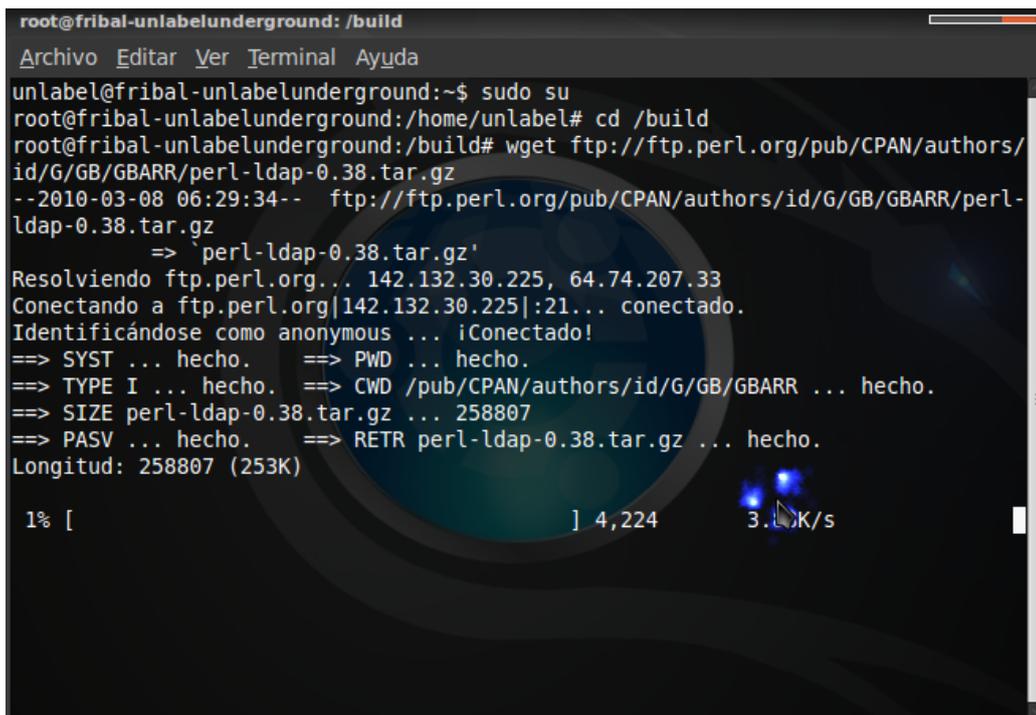
3.2.8 Módulos Perl

Koha requiere de una serie de módulos para su funcionamiento, escritos básicamente en Perl. Instalaremos los mismos, vía CPAN.

El primero que instalaremos será Net::LDAP, básico para el funcionamiento de Koha. Este módulo permite al SIGB un acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red y es implementado para programas escritos en Perl. Net::LDAP no puede ser instalado vía CPAN, por lo que haremos la instalación de manera manual.

Lanzamos la terminal y ejecutamos los siguientes comandos uno a uno en orden:

```
cd /build
sudo wget ftp://ftp.perl.org/pub/CPAN/authors/id/G/GB/GBARR/perl-ldap-0.38.tar.gz
sudo tar -xvzf perl-ldap-0.38.tar.gz
cd perl-ldap-0.38
perl Makefile.PL
make
make test
make install
```



```
root@fribal-unlabelunderground: /build
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ sudo su
root@fribal-unlabelunderground:/home/unlabel# cd /build
root@fribal-unlabelunderground:/build# wget ftp://ftp.perl.org/pub/CPAN/authors/id/G/GB/GBARR/perl-ldap-0.38.tar.gz
--2010-03-08 06:29:34-- ftp://ftp.perl.org/pub/CPAN/authors/id/G/GB/GBARR/perl-ldap-0.38.tar.gz
=> `perl-ldap-0.38.tar.gz'
Resolviendo ftp.perl.org... 142.132.30.225, 64.74.207.33
Conectando a ftp.perl.org|142.132.30.225|:21... conectado.
Identificándose como anonymous ... ¡Conectado!
==> SYST ... hecho. ==> PWD ... hecho.
==> TYPE I ... hecho. ==> CWD /pub/CPAN/authors/id/G/GB/GBARR ... hecho.
==> SIZE perl-ldap-0.38.tar.gz ... 258807
==> PASV ... hecho. ==> RETR perl-ldap-0.38.tar.gz ... hecho.
Longitud: 258807 (253K)

1% [ ] 4,224 3.1 K/s
```

Los siguientes módulos Perl, los instalaremos vía CPAN. La primera vez que ejecutemos CPAN, nos preguntará lo siguiente:

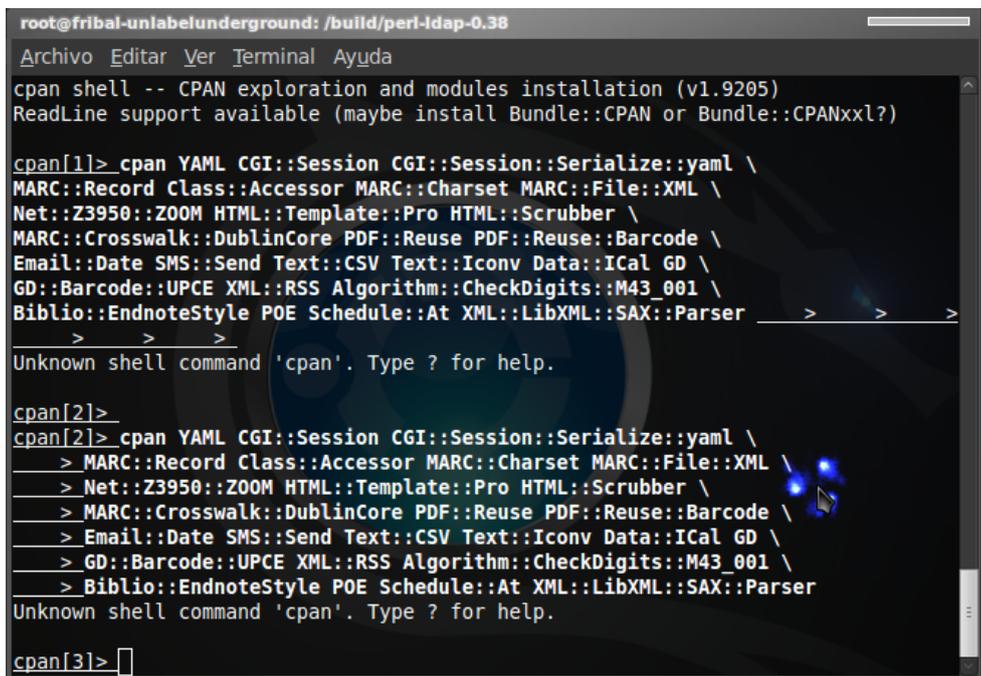
```
Would you like me to configure as much as possible automatically? [yes]
```

Damos ENTER a la opción predeterminada “si” y tendremos por defecto la configuración automática, la cual funciona sin problemas.

Para instalar los módulos requeridos por Koha, ejecutaremos el siguiente comando, incluyendo la diagonal invertida en cada uno, el cual instalara una larga lista de módulos Perl necesarios y requeridos por el sistema; algunos como el formato MARC, DublinCore, soporte para lectores de códigos de barra, etc. La instalación de dichos módulos tomará tiempo, y es necesario estar al tanto en cuanto el sistema requiera de aceptar algún término, para continuar con la instalación.

```
cpan YAML CGI::Session CGI::Session::Serialize::yaml \  
MARC::Record Class::Accessor MARC::Charset MARC::File::XML \  
Net::Z3950::ZOOM HTML::Template::Pro HTML::Scrubber \  
MARC::Crosswalk::DublinCore PDF::Reuse PDF::Reuse::Barcode \  
Email::Date SMS::Send Text::CSV Text::Iconv Data::ICal GD \  
GD::Barcode::UPCE XML::RSS Algorithm::CheckDigits::M43_001 \  
Biblio::EndnoteStyle POE Schedule::At XML::LibXML::SAX::Parser
```

En la terminal GNOME lo veremos de la siguiente manera:



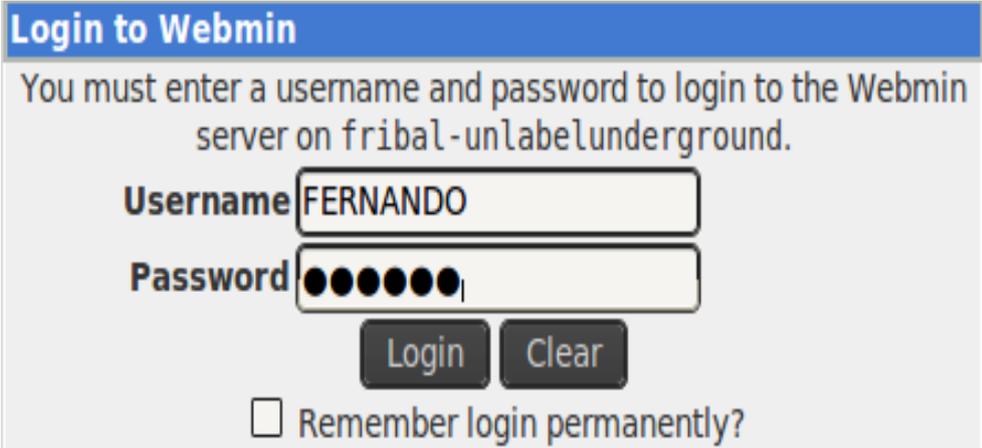
Con esto habremos terminado de instalar los prerequisites necesarios para la instalación de Koha. Lo siguiente será configurar permisos en el sistema, con lo cual determinaremos quién y a qué bases de datos puede acceder cada usuario del mismo; esto enfocado concretamente al staff de la biblioteca. Para concluir con la manera de compilar el código fuente de Koha, realizar la instalación y ligar el sistema al servidor Web apache.

3.2.9 Koha. Configurando permisos en Linux

Usaremos Webmin (herramienta que hemos instalado previamente) para configurar los permisos en el sistema, crear usuarios, acceso a bases de datos, etc.

Para acceder a la interfaz de Webmin e iniciar sesión a través de Firefox, tecleamos nuestra URL local en el navegador, que hemos obtenido al instalar y configurar Webmin por primera vez.

Ingresamos nuestro usuario y contraseña para acceder a la interfaz del administrador.



Login to Webmin

You must enter a username and password to login to the Webmin server on fribal-unlabelunderground.

Username

Password

Remember login permanently?

Tendremos lo siguiente:

The screenshot shows the Webmin 1.441 interface. On the left, there is a navigation menu with categories like Webmin, System, Servers, Others, Networking, Hardware, Cluster, and Un-used Modules. Below the menu is a search box and several action links: View Module's Logs, System Information, Refresh Modules, and Logout. The main content area displays system information in a table-like format:

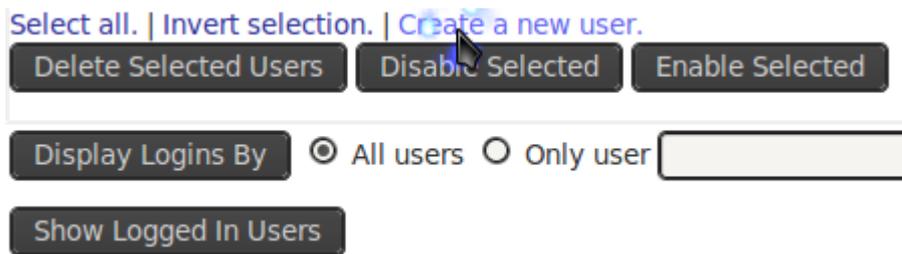
System hostname	fribal-unlabelunderground
Operating system	Ubuntu Linux 9.04
Webmin version	1.441
Time on system	Mon Mar 8 05:42:42 2010
Kernel and CPU	Linux 2.6.28-18-generic on i686
System uptime	2 hours, 11 minutes
CPU load averages	0.17 (1 min) 0.19 (5 mins) 0.18 (15 mins)
Real memory	1.96 GB total, 445.51 MB used
Virtual memory	3.72 GB total, 8.23 MB used
Local disk space	271.44 GB total, 222.48 GB used

El primer paso es crear un usuario llamado “Koha” y agregarlo a un grupo en particular también llamado con el mismo nombre, siguiendo la siguiente secuencia de pasos:

1. Ir a *sistema>Usuarios y grupos*.

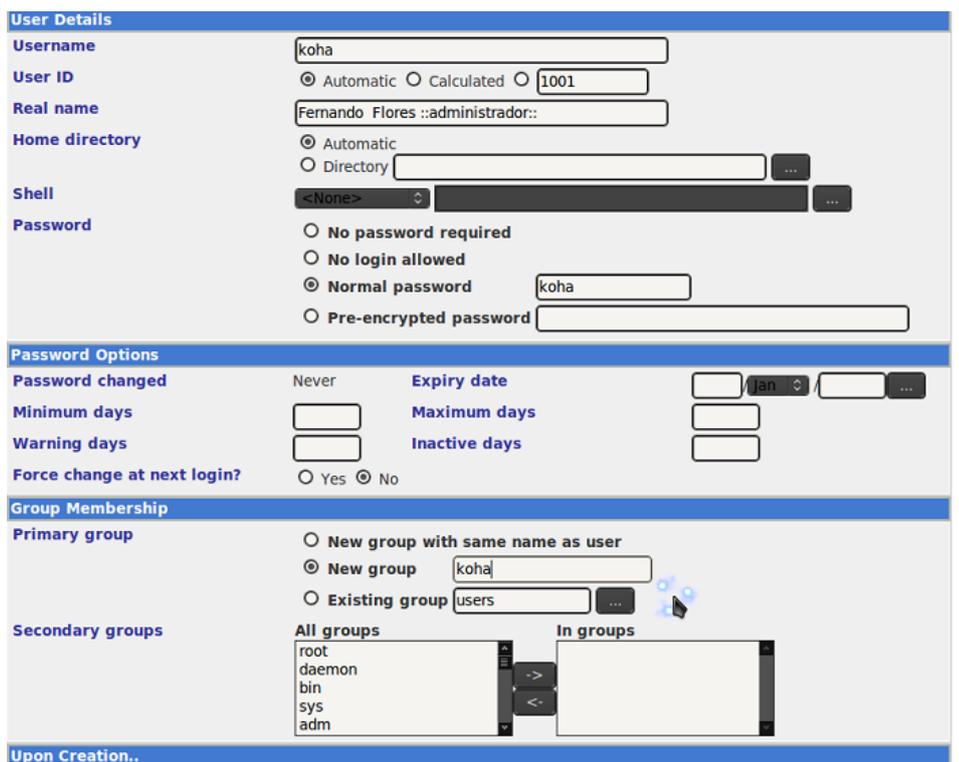
The screenshot shows the Webmin interface with the 'System' menu expanded. The 'Users and Groups' option is highlighted with a mouse cursor. The menu items listed are: Webmin, System, Bootup and Shutdown, Change Passwords, Disk and Network, Filesystems, Filesystem Backup, Log File Rotation, MIME Type Programs, PAM Authentication, Running Processes, Scheduled Commands, Scheduled Cron Jobs, Software Packages, System Documentator, System Logs, and Users and Groups.

2. Dar clic en el botón “crear nuevo usuario”.



3. Rellenar los campos con la siguientes características:

- a) Nombre de usuario: Koha.
- b) Nombre real: usuario de Koha o administrador (según sea el caso).
- c) Establecer un password personal: *****.
- d) Establecer un nuevo grupo primario, llamado Koha.
- e) Dar clic en crear.



3.2.10 Configurando MySQL

Antes de poder configurar MySQL, el sistema necesitará que ingreses la contraseña creada anteriormente para el usuario "root", durante la instalación del servidor LAMP.

Una vez logrado el acceso, lo primero que haremos será crear un usuario administrativo para Koha, con todos los privilegios y permisos en el servidor; lo haremos siguiendo la siguiente ruta: *Servers > MySQL Database Server > User Permissions > Create new user.*

Rellenaremos con los siguientes campos:

- Nombre de usuario: *(escoge un nombre para el administrador de Koha.*
- Contraseña: *(crea un password para el administrador).*
- Hosts: localhost.
- Permisos: seleccionar todos.

Por último, hacemos clic en crear.

Lo siguiente será crear la base de datos que utilizará Koha; para esto seguimos la ruta *Servers > MySQL Database Server > Create a New Database*. Dejamos todas las opciones como están predeterminadas, excepto por el nombre de la base de datos, el cual sustituimos por koha, y damos clic en crear.

MySQL Databases
Select all. | Invert selection. | [Create a New database.](#)

Module Index
Help..

Create Database

New database options

Database name:

Character set:

Initial table: None Named with fields below...

Field name	Data type	Type width	Key?	Autoinc?	Allow nulls?	Unsigned?
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No	<input type="checkbox"/> Yes <input type="checkbox"/> No

Finalmente y antes de proceder con la instalación de Koha, le daremos al usuario administrativo que hemos creado, permisos totales para el uso y manejo de la base de datos "koha" en MySQL; para esto seguimos la ruta *Servers > MySQL Database Server > Database Permissions* (bajo "Global Options")> *Create New Database Permissions*.

Servers
[Apache Webserver](#)
[MySQL Database Server](#)
[Read User Mail](#)


[Database Permissions](#)

Y llenamos los campos con la siguiente información:

- Bases de datos: seleccionamos “koha”.
- Nombre de usuario: (introducimos el nombre para el usuario “administrador” que hemos creado previamente).
- Hosts: from host permissions.
- Permisos: seleccionamos todos.

Module Index
Help..

Create Database Permissions

Database permission options

Databases Any Selected Matching pattern

Username Anonymous user

Hosts From host permissions Any

Permissions

- Select table data
- Insert table data
- Update table data
- Delete table data
- Create tables
- Drop tables
- Grant privileges
- Reference operations
- Manage indexes
- Alter tables

[Return to database permissions](#) | [Return to database list](#)

Para concluir hacemos clic en el botón “crear”.

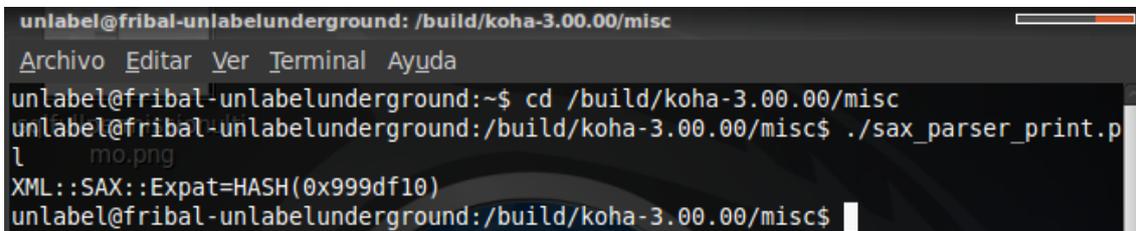
3.2.11 Instalación de Koha

Finalmente y después de llevar a cabo el proceso de configuración de los pre-requisitos necesarios para instalar y ejecutar Koha en Ubuntu 9.04, estamos listos para llevar a cabo la instalación del SIGB mediante el “tarball” que hemos descomprimido previamente.

Verificar el analizador de sintaxis SAX.

Confirmaremos si el sistema está utilizando el correcto analizador de sintaxis (XML::LibXML). Esto evitará que Koha tenga problemas con caracteres extendidos como acentos, signos, etc. Si después de ejecutar en la terminal el siguiente comando, nos aparece lo siguiente: "XML::SAX::Expat=HASH(0x999df10)". entonces tendremos todo en orden para comenzar.

```
cd /build/koha-3.00.00/misc
./sax_parser_print.pl
```



```
unlabel@fribal-unlabelunderground: /build/koha-3.00.00/misc
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ cd /build/koha-3.00.00/misc
unlabel@fribal-unlabelunderground:/build/koha-3.00.00/misc$ ./sax_parser_print.p
l
mo.png
XML::SAX::Expat=HASH(0x999df10)
unlabel@fribal-unlabelunderground:/build/koha-3.00.00/misc$
```

Instalando los archivos de Koha.

El primer paso es ejecutar el comando **Makefile.pl**, el cual generará un archivo **make** para instalar koha, y a su vez diagnosticará si alguna librería no ha sido instalada o es requerida para poder continuar. Cuando el sistema lo requiera, se deberá especificar el nombre de usuario encargado de administrar la base de datos que será usada por Koha, además de proveer la información de acceso para MySQL que hemos creado en Webmin previamente. Los comandos **make test** y **make install** se ejecutarán por algunos minutos.

Para instalar koha, que previamente hemos descargado, ejecutaremos el siguiente comando siguiendo cada una de las líneas, en su mismo orden:

```
sudo su
cd /build/koha-3.00.00
perl Makefile.PL
make
make test
make install
```

Tendremos pantallas como las siguientes, al ejecutar cada línea del comando:

```
root@fribal-unlabelunderground: /build/koha-3.00.00
Archivo Editar Ver Terminal Ayuda
cp koha-tmpl/opac-tmpl/prog/bg-Cyrl/modules/opac-authoritiessearchresultlist.templ blib/OPAC_TMPL_DIR/prog/bg-Cyrl/modules/opac-authoritiessearchresultlist.templ
cp koha-tmpl/opac-tmpl/prog/el-GR/modules/opac-opensearch.templ blib/OPAC_TMPL_DIR/prog/el-GR/modules/opac-opensearch.templ
cp koha-tmpl/opac-tmpl/prog/nl-NL/modules/opac-ISBDdetail.templ blib/OPAC_TMPL_DIR/prog/nl-NL/modules/opac-ISBDdetail.templ
cp koha-tmpl/intranet-tmpl/prog/ar-Arab/modules/cataloguing/value_builder/labs_theses.templ blib/INTRANET_TMPL_DIR/prog/ar-Arab/modules/cataloguing/value_builder/labs_theses.templ
cp etc/pazpar2/pz2-ourl-marc21.xsl blib/PAZPAR2_CONF_DIR/pz2-ourl-marc21.xsl
cp koha-tmpl/intranet-tmpl/prog/sr-Cyrl/includes/reports-menu.inc blib/INTRANET_TMPL_DIR/prog/sr-Cyrl/includes/reports-menu.inc
cp koha-tmpl/intranet-tmpl/prog/nl-NL/modules/cataloguing/z3950_search.templ blib/INTRANET_TMPL_DIR/prog/nl-NL/modules/cataloguing/z3950_search.templ
cp koha-tmpl/intranet-tmpl/prog/lo-Laoo/includes/label-status.inc blib/INTRANET_TMPL_DIR/prog/lo-Laoo/includes/label-status.inc
cp koha-tmpl/intranet-tmpl/prog/sv-SE/modules/help/members/maninvoice.templ blib/INTRANET_TMPL_DIR/prog/sv-SE/modules/help/members/maninvoice.templ
cp koha-tmpl/intranet-tmpl/prog/uk-UA/modules/authorities/auth_finder.templ blib/INTRANET_TMPL_DIR/prog/uk-UA/modules/authorities/auth_finder.templ
cp koha-tmpl/intranet-tmpl/prog/pl-PL/js/tinymce/jscripts/tiny_mce/themes/advanced/images/insert_button_bg.gif blib/INTRANET_TMPL_DIR/prog/pl-PL/js/tinymce/jscripts/tiny_mce/themes/advanced/images/insert_button_bg.gif
```

```
root@fribal-unlabelunderground: /build/koha-3.00.00
Archivo Editar Ver Terminal Ayuda
root@fribal-unlabelunderground: /build/koha-3.00.00# perl Makefile.PL
By default, Koha can be installed in one of three ways:
standard: Install files in conformance with the Filesystem
Hierarchy Standard (FHS). This is the default mode
and should be used when installing a production
Koha system. On Unix systems, root access is
needed to complete a standard installation.
single: Install files under a single directory. This option
is useful for installing Koha without root access, e.g.,
on a web host that allows CGI scripts and MySQL databases
but requires the user to keep all files under the user's
HOME directory.
dev: Create a set of symbolic links and configuration files to
allow Koha to run directly from the source distribution.
This mode is useful for developers who want to run
Koha from a git clone.

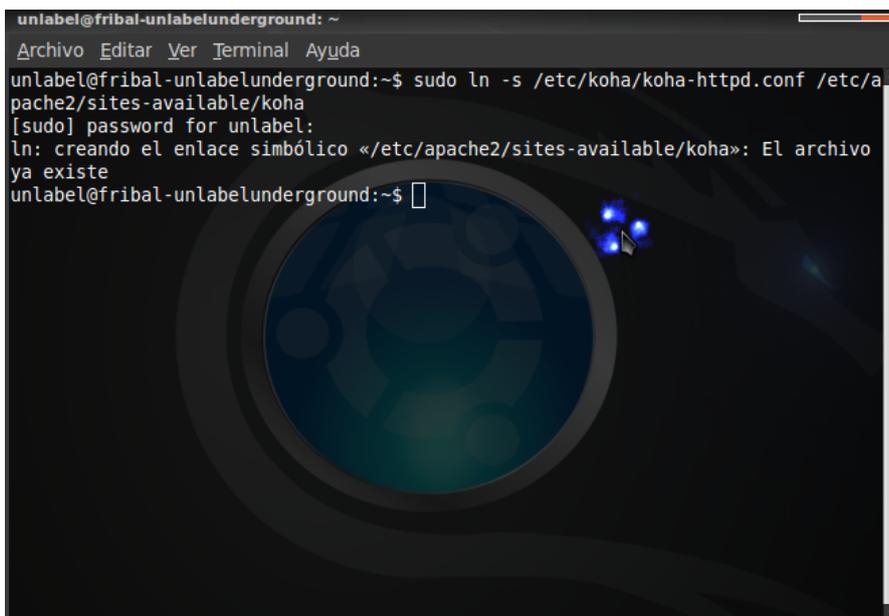
Installation mode (dev, single, standard) [standard] make
Value 'make' is not a valid option.
Please enter a value (dev, single, standard) [standard]
```

Lo siguiente que haremos en este proceso de instalación de Koha, será configurar el servidor Web Apache para habilitar al SIGB como un Website

válido. Lo que haremos será crear un link simbólico entre el archivo de configuración de koha (koha-httpd.conf) y los sitios disponibles para Apache.

