

Software bibliotecario

abierto y gratuito

Córdoba (Argentina)

© 2001 Verónica Lencinas

Este material puede ser distribuido según los términos y condiciones de la Licencia de Publicación Abierta (Open Publication Licence) en su versión 1.0 ó posterior. La última versión esta disponible en <http://www.opencontent.org/openpub/>

Puede obtener este documento en <http://www.procesos-tecnicos.com.ar/docs/OpenSource.htm>

El presente trabajo fue escrito en agosto / septiembre de 2001 y presentado el 15 de Septiembre en las III JORNADAS DE BIBLIOTECAS DE LA PROVINCIA DE CORDOBA Y I JORNADAS DE PROFESIONALES DE LA INFORMACION - Córdoba, 12 al 15 de Setiembre de 2001
FERIA DEL LIBRO DE CORDOBA 2001

Contenido

- ✍ Resumen
- ✍ Introducción

1. Hardware / Software

2. Software abierto

- ✍ Proyecto de Ley para que el Estado use software abierto
- ✍ Software abierto: ejemplos
- ✍ Sistemas operativos abiertos: GNU/Linux
- ✍ Sistemas bibliotecarios abiertos: Avanti
- ✍ Sistemas bibliotecarios abiertos: OSDLS: Open Source Digital Library System (Phyteas)
- ✍ Sistemas bibliotecarios abiertos: OpenBook
- ✍ Herramientas de desarrollo abiertas: OpenIsis

3. Conclusiones

- ✍ Enlaces (links) útiles
 - ✍ Bibliografía
-

Resumen:

El software abierto, conocido como Open Source presenta una serie de ventajas para su implementación en bibliotecas. Se ofrece en forma gratuita y con su código fuente completo que permite corregir errores del software, modificarlo e integrarlo con otros programas. Por estas condiciones el software abierto ofrece mejores posibilidades a las bibliotecas que el software cerrado. Sistemas de gestión bibliotecaria estarán disponibles en poco tiempo y pueden constituir una verdadera alternativa al software comercial. La metodología usada para desarrollar el software abierto, con grupos de desarrolladores descentralizados, puede aplicarse también a desarrollos bibliotecológicos.

Introducción

El presente trabajo fue escrito para las

1. Software / Hardware

Cuando comenzamos a capacitarnos en computación, lo primero que aprendemos es, generalmente, la diferencia entre hardware y software. Aprendemos, que a la colección de placas electronicas, cables, ventiladorcitos, y todos los aparatos que estan enchufados a la compu, se le llama hardware.

Software es mas dificil de definir, se le llama software a los programas, instrucciones que hacen funcionar a los componentes de la computadora. Sin los el software, la computadora realmente no es mas que un rejunte de chatarra electronica.

Desarrollo del software

Para crear un programa, el programador se vale de un lenguaje de programación, que hace de intermediario entre el lenguaje humano y el lenguaje binario de la maquina. Estas instrucciones, escritas en un lenguaje de programación se denomina codigo fuente del programa.

Una vez finalizado el listado de instrucciones, el programa se compila, esto significa que el codigo fuente es transformado en codigo binario. Ya no es legible por una persona y el programador, o la empresa para la cual trabaja, puede vender y distribuir el programa compilado sin temor que alguien pueda modificarlo o reusarlo en otro programa. El código fuente lo guardan en la caja fuerte. Debido a que este tipo de software no se puede modificar, se lo denomina también software cerrado. Esto es lo que generalmente entendemos por programas y lo que tenemos en nuestras computadoras: compilados cerrados que funcionan y que solo sirven para eso: funcionar. Pero este enfoque trae consigo una serie de problemas.

Problemas del software cerrado

Cuando se compra una computadora, generalmente ya trae instalado una serie de programas. El precio de estos programas esta incluido en el precio de venta de la computadora. Pero el comprador no es el dueño de estos programas. Se le entrega una licencia de uso que permite hacer determinadas cosas: usar el software y hacer una copia de seguridad, y prohíbe otras: por ejemplo copiar y distribuirlo.

En caso que el programa tenga una falla, es imposible corregirla debido a que no cuenta con el código fuente. Además de ello se debe confiar en la honestidad del autor del programa, ya que pueden incluir instrucciones ocultas, como por ejemplo abrir un puerto en nuestra computadora por la cual puede entrar un pirata informático y copiar o borrar información, pueden contener virus, o mas comunmente, pueden estar mal escritos y borrar o sobrescribir archivos de configuración. En este caso nada se puede hacer al respecto, solamente avisar a la empresa dueña del software y esperar que la proxima versión venga

corregida.

