

# Automatización de unidades de información: Matriz técnica para la evaluación de software libre

---

## Resumen

En este artículo se presenta una propuesta de matriz de evaluación para software libre orientado a la automatización de unidades de información. Esta matriz está compuesta por una serie de parámetros técnicos que facilitarán la evaluación de diferentes programas de aplicación de software libre, los cuales se agruparán en cuatro distintas clasificaciones. Cada una de ellas permitirá filtrar las diferentes aplicaciones que cumplan uno u otro requisito de acuerdo a las necesidades de información y automatización que tenga la unidad de información. Por tanto esta matriz se convierte en una herramienta útil que servirá al equipo responsable de la automatización de una unidad de información en el proceso de evaluar y seleccionar software libre para este propósito.

**Palabras clave:** automatización de bibliotecas, software libre, evaluación de software

## Automation of Information Units: Matrix Technique for the Evaluation of Free Software

## Abstract

This article presents a proposal evaluation matrix for free software automation oriented information units. This matrix is composed of a number of technical parameters that will facilitate the evaluation of different free software applications, which are grouped into four different classifications. Each will filter the different applications that meet either requirement according to the information and automation needs to have the information unit. Thus this matrix becomes a useful tool that will serve the team responsible for the automation of a unit of information in the process of evaluating and selecting free software for this purpose.

**Keywords:** library automation, free software, software evaluation

**Cómo citar este artículo:** FERNÁNDEZ MORALES, Mynor y CHINCHILLA ARLEY, Ricardo. Automatización de unidades de información: Matriz técnica para la evaluación de software libre. Revista Interamericana de Bibliotecología 2013, vol. 36, n° 3, pp. 207-219.

**Recibido:** 2013-08-14 / **Aceptado:** 2013-10-11

### **Mynor Fernández Morales**

Máster en Administración de Negocios,  
Instituto Tecnológico de Costa Rica.  
Licenciado en Computación e Informática,  
Universidad de Costa Rica.  
Profesor e investigador  
Escuela de Bibliotecología y Ciencias  
de la Información  
San Pedro de Montes de Oca - Costa Rica  
mynor.fernandez@ucr.ac.cr

### **Ricardo Chinchilla Arley**

Máster en Computación, Instituto  
Tecnológico de Costa Rica.  
Licenciado en Bibliotecología  
y Ciencias de la Información,  
Universidad de Costa Rica.  
Profesor e investigador  
Escuela de Bibliotecología y Ciencias  
de la Información.  
San Pedro de Montes de Oca - Costa Rica  
ricardo.chinchilla@ucr.ac.cr

## 1. Introducción

Uno de los principales problemas al que se enfrenta el bibliotecólogo responsable de un proyecto de automatización de una unidad de información es determinar cómo evaluar las distintas alternativas de aplicaciones o paquetes de software para establecer cuál es la mejor opción para su proyecto concreto. De acuerdo con Damsgard J. & Karlsbjerg J. (2010), software empaquetado es una categoría de sistemas de información en la que todas las implementaciones son esencialmente idénticas, es decir, las principales funcionalidades son comunes en todas las instalaciones. Mientras que los componentes básicos de un paquete son idénticos en todas las organizaciones que lo usan, la aplicación personalizada en un sistema de información individual se configura en alguna forma para adaptarse a los requisitos de cada una de ellas. Si bien los parámetros descritos en este artículo pueden ser utilizados para cualquier aplicación, ya sea libre o privativa, el enfoque se realiza sobre las aplicaciones libres, debido a las bondades demostradas, principalmente el disponer del código fuente, tal y como lo indica Vilela (2005, p.7)

La característica principal de los sistemas en software abierto es que se distribuye con su código fuente. Esto permite no solamente que el programa se pueda adaptar a las necesidades locales de cada biblioteca, sino que el programador puede evaluar el código fuente y con ello mejorar la calidad del programa. Estos sistemas pueden en algunos casos tener costo y en otros ser distribuidos gratuitamente

Además, es importante tener presente, tal y como lo expresan Moreira y otros (2011, p. 213), que ninguna aplicación va a responder totalmente a todos los requerimientos de una organización específica; sin embargo al aportar el software libre el código fuente, es posible adaptarlo, actualizarlo e integrarlo a los cambiantes contextos organizacionales.