Ejecutamos en terminal:

```
Sudo ln -s /etc/koha/koha-httpd.conf /etc/apache2/sites-available/koha
```



```
unlabel@fribal-unlabelunderground: ~  
Archivo Editar Ver Terminal Ayuda  
unlabel@fribal-unlabelunderground:~$ sudo ln -s /etc/koha/koha-httpd.conf /etc/a  
pache2/sites-available/koha  
[sudo] password for unlabel:  
ln: creando el enlace simbólico «/etc/apache2/sites-available/koha»: El archivo  
ya existe  
unlabel@fribal-unlabelunderground:~$
```

En nuestro caso ya se tenía el link creado, por lo que la terminal muestra que el archivo ya existe. Pero si es la primera vez que instalas Koha no debería haber problema alguno. Lo siguiente será habilitar un módulo Apache para el sitio de Koha, reescribiéndolo con los siguientes comandos:

```
a2enmod rewrite  
a2ensite koha
```

Predeterminadamente, Koha utiliza el puerto 8080 para su interfaz administrativa, la interfaz Web que el personal bibliotecario utilizará. Para esto es necesario configurar Apache de manera que acepte pedidos de conexión Web, en el puerto 8080, en adición al puerto 80, predeterminado.

Editaremos el archivo de puertos de Apache con “gedit”, agregando la línea del puerto 8080 al final del archivo, como vemos en la imagen.

```
Sudo gedit /etc/apache2/ports.conf
```

```
unlabel@fribal-unlabelunderground: ~
Archivo Editar Ver Terminal Ayuda
unlabel@fribal-unlabelunderground:~$ sudo ln -s /etc/koha/koha-httpd.conf /etc/a
pache2/sites-available/koha
[sudo] password for unlabel:
ln: creando el enlace simbólico «/etc/apache2/sites-available/koha»: El archivo
ya existe
unlabel@fribal-unlabelunderground:~$ a2enmod rewrite
Enabling module rewrite.
Could not create /etc/apache2/mods-enabled/rewrite.load: Permiso denegado
unlabel@fribal-unlabelunderground:~$ sudo a2enmod rewrite
Enabling module rewrite.
Run '/etc/init.d/apache2 restart' to activate new configuration!
unlabel@fribal-unlabelunderground:~$ sudo a2ensite koha
Enabling site koha.
Run '/etc/init.d/apache2 reload' to activate new configuration!
unlabel@fribal-unlabelunderground:~$ sudo gedit /etc/apache2/ports.conf
# README.Debian.gz

NameVirtualHost *:80
Listen 80

<IfModule mod_ssl.c>
# SSL name based virtual hosts are not yet supported, therefore no
# NameVirtualHost statement here
Listen 443
</IfModule>
Listen 8080
```

Guardamos el archivo y cerramos.

Finalmente, con el siguiente comando reiniciamos Apache, con lo cual pondremos en marcha las configuraciones realizadas. Si todo ha salido bien la terminal mostrará el texto “OK”.

```
Sudo /etc/init.d/apache2 restart
```

```
unlabel@fribal-unlabelunderground:~$ sudo /etc/init.d/apache2 restart
* Restarting web server apache2
... waiting [ OK ]
unlabel@fribal-unlabelunderground:~$
```

Llegado este punto, hemos finalizado la instalación de Koha en Ubuntu 9.04; para acceder simplemente tecleamos en nuestro navegador, de preferencia Firefox, la siguiente dirección: <http://127.0.1.1:8080>, y accsaremos al front-end

administrativo de Koha enfocado al staff bibliotecario. El sistema pedirá que seleccionemos el lenguaje preferido y establecerá una conexión con el puerto asignado a MySQL. El usuario y contraseña de inicio para el front-end de Koha, predeterminadamente es:

Credenciales: koha Password: koha



The image shows a login form for Koha. The title is "Log In to Koha". There are two input fields: "Cardnumber:" and "Password:". Below the fields is a blue "Submit" button.

Como seguimiento a este capítulo, en puntos posteriores haremos un énfasis concreto en cada una de las interfaces disponibles en Koha, la interfaz administrativa y el OPAC, describiendo cada una de sus características de funcionalidad.

3.3 Instalación de Koha V 2.29 en Windows XP SP3 arquitectura X86

Afortunadamente, para muchos usuarios que desean utilizar Koha, y que no se encuentran familiarizados con entornos GNU/Linux, se encuentra disponible una versión del sistema integrado para equipos, ejecutando cualquier versión de Windows XP bajo una arquitectura de 32 bits (X86). Podemos afirmar que si bien el proceso de instalación en el entorno Windows es más sencillo que en Linux, debemos cubrir también una serie de pre-requisitos necesarios para el buen funcionamiento del sistema integrado para bibliotecas, es decir, montar un servidor Web (Apache), un gestor de base de datos relacional (MySQL) y llevar a cabo una instalación del lenguaje de programación bajo el cual fue escrito Koha, "Perl". En el siguiente tema describiremos detalladamente cada uno de estos procesos que culminarán con la instalación exitosa de un paquete autoejecutable Win-32, llamado Koha.

3.3.1 Programas requeridos

Apache Webserver

Apache Webserver se encuentra disponible en:
<http://www.apache.org/dist/httpd/binaries/win32/>.

Para la instalación llevada a cabo en este capítulo hemos utilizado la versión 2.2 del servidor Web, la cual trabaja bien y logra conectar sin problemas las dos interfaces disponibles en Koha.

MySQL Database Server

La versión que utilizaremos y con la cual se ejecuta establemente Koha en Windows es la versión 4.1, la podemos descargar desde:

<http://dev.mysql.com/downloads/mysql/4.1.html>

Es importante mencionar que fueron probadas versiones recientes del gestor de bases de datos, las cuales nunca lograron funcionar a la par con el SIGB, por lo que se recomienda utilizar exclusivamente la versión 4.1 de MySQL para lograr resultados óptimos en la instalación de Koha.

Active Perl State

Las librerías Active Perl, necesarias para la ejecución de Koha y sus componentes, se encuentran disponibles en:
<http://www.activestate.com/Products/ActivePerl/>.

Utilizaremos la versión 5.8.8.822 con la que se ejecuta exitosamente Koha en Windows de manera estable.

Archivos de instalación Koha W32

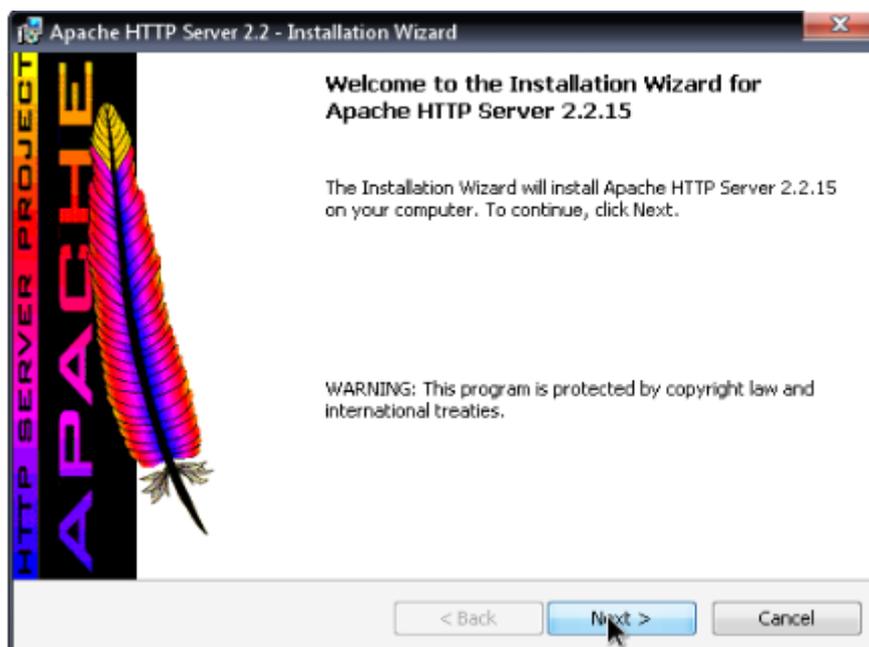
Los paquetes de instalación necesarios se encuentran disponibles en el sitio encargado de dar soporte y ofrecer soluciones enfocadas a la ejecución de Koha bajo el entorno Windows: <http://www.koha.rwjr.com/>.

La última versión liberada y estable para dicho sistema es la 2.2.9, la cual incluye los módulos Perl necesarios para ejecutar Koha con todas sus

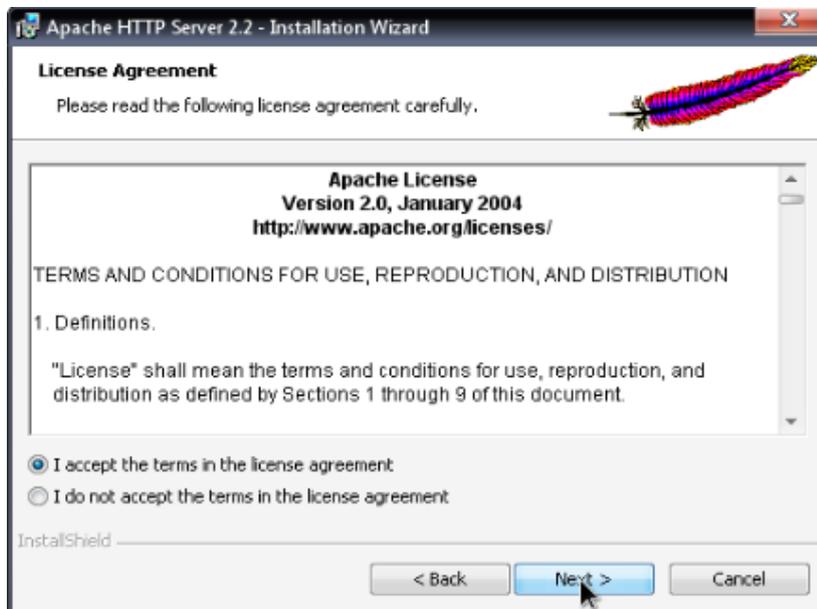
funciones (a excepción de la capacidad para descargar registros bajo Z39.50) en Windows.

3.3.2 Instrucciones de instalación

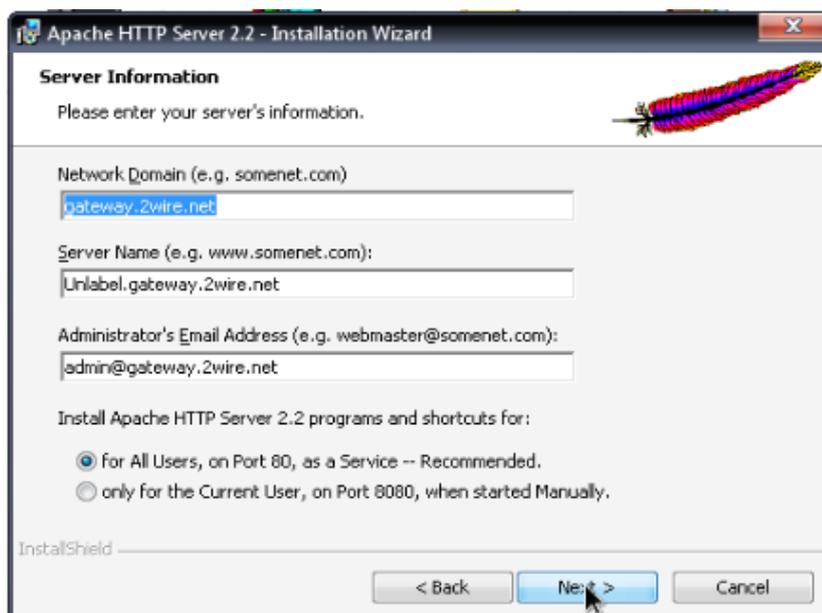
Instalar Apache Webserver. El primer paso será llevar a cabo la instalación del servidor Web Apache. Para esto haremos doble clic en el archivo auto ejecutable que descargamos anteriormente, y tendremos lo siguiente:



Hacemos clic en el botón siguiente (Next) y el instalador nos desplegará la licencia del software. Aceptamos los términos y continuamos.



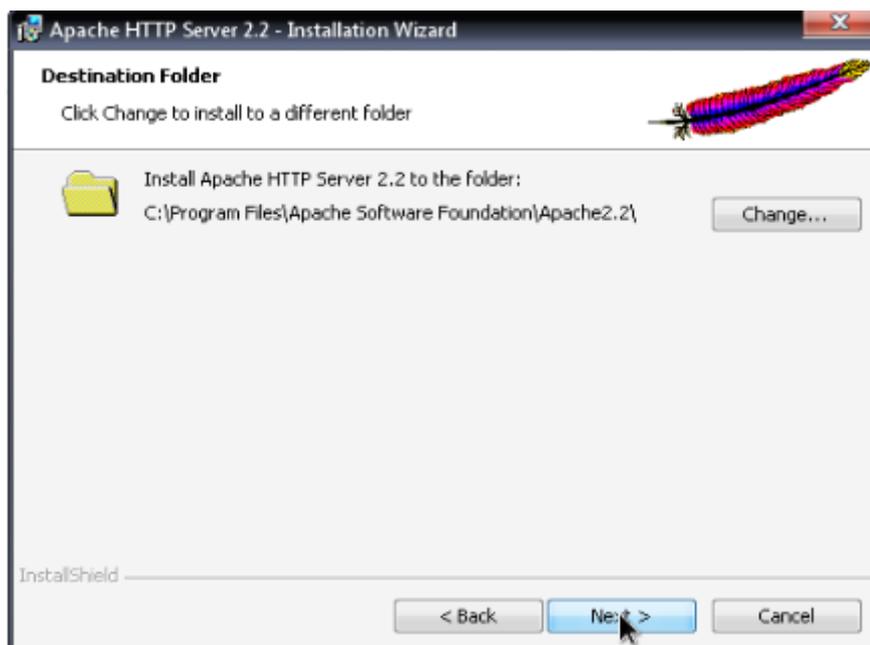
En el siguiente paso, Apache nos mostrará los datos referentes al Host local; al igual que en Linux, tendremos el nombre de nuestro equipo seguido por la dirección de red que utilizarán las aplicaciones locales para acceder al servidor. Se recomienda dejar los datos que aparecen predeterminados, en su momento la instalación de Koha realizará los cambios pertinentes.



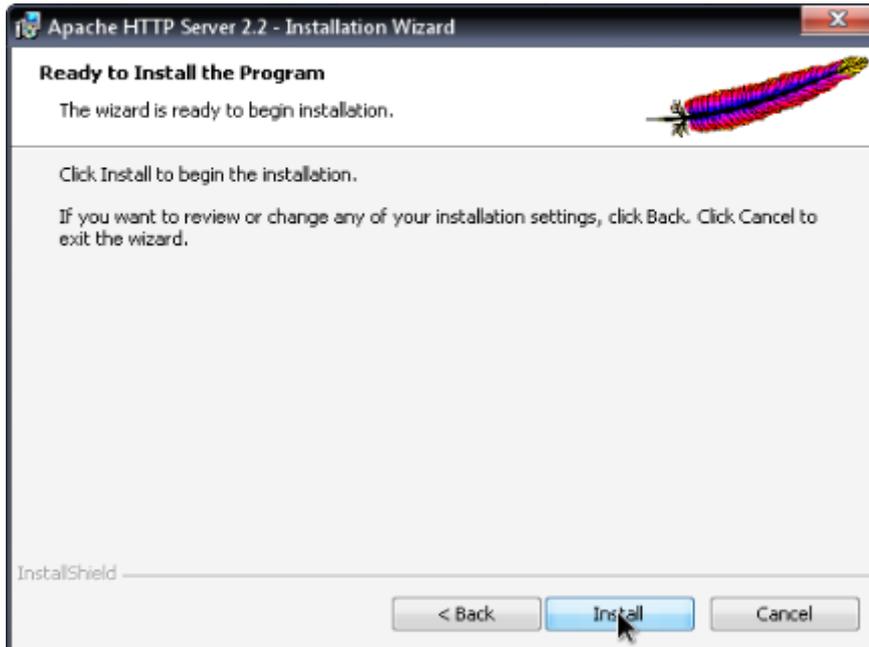
Seleccionamos la opción “instalación típica” y continuamos.



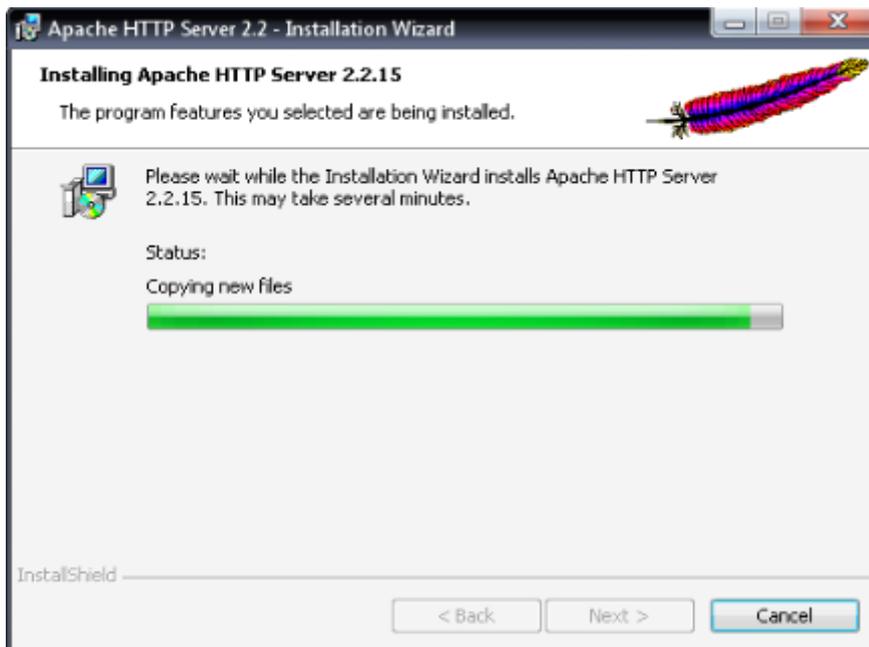
Para la instalación del servidor Web, no será necesario realizar cambios en la ruta predeterminada (a diferencia de las aplicaciones que veremos mas adelante); hacemos clic en continuar, siguiendo la ruta predeterminada que nos muestra el instalador.



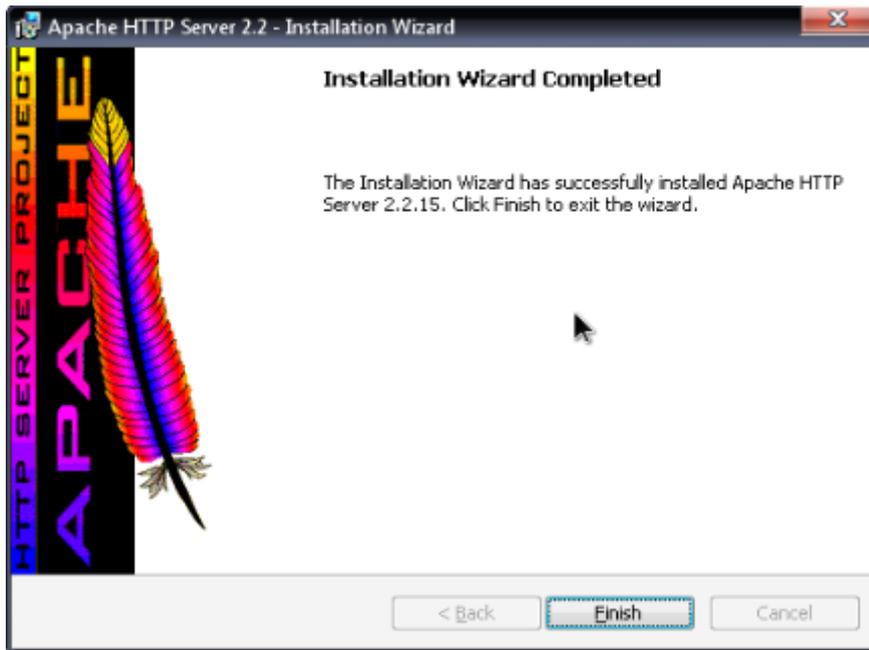
Y finalmente hacemos clic en “instalar”



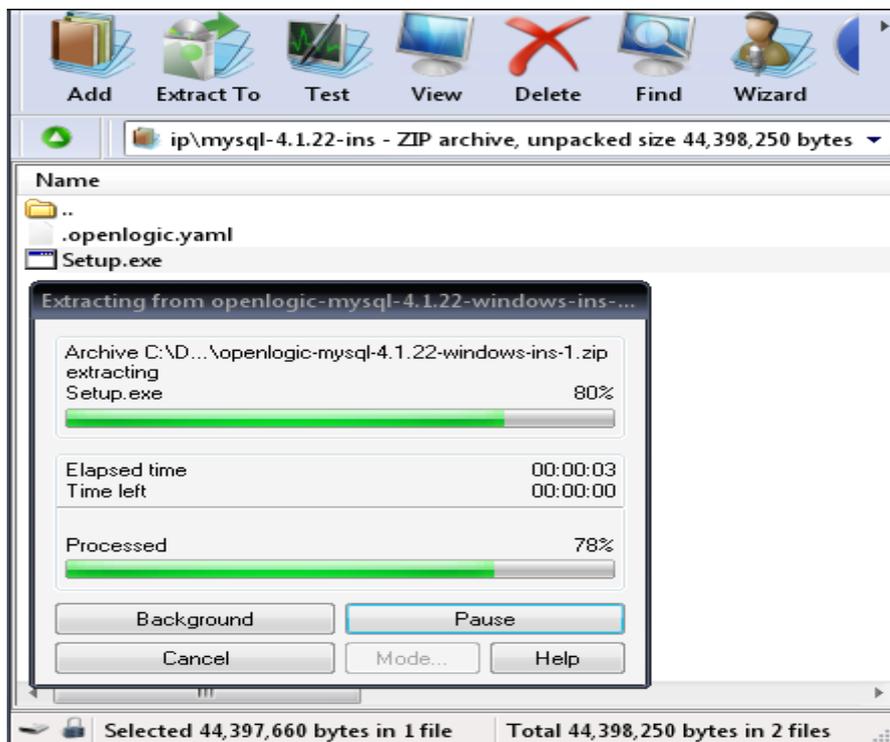
Comenzará la instalación de Apache Web Server.



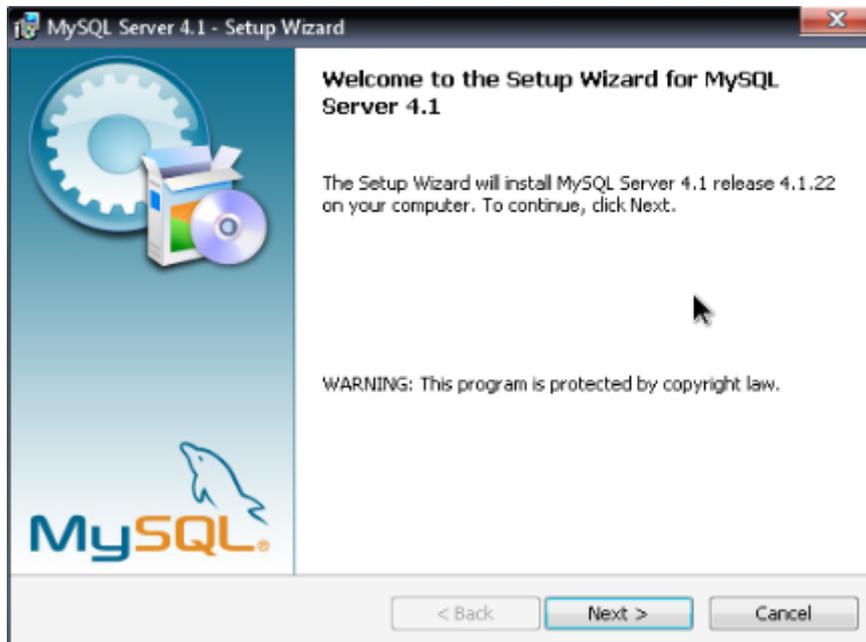
Si todo ha salido bien, el instalador mostrará el siguiente mensaje, haciendo referencia a que ha concluido la instalación de Apache Web Server exitosamente, finalizamos y tendremos un pequeño ícono en la barra del sistema (tray).



Instalación de MySQL Server 4.1. Para comenzar con la instalación de este poderoso gestor de base de datos, necesitaremos descomprimir el archivo que hemos descargado previamente, el cual incluye el archivo de instalación de la aplicación. Podemos utilizar cualquier descompresor que soporte el formato .zip para Windows.



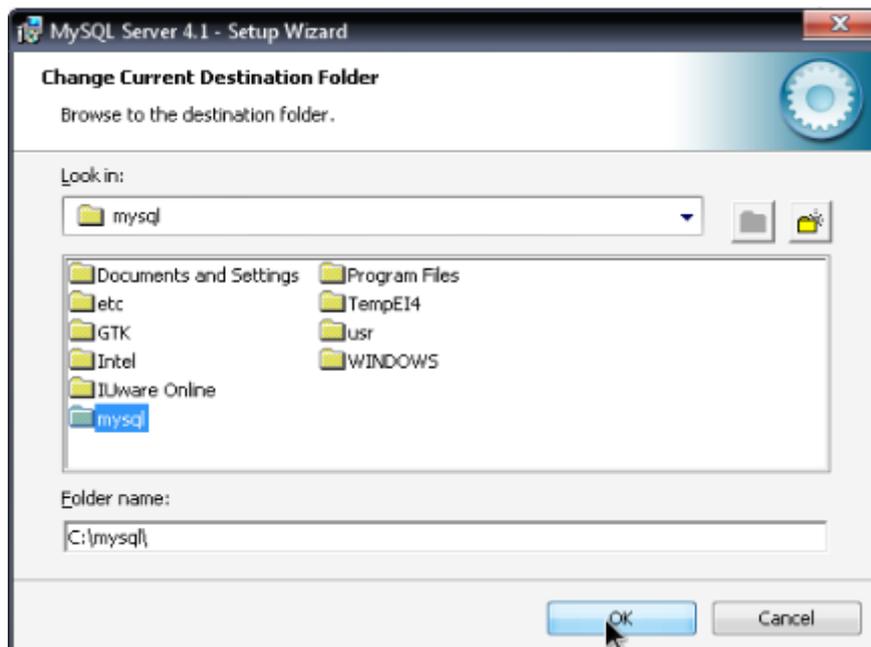
Una vez extraído el archivo de la aplicación, el asistente nos dará la bienvenida a la instalación del software.