2. Software bibliotecario

Ambiente de implementación

El ámbito de implementación en las bibliotecas ofrece dificultades adicionales. Mientras que el software para oficina, tipo "Office" ha alcanzado un alto nivel, no se puede decir lo mismo del software para bibliotecas.

La materia prima de la biblioteca es la información. Un sistema bibliotecario debe almacenar gran cantidad de datos; por un lado datos bibliográficos: título, autores, edición, contenido, etc. y por otro lado los datos relacionados con el soporte y su movimiento en la biblioteca: ubicación, condiciones de prestamos, estadísticas de utilización, etc. Estos datos deben poder recuperarse, tanto para el personal de la biblioteca como para los usuarios. Deben existir diferentes puntos de acceso, búsquedas y formas de visualización.

Los soportes digitales, en disquetes o cd-roms, y los documentos virtuales, como páginas web, cobran cada vez mayor importancia y deben integrarse adecuadamente a los sistemas de acceso de la información de la biblioteca. Estos datos deben cumplir también con las diversas normas bibliotecarias: ISBD, AACR2, CDU, Dewey, Tesauros, Marc, etc. Este ámbito de gran heterogeneidad, ofrece un gran desafío a las bases de datos, desarrolladas para un ambiente de datos más controlados y con otras necesidades de recuperación. Un sistema de gestión bibliotecario que pueda manejar y manipular todos estos datos es también complicado, difícil de programar y propenso a errores. Requiere de potentes bases de datos que a su vez exigen equipos de alto rendimiento, y de Hardware costoso.

Es de esperar que el desarrollo tecnológico siga a un ritmo acelerado. Esto significa para las bibliotecas que seguirán apareciendo tecnologías de la información nuevas que modificarán las formas y el acceso a la información y que requerirán ajustes y adaptaciones del software bibliotecario. Nuestras bibliotecas se desempeñan además en un ámbito de bajos recursos. Dependientes de instituciones, en muchos casos públicas, con presupuestos exiguos y una burocracia ajena a los problemas de la biblioteca.

Software bibliotecario Requerimientos

Un software para bibliotecas debe desempeñarse en un ambiente de altas exigencias y bajo condiciones de inversión restringida. Debe cumplir además con una serie de requerimientos.

- ✍ Debe ser seguro y confiable. Los datos, especialmente todos los datos relacionados con los usuarios y las transacciones de prestamos deben ser a) confidenciales y b) seguros. No deben existir la posibilidad, en el funcionamiento normal, de perder datos.
- ✍ Deben gestionar datos tanto locales como remotos. A la información del catálogo se le suma información proveniente de páginas web y de bases de datos remotas.
- ✍ Toda la información debe cumplir con normas establecidas, esto incluye normas bibliotecológicas y normas del ámbito de la informática. Es preferible adherir a normas internacionales abiertas que a normativas propietarias.
- ✍ El software debe ser altamente adaptable para amoldarse a las necesidades y la organización de cada biblioteca, respetando usos y métodos locales.
- ✍ Debe tener la capacidad de crecer y evolucionar con el avance tecnológico y bibliotecológico. Esto se logra con mayor facilidad, adhieriendo a estándares abiertos.
- ✍ Debe ofrecer servicios tanto locales (catálogo) como remotos (webcat, envío de fotocopias,

reserva remota, etc.)

- ✍ Debe ser económico en su adquisición, no solamente respecto al costo de la licencia de uso, sino también en cuanto a sus requerimientos de hardware y capacitación del personal.
- ✍ Económico en el mantenimiento. Esto es difícil de calcular, debido a que todos los vendedores de software declaran que el que ellos venden tiene menor gastos de mantenimiento que los demás. Una buena fuente para evaluar este costo, es preguntar directamente a bibliotecas que lo usan, los gastos de mantenimiento que tienen.
- ✍ Debe ser económico en la aplicación. Esto incluye el cálculo de licencias para computadoras adicionales, los requerimientos de hardware necesarios y el costo de programas adicionales que pueden ser necesarios.

✍

Algo de todo esto ya lo tenemos y lo usamos. Isis en cualquiera de sus variantes: Microisis, WinIsis, Sigebi, Educa, Biblio, etc. combina una estructura que se adapta a datos textuales heterogéneos con una gran potencia en la recuperación de datos y muy bajos requerimientos de hardware. Si Isis no nos satisface del todo, es por otras razones, por ejemplo por el costo de programas adicionales (circulación) y la falta de integración con otros sistemas.