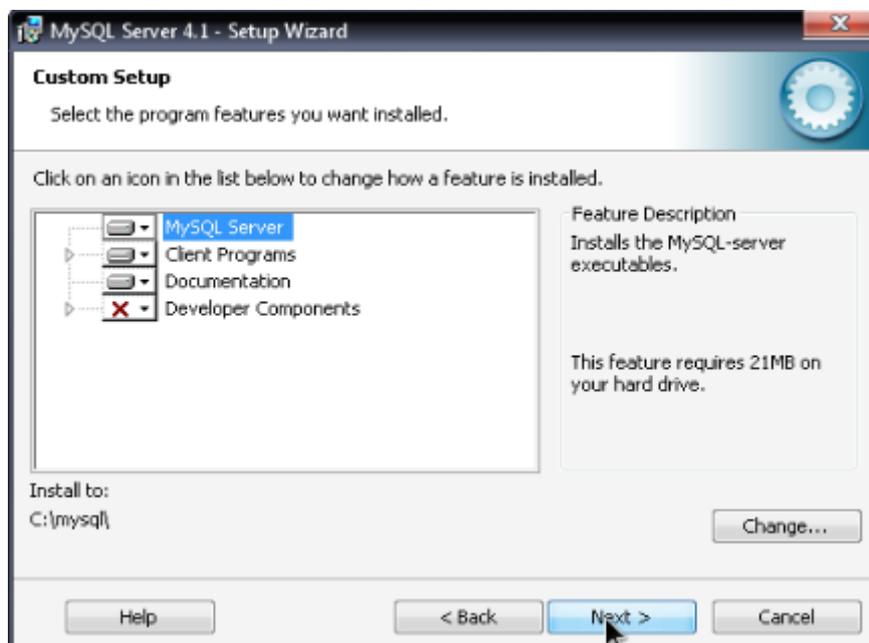
Seleccionamos la opción “instalación personalizada” (custom), que nos permitirá elegir la ruta en donde serán guardados los archivos de la aplicación.



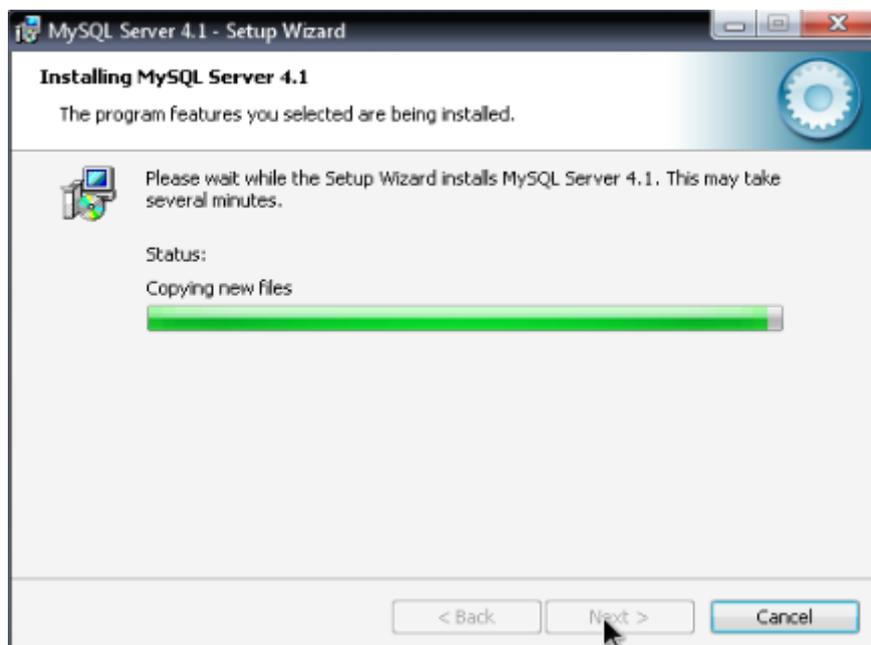
En el siguiente cuadro cambiamos la ruta predeterminada de instalación, a: **C:\mysql** y hacemos clic en “OK”.



Aceptamos las opciones predeterminadas de instalación y continuamos.



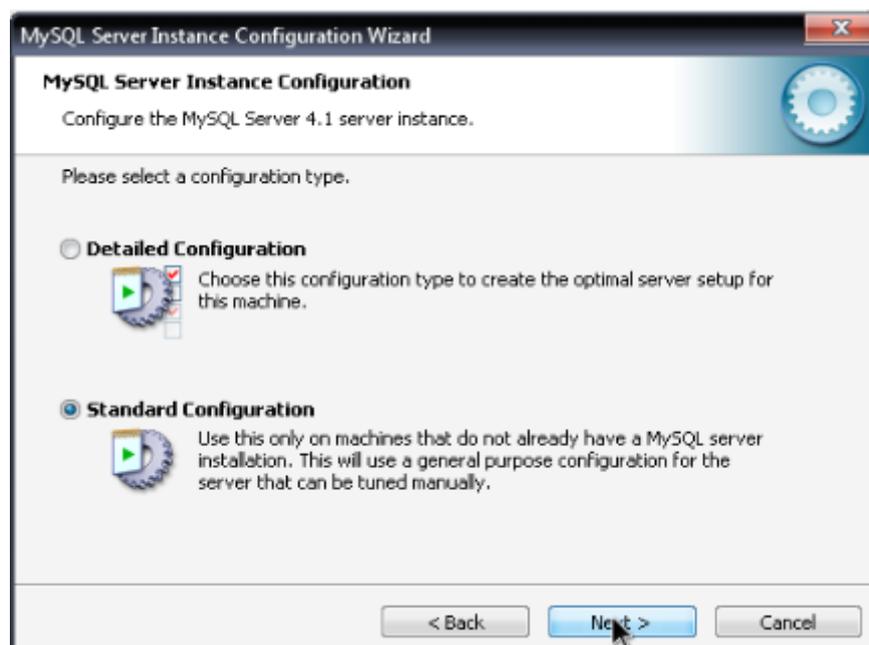
Finalmente hacemos clic en instalar, el asistente comenzará con la instalación de MySQL Server 4.1.



Al término de la instalación el programa preguntará si deseamos crear una cuenta en el sitio MySQL.com, esto no será necesario por el momento, por lo cual omitimos este paso y continuamos.



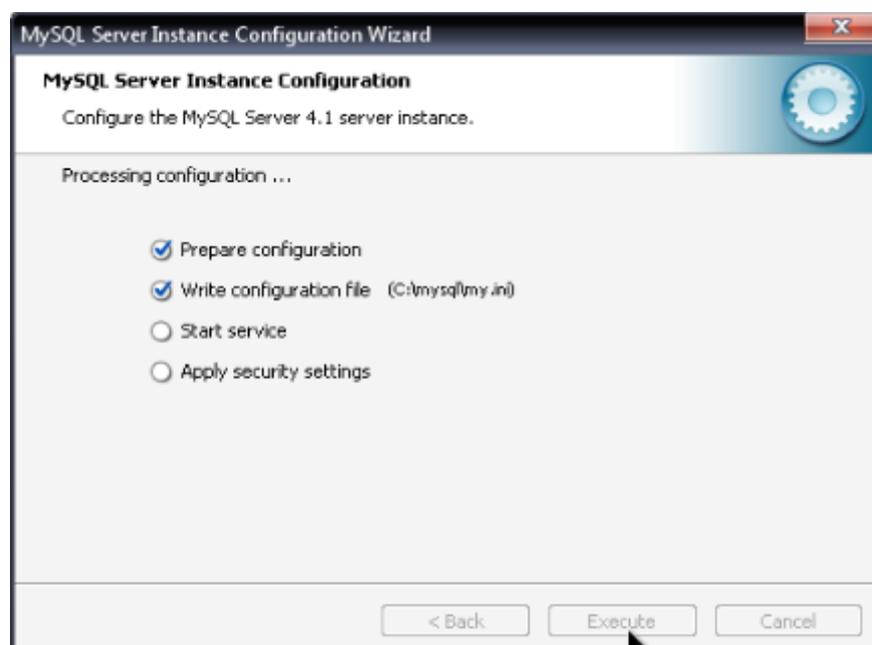
Lo siguiente que haremos será configurar de primera instancia MySQL, para lo cual seleccionamos la opción "configuración estándar", por ser ésta la primera vez que estamos llevando a cabo una instalación del gestor de bases de datos.



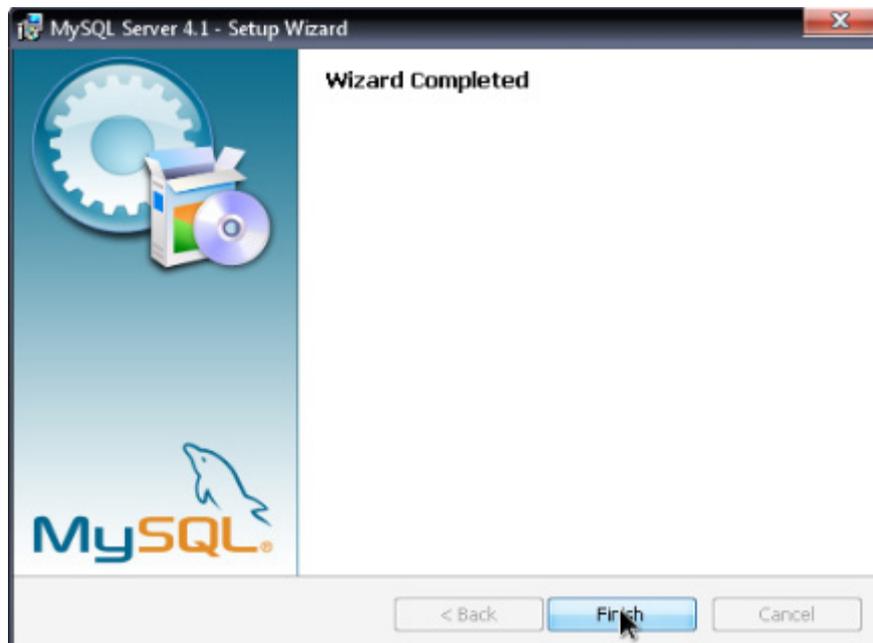
Al igual que en Linux, necesitaremos asignar privilegios de administrador a MySQL, mediante una contraseña, después de asignarla continuamos con el proceso.



Finalmente, ejecutamos la configuración. El asistente creará el archivo de configuración para MySQL, iniciará el gestor como proceso, y aplicará las configuraciones de seguridad pertinentes.



Con esto habremos finalizado la instalación de MySQL en Windows XP.



Instalar ActivePerl 5.8. Después de realizar la descarga de Perl 5.8 mediante los pasos anteriores, notaremos que no contamos con un archivo auto ejecutable .exe o en su defecto .msi; para llevar a cabo la instalación de los scripts escritos en Perl para Koha, ejecutaremos el archivo .BAT llamado "installer" de la carpeta comprimida que hemos descargado.

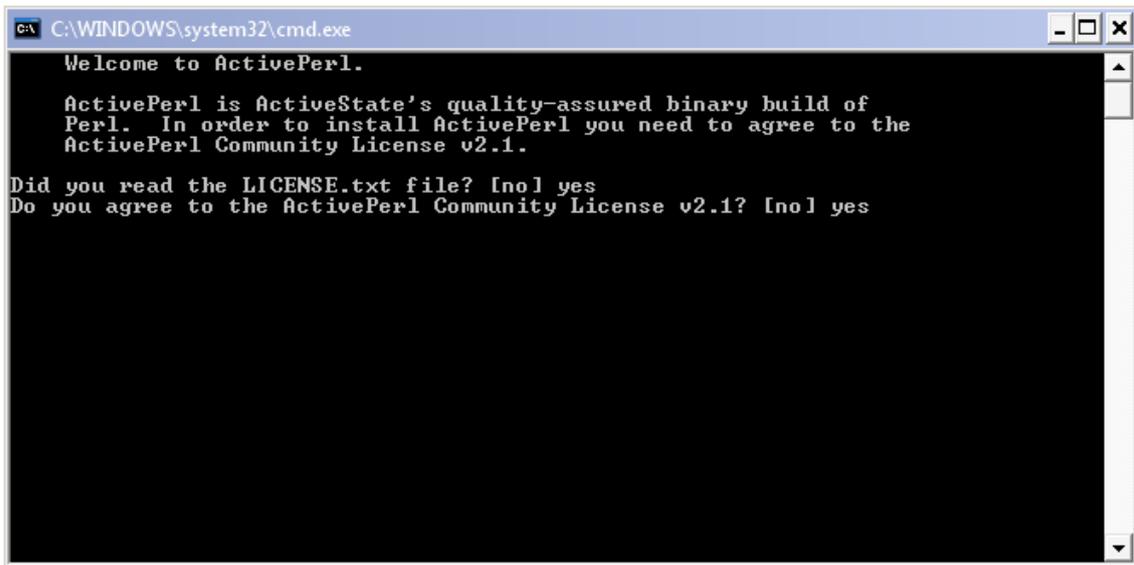
El archivo se ejecutará bajo MS-DOS y tendremos la siguiente pantalla:

```
C:\WINDOWS\system32\cmd.exe
Welcome to ActivePerl.

ActivePerl is ActiveState's quality-assured binary build of
Perl. In order to install ActivePerl you need to agree to the
ActivePerl Community License v2.1.

Did you read the LICENSE.txt file? [no]
```

Escribimos “yes” en la línea de comando, aceptando que hemos leído la licencia del software y que estamos de acuerdo con los términos establecidos.

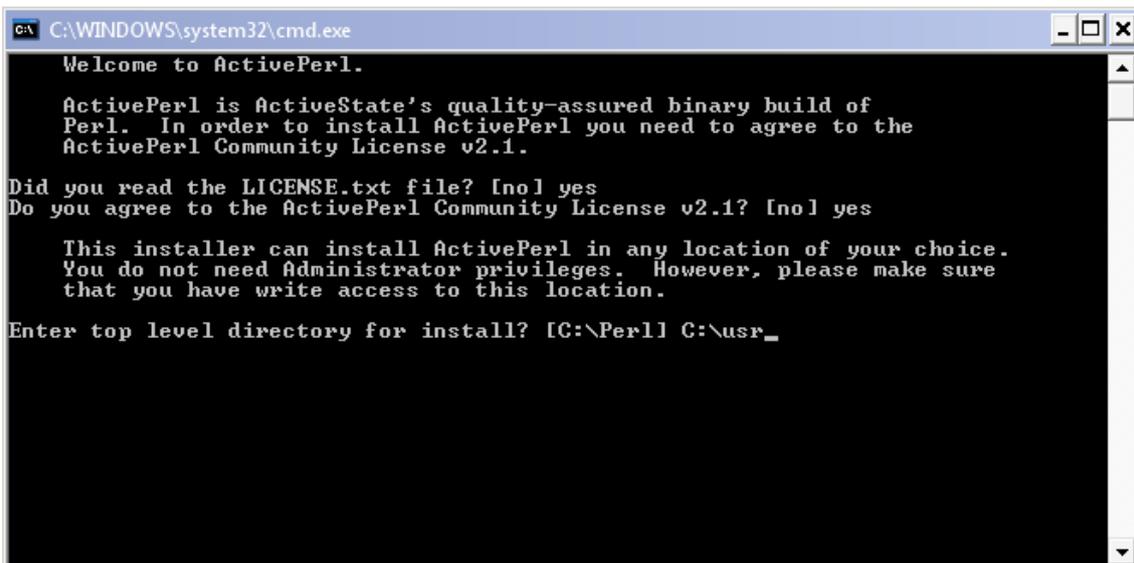


```
C:\WINDOWS\system32\cmd.exe
Welcome to ActivePerl.

ActivePerl is ActiveState's quality-assured binary build of
Perl. In order to install ActivePerl you need to agree to the
ActivePerl Community License v2.1.

Did you read the LICENSE.txt file? [no] yes
Do you agree to the ActivePerl Community License v2.1? [no] yes
```

Es necesario cambiar la ruta de instalación que el archivo incluye predeterminadamente y ubicarla en la misma carpeta que Koha tendrá destinada. Para esto, escribimos la ruta: **C:\usr** dentro de la línea de comandos



```
C:\WINDOWS\system32\cmd.exe
Welcome to ActivePerl.

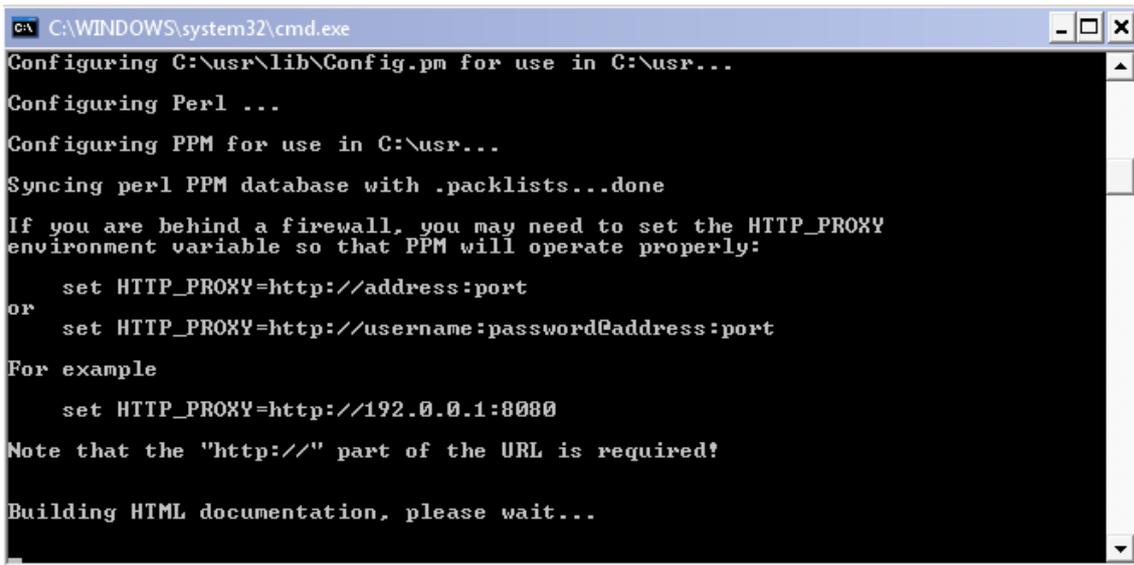
ActivePerl is ActiveState's quality-assured binary build of
Perl. In order to install ActivePerl you need to agree to the
ActivePerl Community License v2.1.

Did you read the LICENSE.txt file? [no] yes
Do you agree to the ActivePerl Community License v2.1? [no] yes

This installer can install ActivePerl in any location of your choice.
You do not need Administrator privileges. However, please make sure
that you have write access to this location.

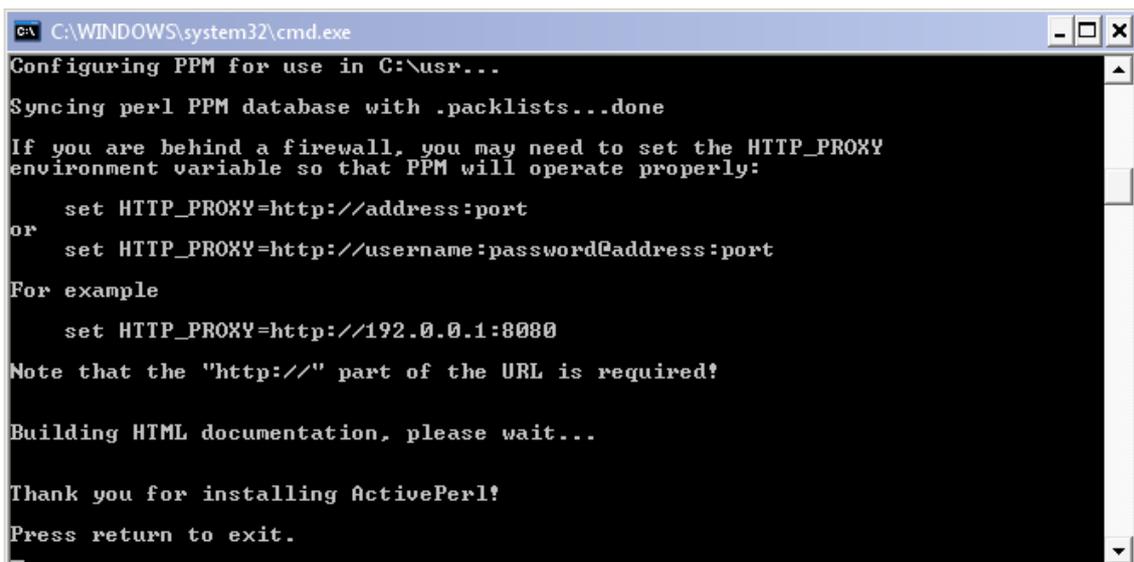
Enter top level directory for install? [C:\Perl] C:\usr_
```

El archivo de ejecución por lotes comenzará con la instalación de los scripts Perl necesarios para el funcionamiento de Koha.



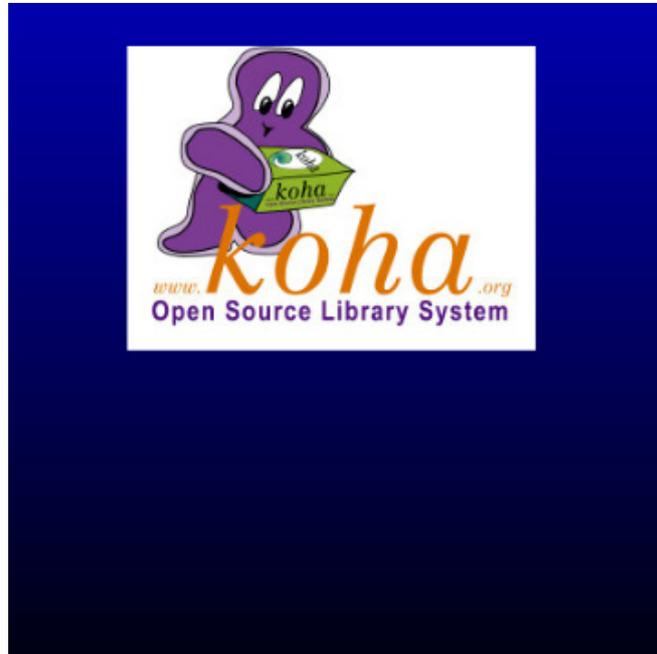
```
C:\WINDOWS\system32\cmd.exe
Configuring C:\usr\lib\Config.pm for use in C:\usr...
Configuring Perl ...
Configuring PPM for use in C:\usr...
Syncing perl PPM database with .packlists...done
If you are behind a firewall, you may need to set the HTTP_PROXY
environment variable so that PPM will operate properly:
    set HTTP_PROXY=http://address:port
or
    set HTTP_PROXY=http://username:password@address:port
For example
    set HTTP_PROXY=http://192.0.0.1:8080
Note that the "http://" part of the URL is required!
Building HTML documentation, please wait...
```

Al final tendremos la siguiente pantalla, la cual hace referencia a una instalación exitosa de ActivePerl 5.8 en Windows XP SP3. Para concluir tecleamos ENTER e inmediatamente saldremos del instalador.

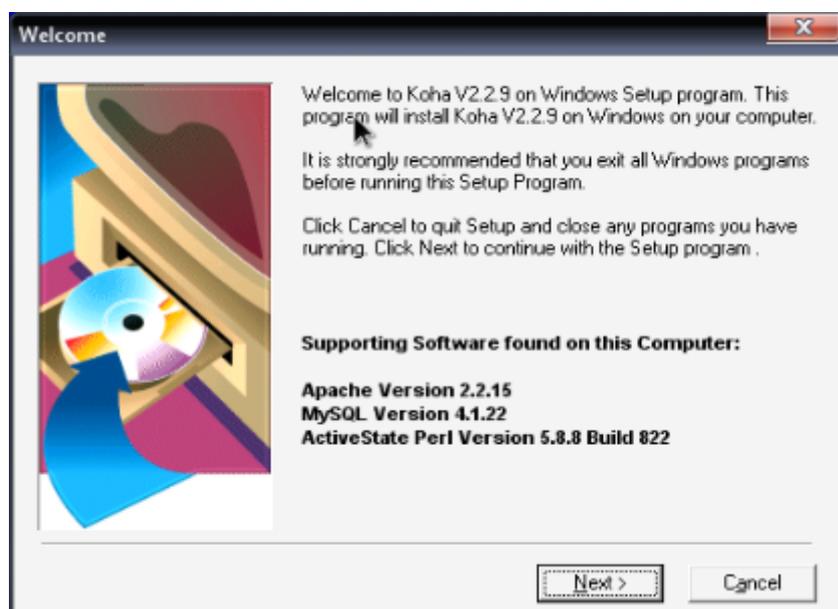


```
C:\WINDOWS\system32\cmd.exe
Configuring PPM for use in C:\usr...
Syncing perl PPM database with .packlists...done
If you are behind a firewall, you may need to set the HTTP_PROXY
environment variable so that PPM will operate properly:
    set HTTP_PROXY=http://address:port
or
    set HTTP_PROXY=http://username:password@address:port
For example
    set HTTP_PROXY=http://192.0.0.1:8080
Note that the "http://" part of the URL is required!
Building HTML documentation, please wait...
Thank you for installing ActivePerl!
Press return to exit.
```

Instalar Koha W32. Ejecutaremos el archivo de auto instalación descargado previamente desde el sitio oficial de Koha para Windows. Tendremos la siguiente pantalla:



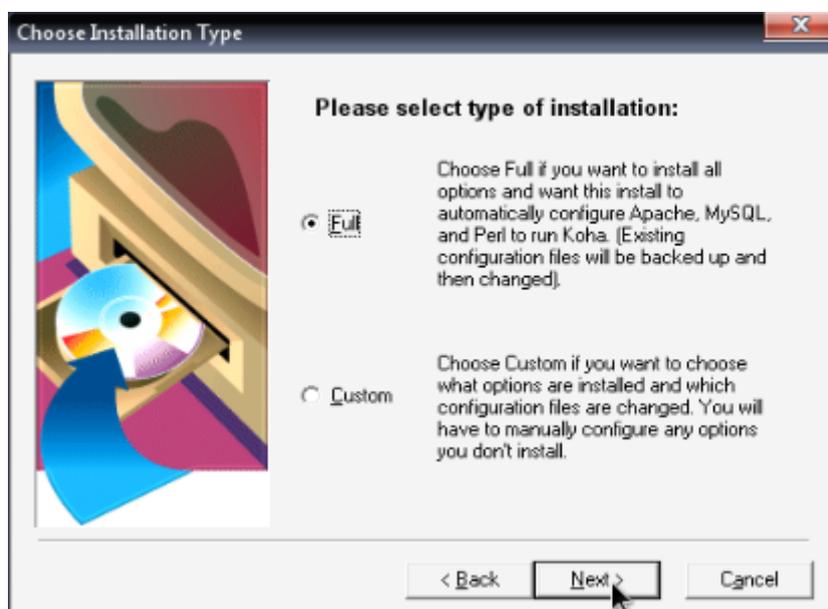
Inmediatamente, el asistente de instalación realizará un diagnóstico que verificará si los pre-requisitos necesarios se encuentran instalados de manera correcta en las rutas correspondientes. Si no hay errores, el asistente mostrará el software que tenemos disponible en nuestro sistema y podremos continuar.



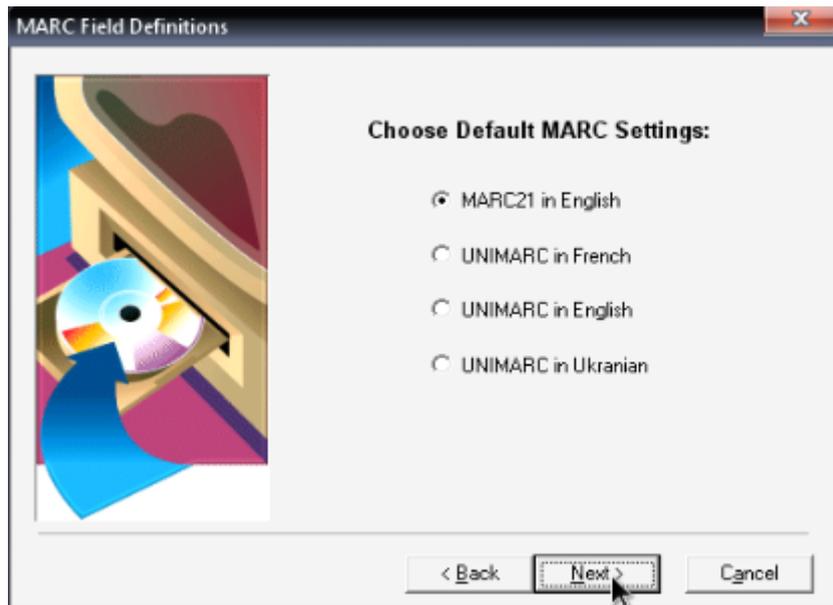
Koha se encuentra liberado bajo la licencia GNU/GPL; aceptamos los términos de uso específicos para Koha y continuamos.



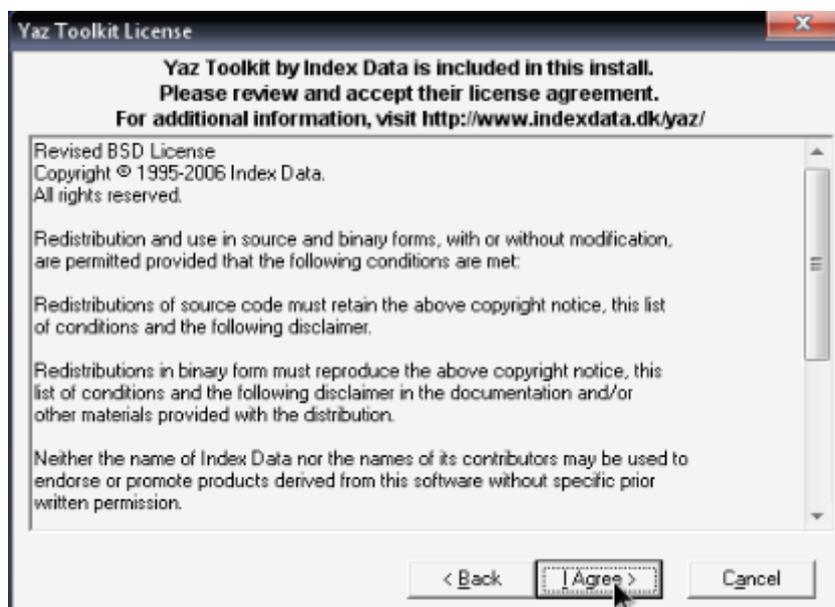
Seleccionamos la opción "instalación completa" (full) del menú y continuamos.



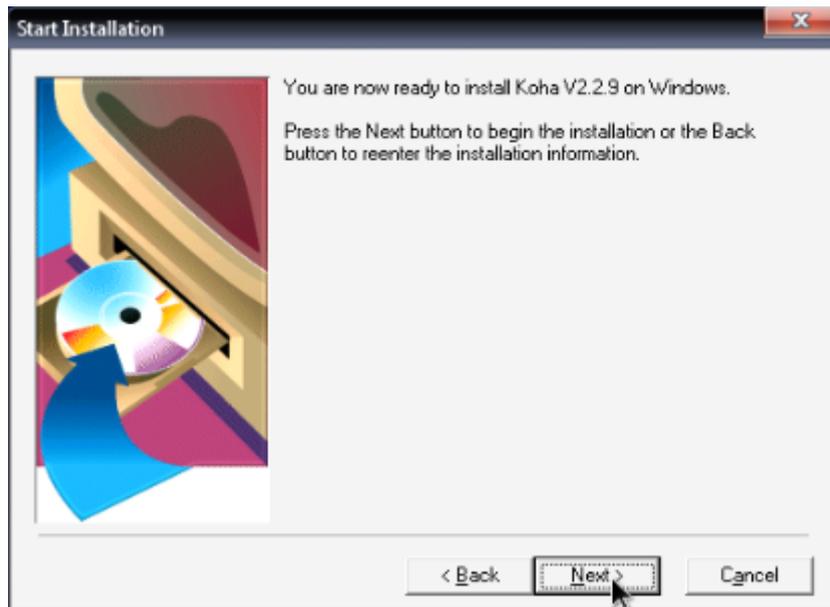
Seleccionamos MARC21 como opción predeterminada para nuestra configuración. Si requiriera, posteriormente se pueden hacer cambios al sistema desde el módulo de administración.



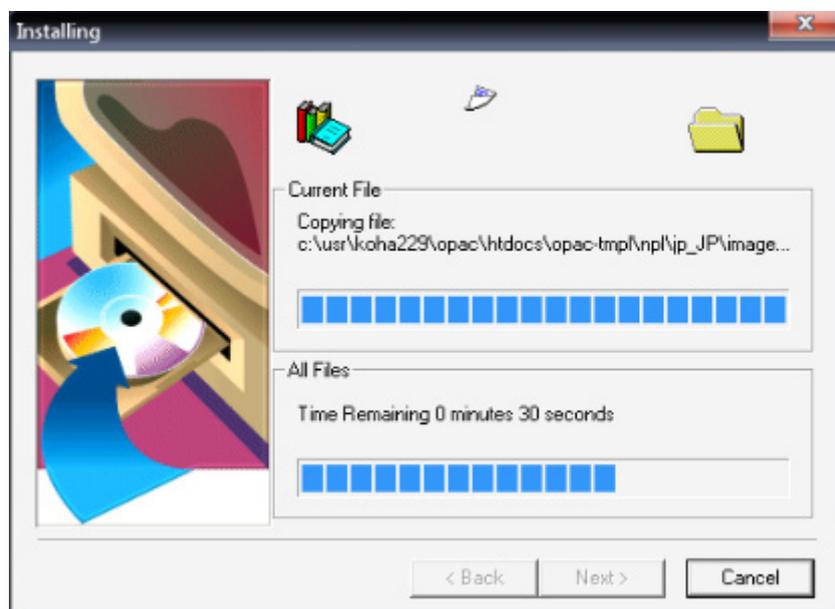
Koha W32 incluye por defecto la herramienta “YAZ” de Index Data en el paquete de instalación; aceptamos la licencia BSD y continuamos.



El asistente muestra el aviso de que una vez verificados los pre-requisitos necesarios y habiendo aceptado las licencias y términos correspondientes, se encuentra listo para proceder con la instalación del SIGB, por lo cual hacemos clic en continuar.

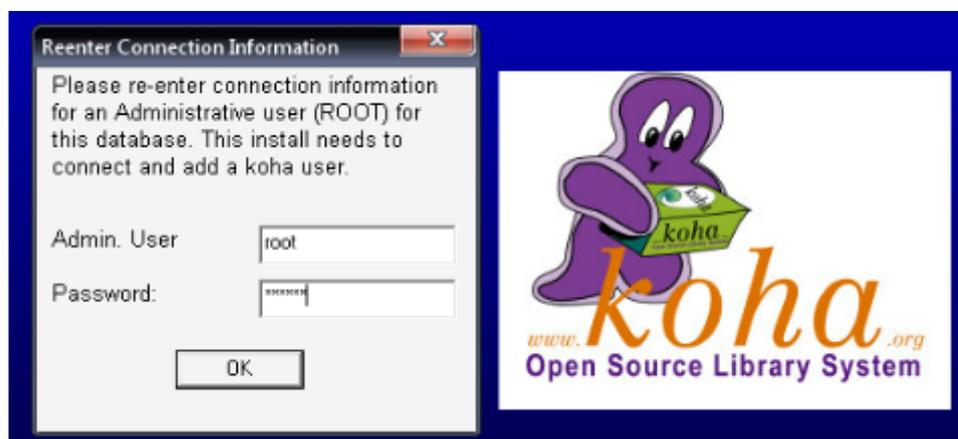


Comenzará la instalación del sistema, incluyendo los módulos Perl necesarios, que hemos instalado en la ruta C:\usr.

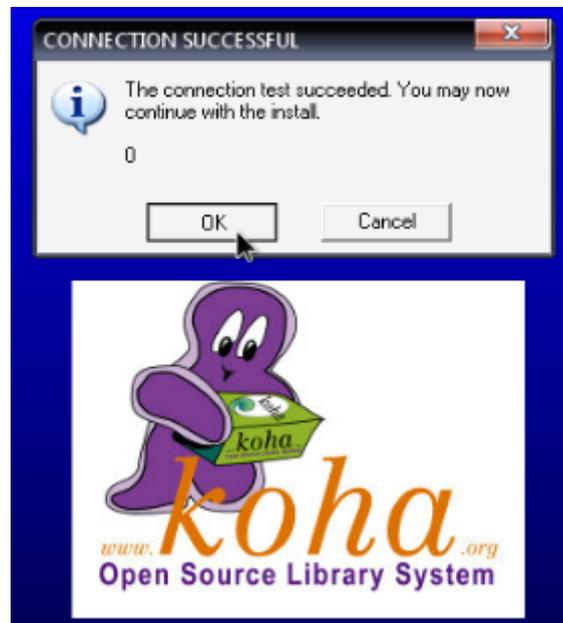




Durante el proceso de instalación Koha intentara conectarse al servidor MySQL, para lo cual será necesario proporcionar la contraseña que hemos establecido previamente durante la instalación del gestor.



No debería haber problema alguno después de ingresar nuestra contraseña; si todo está en orden, el asistente nos mostrará el mensaje de conexión realizada y con esto habremos concluido la instalación de Koha en Windows XP.



Para concluir, el asistente nos mostrará la pantalla de finalizar, aceptamos y con esto habremos llevado a cabo una instalación exitosa del Sistema Integrado para Bibliotecas “Koha” en el sistema operativo Windows XP.



Finalmente, antes de iniciar cualquiera de las dos interfaces del sistema, será necesario reiniciar nuestro equipo con el objetivo de poner en marcha los servicios que hemos instalado en nuestro sistema operativo.

A diferencia de Linux, en el cual accedamos a cada una de las interfaces de Koha tecleando en Firefox la dirección local del sistema para el OPAC y sumando el puerto 8080 para el módulo de administración, en Windows también accederemos mediante el navegador Web, escribiendo en la barra de direcciones la palabra INTRANET, para el módulo de administración (staff bibliotecario), y OPAC, para el catálogo. De igual manera, necesitaremos ingresar el nombre de usuario y la contraseña predeterminados por Koha, los cuales hemos especificado en la instalación sobre Ubuntu y que no varían en lo absoluto para este apartado.



3.4 Funcionalidad del sistema

La intranet de Koha

Como hemos mencionado en la parte que se refiere a la instalación del sistema, Koha divide su funcionalidad bajo dos interfaces de trabajo: el catálogo al público OPAC, que es la parte a la cual tienen acceso los usuarios de la biblioteca, y la intranet, que es la interfaz destinada al equipo de bibliotecarios encargados de llevar a cabo los procesos de la biblioteca. El acceso a cada interfaz se configura durante el proceso de instalación mediante un servidor Web.

Para acceder a la interfaz administrativa será necesario utilizar cualquier navegador (de preferencia Mozilla Firefox), proporcionar la dirección local del sistema, seguida del puerto 8080 y/o escribiendo la palabra Intranet en la barra de direcciones URL, y finalmente, proporcionar el nombre de usuario y contraseña predeterminados por el sistema, los cuales generalmente suelen ser: [koha] [koha].



The image shows a screenshot of the Koha administrative interface. On the left, there is a green sidebar with a white spiral graphic containing the text "Koha" and "Sistema de Biblioteca de Código Abierto Interfaz para bibliotecarios". Below this, it says "Koha : regalo, donación o contribución". To the right of the spiral is a vertical menu with the following items: "Adquisiciones", "Catálogo" (highlighted in purple), "Circulación", "Usuarios", "Autoridades", "Informes", and "Parámetros". The main content area on the right is titled "Catálogo" in a purple header. Below the header, there are several sections: "Buscar en el Catálogo" with a description: "Buscar un registro. Cuando este registro sea encontrado, puede verificar su estado, reservarlo, o ver dónde está"; "Añadir artículo" with a description: "Añadir un artículo directamente al catálogo de la biblioteca"; "Series" with a description: "Administre suscripciones a series y préstamos"; and "Estantes Virtuales" with a description: "Los estantes virtuales pueden ser usados por el bibliotecario o los patronos para construir bibliotecas virtuales dentro de una 'real'. El estante virtual puede ser usado por un profesor para ayudar a sus estudiantes a buscar la información sobre un tema específico. Esto puede ser usado por una biblioteca para definir 'los libros más valiosos para <alguna categoría de patrón>'".

A través de la intranet, los bibliotecarios pueden acceder a las distintas funciones del sistema. Una parte importante es la posibilidad de asignar permisos y niveles para cada elemento del equipo de bibliotecarios. Existen permisos que habilitan el uso de una función predeterminada a un grupo de funciones y permisos de gestión total ("superbibliotecario"). Esto tiene una enorme ventaja y es aplicable, como ejemplo, a un auxiliar bibliotecario o prestador de servicio social, quien puede colocar reservas y realizar préstamos a los usuarios teniendo acceso al sistema, pero que no puede catalogar ni llevar a cabo modificaciones que no le son pertinentes. Koha permite la asignación de los siguientes permisos:

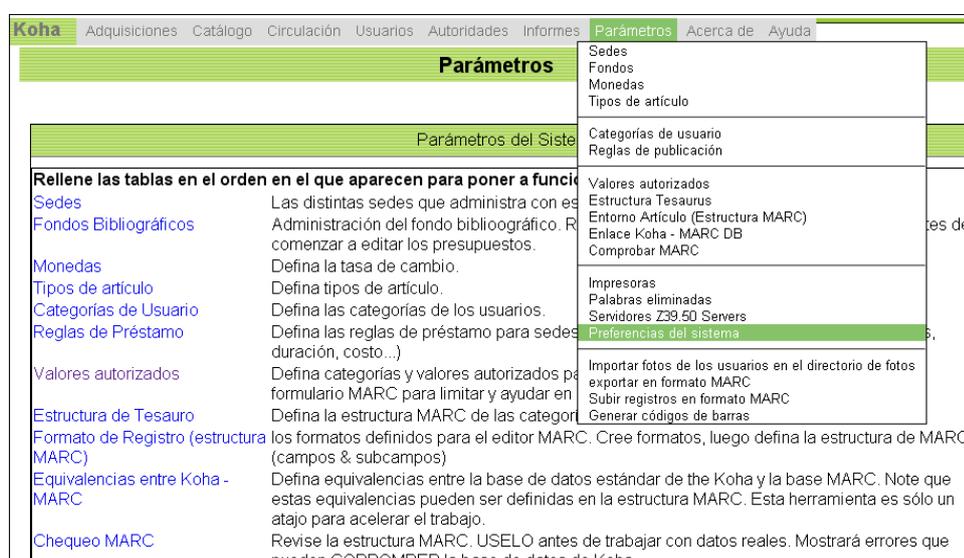
- “superbibliotecario” - acceso a todas las funciones del sistema.
- “Circulación” - acceso único al módulo de circulación.
- “Catálogo”- visualización del catálogo dentro de la intranet.
- “Parámetros” - modificación de los parámetros del sistema.
- “Socios” - alta, baja y modificación de registros de usuarios.
- “Permisos” - fija permisos de usuarios.
- “Reservas para otros” - permite colocar reservas para otros usuarios.
- “Prestamos” - Realiza préstamos.
- “Reserva para sí mismo” - permite colocar reservas para sí mismo.
- “Editar catálogo” - acceso al módulo de catalogación.
- “Actualizar demoras” - permite colocar sanciones por incumplimiento.
- “Adquisición” - gestión de adquisiciones.
- “Herramientas” - permite el uso de herramientas como importación y exportación de registros, impresión, códigos de barra, etc.

PERMISOS para Salinas de Flores, Valentina	
<input type="checkbox"/>	superlibrarian Access to all librarian functions
<input type="checkbox"/>	circulate Circulate books
<input type="checkbox"/>	catalogue View Catalogue (Librarian Interface)
<input type="checkbox"/>	parameters Set Koha system paramters
<input type="checkbox"/>	borrowers Add or modify borrowers
<input type="checkbox"/>	permissions Set user permissions
<input type="checkbox"/>	reserveforothers Reserve books for patrons
<input type="checkbox"/>	borrow Borrow books
<input type="checkbox"/>	reserveforself Reserve books for self
<input type="checkbox"/>	editcatalogue Edit Catalogue (Modify bibliographic/holdings data)
<input type="checkbox"/>	updatecharges Update borrower charges
<input type="checkbox"/>	acquisition Acquisition and/or suggestion management
<input type="checkbox"/>	management Set library management parameters
<input type="checkbox"/>	tools Use tools (export, import, barcodes)
Establecer Permisos	

Podemos ver que se pueden asignar permisos de manera muy específica, con la posibilidad de seleccionar varios de ellos y combinarlos para formar un grupo de funciones específicas para cada elemento de nuestro equipo de bibliotecarios.

3.4.1 Módulo de administración

El módulo de administración de Koha es uno de los elementos claves en la funcionalidad del sistema, ya que desde aquí se pueden configurar cada una de las características que le preceden. Será necesario llevar a cabo cada una de las configuraciones pertinentes, en relación con las características específicas de nuestra biblioteca, antes de comenzar a utilizar el sistema. Para acceder nos situamos dentro de la barra de Menú>Parámetros>Preferencias del sistema.



Preferencias del sistema

La configuración principal se encuentra en "preferencias del sistema". Cada variable de configuración está explicada y, según el tipo, puede ser una opción en una lista de opciones o un campo de texto libre, con la posibilidad de agregar preferencias personalizadas creando las variables correspondientes.

Administración de Preferencias del Sistema	
Administración	Adquisiciones Autoridades Intranet Catálogo Circulación Usuarios OPAC Otros
delimiter: ;	(Define el caracter separador para exportacion de reportes) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
IndependantBranches: 0	(permitir gestion de sucursales) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
KohaAdminEmailAddress: fernando_flores@koha.org	(Ingresa el Email del administrador del sistema) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
MIME: OPENOFFICE.ORG	(Define la aplicacion predeterminada para la lectura de reportes) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
MinPasswordLength: 5	(Numero de caracteres minimos para administradores) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
timeout: 1200	(Tiempo de inactividad para "cookies del sistema" (en segundos)) <input type="button" value="Modificar"/> <input type="button" value="Borrar"/>
<input type="button" value="Añadir una Preferencia del Sistema"/>	

Las variables más importantes a configurar son el idioma para el “OPAC”, donde podemos configurar también el nombre de la biblioteca, los logotipos usados, la disposición de elementos en el “OPAC”, la hoja de estilos css (para configuración de colores) y la activación de los servicios del “OPAC” para usuarios registrados.

En la pestaña Intranet, se configuran las opciones de visualización para la interfaz administrativa, de manera que los bibliotecarios que cuenten con nociones básicas de programación en HTML, puedan llevar a cabo una total personalización para la biblioteca, agregando textos a la barra de navegación, creando enlaces al sitio Web de la institución y/o a servicios Web ofrecidos por la misma, así como la posibilidad de crear diversas interfaces con distintos esquemas de colores y enlistar el orden en que cada una será presentada con sus diversas características.

Administración de Preferencias del Sistema								
Administración	Adquisiciones	Autoridades	Intranet	Catálogo	Circulación	Usuarios	OPAC	Otros
IntranetNav:						Modificar	Borrar	
<i>(Utiliza tablas HTML para agregar links de navegacion a la barra de menu)</i>								
TemplateEncoding: utf-8						Modificar	Borrar	
<i>(Especifica el tipo de codificacion de caracteres para Koha)</i>								
intranetcolorstylesheet: colors.css						Modificar	Borrar	
<i>(Especifica el nombre de la tabla de colores que sera usada por la Intranet)</i>								
intranetstylesheet:						Modificar	Borrar	
<i>(Ingresa la direccion URL alternativa para el esquema de colores)</i>								
template: default						Modificar	Borrar	
<i>(Orden preferencial para las interfaces de la Intranet)</i>								
Añadir una Preferencia del Sistema								

En las demás pestañas se configuran opciones específicas de cada uno de los módulos.

Sucursales

En la configuración de sucursales (sedes) se configuran los nombres, códigos y direcciones de cada una de las sucursales que son administradas con Koha. También se pueden definir categorías diferentes, por ejemplo, “sede principal” y/o “sede alterna”, según convenga.

Sedes					
Sedes					
Nombre	Código	Dirección	Categorías		
Colegio de Ciencias de la Información en México	002	Av. Insurgentes Nte. #41 Col: Tepeyac Delegación: Gvo. A. Madero Teléfono: 53259000 Fax: 56581111 Correo: Info@cimex.net	Sede alterna	Editar	Borrar
Escuela Nacional de Biblioteconomía y Archivonomía	003	Calz. Ticoman #645 Col. Sta Maria Ticoman Delegación: Gvo. A. Madero Teléfono: 53259000 Fax: 56581111 Correo: enba@sep.gob.mx	Sede Principal	Editar	Borrar

Fondos para la compra

Esta característica permite dar de alta los diferentes fondos que estarán disponibles en cada una de las sedes que son administradas con Koha y ligarlos directamente hacia “adquisiciones”. Cada fondo maneja su propio presupuesto y el tiempo que estará vigente para su utilización. De esta manera, Koha puede administrar ingresos provenientes de diversas fuentes, con montos específicos determinados para cada sucursal.

Administración del Fondo Bibliográfico				
Borrar				
Presupuestos 1 a 2 fuera de 2				
Fondo Bibliográfico	Nombre	Sede	Editar	Borrar
002	Fondo Tecnológico 2010	Escuela Nacional de Biblioteconomía y Archivonomía		
	Presupuesto	Sede	Añadir Presupuesto	
	12/20/2009 - 03/03/2010 :			
	65000.00			
01	Fondo Hemerografico	Colegio de Ciencias de la Información en México		
	Presupuesto	Sede	Añadir Presupuesto	
	12/20/2009 - 03/03/2010 :			
	45000.00			
Añadir Fondo				

Monedas

Aquí se dan de alta diferentes monedas, con su respectivo tipo de cambio, frente a la moneda local. Esto será necesario si se pretenden realizar adquisiciones de proveedores foráneos, e incluso locales, que bajo sus términos de compra, establezcan la adquisición de su material mediante monedas como el dólar y/o el euro.

Administración de monedas						
Moneda	Equivalencia					
DOLAR	13.42000	Editar		Borrar		
EURO	17.80000	Editar		Borrar		
PESO	1.00000	Editar		Borrar		

Añadir moneda

Tipos de Ítems

Aquí llevaremos a cabo la configuración para los distintos tipos de materiales disponibles en la biblioteca. Conviene, por obvias razones, configurarlo antes de comenzar con el proceso de catalogación. A cada tipo de ítem se le asigna un código que estará disponible en el módulo de catalogación, así como indicaciones sobre renovación y costos (en el caso que la biblioteca cobre algún costo por el préstamo de material). El tipo de ítem especificado, permitirá visualizar estadísticas en el módulo de informes.

Administración de tipos de artículo						
Código	Descripción	no prestable	Renovable	Costo	Editar	Borrar
AV	Materiales Audiovisuales		3 vez(veces)	0.00		
BOOK	Monografías		5 vez(veces)	0.00		
CC	Colección de consulta	Si				
CE	Colecciones especiales	Si				
SER	Publicaciones periódicas y seriadas	Si				

Añadir tipo de artículo

Categorías de usuarios

De manera similar a los tipos de ítems que hemos especificado, será necesario definir categorías de usuarios antes de comenzar a trabajar con el sistema. Para cada categoría se indica: código, periodo de alta (en años), límites de edad (máxima y mínima), precio de asociación, costo de las reservas (en caso de que se cobren) y aviso de no demora en materiales.

Administración de Categorías								
							ACEPTAR	
Categoría	Descripción	Período de asociación	Costo de asociación	Límite de edad superior	Edad Requerida	Retraso Reserva		
UI	Usuarios internos de la institución	3	0.00	18	18	Si	0.00	 
UE	Usuarios externos de la institución	0	0.00	18	18	Si	0.00	 
IA	Institución asociada	5	500.00	18	18	No	50.00	 
Añadir Categoría								

Reglas de préstamo

Las reglas de préstamo son la base para el módulo de circulación. Aquí se configuran para cada sucursal (o para el sistema de bibliotecas de manera general):

- Extensión del préstamo en días y por tipo de material.
- Cantidad de ítems que puede tener un usuario, según el tipo de material o en total.
- Cobro de multas por tipo de ítem y categoría de usuario (monto y vigencia del mismo).

En caso de no haber llevado a cabo una configuración previa para cada tipo de material, Koha tratará de encontrar un valor realizando una búsqueda en la configuración de otros tipos de ítems, categorías de socios, sucursales, y en caso de no estar configurado nada, realiza los préstamos con valores predeterminados que abarcan un máximo de 5 ítems por el término de 21 días.