Mercado del software bibliotecario

El mercado del software bibliotecario tiene características propias que deben tenerse en cuenta. En primer lugar, es un mercado pequeño, aún en países desarrollados en los que las bibliotecas disponen de mayores fondos. Muchos sistemas son incompatibles entre sí. Esto significa que los módulos deben adquirirse en bloque y no pueden mezclarse con módulos provenientes de otros sistemas. Esto significa en la práctica que una biblioteca no puede elegir el software que más le convenga, sino el que vende la empresa al que ya se le compraron módulos.

Los softwares comerciales para bibliotecas muestran en general poca adaptabilidad a necesidades locales. En el mejor de los casos se cuenta con pantallas en castellano, en vez de inglés, pero exigen una organización del trabajo determinada, poco adaptable y configurable a las necesidades puntuales de cada biblioteca.

Los costos suelen ser elevados. Esto puede justificarse con la inversión necesaria en programación para adaptar a bases de datos estándares a las necesidades de la biblioteca, lo restringido del mercado existente y la necesidad de obtener ganancias. Por esta última razón las licencias también suelen ser muy restrictivas. Una condición normal en este mercado es, por ejemplo, exigir licencias adicionales por cada computadora conectada.

Lo mismo rige para los gastos de migración de sistemas. A pesar de que los sistemas generalmente permiten la exportación de registros bibliográficos en formato Marc21, éstos no incluyen los datos locales, como por ejemplo número de inventario, esenciales para el funcionamiento de los servicios bibliotecarios. Las empresas propietarias del software suelen cobrar fuertes sumas para exportar datos completos. Generalmente estos datos requieren también un reformato, igualmente costoso. Es por ello que muchas bibliotecas en EEUU y Europa se ven atrapadas con softwares deficientes y desactualizados, que no pueden cambiar por la falta de fondos y tiempo necesario para la migración del sistema. Las empresas vendedoras del software han demostrado muy poco incentivo para actualizar sus softwares e integrar tecnologías nuevas.

Si nos fijamos en software no desarrollados específicamente para bibliotecas o centros de documentación, encontramos todo un grupo de software que funciona en máquinas viejas, es adaptable,

se puede combinar fácilmente con otros programas y es gratuito: es el software de fuentes abiertas o software abierto.

3. Software abierto

La característica principal del software abierto es el hecho que se distribuye con su código fuente. Esto permite no solamente adaptar el programa a las necesidades que tenemos. Un programador puede también evaluar el código fuente y con ello la calidad del programa. Esto es algo muy similar a lo que sucede con las publicaciones científicas, en las que cada artículo es revisado por científicos de esa especialidad evaluando la importancia del tema, la rigurosidad científica, la metodología empleada, etc. El software abierto no permite, o por lo menos lo hace muy difícil de esconder funciones como virus, puertas traseras y otros mecanismos no desados por el usuario del programa.

Problemas que surgan, fallas, los así llamados bugs se pueden corregir fácil y rápidamente. Una vez detectado el error, se busca en el código fuente la instrucción que lo causa. Después de corregirla, se vuelve a compilar el programa y se solucionó la falla. Pero incluso cuando el usuario no puede solucionar el problema porque carece de los conocimientos técnicos necesarios, es muy probable que los desarrolladores u otro usuario capacitado le puedan ayudar.

El desarrollo de estos programas no está en manos de empresas poderosas, sino que se realiza a través de una comunidad de programadores que están en contacto mediante internet. Estos desarrollos cooperativos han demostrado ser sumamente eficientes. Los usuarios del software son incluidos en la comunidad de desarrollo, generalmente mediante las listas de discusión. Se aprovecha toda la experiencia de los usuarios del software, corrigiendo defectos, siguiendo las sugerencias y mejorando en todo el programa. Como regla general los proyectos nuevos tardan en arrancar. Una vez que suficientes personas usen el programa y se haya alcanzado una masa crítica, el programa toma envión y llega a una madurez relativamente rápida. Según un estudio realizado por el empresa Microsoft, el resultado se compara o supera los programas comerciales.

Estos programas aprovechan al máximo desarrollos previos y utilizan librerías de funciones, módulos e incluye partes del código de otros programas. Esto es una característica deseada y muchos programadores se esfuerzan de escribir código reutilizable para no reinventar constantemente la rueda. Esta característica le otorga a los softwares abiertos una gran facilidad para interactuar con otros programas e integrarse en sistemas más complejos.

El usuario cumple una importante función en el desarrollo del programa. Es el que detecta primero los errores y el que interactúa con los desarrolladores para orientar el desarrollo del programa a lo que necesita para su trabajo diario.