Seleccione una sede : Escuela Nacional de Biblioteconomía y Archivonomía ▾

	Institución asociada Préstamo / Cuenta	Usuarios externos de la institución Préstamo / Cuenta	Usuarios internos de la institución Préstamo / Cuenta	* Préstamo / Cuenta
Colección de consulta	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>
Colecciones especiales	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>
Materiales Audiovisuales	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>
Monografías	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>
Publicaciones periódicas y seriadas	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>
*	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>	<input type="text"/> / <input type="text"/>

Guardar

Estructura bibliográfica

Aquí se configuran las plantillas predeterminadas de etiquetas para el editor MARC. De igual manera, se pueden dar de alta nuevos campos, los subcampos que serán utilizados en cada plantilla, su ubicación, relación con otros campos, valores por defecto, plantillas personalizadas, así como enlaces con bases de datos de autoridades.

Podemos afirmar que esta es una de las partes más complejas y que consume mayor tiempo durante la configuración del sistema, pero es básica para poder adaptar Koha a necesidades de catalogación específicas de la biblioteca.

El editor incluye una estructura predeterminada, la cual sirve de base para la configuración de las diferentes plantillas del editor MARC. Conviene dejarlo lo más extenso posible ya que también configura la importación de registros MARC a Koha; si realizamos una reducción en las etiquetas incluidas por defecto los campos suprimidos no serán importados al sistema.

administración de estructura de etiquetas MARC para formato MARC por defecto

Select a framework

Predeterminado ▾ 0 ACEPTAR

Display only used Tags/Subfields

NOTE : if you change the link between a MARC subfield and a non-MARC field, ask your administrator to run misc/rebuildnonmarc.pl script.

Etiqueta	Lib	Repetible	Obligatorio	Autorizado valor	Subcampos	Editar	Borrar
000	LEADER	No	Si		subcampos		
001	CONTROL NUMBER	No	No		subcampos		
003	CONTROL NUMBER IDENTIFIER	No	Si		subcampos		
005	DATE AND TIME OF LATEST TRANSACTION	No	Si		subcampos		
006	FIXED-LENGTH DATA ELEMENTS--ADDITIONAL MATERIAL CHARACTERISTICS--GENERAL INFORMATION	Si	No		subcampos		
007	PHYSICAL DESCRIPTION FIXED FIELD--GENERAL INFORMATION	Si	No		subcampos		
008	FIXED-LENGTH DATA ELEMENTS--GENERAL INFORMATION	No	Si		subcampos		
009	PHYSICAL DESCRIPTION FIXED-FIELD FOR ARCHIVAL COLLECTION (VM) [OBSOLETE]	Si	No		subcampos		
010	LIBRARY OF CONGRESS CONTROL NUMBER	No	No		subcampos		

Enlaces Koha - base MARC

En este apartado es posible configurar las relaciones entre los diferentes campos que se van utilizar en el módulo de circulación y la base bibliográfica. El sistema de préstamos de Koha funciona sobre un grupo de campos predeterminados que identifican al ítem al momento de ser prestado. De esta manera, en el módulo de circulación no visualizamos el registro MARC en forma completa (aunque se puede acceder con un sólo clic a la visualización en el OPAC). Conviene dejar los valores predeterminados de la manera que se encuentran, a menos que se desee implementar un nuevo formato. De ser así, esta es la herramienta para adaptar el módulo de circulación a cualquier formato bibliográfico, para lo cual ejecutaremos el script **misc/rebuildnonmarc** en modo administrador, llevando a cabo el enlace de los distintos campos creados.

enlaces MARC

NOTAS :
 Si cambia el enlace entre una un subcampo MARC y un campo no-MARC, pídale a su administrador que ejecute el script misc/rebuildnonmarc.
 Si cambia un enlace aquí, TODOS los formatos serán modificados.

	Etiqueta	Subcampo	Lib	
biblioitems	090	d	Koha biblioitemnumber	
volume	440	v	Volume number/sequential designation	
number	440	n	Number of part/section of a work	
classification	942	j	Location (call number prefix code)	
itemtype	942	c	Item type	
isbn	020	a	International Standard Book Number	
issn	022	a	International Standard Serial Number	
dewey	942	k	Classification base (DDC to decimal or LCC letter class padded after single letter classes with trailing 0)	
subclass	942	l	Classification subclass (DDC after decimal or LCC number after letters)	
publicationyear				
publishercode	260	b	Name of publisher, distributor, etc	

Comprobar MARC

Esta herramienta será necesaria después de haber llevado a cabo alguna modificación de relaciones entre la base Koha y la base bibliográfica. Este diagnóstico permitirá detectar errores en la configuración, que de alguna manera pueden afectar en el desempeño del sistema. Si después de realizar el diagnóstico el resultado es “OK”, el sistema puede ser usado.

Revisando configuración MARC	
Prueba	Resultado
ACEPTAR	<i>número de artículo : el campo número de artículo está mapeado a un campo en la sección -1</i>
ACEPTAR	<i>Todos los campos de artículo estan en el mismo campo y en la etiqueta artículo</i>
ACEPTAR	<i>Solamente 1 etiqueta de marc se ha mapeado a los artículos</i>
ACEPTAR	<i>tabla de tipos de artículo llena con al menos 1 valor</i>
ACEPTAR	<i>biblioitems.itemtype definido</i>
ACEPTAR	<i>tabla de sedes contiene por lo menos un valor</i>
ACEPTAR	<i>sede propietaria definida</i>
ACEPTAR	<i>sede definida</i>
ACEPTAR	<i>biblionumber y biblioitemnumber mapeados correctamente</i>
ACEPTAR	<i>ningún valor NULO en el código del formato</i>
La configuración está OK, no hay errores en su tabla de parámetros MARC	

Parámetros extras

En esta sección se configuran las impresoras que el sistema utilizará, el listado de términos vacíos (palabras que no serán indexadas en el catálogo para su búsqueda) y los servidores Z39.50 que usará Koha para la catalogación por importación.

Administración de Servidores Z39.50

Sitio	Hostname	Puerto	Base de Datos	ID de usuario	Clave	Marcado	Orden	Sintaxis		
AMICUS	amicus.collectionscanada.ca	210	NL		1	1	1	USMARC		
LIBRARY OF CONGRESS	z3950.loc.gov	7090	voyager		1	2		USMARC		

Administración de Impresora

Nombre	Cola	Tipo		
DATAMAX E4203	4	Impresora de códigos	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
EPSON stylus color c79	2	Blanco y negro / col	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
HP laser Deskjet MF	1	Color	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
Star sp 512	3	Impresora de puntos	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>

Administración de Palabras eliminadas

Palabra	
ALGUNOS	<input type="button" value="Borrar"/>
AQUEL	<input type="button" value="Borrar"/>
CUAL	<input type="button" value="Borrar"/>
DESDE	<input type="button" value="Borrar"/>
EL	<input type="button" value="Borrar"/>
LA	<input type="button" value="Borrar"/>
LAS	<input type="button" value="Borrar"/>
LOS	<input type="button" value="Borrar"/>

Herramientas

Esta sección ofrece una serie de herramientas como complemento al sistema. Aquí se pueden exportar registros bibliográficos, de acuerdo al rango de adquisición, tipo de ítem, sucursales, signaturas topográficas, cantidad, etc. De igual manera, podemos importar registros, aunque es conveniente utilizar la opción disponible en el módulo de catalogación.

Una herramienta extra incluye la posibilidad de generar códigos de barra basados en las características de los registros disponibles, e imprimirlos si contamos con el hardware necesario.

Generador de Código de barras

Generar Códigos de Barras a partir de códigos de Inventario

- Seleccionar un rango de códigos de inventario. Puede seleccionar un rango continuo o códigos individuales
- Seleccione el tipo estándar para generar códigos de barra.
- Defina el tamaño de página para la salida PDF.
- Dependiendo del tamaño de la página, Koha mostrará cómo se acomodarán los códigos de barra. Puede definir en qué punto de la página comenzar a imprimir.

Tipo de intervalo

Código de País

Texto bajo la etiqueta

Desde

Hasta:

Tamaño de Página
 [\[Ir a Configuración de Impresora\]](#)

Si este campo esta vacío, el autor y el título serán usados en su lugar

Número de la etiqueta para iniciar la impresión				
<input checked="" type="radio"/> Etiqueta 1	<input type="radio"/> Etiqueta 2	<input type="radio"/> Etiqueta 3	<input type="radio"/> Etiqueta 4	<input type="radio"/> Etiqueta 5
<input type="radio"/> Etiqueta 6	<input type="radio"/> Etiqueta 7	<input type="radio"/> Etiqueta 8	<input type="radio"/> Etiqueta 9	<input type="radio"/> Etiqueta 10

3.4.2 Módulo de circulación

El módulo de circulación de Koha es el módulo desde el cual se llevan a cabo los préstamos. Este módulo se encarga también de gestionar a los usuarios del sistema (nombrados “socios” en Koha).

Préstamos

El procedimiento de préstamo comienza con la búsqueda del usuario. Es posible realizar búsquedas por apellido y por número de carnet. Si la búsqueda recupera un sólo usuario, de manera inmediata se le puede prestar el material. Si existen varios usuarios con el mismo apellido, Koha devuelve un listado de usuarios para seleccionar el buscado.

Una vez encontrado el usuario, habrá que ingresar el número de inventario del ítem o en su defecto el código de barras (si se ha generado previamente), y oprimir el botón “préstamo” para registrarlo en el sistema.

Circulación: Préstamos

Parámetros

Sede: Escuela Nacional de Biblioteconomía y Archivonomía, **Impresora:** HP laser Deskjet MF [Cambiar configuración](#)

Información de usuario

Valentina Salinas de Flores (1) [detalle](#) [editar](#)

El usuario caducará pronto

Calle 23 #6 Lindavista México
53259000
Categoría: UI
0 Préstamos

Inserte el Código de Barras del ejemplar

Código de barras del ejemplar:

Día Mes Año

Mantener Fecha de Devolución

Préstamos

Fecha de Devolución	Código de barras	Título	¿Renovar?	¿Devolver?
14/02/2009	3456 7890	Open...	<input type="button" value="Renovar"/>	

La fecha de devolución es calculada automáticamente a partir de las reglas de préstamos que combinan “categoría de usuario” contra “tipo de material”, y las características previamente definidas, aunque también es posible asignar una fecha de devolución diferente a la que el sistema establece por defecto.

Si el usuario tiene materiales vencidos el sistema no realiza el préstamo y de inmediato genera un aviso para el bibliotecario. Sin embargo, es posible realizar el préstamo en una segunda instancia; lo mismo sucede si el usuario ya tiene en préstamo la máxima cantidad de ítems permitidos.

Una vez realizado el préstamo, es posible imprimir un ticket o una página con toda la información referente a los préstamos otorgados al usuario, incluyendo los ejemplares con los que cuenta en el momento, fechas de devolución de cada ejemplar y reservas pendientes. Es posible automatizar este proceso en Koha de manera que se auto ejecute después de cada préstamo.

Devoluciones

Los procesos de préstamo y devolución son de muy fácil uso en el sistema. De igual manera, aquí lo único que debe ingresar el bibliotecario es el número de inventario del ejemplar y/o escanear el código de barras para dar de baja el ítem prestado. En caso que corresponda a cualquier material sobre el cual está pendiente una reserva, el sistema avisa al bibliotecario para que separe el

ejemplar. Cabe destacar que estos procesos se agilizan si se cuenta con un lector de códigos de barra.

Circulación: Devoluciones

Sede: Escuela Nacional de Biblioteconomía y Archivistomía, Impresora: HP laser Deskjet MF Cambiar configuración

Inserte el Código de Barras del ejemplar

Código de barras del ejemplar:

Mensajes

No Está Prestado

Items Devueltos

Fecha de Devolución	Código de barras	Título	Autor	Tipo	Usuario
No Está Prestado	3456 7890	Software libre para bibliotecas		CC	

Listas de reservas

Dentro del módulo de circulación el personal bibliotecario tiene acceso constante a la lista de materiales reservados. El sistema dará de baja de manera automática a aquellos ítems que hayan sobrepasado el tiempo estipulado por la biblioteca.

Transferencias

El módulo de circulación también gestiona la transferencia de materiales entre diferentes sedes. Cabe destacar que una transferencia no es un préstamo entre cada sede, sino una asignación permanente a una sucursal determinada.

Módulo de usuarios

El módulo conocido en Koha como “socios” se encarga de todo lo que conlleva la administración de los usuarios. Permite realizar búsquedas por apellido y número de credencial (carnet), ingresar nuevos usuarios al sistema, asignar permisos de acceso para cada uno, modificar registros de usuarios y dar de baja a los mismos en caso de ser necesario.

Alta y modificación de usuarios

El formulario diseñado para dar de alta y modificar registros de usuarios existentes en el sistema, es muy completo; los datos que se registran son:

- Asignación de folio para credencial.
- Apellido, nombres, iniciales, nombre preferido.
- Sexo.
- Fecha de nacimiento.
- Dirección.
- Categoría de usuario.
- Teléfono de domicilio.
- Correo electrónico.
- Mensaje personal (el usuario vera este mensaje al realizar búsquedas en el OPAC).
- Nota de circulación.
- Datos laborales.
- Contactos alternativos.
- Sede a la que pertenece el usuario.
- 2 campos de control libres que pueden ser usados para fines estadísticos.

Añadir Usuario			
Member# , Número de carnet* 2			
USUARIO PERSONAL DETALLES			
Saludo Sin Titulo	Iniciales	Nombre*	Apellidos*
Nombre preferido	Fecha de Nacimiento (mm/dd/yyyy)	Categoría Institución asociada	
<input type="radio"/> Female <input checked="" type="radio"/> Male			
DIRECCIÓN DEL USUARIO			
Dirección Postal*	Cod. Postal	Ciudad*	Sede de inscripción
Dirección alternativa			Escuela Nacional de Biblioteconomía y Archivonomía
Teléfono de casa	Fax	Correo electrónico	
Teléfono móvil			
DETALLES DE CONTACTO ALTERNATIVOS			
Nombre completo	Teléfono	Relación	
Notas			
USO INTERNO			
Fecha de alta (dejar en blanco para hoy)	Fecha de Vencimiento (dejar en blanco para autocalcular)		
Mensaje para el usuario (aparece en el OPAC)			
Nota de Circulación (aparece durante los préstamos/devoluciones)			
PERMISOS	<input type="radio"/> Si <input checked="" type="radio"/> No		
Same no address	<input type="radio"/> Si <input checked="" type="radio"/> No		
Debarred	<input type="radio"/> Si <input checked="" type="radio"/> No		
Lost	<input type="radio"/> Si <input checked="" type="radio"/> No		
Los siguientes 2 campos están disponibles para su uso interno. Pueden ser útiles para propósitos estadísticos			
Campo de control 1			
Campo de control 2			
<input type="button" value="Guardar"/>			

Información del usuario

Además de los datos del usuario (socio), el sistema permite administrar opciones adicionales a partir de la página de información de cada usuario.

Valentina Salinas Ruiz	
Registro de Membresía Valentina Salinas Ruiz (Valentina, VSF) Número de Carnet: 1 Dirección Postal: Calle 23 #6 Lindavista, 06050 México Dirección: . Teléfono de casa: 53259000 Teléfono móvil: . Fax: . Correo: pekitashermosa@yahoo.com Mensajero de texto: Bienvenida Valentina Número de Membresía: 1 Categoría de Membresía: Usuarios internos de la institución Registrado: . Vencimiento: . Sede de inscripción: Escuela Nacional de Biblioteconomía y Archivonomía Fecha de Nac: %s 02/14/1987 Sexo: F Contacto: Teléfono: Relación: Notas: Garantías: Nota de la circulación: Plugin :%s (no hay clave) Modificar Borrar Préstamo de libros Cambio de clave Añadir Hijo Modificar permisos de usuario Imprimir Página Imprimir	Permisos de Usuario <ul style="list-style-type: none">El carnet de usuario aparece como perdido. Cuentas & Cargos Deuda Total 0.00 Ver Cuenta Pagar Deudas Ejemplares reservados Registro de Lectura Ninguno Ejemplares actualmente prestados Ninguno

A partir de ésta, podremos definir el nombre de acceso y contraseña a utilizar por el usuario para acceder al OPAC y a sus funciones extendidas. Esta página permite también establecer cargos al usuario por multas, renovar materiales simultáneamente, visualizar los materiales que ha pedido a reserva, así como un historial de lectura y la posibilidad de agregar un hijo, al registro del usuario.

Valentina Salinas Ruiz, Valentina	
Nuevo Id de Usuario:	Valentina
Nueva Clave	XXXXXXXXXX
Confirmar Password	

Multas e inhabilitaciones

Koha permite calcular multas de manera automática. El monto de las mismas y la manera de calcularlas se configuran en el módulo de administración del sistema. En el caso de las inhabilitaciones y cargos por diferentes conceptos, el bibliotecario deberá colocar las mismas manualmente en el registro del usuario y deberán ser removidas también manualmente.

Factura Manual	
Número de usuario	1
Tipo	Ejemplar perdido
Número de ejemplar	
Descripción	
Monto	
<input type="button" value="Añadir"/>	

3.4.3 Módulo de catalogación

Koha nos da la opción de catalogar el material bibliográfico de nuestra biblioteca de diversas formas. Es necesario que el catalogador defina si utilizará o no el formato MARC.

- **Catalogación simplificada.** La catalogación simplificada permite la aplicación del módulo en una biblioteca que no cuente con personal profesional del área. Los campos son pocos y el único campo obligatorio es el de título.
- **Catalogación en MARC.** Si el bibliotecario opta por llevar a cabo la catalogación en MARC, Koha dispone del formato MARC 21, en el cual los campos ya vienen previamente cargados y configurados. De igual manera, es posible implementar un nuevo formato no MARC de carácter extenso con la información que convengamos necesaria.

Plantillas de carga. El editor de catalogación permite también una configuración específica, según las necesidades y características de cada biblioteca, conforme a esto será posible definir diferentes plantillas de carga.

Por ejemplo, se pueden definir plantillas diferentes para cada tipo de material (monografías, videos, dvd's, mapas, etc.) que incluyan sólo aquellos campos que aplican al material catalogado y que respondan a la política de la biblioteca respecto a un tipo de material específico. En este caso, lo conveniente será definir únicamente los campos que realmente serán usados de manera habitual. En caso de requerir posteriormente de algún otro campo, es posible también editar el registro utilizando la plantilla general (que incluye todos los campos MARC predeterminados) y agregar datos extras a la plantilla personalizada.

Editor MARC

Una característica de peso en Koha es el potente editor MARC incluido en el módulo de catalogación. El editor carga todos los campos y subcampos definidos previamente en la plantilla y son organizados por pestañas que despliegan, una a una, la estructura del formato. Como ejemplo, en la siguiente imagen se muestra la pestaña correspondiente a los 200, ingresando datos para catalogación referentes al título y mención de responsabilidad, así como al pie de imprenta del ítem en cuestión.

0	210	- ABBREVIATED TITLE +	
	2 a b		
1	222	- KEY TITLE +	
	a b		
2	240	- UNIFORM TITLE	
	a h l r s		
3	243	- COLLECTIVE UNIFORM TITLE	
	a h l s		
	245	00 - TITLE STATEMENT	
4	▲ a	Title *	Software libre para bibliotecas /
5	▲ c	Statement of responsibility, etc	editado por Oscar Arriola Navarrete ... [et al.].
6	▲ b	Remainder of title	
	▲ h	Medium	
7	246	- VARYING FORM OF TITLE +	
	a b h i		
8	247	- FORMER TITLE +	
	a b h		
9	250	- EDITION STATEMENT	
	▲ a	Edition statement	
	▲ b	Remainder of edition statement	
	260	- PUBLICATION, DISTRIBUTION, ETC. (IMPRINT) +	
	▲ a	Place of publication, distribution, etc	México : +
	▲ b	Name of publisher, distributor, etc	Network Solutions. +
	▲ c	Date of publication, distribution, etc	2006. +
	270	- ADDRESS +	

Antes de proceder a agregar el ítem al catálogo, el sistema realiza una revisión del registro y verifica que hayan sido ingresados todos los datos en los campos y subcampos previamente definidos. En el caso que falten campos obligatorios, el editor informará qué campos es necesario llenar antes de proceder, coloreándolos para su fácil identificación.

0	000 - LEADER	
1	001 - CONTROL NUMBER	
2	003 - CONTROL NUMBER IDENTIFIER	
3	005 - DATE AND TIME OF LATEST TRANSACTION	
4	006 - FIXED-LENGTH DATA ELEMENTS--ADDITIONAL MATERIAL CHARACTERISTICS--GENERAL INFORMATION +	
5	007 - PHYSICAL DESCRIPTION FIXED FIELD--GENERAL INFORMATION +	
6	008 - FIXED-LENGTH DATA ELEMENTS--GENERAL INFORMATION	
7	010 - LIBRARY OF CONGRESS CONTROL NUMBER	
8	▲ a LC control number	b z
9	015 - NATIONAL BIBLIOGRAPHY NUMBER +	
	▲ 2 Source	
	▲ a National bibliography number	

Editor MARC avanzado

Una opción extra, es el editor avanzado de MARC, el cual no incluye la descripción de los campos y subcampos, únicamente indica los números y códigos pertenecientes a cada uno. Esta opción está enfocada a catalogadores experimentados que conocen los campos y subcampos que van a utilizar.

0	100	▲ 4		+
1		▲ a		
2		▲ d		
3		▲ e		+
4		▲ q		
5	110	▲ 4		+
6		▲ a		

Plantillas para campos codificados

Las plantillas adicionales son de gran ayuda en los campos y subcampos codificados, en los cuales es necesario contar con un mayor control sobre cada una de las posiciones y opciones disponibles con las que éstos cuentan. Mediante este tipo de plantillas que son llamadas desde el editor, usuarios novatos del sistema pueden llevar a cabo sin dificultades la codificación correcta del campo “leader” o los relacionados con el tipo de control, por ejemplo.

Listas de valores autorizados

Koha incluye un editor para crear listas de valores autorizados tales como códigos de países, idiomas, etc. Éste puede ser configurado de manera que en determinados subcampos sólo se carguen los datos de estas listas.

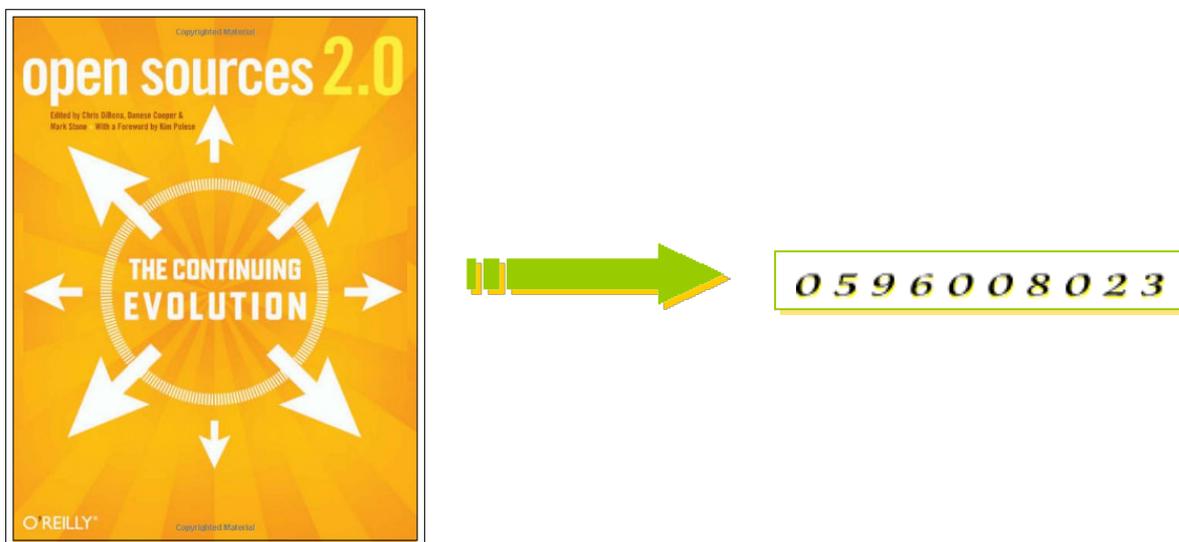
Catalogación por copia y protocolo Z39.50

Una de las características interesantes de Koha es la posibilidad de catalogar material, importando registros desde servidores Z39.50. De igual manera, la catalogación por copia puede llevarse a cabo subiendo registros bibliográficos al repositorio. El repositorio es un almacén de registros bibliográficos externo al catálogo; cuando se desea agregar un registro nuevo, el sistema busca en el repositorio un registro que cumpla con las condiciones de búsqueda ingresadas. Una vez seleccionado un registro desde el repositorio, es posible agregarlo al catálogo, editarlo y agregarle ejemplares.

Koha tiene preestablecidos por defecto, algunos servidores de acceso para importar registros remotamente mediante Z39.50; para el siguiente ejemplo utilizaremos el servidor de la Library of Congress en conjunto con el catálogo de registros bibliográficos canadiense AMICUS.

El proceso es muy sencillo y requiere al menos de un dato para localizar el ítem en el servidor remoto, e importarlo completamente de manera muy fácil a nuestro sistema, ahorrando con esto tiempo valioso en los procesos de catalogación de nuestra biblioteca.

Supongamos que el siguiente título, entre otros, llega a nuestra biblioteca y queremos realizar la catalogación importando las etiquetas MARC mediante Z39.50. Utilizaremos como dato principal el ISBN del ítem para llevar a cabo la búsqueda.



Ingresamos el ISBN dentro de la etiqueta 020 de la plantilla MARC y hacemos clic en el botón búsqueda Z39.50 de la parte superior.

020 - INTERNATIONAL STANDARD BOOK NUMBER +

▲ a International Standard Book Number

▲ c Terms of availability

022 - INTERNATIONAL STANDARD SERIAL NUMBER +

▲ 2 Source

▲ a International Standard Serial Number

024 - OTHER STANDARD IDENTIFIER +

▲ 2 Source of number or code

▲ a Standard number or code

▲ c Terms of availability

▲ d Additional codes following the standard number or code

De acuerdo a los parámetros de configuración establecidos para este tipo de servidores, el sistema nos mostrará a cuáles tenemos acceso para llevar a cabo la búsqueda. Seleccionamos los deseados y oprimimos el botón “búsqueda en Internet”.

Buscar

amicus.collectionscanada.ca

z3950.loc.gov

Los resultados nos muestran que los datos del ítem están disponibles para importar desde el servidor perteneciente a la Library of Congress, verificamos y hacemos clic en “Importar”.

Resultados			
Servidor	Título	Autor	ISBN
z3950.loc.gov	Open sources 2.0 :		0596008023(pbk) Importar

Podremos ver que el sistema ingresa de manera automática, los datos pertenecientes a cada una de las etiquetas correspondientes en la plantilla MARC y a las cuales podemos acceder mediante el sistema de pestañas que divide al formato.