Otra característica es la licencia de uso que acompaña estos programas. Mientras que las licencias comerciales, sumamente restrictivas permiten por lo general usar el programa en una sola máquina y hacer una copia de seguridad del programa. Los programas abiertos se distribuyen con diferentes tipos de licencia. La más difundida es la Licencia Pública General (GPL = General public Licence) que permite no solamente el uso irrestricto del software, sino también su copia, distribución y modificación. La única condición que establece, es la obligación de mantener bajo la misma licencia el software obtenido a partir de la modificación del código fuente. Esto significa que todos los derivados deben permanecer libre para su uso, copia y distribución.

Científicos e investigadores están advirtiéndolo que el uso de software cerrado atenta contra uno de los pilares del método científico: la verificabilidad. Para aceptar como válido todos los cálculos usados para

apoyar una asunción científica determinada, deben estar disponibles a la verificación pública. Esto es posible únicamente con software abierto.

Proyecto de Ley para que el Estado use Software abierto

En Argentina existe un proyecto de ley para que el estado use software abierto. Las razones esgrimidas son: la seguridad nacional, porque el software abierto permite evaluar los la seguridad de los programas. La dependencia tecnológica, ya que el software abierto posibilita la actualización y el desarrollo local del mismo. El acceso a los datos que en formatos propietarios solamente pueden ser descifrados por programas propietarios para los cuales no existe ninguna seguridad que en el futuro sigan disponibles. Existen empresas que tienen una posición monopólica en el mercado del software y que, debido al gran número de clientes cautivos, pueden dictar las condiciones que desean referidos a precios, condiciones de uso e incluso el hardware que debe usar el usuario.

El software libre fomenta el desarrollo local, permitiendo a profesionales locales ofrecer un apoyo técnico real, donde pueden corregir fallas del sistema, desarrollar aplicaciones adaptadas a necesidades locales y a través de estas actividades contribuir al desarrollo nacional y regional.

GNU/Linux

Linux es un sistema operativo abierto. Acaba de cumplir 10 años de edad. En estos 10 años ha pasado de un juguete de programadores a una alternativa viable para cualquier usuario de computación y una seria competencia para MS Windows.

Sus características principales son su gran robustez, es conocido porque no se cuelga como MS Windows, funciona sobre una gran variedad de hardware: en las PC corre a partir del chip 80386, lo que comunmente se denomina "386, pero también en otros tipos computadoras como SPARC, DEC Alpha, PowerPC/PowerMac, Mac/Amiga, etc. Las redes constituyen su hábitat natural, es multiusuario y multitarea por naturaleza. Toda la estructura de Internet se construyó sobre este tipo de sistemas por lo que existen herramientas probadas para todos los servicios de Internet.

Para el usuario acostumbrado a MS Windows, Linux también ofrece una interfaz gráfica, el sistema X-Windows. A diferencia de MS Windows es un sistema mucho más configurable, donde el usuario puede elegir el tamaño del escritorio y se puede elegir la apariencia de las ventanas. Mientras que Linux todavía tiene fama de ser difícil de usar, las distribuciones actuales son muy amigables al usuario promedio.

Linux ha comenzado a impactar también en el ámbito empresarial. En los Estados Unidos viene de la mano de servicios web, como servidor barato y poco complicado, expandiéndose posteriormente en máquinas de escritorio. Según una encuesta de Infoworld CTO Network, el motivo principal de la introducción de software abierto es la reducción de costos. Pero le siguen con un 72% la reducción de tiempo de desarrollo e implementación y un 45% de los gerentes encuestados lo justificaron con la calidad superior del software.

En Europa el criterio de ahorro de costos también es un motor poderoso para la implementación de Linux, lo es además el miedo a la dependencia tecnológica de los EEUU. Linux es usado por ejemplo en empresas como el correo alemán, Daimler Chrysler, los ferrocarriles alemanes, IKEA (una gran distribuidora de muebles suecos), bancos, empresas de seguro y muchos más. Alemania, Francia y los países escandinavos son los líderes en el uso y desarrollo de Linux en Europa.

En México Linux se está instalando en 140.000 escuelas primarias y secundarias para formar una red nacional escolar. Usando Linux en lugar de MS Windows ha permitido ahorros de más de 100 millones de dólares que pueden ser usados para la compra de computadoras.

Para una biblioteca, Linux constituye un sistema operativo ideal. Es un sistema escalable, que se adapta a redes pequeñas como a grandes. Permite definir diferentes tipos de usuarios con diferentes atribuciones y permisos. Es posible configurar una red Linux para realizar todos los trabajos bibliotecarios interconectados en red, a la vez tener un servidor web para el catálogo y los demás servicios en Internet, y una subred para los usuarios de la biblioteca que pueden contar en sala de lectura con servicios de computación general, como procesadores de texto y planillas de cálculo como también servicios de internet.