0	210	- ABBREVIATED TITLE +	
1	222	- KEY TITLE +	
2	240	- UNIFORM TITLE	
3	243	- COLLECTIVE UNIFORM TITLE	
4	245 00	- TITLE STATEMENT	
5	▲ a	Title *	Open sources 2.0 :
6	▲ b	Remainder of title	the continuing evolution /
7	▲ c	Statement of responsibility, etc	edited by Chris DiBona, Danese Cooper, Mark Stone.
8	▲ h	Medium	
9	246	- VARYING FORM OF TITLE +	
0	247	- FORMER TITLE +	
1	250	- EDITION STATEMENT	
2	▲ a	Edition statement	1st ed.
3	▲ b	Remainder of edition statement	
4	260	- PUBLICATION, DISTRIBUTION, ETC. (IMPRINT) +	
5	▲ a	Place of publication, distribution, etc	Beijing ;
6	▲ a	Place of publication, distribution, etc	Sebastopol, CA :
7	▲ b	Name of publisher, distributor, etc	O'Reilly,
8	▲ c	Date of	c2006.

Finalmente, añadimos el ítem a nuestra colección; éste se encontrará disponible de manera inmediata en el OPAC y podremos también realizar modificaciones posteriores, si es pertinente, además de poder visualizarlo de distintas maneras según nos convenga.

- **Registro completo.** Etiquetas MARC.
- **Registro normal.** Método simplificado.
- **ISBD.** Ficha catalográfica tradicional.

Open sources 2.0	
Editar artículo Duplicar Editar ejemplar borrar Completo Normal ISBD Añadir a estante Imprimir	
Open sources 2.0 : (registro 2)	
Subtítulo(s):	the continuing evolution / ;
Autor Adicional:	Stone, Mark.
Ilustrador:	ill. ;
publicado por:	O'Reilly, xl, 445 p. : , 24 cm.
ISBN:	0-596-008023 (p-b)
Temas:	- Open source software.
Año :	2006
Notas:	- Includes index.
URL:	http://www.loc.gov/catdir/enhancements/fy0715/2006275815-d.html
Tipo de artículo :	Colección de consulta

Agregar ejemplares

Una vez completado el registro bibliográfico, es necesario agregar información sobre ejemplares disponibles, es decir los “ítems”.

Aquí es posible añadir los siguientes datos:

- Tipo de material (posteriormente utilizado para definir extensión del préstamo).
- Sucursal de origen.
- Signatura topográfica.
- Inventario (utilizado para el sistema de préstamos).
- Número consecutivo.
- Condición física del ítem.
- Restricciones de uso.
- No préstamo (independiente al tipo de material y sus características, podemos especificar si un ítem en específico no sale a préstamo).
- Fuente de adquisición.
- Costo de adquisición.
- Precio por reposición.
- Situación del ítem .
- Código de país.
- Sub-localización (en caso de pertenecer a una colección).

Nuevo Ejemplar	
0 - Item status (withdrawn)	Nuevo
1 - Item status (lost)	
2 - Source of classification or shelving scheme	En línea (Sistema de Clasificación Decimal Dewey)
3 - Materials specified	Monografía
4 - Use restrictions	No
6 - Linkage	
8 - Sequence number	
9 - Cost, normal purchase price	\$240.00
a - Canceled barcode	
b - Location (homebranch)	
c - Shelving location	Colegio de Ciencias de la Información en México Escuela Nacional de Biblioteconomía y Archivonomía
d -	

Cada ejemplar de un título genera un registro distinto, con lo cual es posible especificar los datos pertenecientes a cada uno de los ejemplares disponibles de manera particular. Respecto al número de inventario, el sistema sugiere el número indicado en base al último registrado, siguiendo con esto un orden y no permitiendo dar de alta números ya existentes.

La personalización de la plantilla “ítems” se realiza desde la configuración del formato MARC, donde el campo 852 (Locación) permite incluir más o menos subcampos con relación a diferentes datos. Es posible, por ejemplo, manejar información referente a las medidas necesarias de preservación para cada tipo de material, costo total (si el ítem pertenece a una colección) y, de manera general, la cantidad de datos específicos que necesitemos.

3.4.4 Módulo de informes

El módulo de informes de Koha ofrece la posibilidad de crear informes preconfigurados para algunos procesos, así como un asistente para generar informes personalizados.

Informes preconfigurados

Los informes preconfigurados disponibles en Koha, son:

- Préstamos
 - Usuarios con más préstamos.
 - Materiales más prestados.
 - Informes diarios de préstamo vs. informes anteriores.
- Estadísticas de catálogo
 - Inventario – permite generar listas de estadística para el control de inventario.
 - Catálogo por tipo de material – crea un informe sobre la cantidad de registros en el sistema, jerarquizando de acuerdo al tipo de material.
- Inactividad
 - Materiales sin préstamos.
 - Socios sin préstamos.

- Estadísticas de circulación
 - Préstamos por categoría de usuario.
 - Tiempos promedio de préstamo.
 - Materiales en devolución.
 - Materiales retrasados en su devolución.

Asistente para crear informes

El asistente para crear informes personalizados viene preconfigurado para generar estadísticas sobre:

- Adquisiciones.
- Usuarios.
- Catálogo.
- Circulación.

Para cada una de las estadísticas, el asistente incluye una serie de opciones que nos permitirán crear una consulta específica, obteniendo como resultado el informe adecuado, mismo que puede ser visualizado en pantalla y/o exportado a un archivo de texto.

Informes		
Estadísticas Compuestas <ul style="list-style-type: none"> • Adquisiciones • Usuarios • Catálogo • Circulación 	Infomes principales <ul style="list-style-type: none"> • Los usuarios con más préstamos • Los ejemplares más prestados Forasteros <ul style="list-style-type: none"> • Usuarios sin préstamos • El ejemplar no está prestado 	Otro estado <p>sobre catálogo</p> <ul style="list-style-type: none"> • Inventario • Catálogo por tipo de artículo <p>sobre préstamos</p> <ul style="list-style-type: none"> • Till Reconciliation: Informe diario (ayer) • Till Reconciliation: Informe diario (hoy) • Ejemplares atrasados • Préstamos por categoría de usuario • Tiempo de préstamo medio

3.4.5 Módulo OPAC (catálogo al público de acceso en línea)

Koha incluye un OPAC muy completo y de amplia funcionalidad, el cual se encuentra disponible tanto para el usuario de la biblioteca como para el bibliotecario. Al igual que todo el sistema, el OPAC funciona con tecnologías Web, por lo que no requiere de módulos adicionales para publicar el catálogo; el acceso se realiza a través de un navegador Web.

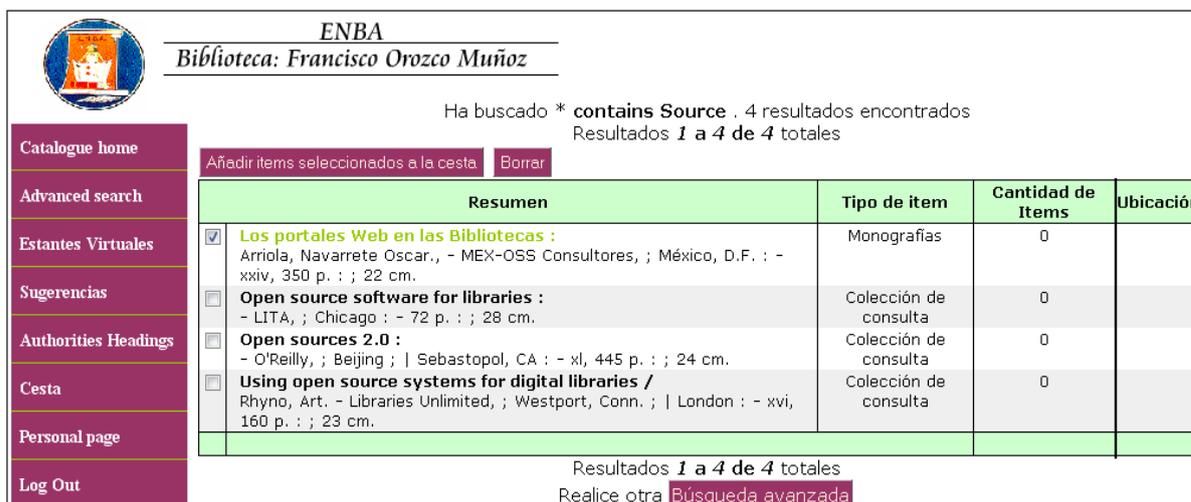
El OPAC del sistema es compatible con los estándares World Wide Web, por lo cual puede ser visualizado utilizando cualquier navegador, aunque es recomendable el uso de Mozilla Firefox por sus características, estabilidad, y personalización. Para ello existe además un “add on” que integra al OPAC en la barra de búsquedas, contando con la ventaja de tenerlo siempre disponible. Cabe señalar que al igual que todo el sistema, este módulo es completamente personalizable, ya que incluye la opción de añadir el logo de nuestra biblioteca, el nombre de la misma, editar el tema de colores, agregar opciones a la tabla de navegación, etc.



The screenshot displays the OPAC interface for ENBA (ENBA Biblioteca: Francisco Orozco Muñoz). On the left is a vertical navigation menu with the following items: Catalogue home, Advanced search, Estantes Virtuales, Sugerencias, Authorities Headings, Cesta, Personal page, and Log Out. The main content area features a search section titled "Buscar en el catálogo" with a "Búsqueda rápida" input field containing "Software libre bibliotecas" and an "ACEPTAR" button. Below this is a link for "Búsqueda Avanzada, Más Opciones". The "Últimas adquisiciones" section includes a dropdown for "Colección de consulta", a text input for "acquired in the last 30 days", and an "ACEPTAR" button. A link for "Search recent acquisitions by branch" is also present. The "Otras opciones" section contains buttons for "Estantes Virtuales", "Sugerencias", "My details", "My reading history", and "Log Out", along with a language dropdown set to "en" and a "Cambiar Idioma" button.

Búsqueda simple y Visualización

La búsqueda simple es la búsqueda predeterminada del OPAC. A través de la misma es posible llevar a cabo búsquedas mediante palabras claves, autores, títulos, temas, etc.; cuando se recupera más de un ítem el sistema muestra los registros ordenados mediante una tabla, con datos disponibles hasta el área de descripción física del material. Aquí es posible visualizar también en qué área se encuentra el material, de acuerdo al tipo de ítem, la cantidad de ejemplares disponibles, y la sede en la que podemos encontrarlo. También es posible agregar una reserva sobre el ítem o agregarlo a la canasta personal de libros. En caso de recuperar únicamente un registro, el sistema pasa directamente a la visualización configurada por defecto.



ENBA
Biblioteca: Francisco Orozco Muñoz

Ha buscado * contains Source . 4 resultados encontrados
Resultados 1 a 4 de 4 totales

Añadir ítems seleccionados a la cesta Borrar

Resumen	Tipo de ítem	Cantidad de ítems	Ubicación
<input checked="" type="checkbox"/> Los portales Web en las Bibliotecas : Arriola, Navarrete Oscar., - MEX-OSS Consultores, ; México, D.F. : - xxiv, 350 p. ; ; 22 cm.	Monografías	0	
<input type="checkbox"/> Open source software for libraries : - LITA, ; Chicago : - 72 p. ; ; 28 cm.	Colección de consulta	0	
<input type="checkbox"/> Open sources 2.0 : - O'Reilly, ; Beijing ; Sebastopol, CA : - xl, 445 p. ; ; 24 cm.	Colección de consulta	0	
<input type="checkbox"/> Using open source systems for digital libraries / Rhyno, Art. - Libraries Unlimited, ; Westport, Conn. ; London : - xvi, 160 p. ; ; 23 cm.	Colección de consulta	0	

Resultados 1 a 4 de 4 totales
Realice otra [Búsqueda avanzada](#)

Partiendo del listado de resultados, es posible visualizar el registro seleccionado de diferentes formas:

- **Visualización normal.** Cada tema es un enlace (link) y permite recuperar todos los registros que incluyan el mismo.
- **Visualización ISBD.** El sistema presenta el registro en el formato de ficha bibliográfica tradicional.
- **Visualización completa.** El sistema muestra todos los campos y subcampos en formato MARC; esta visualización se encuentra predeterminada por defecto en Koha, pero el usuario siempre tiene acceso a las demás.

Simple	Completo	ISBD	Añadir a estantería	Imprimir	Reserva	Añadir a cesta
0						
1	245 -TITLE STATEMENT					
2	a Title	Los portales Web en las Bibliotecas :				
3	b Remainder of title	Importancia, estructura y creación /				
4	c Statement of responsibility, etc	Oscar Arriola Navarrete.				
5	260 -PUBLICATION, DISTRIBUTION, ETC. (IMPRINT)					
6	a Place of publication, distribution, etc	México, D.F. :				
7	b Name of publisher, distributor, etc	MEX-OSS Consultores,				
8	c Date of publication, distribution, etc	c2009.				
I t e m s						

En la visualización del registro simple, Koha también muestra al usuario datos referidos a la localización del ítem (sede, aunque se puede configurar para que muestre la signatura topográfica), disponibilidad (prestado, disponible, retirado, perdido), fecha de devolución (en caso que se encuentre prestado), última visualización (última fecha que el ítem circuló), y número de inventario o código de barras. Estos datos son mostrados al usuario común de la biblioteca; cuando la visualización se realiza a través de la intranet –interfaz para bibliotecarios– es posible identificar al usuario que se le ha prestado el ítem.

Simple	Completo	ISBD	Añadir a estantería	Imprimir	Añadir a cesta
Los portales Web en las Bibliotecas :					
Subtítulo(s):	Importancia, estructura y creación / -				
AUTOR:	Arriola, Navarrete Oscar.,				
Illus:	ill. ;				
publicado por:	MEX-OSS Consultores,				
	xxiv, 350 p. : , 22 cm.				
ISBN:	978-15-73872				
Subjects:	<ul style="list-style-type: none"> - Libraries and the Internet., - Open Source., - Information technology., - Electronic reference services (Libraries)., - Libraries., - Blogs., 				
Año :	2009				
Tipo de ítem :	Monografías				
Item(s)					
localidad	situación	fecha límite	last Seen	código de barras	

Búsqueda avanzada

La búsqueda avanzada se lleva cabo de dos maneras distintas: usuario común y bibliotecario. Para el usuario, la búsqueda avanzada permite seleccionar entre diferentes campos para realizar búsquedas. El sistema permite restringir los resultados obtenidos mediante diversos filtros (tipo de ítem, fechas de publicación, editor, serie, etc.). El usuario puede elegir también la cantidad de registros mostrados por página y el orden, indicando el campo primario por el cual se mostrarán. En la intranet, el bibliotecario puede llevar a cabo búsquedas avanzadas especificando los campos MARC y utilizando operadores booleanos.

Búsqueda en el Catálogo

Buscar en

Cualquier palabra

Título

AUTOR

Tema

Series title

Tipo de ítem

Biblioteca

Otras opciones

Código de barras del ítem

Call Number

ISBN

Editor

Publicado entre y

Resultados por página

Exact search

Ordenado por

Diccionario

El acceso al diccionario se realiza mediante los diferentes campos, a partir de la búsqueda avanzada. Es necesario tener habilitadas las “pop ups” en el navegador ya que el diccionario aparece en una nueva ventana emergente. Una vez seleccionado el término de búsqueda, la ventana del diccionario se cierra y el término aparece en el formulario de búsqueda avanzada.

Búsqueda en diccionario

Buscar en

(add % at the end of your word if you want to use right-truncation :
theor% will find theory & theories)

Resultados de búsqueda

Values in biblios

Resultados 1 a 4 de 4

Resumen	Usado en	Seleccionar	
an open source for libraries collaboration.	1 biblio(s)	<input type="button" value="Select"/>	<input type="button" value="Sel&close"/>
Open source software for libraries :	1 biblio(s)	<input type="button" value="Select"/>	<input type="button" value="Sel&close"/>
Open sources 2.0 :	1 biblio(s)	<input type="button" value="Select"/>	<input type="button" value="Sel&close"/>
Using open source systems for digital libraries /	1 biblio(s)	<input type="button" value="Select"/>	<input type="button" value="Sel&close"/>

Resultados 1 a 4 de 4

Acceso a usuarios

Si bien muchas funciones se encuentran disponibles para todos los usuarios, aquellos que disponen de una cuenta (previamente configurada por el bibliotecario) tienen acceso a opciones extendidas del OPAC. Esta opción es configurable a través del módulo de administración de Koha, teniendo la opción de habilitarla o inhabilitarla, según convenga.



ENBA
Biblioteca: Francisco Orozco Muñoz

Catalogue home	Entrar a Koha
Advanced search	You want to log in to get personal services, or you must be logged in to get the page you requested. If you ignore your login and password, you must ask your library
Authorities Headings	Número de <input type="text"/>
Log in	Carnet <input type="text"/>
	Clave <input type="text"/>
	<input type="button" value="Ingresar"/>

Otra posibilidad para acceder es utilizar un sistema basado en LDAP. De esta manera, es posible integrar Koha con otros sistemas mediante un sólo ingreso; el usuario inicia sesión una sola vez y puede acceder a distintos servicios, como sistemas de referencia virtual, bases de datos, etc.

Canasta de libros.

La “Canasta de libros”, o también llamado “Cesto”, es un carrito virtual de ítems del OPAC. Aquí se pueden ubicar temporalmente y reunir los registros bibliográficos recuperados de distintas búsquedas. A partir de la canasta de libros, es posible reservar material, enviar el contenido de la canasta por correo electrónico y/o imprimir el listado con toda la información.

Catalogue home	Empty	<input type="checkbox"/>	Open source para bibliotecarios	
Advanced search	Remove selection		Autor: Fernando Flores Más autores: Ruiz Valentina Publicado por: Networks editions, , 300 p. Fecha Copyright: 2009	
Estantes Virtuales	Send		Ubicación	Código de barras
Sugerencias	Print	<input type="checkbox"/>	Software libre: una alternativa real	
Authorities Headings	Close		Autor: Arriola Navarrete Oscar Publicado por: ENBA ediciones , 275 p. Fecha Copyright: 2009	
Cesta (2)	Details		Ubicación	Código de barras
Personal page				
Log Out				

Estanterías virtuales

Otro concepto de Koha son las estanterías virtuales. Éstas, a diferencia de la canasta que está disponible únicamente durante una sesión, permiten coleccionar registros bibliográficos de manera permanente durante varias sesiones de búsqueda. El sistema permite ingresar una cantidad ilimitada de estanterías virtuales por usuario, y ésta puede ser marcada como privada –sólo el usuario que la creó puede usarla–, como pública –todos los usuarios pueden ver la estantería– o como libre –en este caso cualquier usuario puede visualizar la estantería y agregar registros a la misma. Los usos interesantes de las estanterías virtuales pueden ser aplicados para crear bibliografías de ciertas materias académicas, consultables por los estudiantes, recomendaciones de libros sobre ciertas áreas del conocimiento, listas privadas de materiales necesarios, etc., una utilidad mediante la cual los usuarios pueden organizar grupos de registros de la manera que lo requieran.

Catalogue home	Lista de Estantes	Agregar o Eliminar Estantes								
Advanced search	<table border="1"> <thead> <tr> <th>Nombre de Estante</th> <th>Categoría</th> <th>Tamaño de Contenido</th> <th>Modificar</th> </tr> </thead> <tbody> <tr> <td>Fernando estante</td> <td>Público</td> <td>0 vez(veces)</td> <td style="background-color: #800040; color: white;">Modificar</td> </tr> </tbody> </table>	Nombre de Estante	Categoría	Tamaño de Contenido	Modificar	Fernando estante	Público	0 vez(veces)	Modificar	
Nombre de Estante	Categoría	Tamaño de Contenido	Modificar							
Fernando estante	Público	0 vez(veces)	Modificar							
Estantes Virtuales	<h3 style="text-align: center;">Modificar estante</h3> <p>Nombre: <input type="text" value="Fernando estante"/></p> <p>Propietario: <input type="text" value="koha"/></p> <p>Categoría: <input type="text" value="Público"/></p> <ul style="list-style-type: none"> • un estante privado es administrado por usted y sólo puede ser visto por usted. • Un estante público puede ser visto por cualquiera, pero administrado solamente por usted. • Un estante libre puede ser administrado por cualquier usuario <p style="text-align: center;">Guardar</p>									
Sugerencias										
Authorities Headings										
Cesta (2)										
Personal page										
Log Out										

Cambio de idioma

El OPAC de Koha permite cambiar la interfaz a distintos idiomas. Esto depende de la disponibilidad de traducciones por defecto o creadas por el administrador del sistema. Hay traducciones disponibles en inglés, francés, italiano, y otros idiomas.

Configuración del OPAC

Dentro del módulo de administración de Koha (interfaz para bibliotecarios) es posible realizar una serie de personalizaciones al OPAC. De primera instancia, es posible modificar fácilmente los íconos de Koha, la estructura de los elementos del OPAC, así como los colores de texto, fondos y botones de acceso. De igual manera, el sistema incluye espacios destinados al uso particular de la biblioteca; de esta manera se pueden agregar textos adicionales y/o ligas a fuentes de información que la biblioteca considere importantes. Una opción de Koha para el módulo del OPAC, es la integración de contenidos del sitio AMAZON (<http://www.amazon.com>), permitiendo con esto la visualización de portadas de libros y de otros contenidos en el portal.

Administración de Preferencias del Sistema								
Administración	Adquisiciones	Autoridades	Intranet	Catálogo	Circulación	Usuarios	OPAC	Otros
AmazonAssocTag:	associates.amazon.com/gp/flex/associates/apply-login.html					Modificar	Borrar	
	<i>(Ver asociaciones con el portal AMAZON)</i>							
AmazonContent:	1					Modificar	Borrar	
	<i>(Habilitar conexión con el sitio AMAZON)</i>							
AmazonDevKey:						Modificar	Borrar	
	<i>(see: aws-portal.amazon.com/gp/aws/developer/registration/index.html)</i>							
BiblioDefaultView:	normal					Modificar	Borrar	
	<i>(Define la vista bibliográfica para cada ítem)</i>							
Disable_Dictionary:	0					Modificar	Borrar	
	<i>(Habilitar o deshabilitar diccionario)</i>							
LibraryName:	Biblioteca: Francisco Orozco Muñoz					Modificar	Borrar	
	<i>(Define el nombre de la biblioteca, como sera mostrado en el OPAC)</i>							
OpacNav:						Modificar	Borrar	
	<i>(Usar tablas HTML para modificar la barra de navegación de OPAC)</i>							
opaclargeimage:						Modificar	Borrar	
	<i>(Ingresar la imagen que desees usar para el OPAC, logo personalizado)</i>							

3.4.6 Módulo de adquisiciones

Al igual que los módulos descritos con anterioridad, el módulo de adquisiciones de Koha también es muy completo. Este nos permite generar distintas bases de datos entre las que encontramos: presupuestos disponibles en nuestra biblioteca, proveedores, tablas de moneda y su tasa de cambio, pedidos, facturas y una lista de sugerencias a revisar para posible adquisición.

Adquisiciones

Iniciar, recibir, modificar pedidos

Debe definir un presupuesto en parámetros.

tasas de cambio

PESO	1.00000
DOLAR	13.42000
EURO	17.80000

Sugerencias para revisar

No hay sugerencias esperando

presupuestos y fondos de libros

Debe definir un presupuesto en parámetros.

AYUDA sobre Ordenes

Para comenzar una adquisición, si una orden, una compra local o una donación primero busque en el suministrador, añada uno nuevo si lo necesita.
Entonces seleccione el anterior carrito de compra o cree uno nuevo. Dese cuenta que el carrito será asignado al usuario con el que se ha registrado
Para ordenar un artículo necesita establecer una biblio ya existente, y luego añadir un artículo, o establezca una nuevo biblio y luego añada el artículo.
Para cerrar un carrito de compra pulse sobre "ver carrito de compra" o busque encima, y luego pulse sobre "confirmar carrito".

Presupuestos

Esta opción permite crear y asignar diferentes tipos de presupuesto, con la posibilidad de enfocarlos de manera directa a un fondo específico de la biblioteca. En este módulo se puede llevar el control de todos los presupuestos que maneje la biblioteca en sus distintas sedes, los cuales pueden ser presupuestos asignados a la biblioteca para compra de libros, videos, revistas, software informático, etc.

Fondo Bibliográfico	Nombre	Sede	Editar	Borrar
002	Fondo Tecnológico 2010	Escuela Nacional de Biblioteconomía y Archivonomía		
	Presupuesto	Sede	Añadir Presupuesto	
	12/20/2009 - 03/03/2010 : 65000.00			
01	Fondo Hemerografico	Colegio de Ciencias de la Información en México		
	Presupuesto	Sede	Añadir Presupuesto	
	12/20/2009 - 03/03/2010 : 45000.00			

Añadir Fondo

Algunas de las ventajas de contar con este módulo, son:

- Permite manejar diversos presupuestos, con distintas fechas de vigencia, y determinar para qué sede (en caso de que la biblioteca cuente con más de una) estará disponible.
- Permite asignar claves para cada fondo de manera libre, utilizando caracteres alfanuméricos y generando un control consecutivo para cada uno.
- Presenta el presupuesto de manera detallada, contando con la posibilidad de modificarlo en cualquier momento, redistribuirlo, o añadir un nuevo monto para un mismo fondo.
- Las claves pueden ser actualizadas y establecer nuevos periodos de vigencia, de manera libre.

Proveedores

Permite ingresar los datos completos de proveedores y generar una lista que puede ser consultada, actualizada o corregida, según las necesidades de la biblioteca. Entre las opciones que destacan, se encuentra la posibilidad de ingresar datos sobre el tipo de material específico para cada proveedor, tipo de factura expedida, tipo de moneda en la cual basa sus precios de lista, editores y editoriales con las que tiene convenios, y notas extras como convenios para descuentos sobre precio final de la factura.