Linux permite a las bibliotecas comenzar con pocas máquinas e ir integrando nuevas máquinas a medida que se incorporen. Permite integrar en red máquinas con diferentes sistemas operativos. Pueden coexistir pacíficamente computadoras con Linux, Windows NT, Windows 98, Windows 2000 y Macintosh en red. Máquinas más viejas, con discos duros pequeños, e incluso sin disco duro, se pueden reciclar para ofrecer terminales de consulta del catálogo y de otros recursos de la biblioteca. Computadoras viejas, "386" y "486" sin monitor pueden emplearse como servidores de archivo y servidores de impresión, aliviando la tarea de las máquinas de escritorio.

Una vez instalado y configurado el sistema, se produce también un ahorro significativo en gastos de mantenimiento. Marcelo Baldi, informático a cargo de la red de la Academia Nacional de Ciencias (en Córdoba) relata que con computadoras con MS Windows, tenía que recurrir cada dos a tres días a solucionar problemas de configuración, archivos borrados accidentalmente, virus, etc. Actualmente con Linux, el tiempo destinado al mantenimiento del sistema ha disminuído a una visita rutinaria por semana.

Por qué entonces no se usa Linux en las bibliotecas? En algunas bibliotecas ya ha comenzado a usarse. Aquí en Córdoba en la Academia Nacional de Ciencias, en otras localidades también se está comenzando a implementarse. En muchos casos es simplemente desconocimiento. También faltan informáticos capacitados. Las instituciones de capacitación enseñan generalmente MS Windows. En Córdoba capacitan en Linux la UTN y Famaf de la UNC.

Sistemas bibliotecarios abiertos

Dentro del movimiento de software de fuente abierta también hay proyectos para desarrollar sistemas de gestión bibliotecario. Existe un sitio dedicado a software abierto para bibliotecas: en <http://www.oss4lib.org> figuran tres proyectos:

☞ Avanti:

Avanti inicia en Octubre de 1998 por iniciativa de Peter Schlump, especialista en sistemas de información del Sistema de Bibliotecas Suburbanas Norte en Wheeling, Illinois.

Igual que ODSL enfoca sobre un sistema escalable que se adapta a cualquier tamaño de biblioteca. Mientras que los demás sistemas han sido desarrollados sobre Linux, Avanti enfoca sobre una implementación neutral e independiente del sistema operativo. Para ello usa el lenguaje de programación Java que permite usar el mismo programa en diferentes sistemas operativos.

El diseño del programa se orienta en un modelo general de biblioteca, no en una biblioteca en

particular. Esto es un problema recurrente en sistemas comerciales que generalmente nacen en una biblioteca determinada y después se venden a otras bibliotecas que pueden tener flujos de trabajos muy diferentes. Los módulos actuales incluyen circulación, catalogación y catálogo al público (OPAC). Avanti usa una base de datos Java de desarrollo propio: PicoDB.

✍ **OSDLS: Open Source Digital Library System / Phyteas**

Se inicia a comienzos de 1999 con bibliotecarios e informáticos de las Universidades de Arizona, Yale y de Windsor y con el objetivo de crear un sistema de gestión bibliotecaria de próxima generación. A diferencia de los sistemas comerciales actuales que apuntan a bibliotecas de determinado tamaño este sistema intenta ser independiente del tamaño de la biblioteca.

Actualmente corre bajo Linux. Integran tecnologías como XML, RDF, MySQL, Javascript, Java, RMI, LiveConnect, y otras en una arquitectura

Los módulos disponibles actualmente incluyen a: Phyteas, un editor Marc y OPAC. Actualmente se buscan interesados para integrar grupos de trabajo para: módulo de adquisiciones, circulación y publicaciones seriadas.

✍ **OpenBook**

OpenBook es un sistema de gestión bibliotecaria desarrollada para pequeñas bibliotecas escolares y públicas por la Technology Resource Foundation. Se basa en un sistema ya existente: Koha, que fue desarrollado en Nueva Zelanda para una biblioteca pública. Incluye: 3 módulos: OPAC, módulo de catalogación, módulo de circulación. Para el futuro se espera implementar un módulo de adquisición.

Características incluye pantallas (Interfaces) basadas en Web. Esto significa que se usa mediante el navegador estándar del sistema. Es multilingüe: inglés, francés, español. Incluye control de autoridades y se está desarrollando un cliente Z39.50 para integrarlo con su módulo de catalogación para importar registros de otras bases de datos.

Está optimizado para Linux, pero funciona también con Windows NT. A diferencia de Koha, es compatible con Marc21. Utiliza tecnologías como: PHP, Perl, MySQL, servidor Apache.

✍ **OpenIsis**

Para qué una nueva herramienta para Isis, si ya se cuenta con toda una gama de opciones para trabajar en Isis. Existe el viejo MicroIsis para máquinas viejas que no funcionan con Windows. Ahora se está usando WinIsis y hay una serie de utilitarios y herramientas para desarrolladores, programadas por el equipo de Bireme (Biblioteca Regional de Medicina de la PAHO).

Sin embargo, después de una revisión exhaustiva del software existente se llegó a la conclusión que el software actual para bases de datos Isis no cubre todo el espectro de aplicaciones requeridas. Falta, por ejemplo, un potente servidor web que pueda atender varias consultas simultáneamente. Además de ello, las herramientas de desarrollo disponibles (utilitarios y librerías de Bireme) son propietarias y cerradas. Para Linux hay muy pocas opciones y con la excepción del cliente de JavaIsis y una librería básica, no hay software abierto para Isis.

La distribución es deficiente. Internet ha demostrado ser el medio de distribución de software más

eficiente en la actualidad. Para muchas aplicaciones y herramientas, a excepción de las de la Unesco, no se dispone de licencias. No se sabe que está permitido y que no. Para el desarrollador que desea usar Isis para crear nuevo software, existe una gama muy limitada de lenguajes de programación disponibles. Sobre todo nuevos lenguajes de amplia difusión en Internet, como Perl, Python y Java no cuentan con módulos o asociaciones a las herramientas de desarrollo existentes.

OpenIsis actualmente

OpenIsis se presenta actualmente como una librería de funciones para el lenguaje de programación C. No es una aplicación, lo que generalmente conocemos por programa, sino una herramienta para programadores que desean desarrollar programas para Isis, sobre todo en ambiente Web.

La versión actual de OpenIsis es la versión 0.8.2. A pesar de es desarrollada por dos programadores alemanes: Klaus (Paul) Ripke y Erik Grziwotz, OpenIsis nació en mayo de este año (2001) en la Ciudad de Córdoba, Argentina. Maneja bases de datos más grandes que MicroIsis y WinIsis y no permite manipular bases de datos para Windows o para Unix indistintamente. Sus funciones actuales incluyen la visualización de registros en forma secuencial y la búsqueda en texto libre. Maneja campos y subcampos repetibles y cuenta con asociaciones para ser utilizados con los lenguajes de programación Perl y Java.

OpenIsis objetivos a corto plazo

Entre los objetivos que se tienen para OpenIsis a corto plazo el más importante es sin lugar a dudas la búsqueda por archivo invertido. Una vez implementado ésta, OpenIsis estaría habilitado para ser usado como servidor web para bases de datos Isis. Otro objetivo es implementar el lenguaje de búsqueda propio de Isis.

Con estos desarrollos, se estará en condiciones de desarrollar un servidor Java con dos aplicaciones inmediatas: un servidor web de alto rendimiento que puede manejar varias conexiones simultáneas permitirá mejorar el servicio de catálogo en web (webcat) de bibliotecas importantes que actualmente sufren la limitada capacidad de los servidores web para Isis existentes. La segunda aplicación consiste en desarrollar un servidor para JavaIsis. Este software es distribuido por la Unesco con todo su código fuente, salvo el servidor web que es wwwisis 3.x de Bireme. Un fuerte servidor web para JavaIsis permitirá ampliar el uso de JavaIsis para carga de datos remotas e importación de registros bibliográficos.

Finalmente también se deberá implementar el lenguaje de formateo propio de Isis en OpenIsis para mantener la compatibilidad con los sistemas existentes.

OpenIsis objetivos a mediano plazo

Más allá de estos objetivos, OpenIsis deberá ofrecer en un futuro intermedio también funciones realizadas actualmente mediante otras aplicaciones. Entre ellas se pueden mencionar la carga y edición de registros y la actualización y generación del archivo invertido. Esto permitirá emplear OpenIsis en sistemas de gestión bibliotecarios abiertos en su totalidad y fomentar el uso de Isis en lugar de bases de datos relacionales que son más difíciles de programar y mantener para datos bibliográficas.

4. Consecuencias para las bibliotecas (Conclusiones)

En primer lugar el software abierto ofrece mejores posibilidades de evolución y desarrollo para el futuro. En cuanto a estabilidad, adaptabilidad, integración con sistemas existentes y costos, los sistemas abiertos son claramente superiores a los sistemas cerrados.