Información para :Sociedad Libre Publicaciones	
Editar Suministrador	Administrar Pedidos Recibir paquetes
DETALLES DE LA COMPAÑÍA	
NOMBRE DE Sociedad Libre Publicaciones	
LA	
COMPañÍA	
Dirección Postal	Apartado Postal 413 México D.F.
Dirección Física	Av. Insurgentes Nte. #8290 Colonia: Lindavista, delegación Gvo. A. Madero
Teléfono	53259000
Fax	56581111
Sitio web	http://soclib.com
DETALLES DE CONTACTO	
Nombre del Contacto	Lic. Ricardo Torvaldo
Posición	Depto. Ventas
Teléfono	56256700
teléfono alternativo	
Fax	
Correo	ric_torvaldo@soclib.com
Notas	
ESTADO ACTUAL	
El proveedor es	Activo
INFORMACION DE PEDIDOS	
Editores y Editoriales	Mc-Graw Hill Porrúa Libertades Limusa
Los precios de lista son	PESO
Precios de la Factura son	PESO
GST Registrado	Si
Precio unitario de lista incluye GST	Si
Precio unitario de la factura incluye GST	Si
Descuento	20.0000 %
Notas	

Pedidos

Para realizar el pedido de materiales, el sistema proporciona un formulario dividido en dos secciones:

- Detalles del material.
- Detalles contables.

El módulo interactúa con información existente en el sistema, por ejemplo, se puede tomar información del catálogo bibliográfico e importar los datos a la base de pedidos, realizando los cambios pertinentes. El módulo permite contar con información actualizada por cada movimiento efectuado, afectando por cada compra a cada uno de los distintos presupuestos asignados, para cada tipo de material. De igual manera, el módulo evita duplicidad en la adquisiciones al realizar una búsqueda mediante los distintos datos de un registro y proporcionando el estado actual de cada pedido (solicitado, pagado, facturado, etc.).

Nueva línea de pedido	
Carrito de Compra para Sociedad Libre Publicaciones	
Detalles de catálogo	
Título *	ish Edition) (Paperback)
Autor	Peter Wayner
Editor	Ediciones Granica, S.A.
Fecha de Copyright	2001
Formato	Colecciones especiales
ISBN	8475778593
Series	
Sede	Escuela Nacional de Biblioteconomía y Archivonomía
Detalles contables	
Cantidad	3
Fondo Bibliográfico	
Precio de Lista del Proveedor	\$44.57
Costo de Restitución	NaN
Costo presupuestado	NaN
GST presupuestado	0
TOTAL PRESUPUESTADO	NaN
Costo Real	\$44.57
<i>código de paquete</i>	0001-2009
Notas	Proveedor: in stock
Los siguientes 2 campos están disponibles para su uso interno. Pueden ser útiles para propósitos estadísticos	
Campo de control 1	
Campo de control 2	
ACEPTAR	

CONCLUSIONES

En el ámbito bibliotecario, los nuevos proyectos se construyen muchas veces sobre la marcha del día a día para adaptarse a los nuevos cambios organizacionales y tecnológicos que invaden nuestro entorno. Es así como algunas bibliotecas han evolucionado consciente o inconscientemente, mientras que otras, las que no han pensado en su adaptación, viven momentos críticos que ponen en jaque su supervivencia.

Hoy en día, la gestión integral y la automatización de las bibliotecas son un factor clave en el desarrollo e impacto de las mismas. En el área tecnológica los bibliotecarios deben conocer todas las alternativas existentes a nivel global, relacionadas con estos procesos, y lo que conlleva desarrollarlos, para así poder enfrentar cualquier situación que se presente en su unidad de información.

El software de gestión integral es la herramienta informática básica que marca la pauta para la organización y buen funcionamiento de muchas bibliotecas; es por esto que el bibliotecario debe conocer los conceptos básicos del software de código abierto, su funcionalidad, sus características y la aplicación y desarrollo que se le da para su utilización en las bibliotecas mediante los sistemas integrales de gestión bibliotecaria de código abierto, que sin duda son una alternativa real para la automatización integral de la biblioteca. Esto, sin duda, será un plus y no habrá impedimento alguno para gestionar de manera automatizada cualquier unidad de información.

El punto central de este trabajo era la instalación del sistema integrado para bibliotecas basado en el software libre llamado “Koha”, así como la descripción de su funcionalidad de acuerdo a los módulos con los que cuenta. Cabe destacar que lo anterior se logró con éxito, siendo un trabajo fundamentado que servirá como base para bibliotecarios novatos y/o expertos en el área de informática, que deseen implementar dicho software en sus bibliotecas. Es importante mencionar que este es el primer acercamiento con software libre

para bibliotecas, habiendo trabajado previamente con sistemas Linux en otras áreas; sin embargo, durante el transcurso del tiempo se pudo detectar que las facilidades que se dan tanto de instalación como de uso en este tipo de software, son muchas, razón por la cual no comprendemos cómo no se ha generalizado su uso en las unidades de información mexicanas. Cabe mencionar que como bibliotecarios debemos adquirir mayor capacitación en áreas referentes a las tecnologías de la información y la comunicación, que nos prepararán para enfrentar nuevos retos, nuevas metas, y nuevas perspectivas que en primera instancia pudieran parecer ser complicadas o externas a nuestro ámbito, de manera literal; pero que sin duda alguna serán un plus en nuestra formación académica y profesional, que nos dotará de conocimiento y entendimiento sobre los procesos tecnológicos llevados a cabo en la biblioteca, y su correcto tratamiento.

Koha es un sistema robusto que está ganando adeptos entre la comunidad bibliotecaria internacional que requiere de un sistema de gestión integral, totalmente libre y adaptable, pero sobre todo funcional y en constante desarrollo, factor que ha influido de manera determinante para que le sea comparado con sistemas propietarios excesivamente costosos, y 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 tradicionalista basado en mercado, oferta y demanda, eliminando la idea de que lo más caro siempre será lo mejor, forzando 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, contando con la posibilidad de migrar hacia cualesquiera de los mismos, con todas las ventajas que esto representa.

El software libre, sin duda alguna, cumple con las necesidades que la biblioteca demanda, enmarcándose dentro de un ámbito de cooperación entre colegas y personal de otras áreas, con el objetivo de gestionar de manera automatizada los procesos bibliotecarios, disminuyendo costos y a la par mejorando la

productividad y los servicios ofrecidos, llevando a cabo con esto la misión que la biblioteca es llamada a desarrollar.

Hay que romper muchos miedos y desconocimientos, ya que software libre no es sinónimo de software pirata; lamentablemente esta idea es muy generalizada en nuestro ámbito profesional. Por esto, es importante leer el segundo capítulo de este trabajo y revisar lo referente a las libertades que el software libre nos brinda para su uso, para darnos cuenta que la implementación de este tipo de sistemas es completamente legal y se encuentra respaldada en licencias legales que nos 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 ofrecen los sistemas libres y todo el panorama que se desarrolla alrededor de los mismos. Esto implica que se adquieran nuevos conocimientos y nuevas perspectivas, de manera que amplíen su visión más allá del enfoque tradicionalista de trabajo, centrado en el hecho de utilizar únicamente sistemas comerciales y propietarios. Koha, al igual que muchos otros sistemas libres, son una opción real para las bibliotecas mexicanas y del mundo; la decisión de avanzar está en cada uno de nosotros como bibliotecarios; las herramientas, como lo hemos visto, están ahí disponibles en el cyberspacio.

BIBLIOGRAFÍA

ARRIOLA NAVARRETE, Oscar. *Creación de un portal: el caso de la biblioteca Daniel Cosío Villegas de El Colegio de México*. Tesis. México: El autor, 2002.

ARRIOLA NAVARRETE, Oscar. *Proyecto de desarrollo bibliotecario: anexo técnico*. Guanajuato: Instituto Estatal de la Cultura, Coordinación Estatal de Bibliotecas Públicas, 2007.

ARRIOLA NAVARRETE, Oscar. "Referencia virtual: un nuevo reto para las bibliotecas". En: *Biblioteca universitaria: revista de la Dirección General de Bibliotecas de la UNAM*. Nueva época, v. 10, no.2, julio-diciembre de 2007, p. 138-152.

ARRIOLA NAVARRETE, Oscar y ÁVILA GONZÁLEZ, Armando. "El software libre y la enseñanza de la catalogación: una relación amistosa". En: *Revista CODICE*. Vol. 4, no. 2 (Julio-Diciembre 2008). p. 21-32.

ARRIOLA NAVARRETE, Oscar y BUTRÓN YÁÑEZ, Katya. "Sistemas integrales para la automatización de bibliotecas basados en software libre". [en línea]. En: *ACIMED*, 2008, vol. 18, no. 6. [Consultado: febrero 06, 2009]. Disponible en Internet: http://bvs.sld.cu/revistas/aci/vol18_6_08/aci091208.htm.

ARRIOLA NAVARRETE, Oscar y GARMENDIA BONILLA, Lovania. "Evaluación de software para bibliotecas: requerimientos técnicos" [en línea]. En: *Bibliotecas y archivos: órgano de la Escuela Nacional de Biblioteconomía y Archivonomía*. 1(4): p. 23-31. [Consulta: 04 de Agosto de 2009]. Disponible en Internet: http://eprints.rclis.org/archive/00012967/01/Evaluaci%C3%B3n_software.pdf.

BIRD, Drew. *Network+ Training guide*. USA: QUE Certification, 2002.

BOLIN, M. K. "Automating Idaho's libraries". En: *Resource Sharing & Information Networks*, v.9, no.1, 1993. p.79-94.

BOWMAN, J.H. *British Librarianship and Information Work 1991-2000*. USA: Ashgate, 2006.

CANO, V. "Information technology and the future of professional Library Practice". En: H. P. GEH y H., WALCKIERS, M. *Library Networking in Europe*. London: TFPL, 1996. p.51-65.

CARRILLO LÓPEZ, María Eugenia. *Automatización de bibliotecas de la Unidad de Torreón de la Universidad Autónoma de Coahuila*. Monterrey: El autor, 1992.

CHACON ALVARADO, Lucía. *Automatización de la biblioteca*. Costa Rica: EUNED, 2002.

CHAIN NAVARRO, Celia. *Gestión de Información en las Organizaciones*. España: ICE-Universidad de Murcia, 2000.

Class Lecture Notes: H. P. Luhn and Automatic Indexing; References to the Early Years of Automatic Indexing and Information Retrieval. [en línea]. [Consulta: 30 de Noviembre de 2009]. Disponible en Internet: <http://www.ischool.utexas.edu/~ssoy/organizing/luhn.htm>.

CLAYTON, Marlene. *Gestión automatizada de bibliotecas*. Madrid: Fundación Germán Sánchez Ruipérez, 1998.

CULEBRO JUÁREZ, Monserrat. [et al.] *Software libre VS software propietario*. [en línea]. [Consulta: 21 de Diciembre de 2009]. Disponible en Internet: www.softwarelibre.cl/drupal//files/32693.pdf.

DE ICAZA, Miguel. *El impacto al sistema e-México de la incorporación de software libre*. [en línea]. [Consulta: 19 de Diciembre de 2009]. Disponible en Internet: <http://www.emexico.gob.mx/>.

DE PABLOS HEREDERO, Carmen. *Informática y Comunicaciones en la Empresa*. España: Esic, 2004.

DEGANI, Asaf. *Taming HAL: Designing interfaces beyond 2001*. USA: Mcmillan, 2003.

DRAKE, Miriam. *Encyclopedia of Library and Information Science*. USA: Taylor & Francis Group, 2003.

FAISAL, S. y SURENDAN, B. *Report on Automation of Library at Kendriya Vidyalaya Pattom 2008*. [en línea]. [Consulta: 12 de Agosto de 2009]. Disponible en Internet: <http://librarykvpattom.files.wordpress.com/2008/07/library-automation.pdf>. p.14-15.

GONZÁLES, Jesús. "Cómo van los proyectos de software libre". En: GRUPO DE SISTEMAS Y COMUNICACIONES. *Compilación de ensayos sobre software libre*. [en línea]. [Consulta: 17 de Diciembre de 2009]. Disponible en Internet: <http://gsyc.escet.urjc.es/~grex/sobre-libre/libro-libre.pdf>.

GROSCH, Audrey. *Library Information Technology and Networks*. USA: Marcel Dekker, 1995.

HAYES, Robert y BECKER, Joseph. *Handbook of Data Processing for Libraries*. USA: Becker and Hayes, 1970.

HERRERA MORALES, José Román. *Software de Acceso a la Biblioteca Digital*. México: Facultad de Telemática: Universidad de Colima, 2004.

La historia de Labordoc. [en línea]. [Consulta: 10 de Diciembre de 2009]. Disponible en Internet: <http://www.oit.org/public/spanish/support/lib/labordoc/history.htm>.

History of the OSI. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.opensource.org/history>.

IDEAS PROPIAS PUBLICIDAD. *Introducción a la informática y al uso y manejo de aplicaciones comerciales*. España: GESBIBLO, 2006.

JACQUESSON, A. *L'informatisation des bibliothèques: historique, stratégie et perspectives*. Paris: Cercle de la Librairie, 1995.

KIMBER, Richard. *Automation in libraries*. USA: Pergamon, c1974. p.19-27.

LANCASTER, F. W. *Evaluación y medición de los servicios bibliotecarios*. México: UNAM, 1983.

LANKES, D. R. "Lesson learned from K-12 digital reference services". En: *Reference & User Services Quarterly*. v. 38, no. 1, 1998. p.63-71.

LASS, Andrew. *Library automation in transitional societies: lessons from Eastern Europe*. USA: Oxford University Press, 2000.

M. REITZ, Joan. "Library automation". En: *On-line dictionary of library and information science*. [en línea]. [Consulta: 02 de Agosto de 2009]. Disponible en Internet: <http://lu.com/odlis/search.cfm>.

MAGAL WALS, José. *Temas de biblioteconomía universitaria y general*. España: Editorial complutense, 2001.

MOYA ANEGÓN, F. *Los sistemas integrados de gestión bibliotecaria*. Madrid: ANABAD, 1995.

MURCIA FLORIAN, Jorge. *Redes del saber, investigación virtual, proceso educativo y autoformación*. Colombia: Cooperativa Editorial Magisterio, 2004.

The Open Source Definition. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.opensource.org/docs/osd>.

PANDEY, S.K. *Encyclopaedia of Library Automation Systems and Networks. India*: ANMOL Publications, 1999.

PARDO CLEMENTE, Ezequiel. *Microinformática de Gestión*. España: Servicio de Publicaciones-Universidad de Oviedo, 1993.

PARKER, AL. "La Biblioteca Virtual en Salud para América Latina y el Caribe". En: *Liaison*, v.10, no.3, 1999, p. 2-5.

Philosophy of the GNU Project. [en línea]. [Consulta: 15 de Noviembre de 2009]. Disponible en Internet: <http://www.gnu.org/philosophy/philosophy.html>.

PORCEL ITURRALDE, María Laura y RODRÍGUEZ MEDEROS, Mabel. "Software libre: una alternativa para las bibliotecas". [en línea]. En: ACIMED. 2005,13 (6). [Consulta: 25 de Diciembre de 2009]. Disponible en Internet: http://www.bvs.sld.cu/revistas/aci/vol13_6_05/aci090605.htm.

Referencia virtual. [en línea]. [Consulta: 06 de Agosto de 2009]. Disponible en Internet: <http://www.oclc.org/americalatina/es/questionpoint/about/virtual/default.htm>.

REYNOLDS, Dennis. *Automatización de bibliotecas*. Madrid: Fundación Germán Sánchez Ruipérez, 1998.

RODRÍGUEZ BRIZ, Fernanda. Los servicios de la referencia virtual : surgimiento, desarrollo y perspectivas a futuro. Buenos Aires: Alfagrama, 2005.

RODRÍGUEZ DURÁN, Lluís. *Sistemas Operativos*. España: Boixareu Editores, 2000.

SAORÍN PÉREZ, Tomas. *Modelo conceptual para la automatización de bibliotecas en el contexto digital*. [en línea]. España: El autor, 2002 [Consulta: 06 de Agosto de 2009]. Disponible en Internet: <http://www.tesisenred.net/TDR-0725106-121514/>.

SILVA ZAMORA, Oscar. *La automatización de Bibliotecas en México*. Tesis. México: El autor, 1989.

STOVER, Mark. *Theological Librarians and the Internet: Implications for practice*. USA: The Haworth Press Inc, 2001.

STURMAN, Robi. "Software open source per la gestione integrata delle biblioteche: una nuova risorsa?". En: *Bollettino AIB* N° 3 p.257-270.

TEDD, Lucy A. *An introduction to computer-based library systems*. London: John Wiley, 1985.

Usability 101: Introduction to Usability. [en línea]. [Consulta: 14 de Diciembre de 2009]. Disponible en Internet: <http://www.useit.com/alertbox/20030825.html>.

VON COTTA-SCHONBERG, Michael. "Automation and academic library structure". En: *Libri*, 1989, 39 (1) p.47-63.

VOUTSSÁS MÁRQUEZ, Juan. "Historia de la automatización de bibliotecas en México". En: LICEA DE ARENAS, Judith. *Cuarenta y cinco años de estudios universitarios en bibliotecología*. México: UNAM, Facultad de Filosofía y Letras, 2001.

WEDGEWORTH, Robert. *World Encyclopedia of Library and Information Services*. USA: American Library Association, 1993.

WILSON, Katie. *Computers in libraries: an introduction for library technicians*. USA: The Haworth information Press, 2006.

WORKING GROUP ON LIBRE SOFTWARE. *Free software history*. [en línea]. [Consulta: 15 de Diciembre de 2009]. Disponible en Internet: <http://eu.conecta.it/paper.pdf>.

YEE, Martha. "System design and cataloging meet the user: user's interfaces to online public access catalogs". En: *Journal of American society for information science* N° 42 (2) p.78.

ZURITA SÁNCHEZ, Juan Manuel. "Software libre: una alternativa para la gestión de recursos de información en bibliotecas". En: *Memorias de la Séptima Conferencia Iberoamericana en Sistemas, Cibernética e Informática*. Orlando, Fl.: International Institute of Informatics and Systemics, 2008. p.10-15.

ANEXO. INSTALACIÓN DE UBUNTU 9.04 “JAUNTY JACKALOPE”

En el siguiente anexo se explica paso a paso como llevar a cabo una instalación exitosa de Ubuntu, sobre Windows, contando con la posibilidad de crear un inicio dual, el cual nos permita seleccionar que sistema operativo deseamos utilizar al prender nuestro equipo.

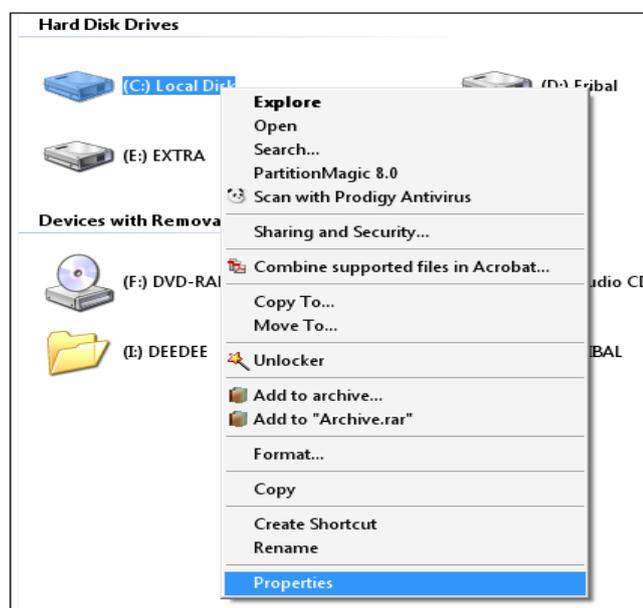
Ubuntu 9.04 puede ser instalado fácilmente en un portátil o en una PC de escritorio, independientemente si se cuenta ya con alguna versión de Windows, como Vista o XP.

Descargar Ubuntu 9.04

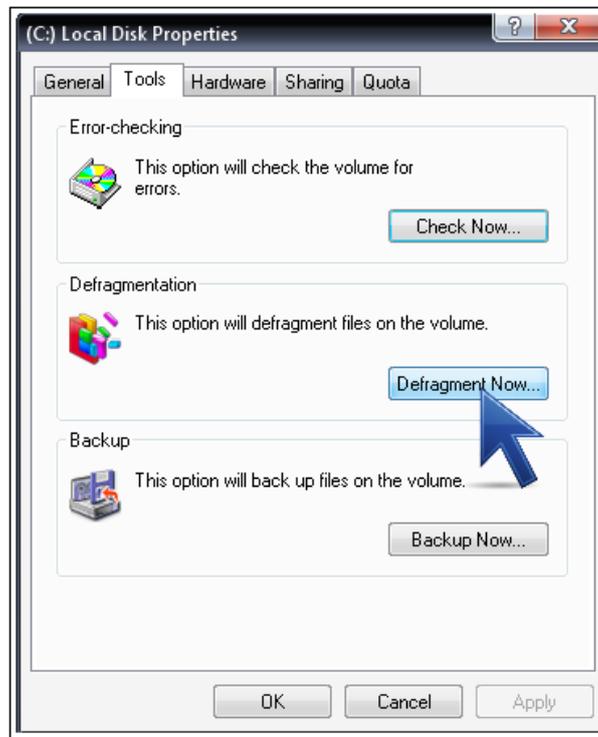
Para instalar Ubuntu es necesario contar con el CD de la distribución; podemos descargar una imagen .ISO y grabarla en un CD o DVD, desde esta dirección: <http://www.ubuntu.com/getubuntu/download> o solicitarlo directamente a Canonical, sin costo alguno, en: <https://shipit.ubuntu.com/> y recibirlo de 6 a 10 semanas, una vez realizado el pedido.

Preparando Windows

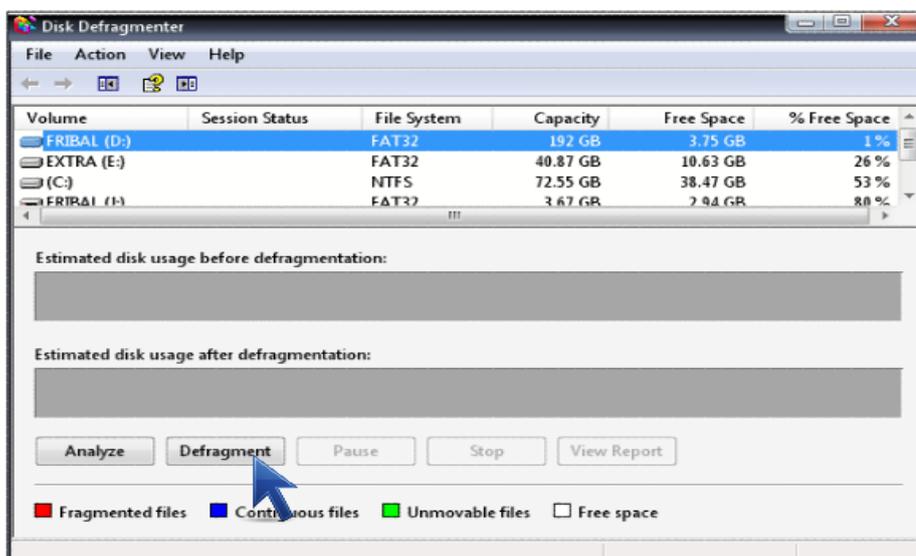
Antes de proceder a instalar Ubuntu, es necesario desfragmentar el disco duro de Windows con el objetivo de darle estabilidad y orden a los archivos que ahí se encuentran y evitarnos futuras complicaciones a la hora de redimensionar las particiones. Para llevar a cabo este proceso haremos clic sobre la unidad a desfragmentar y pulsamos **propiedades**.



En la siguiente ventana, hacemos clic en la pestaña **herramientas**, tomando interés por la segunda opción, **desfragmentar ahora**.



En la siguiente ventana seleccionamos la unidad que se desea desfragmentar, en nuestro caso será la unidad D únicamente, ya que es un proceso lento y conlleva tiempo; pulsamos **desfragmentar** y el sistema comenzará con el proceso sobre nuestro disco o partición.



Una vez concluido este proceso, insertamos el CD de Ubuntu y reiniciamos el equipo.

Instalando Ubuntu

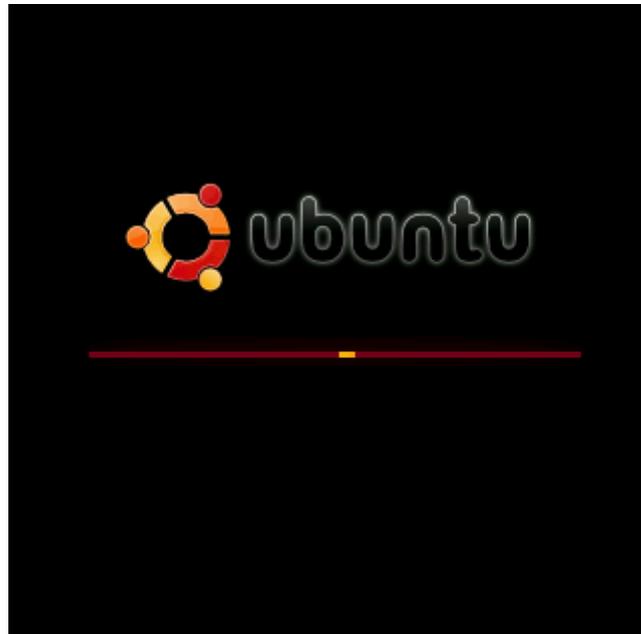
Al reiniciar el equipo con el CD de Ubuntu en el lector, lo primero que se muestra es un listado de idiomas, donde seleccionaremos **español**, y seguido pulsamos **Enter**.

Language		
Amharic	Hebrew	Polski
Arabic	Hindi	Português do Brasil
Беларуская	Hrvatski	Português
Български	Magyar	Română
Bengali	Bahasa Indonesia	Русский
Bosanski	Italiano	Sámegiellii
Català	日本語	Slovenčina
Čeština	ភាសាខ្មែរ	Slovenščina
Dansk	Khmer	Shqip
Deutsch	한국어	Svenska
Dzongkha	Kurdî	Tamil
Ελληνικά	Lietuviškai	Thai
English	Latviski	Tagalog
Esperanto	Македонски	Türkçe
Español	Malayalam	Українська

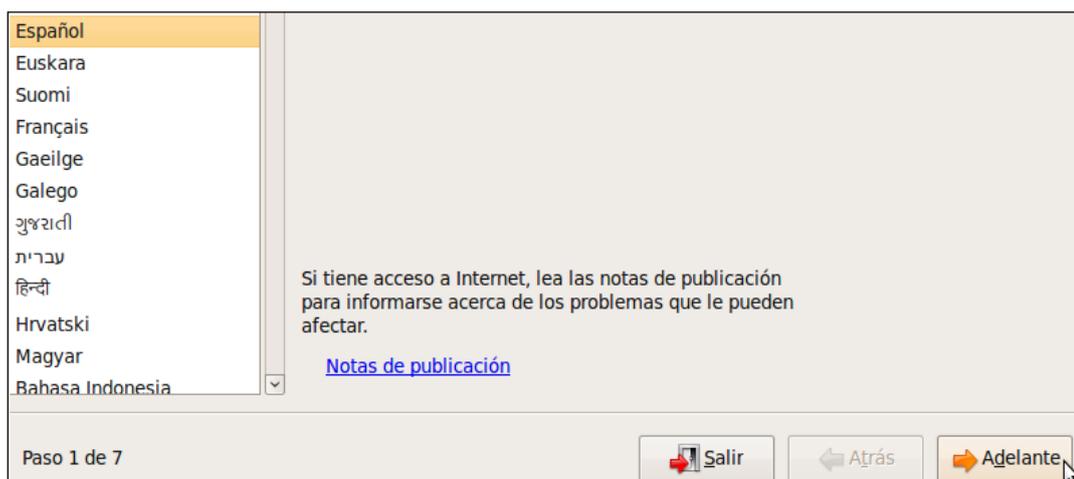
La siguiente pantalla muestra el menú de arranque del disco de Ubuntu; en el mismo contamos con diferentes opciones, como probar el sistema sin instalarlo (live), verificar el CD o comenzar con la instalación. Nos situamos sobre **instalar Ubuntu** y pulsamos **Enter**.