El uso en el ámbito bibliotecario requiere de desarrollos específicos: OpenBook, Avanti, OSDLS son proyectos de sistemas de gestión bibliotecaria abiertos. Estos sistemas están disponibles en corto tiempo (OpenBook ya tiene sus tres módulos básicos funcionando) y comienzan a constituir una verdadera alternativa al software comercial.

Los bibliotecarios necesitan comprometerse aún más en el desarrollo de software para bibliotecas. Es necesario que tomen conciencia que deben trabajar conjuntamente con los informáticos en el desarrollo, mejoramiento y aplicación del software bibliotecario. Se requiere aún más capacitación en las tecnologías de la información para constituir interlocutores válidos.

Y finalmente, más allá del desarrollo de software, la metodología empleada en el desarrollo de software abierto puede usarse igualmente para desarrollar herramientas bibliotecológicas como normas abiertas de catalogación, sistemas de clasificación o tesauros. El desarrollo cooperativo ha sido empleado en las bibliotecas con resultados sobresalientes. Las posibilidades actuales permiten realizar desarrollos cooperativos descentralizados, extendiendo y ampliando los límites territoriales de las redes bibliotecarias. La combinación precisa de herramientas de internet para el desarrollo de sistemas sitios web de desarrollo (sourceforge.net) / servidores ftp / listas de discusión / foros de discusión / grupos de noticias permite combinar y ampliar las capacidades humanas en nuestras bibliotecas.

Enlaces (links) útiles

Oss4lib : open source systems for libraries - <http://www.oss4lib.org>

Sitio dedicado al software abierto para bibliotecas. Novedades, lista de discusión, artículos, enlaces a proyectos de software abierto para bibliotecas (en inglés).

GrULiC - <http://www.grulic.org.ar>

El Grupo de Usuarios de Linux de Córdoba (Argentina) ofrece en su sitio información básica sobre Linux, una lista de discusión con acceso a través de un buscador y contactos Linux en Córdoba.

Fundación Via Libre - <http://www.vialibre.org.ar>

La Fundación Via Libre tiene como objetivo la difusión de Linux en Pymes Instituciones. Ofrecen capacitación en Linux.

L.U.G.Ar (Linux Users Group Argentina). - <http://www.linux.org.ar/>

Información, artículos, tutoriales y mucho más. Recomendable: AULA.

Free Software Foundation - <http://www.free-soft.org/>

Fundadores del movimiento del software libre. Información, documentos sobre el software libre (en inglés).

Gnome - <http://www.gnome.org>

El proyecto Gnome abarca tres áreas: un escritorio simple de usar con interfaz grafica (similar a MS Windows), una plataforma de desarrollo para crear nuevos programas Gnome y, Gnome-Office, una coleccion de programas para oficina. Información, enlaces y aplicaciones para bajar e instalar (en inglés).

Open Source Education Foundation - <http://www.osef.org/>

Sitio dedicado a la promoción de software abierto en escuelas (en inglés).

Lecturas recomendadas:

Padula, Diego A. Sobre Open Source [en línea]. Buenos Aires: Linux Users Group Argentina, 1999.

<http://www.linux.org.ar/AULA/docs/opinion/sobre_open_source/sobre_open_source.htm>

Explica los diferentes paradigmas entre el software cerrado y el software abierto. Información sólida y entendible para el no-informático.

Ali, Adrian Pablo. La era de la estupidez [en línea]. Tucumán : Linux User Group Tucumán, 2000.

<<http://www.linux.org.ar/AULA/docs/opinion/estu/estu.htm>>

Comparación entre MS Windows y Linux.

Bibliografía

Ali, Adrian Pablo. La era de la estupidez [en línea]. Tucumán : Linux User Group Tucumán, 2000.

<<http://www.linux.org.ar/AULA/docs/opinion/estu/estu.htm>> [Consulta: 5. Sept. 2001]

Cave, Damien. Microsoft to schools: Give us your lunch money! [en línea]. Salon.com. 10 Julio 2001.

<http://www.salon.com/tech/feature/2001/07/10/microsoft_school/index.html> [Consulta: 7. Sept. 2001]

Chudnov, Dan. Open Source Library Systems: Getting Started [en línea]. Yale, Oss4lib, 1999.

<<http://www.oss4lib.org/readings/oss4lib-getting-started.php>> [Consulta: 1. Sept. 2001]

Chudnov, Dan. Oss4lib : open source system for libraries [en línea]. Yale : Dan Chudnov, 2001.

<<http://www.oss4lib.org>> [Consulta: 1. Sept. 2001]

Heinz, Federico. Razones por las que el Estado debe usar software libre [en línea]. Córdoba

(Argentina) : Grulic, 2001. <<http://www.grulic.org.ar/proposicion/doc/razones.html>> [Consulta: 2. Sept. 2001]

Kahney, Leander. Mexican Schools Embrace Linux [en línea]. Wired News. 6 Noviembre 1998.

<<http://www.wired.com/news/technology/0,1282,16107,00.html>> [Consulta: 4. Sept. 2001]

Marqués, Martín Argentino. Por qué usa la gente Windows? [en línea]. Buenos Aires : Linux Argentina,

[2000?]. <<http://www.linux.org.ar/AULA/docs/opinion/MMarques-1/MMarques-1.htm>> [Consulta: 7. Sept. 2001]

Moisset, Daniel. Qué es Linux? [en línea]. Córdoba : Grulic, 1999.

<<http://www.grulic.org.ar/linux.html>> [Consulta: 5. Sept. 2001]

Open Source Initiative. Opensource.org [en línea]. [s.l.]: Open Source Initiative, 2001

<<http://www.opensource.org>> [Consulta: 7. Sept. 2001]

The Open Source Digital Library System Project [en línea]. Tucson : OSDLS, [1999?].

<<http://osdls.library.arizona.edu/>> [Consulta: 5. Sept. 2001]

Padula, Diego A. Sobre Open Source [en línea]. Buenos Aires : Linux Argentina, 1999.

<http://www.linux.org.ar/AULA/docs/opinion/sobre_open_source/sobre_open_source.htm> [Consulta:

2. Sept. 2001]

Pfaffenberger, Bryan. Linux in Higher Education: Open Source, Open Minds, Social Justice [en línea]. Linux Journal. 21 Marzo 2000.

<<http://www2.linuxjournal.com/articles/currents/017.html>> [Consulta: 2. Sept. 2001]

Raymond, Eric S. The Cathedral and the Bazaar [en línea]. Boston, Free Software Foundation, 1998.

<<http://www.free-soft.org/literature/papers/esr/cathedral-bazaar/>> [Consulta: 5. Sept. 2001]

Ripke, Klaus. Welcome to OpenIis.org [en línea] / Klaus Ripke, Erik Grziwotz. Berlin : Openisis.org, 2001. <<http://www.openisis.org>> [Consulta: 1. Sept. 2001]

Scoville, Thomas. OSS Europe: Open Source over there, continental savoir-faire [en línea]. Cambridge : O'Reilly Opensource Center, 2001.

<http://opensource.oreilly.com/news/oss_europe_0299.html> [Consulta: 5. Sept. 2001]

Schumpf, Peter. Avanti : an open source library automation system [en línea]. [Wheeling] : Peter Schlumpf, 2001. <<http://www.avantilibrarysystems.com/>> [Consulta: 6. Sept. 2001]

Stallmann, Richard. Copyleft: Pragmatic Idealism [en línea]. Boston : Free Software Foundation, 1999.

<<http://www.free-soft.org/literature/papers/gnu/pragmatic.html>> [Consulta: 7. Sept. 2001]

Stallmann, Richard. The GNU Manifesto [en línea]. Boston : Free Software Foundation, 1985.

<<http://www.free-soft.org/literature/papers/gnu/manifesto.html>> [Consulta: 7. Sept. 2001]

Stoltz, Mitch. The case for government promotion of Open Source software : a NetAction white paper [en línea]. San Francisco : NetAction, 1999.

<<http://www.netaction.org/opensrc/oss-report.html>> [Consulta: 4. Sept. 2001]

Technology Resource Foundation. Projects : OpenBook [en línea]. Seattle, Technology Resource Foundation, 2001. <<http://www.trfoundation.org/projects/openbook.html>> [Consulta: 5. Sept. 2001]

The time of the penguin [en línea] / by CNET news.com Staff. San Francisco : Cnet News.com, 2001.

<<http://news.cnet.com/news/0-1003-201-6947032-0.html>> [Consulta: 7. Sept. 2001]

Yager, Tom. Open source takes hold [en línea]. InfoWorld. 24 Agosto 2001.

<<http://www.infoworld.com/articles/tc/xml/01/08/27/010827tcintro.xml>> [Consulta: 7. Sept. 2001]

Adiciones:

Cuando hice este trabajo no incluí al sistema Koha (<http://www.koha.org>), suponiendo que se trataba de una versión anterior de OpenBook. No es así, OpenBook fue creado sobre la base de Koha: mientras que Koha está escrito en Perl, OpenBook se hizo con PHP.

Por alguna razón misteriosa se me escapó uno de los mejores trabajos sobre Software abierto en español: [La encrucijada digital](#) de Marcelo C. Baldi, Fernando A. Cuenca y Daniel F. Moisset.