Se cargará el entorno gráfico, mediante el cual se realiza la instalación, siguiendo un sencillo asistente de 7 pasos que detallamos a continuación:



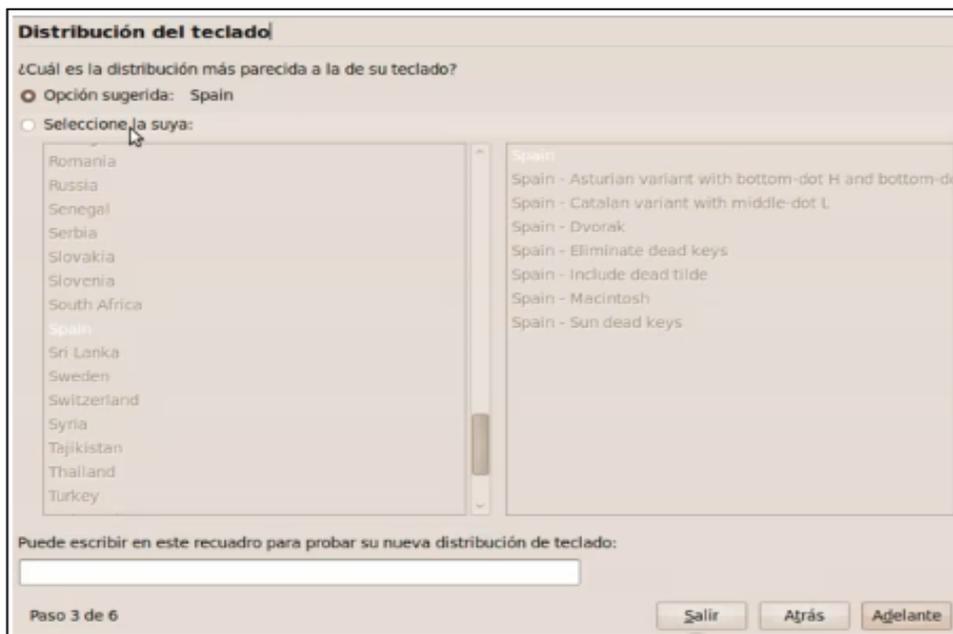
El primer paso consiste en seleccionar el idioma que deseamos utilizar en el asistente de instalación y en el que se instalará Ubuntu por defecto. Como ya hemos seleccionado previamente la opción **español** directamente del menú del CD, ésta aparecerá como predeterminada, por lo cual hacemos clic en el botón **adelante**.



En la siguiente ventana indicaremos el área geográfica donde nos encontramos, lo cual le permitirá al sistema establecer la zona horaria. Para nuestro caso, seleccionamos México y hacemos clic en **adelante**.



Lo siguiente será elegir la distribución de nuestro teclado; el sistema generalmente sugiere la opción correcta, por lo que únicamente hacemos clic en **adelante**.



El siguiente paso es muy importante y se refiere a las particiones del disco, para la cual seguiremos dos opciones:

- Particionado guiado (sencillo pero no configurable).
- Particionado manual (complejo pero configurable en su totalidad).

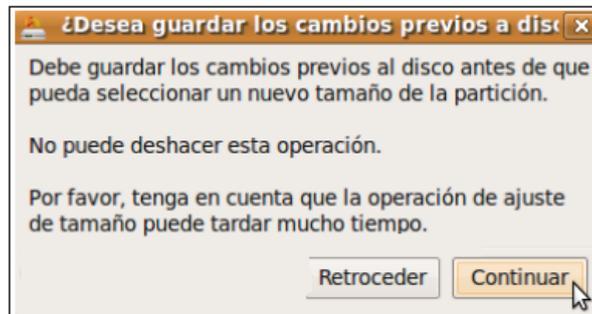
Particionado guiado. Si elegimos el particionado guiado, sólo tendremos que indicar el tamaño de la partición que queremos asignarle a Ubuntu y el instalador creará automáticamente dos particiones:

- Partición raíz (/): contiene los directorios y archivos del sistema, así como los datos de los usuarios (documentos, música, fotografías, etc.).
- Partición de intercambio (swap): ésta será requerida por el sistema, a falta de memoria RAM.

Para comenzar con Ubuntu se recomienda utilizar de 8 a 10 GB. Indicaremos el tamaño deseado moviendo con el puntero de izquierda a derecha la barra de selección, y posteriormente hacemos clic en **adelante**.



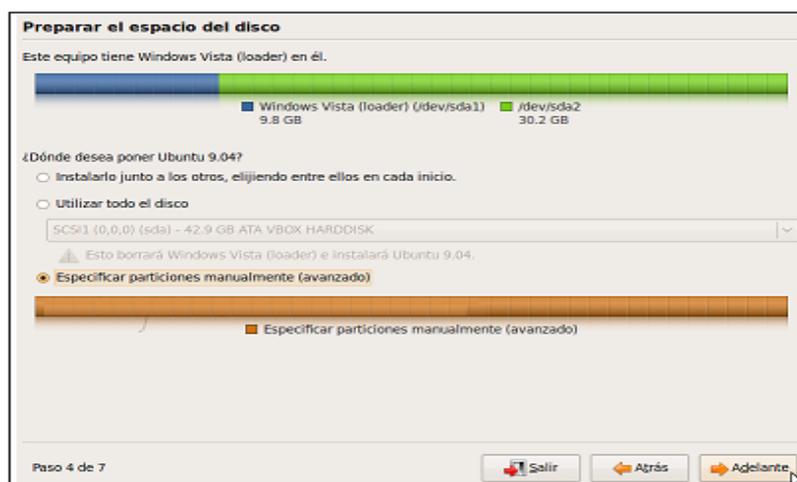
El sistema avisará que necesita guardar los cambios y que dicha operación no se podrá revertir una vez llevada a cabo. Conformes con el tamaño que hemos asignado, hacemos clic en **continuar** y proseguimos.



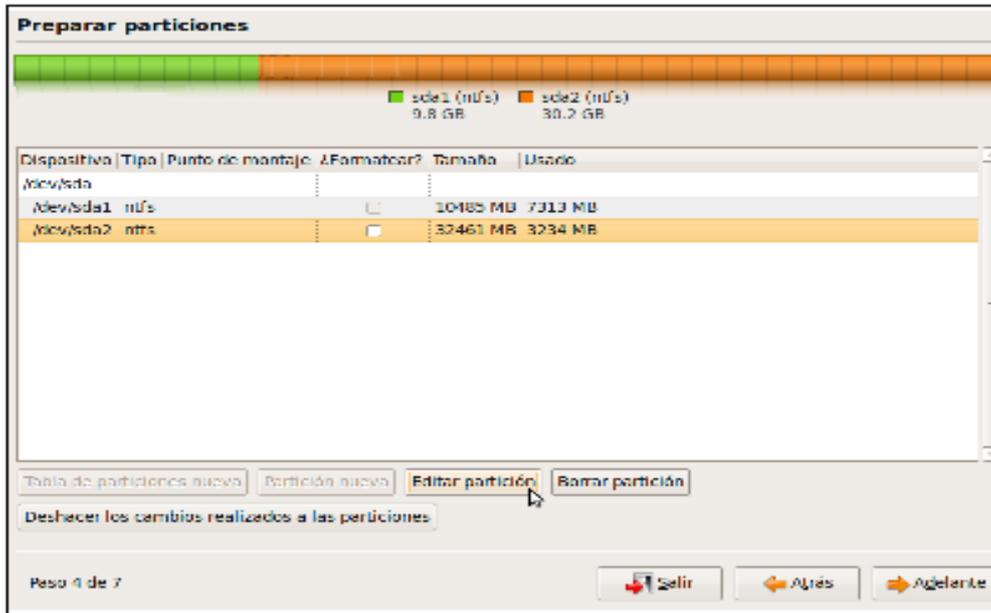
Particionado manual. Si se opta por el particionado manual, es posible seleccionar el número de particiones, tamaños y tipos de sistema de archivos. Conviene tenerlo de la siguiente manera:

- Partición raíz (/): contiene los directorios y archivos del sistema, así como los datos de los usuarios (documentos, música, fotografías, etc.).
- Partición (/home): contiene los archivos y datos de cada usuario, de manera que si se piensa actualizar o cambiar de distribución no será necesario formatearlos, ya que esta partición seguirá ahí, con nuestros datos seguros.
- Partición de intercambio (swap): ésta será requerida por el sistema, a falta de memoria RAM.

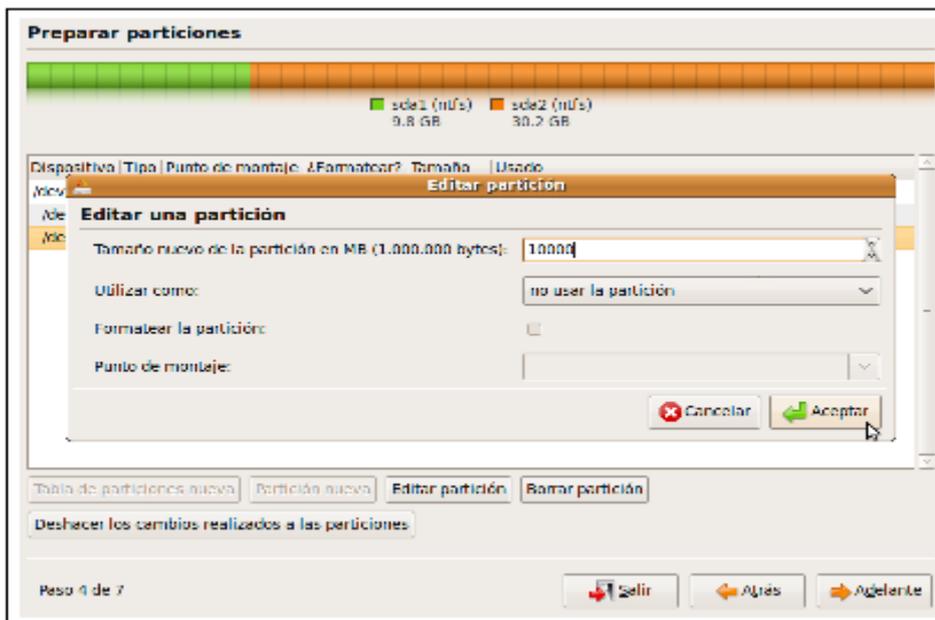
Para comenzar con este tipo de particionado, elegimos **especificar particiones manualmente (avanzado)** y hacemos clic en **adelante**.



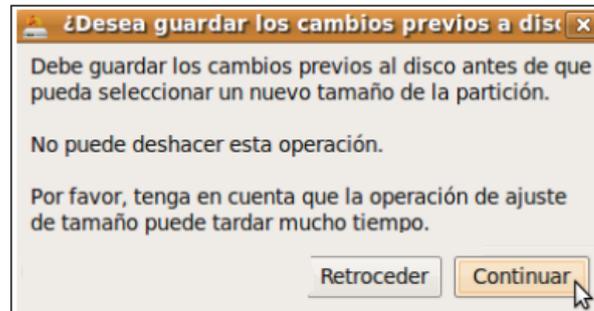
El primer paso es disminuir el tamaño de una de las dos particiones con las que contamos, de esta manera tendremos espacio libre para crear una nueva partición. Seleccionamos la partición deseada y hacemos clic en **editar partición**.



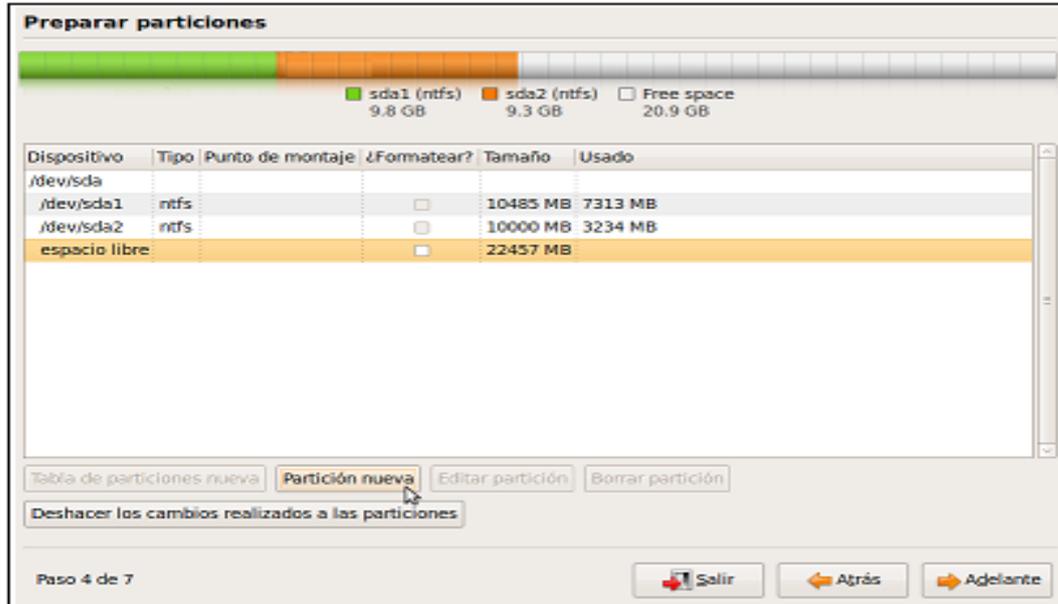
Escribimos el nuevo tamaño y hacemos clic en **aceptar**. Es necesario tener bien definido el espacio que asignaremos para instalar Ubuntu, dependiendo el tamaño del disco.



Antes de seguir es, necesario confirmar el cambio del tamaño en la partición ya que será una operación que no se podrá deshacer de manera posterior, por lo cual hacemos clic en **aceptar**.



El instalador nos devuelve a una pantalla anterior, pero ahora contamos con un nuevo espacio libre que será el utilizado para crear las particiones correspondientes a Ubuntu. Nos situamos en el espacio libre y hacemos clic en **partición nueva**.

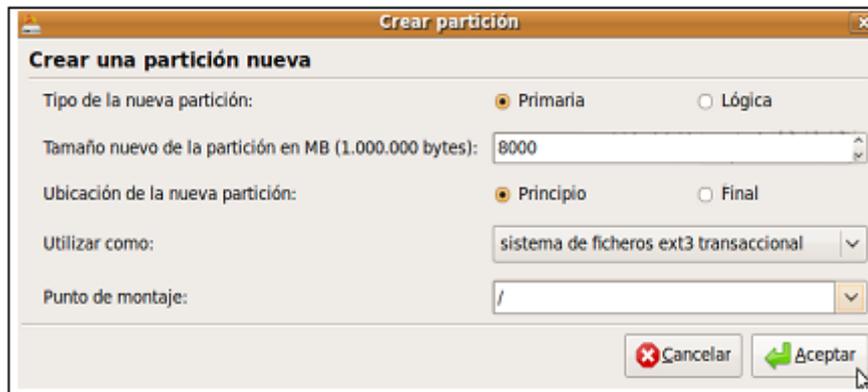


En la siguiente ventana debemos indicar lo siguiente:

- Tipo de partición: la partición raíz (/) debe ir en una partición **primaria**.
- Tamaño en MB: como mínimo debería tener 3000 MB, para este caso lo ajustaremos a 8000 MB que es lo ideal.

- Sistema de archivos: **ext3** es el sistema que utiliza Ubuntu por defecto, también podríamos elegir **ext4** pero no es 100% fiable con esta versión de Ubuntu.
- Punto de montaje: / (la raíz).

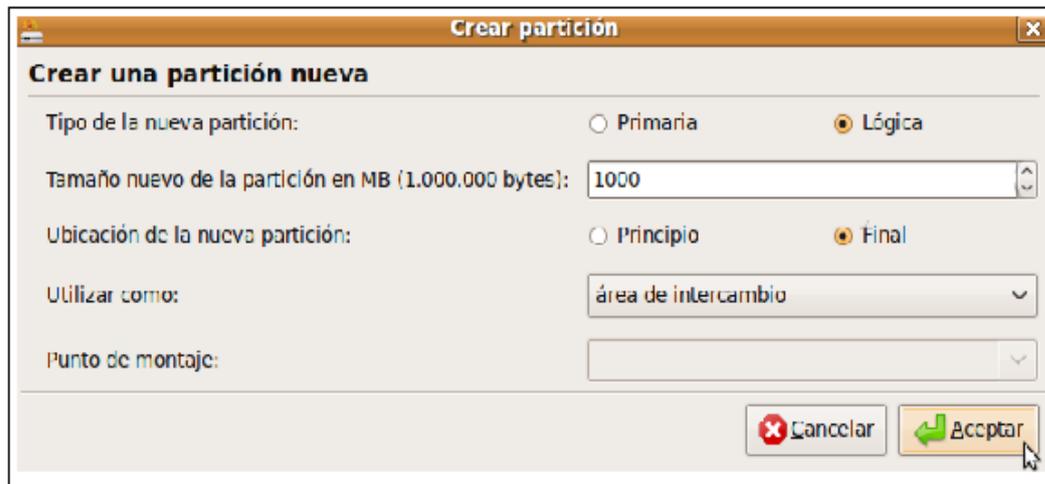
Hacemos clic en **aceptar**.



Con esto hemos creado nuestra primera partición para Ubuntu, llamada partición raíz. Lo siguiente que vamos a definir es la partición (swap), que mantendrá todas las aplicaciones en memoria cuando el sistema no disponga de suficiente memoria RAM. Asignaremos un GB para esta partición y tomaremos en cuenta los siguientes datos:

- Tipo de partición: ésta puede ir indistintamente en una partición primaria o lógica, aunque debemos tomar en cuenta que solamente puede haber 4 particiones primarias en un disco duro, siendo, por el contrario, ilimitadas las particiones lógicas que podemos asignar. Por lo tanto, la seleccionaremos como **lógica**.
- Tamaño en MB: 1000 MB.
- Ubicación: final.
- Sistema de archivos: área de intercambio.

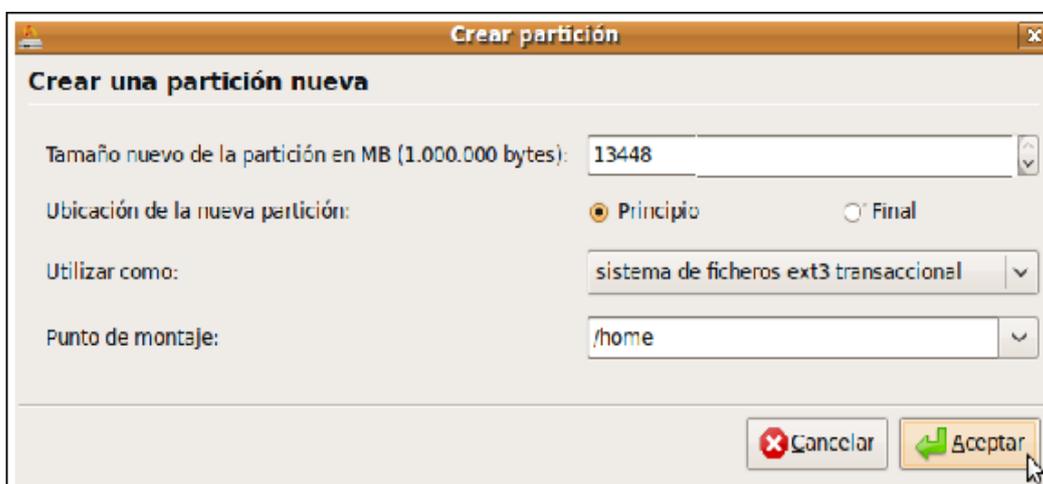
Hacemos clic en **aceptar**.



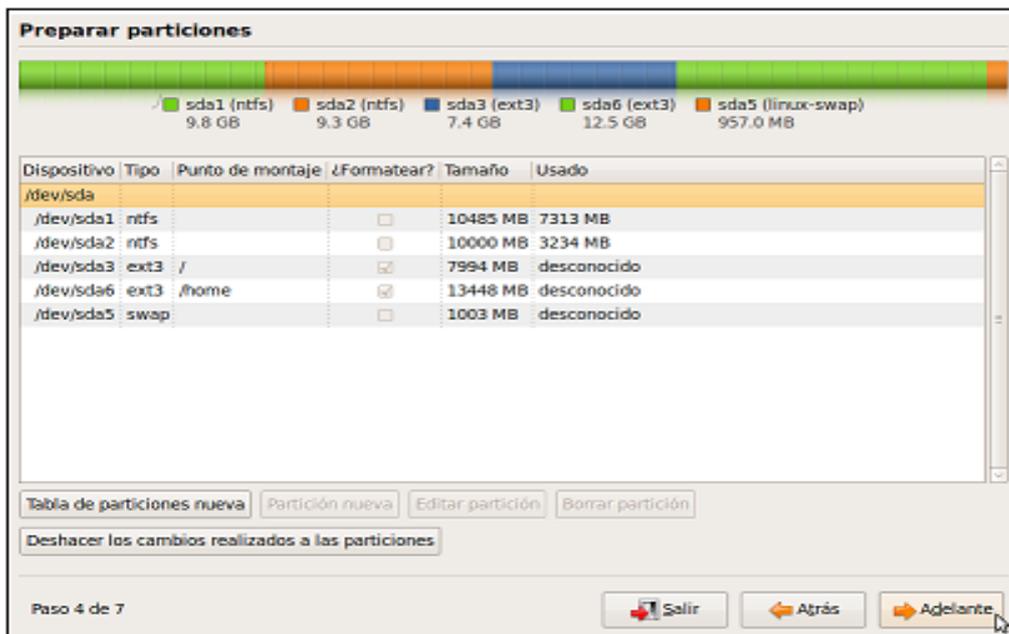
Por último, definimos la partición para los datos de los usuarios, /home. En el espacio libre que queda hacemos clic nuevamente, sobre una nueva partición.

El tamaño de esta partición será el restante de las que hemos creado previamente, tomando en cuenta los siguientes datos:

- Tamaño en MB: todo el que queda disponible.
- Ubicación: principio.
- Sistema de archivos: ext3 por defecto.
- Punto de montaje: /home.



Tras aceptar, el asistente nos mostrará las particiones con las que cuenta nuestro disco duro y podremos continuar pulsando **adelante**.



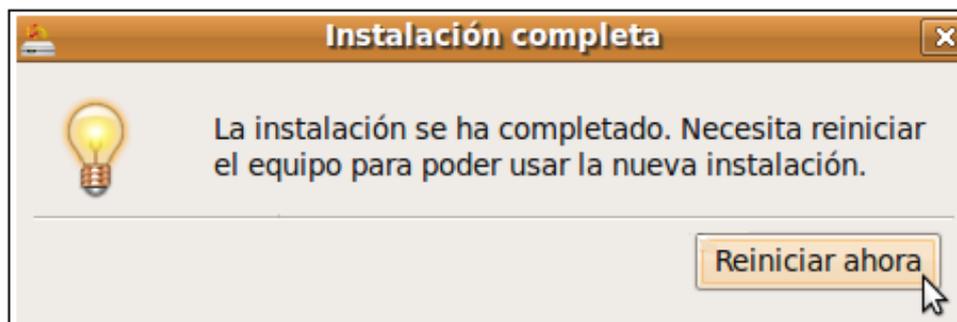
Hemos llevado a cabo la parte más complicada del proceso de instalación de Ubuntu (crear las particiones). En la siguiente pantalla tendremos que introducir nuestro nombre, nombre de usuario, contraseña y nombre del equipo.



Una vez hecho esto, el sistema nos pregunta si deseamos migrar archivos de usuarios Windows. De esta manera podemos importar los favoritos de Internet, el papel tapiz del escritorio, documentos personales, etc.



Finalmente, antes de que el asistente lleve a cabo la instalación de Ubuntu 9.04 en nuestro equipo se mostrará un resumen con los datos que hemos ido completando; si estamos de acuerdo y todo es correcto, hacemos clic sobre el botón **instalar**. Una vez finalizada la instalación es necesario reiniciar el equipo, por lo cual, hacemos clic en **reiniciar ahora**.



Iniciar el equipo

Después de haber instalado Ubuntu, al arrancar el equipo nos aparecerá el siguiente menú, llamado GRUB, en el que podremos elegir iniciar con Ubuntu, o con Windows. En el menú tenemos 3 opciones para Ubuntu.

- **Ubuntu 9.04, kernel 2.6.28-11-generic:** inicia Ubuntu de manera regular, esta opción será la predeterminada el 99.9% de las veces que inicie nuestra computadora.
- **Ubuntu 9.04, kernel 2.6.28-11-generic (recovery mode):** nos permite entrar en modo de recuperación, fundamental para cambiar la contraseña si la hemos olvidado.
- **Ubuntu 9.04, memtest86+:** realiza un test de memoria del sistema.



```
Ubuntu 9.04, kernel 2.6.28-11-generic
Ubuntu 9.04, kernel 2.6.28-11-generic (recovery mode)
Ubuntu 9.04, memtest86+
Other operating systems:
Microsoft Windows XP Professional
```

Use the ↑ and ↓ keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.

Con esto, hemos concluido la instalación de Ubuntu 9.04 con la opción de boteo dual, sin perder la partición del sistema operativo Windows. Es conveniente revisar la literatura relacionada con acciones que se deben llevar a cabo posteriormente a la instalación de nuestra distribución, con el fin de obtener el mejor rendimiento posible.