Regresando al problema planteado, es importante anotar que, si bien es cierto que los proyectos de automatización de unidades de información generalmente son realizados por grupos interdisciplinarios, el bibliotecólogo o bibliotecólogos integrados a ese grupo enfrentan una gran responsabilidad propia de su cargo al momento de evaluar software. De aquí la importancia de que el bibliotecólogo tenga una herramienta de evaluación

que le facilite la realización de esta importante tarea. Este documento, el cual se enmarca dentro del proyecto de investigación inscrito en la Vicerrectoría de Investigación de la Universidad de Costa Rica, titulado *Análisis del software libre disponible en la WEB en idioma español, orientado a la automatización de las unidades de información, en los tres niveles a saber: a. catálogos automatizados, b. repositorios digitales y c. sistematización integral de las funciones operativas de las unidades de información*, propone una matriz de evaluación de software libre orientado a la automatización de unidades de información, la cual pretende convertirse en una herramienta técnica que se le proporcionará al grupo responsable de esta importante tarea. Esta matriz está conformada por una serie de parámetros cuya valoración permitirá obtener una comparación técnica de un conjunto de aplicaciones orientadas a la automatización de unidades de información.

Independientemente de la evaluación del software, en un proyecto de automatización surgen otras consideraciones; en primera instancia, en una unidad de información es importante señalar que:

Las bibliotecas son organizaciones que presentan distintos niveles de complejidad, que puede ir desde un nivel sencillo, pasando por un nivel medio, hasta alcanzar un nivel de alta complejidad. Para realizar esta clasificación, se utilizarán seis variables, con el fin de establecer el nivel de complejidad de la biblioteca. Las variables son las siguientes: tipo de biblioteca, tamaño de la colección, cantidad y tipo de usuarios que maneja, nivel de especialización del recurso humano que tiene la biblioteca y, finalmente, la capacidad técnica y económica de la biblioteca. De acuerdo con esto, y por razones de conveniencia práctica, estableceremos tres subniveles de clasificación de las bibliotecas: nivel sencillo, nivel de mediana complejidad y nivel de alta complejidad (Fernández, 2013, p. 2).

En segunda instancia, un proyecto de automatización implica la realización de otras actividades adicionales a la tarea de evaluación de software, tareas que aquí no son contempladas. Así “*Es una buena medida realizar una evaluación de nuestras capacidades como biblioteca en términos financieros, de recursos humanos y de infraestructura tecnológica. Aquí nos daremos cuenta si un proyecto de automatización es viable o no en nuestras instituciones*” (Sifuentes, 2005, p.1). Por ello, esta matriz de evaluación viene a complementar dicha evaluación, contemplada en el estudio de fac-

tibilidad técnica, operacional y económica que se debe realizar.

## 2. Metodología

Un software para la automatización de unidades de información es un sistema sumamente particular.

Dado que 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 préstamos, estadísticas de utilización, etc. Deben existir además diferentes puntos de acceso, búsquedas y formas de visualización. Los soportes digitales (disquetes o CD ROM) 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. El software debe cumplir también con las diversas normas bibliotecarias: ISBD, AACR2, CDU, Dewey, Tesauros, MARC, etc. (Vilela del Águila, 2005, p.2).

Para lograr un conocimiento al respecto por parte de los responsables de cualquier proyecto de automatización de unidades de información, se propone aquí una matriz de evaluación del software libre, compuesta por una serie de parámetros y que fue confeccionada a partir de una investigación bibliográfica y la experiencia profesional y académica de los investigadores que trabajan en este proyecto de investigación.

Para la confección de la matriz de evaluación se tomaron en cuenta los niveles de automatización de unidades de información, los cuales agrupan el software en categorías comparables, para no caer en el error de comparar software con diferentes propósitos. Estas categorías de software se establecieron de acuerdo con Chinchilla (2011), quien dice que existen tres niveles: automatización de catálogos, repositorios digitales y automatización integral de bibliotecas. Sin embargo, es importante tener presente que, según nos dice Bertot (2004), no hay definiciones estándar o enfoques de evaluación de unidades de información establecidos, así como estrategias o prácticas definidas. Cada enfoque de evaluación ofrece información potencial particular a un área específica de la unidad de información.

Para facilidad de manejo y comparación de los parámetros de la matriz, se decidió agrupar estos en cuatro categorías, a saber: 1. Generalidades: agrupa todos los parámetros relacionados a aspectos generales de la aplicación; 2. Gestión de Estándares: abarca todos los parámetros relacionados con el manejo de estándares y que permiten la interoperabilidad del software; 3. Aspectos Técnicos varios: comprende características técnicas importantes de diversa índole que identifican el software, y finalmente, 4. Funcionalidad: que son los parámetros técnicos que califican las operaciones que caracterizan la funcionalidad de cada uno de los módulos que implementa el software.

Al estar basada esta matriz en la propuesta de Chinchilla, el cuarto grupo de parámetros, relativos a la funcionalidad de los módulos para cada área funcional de una unidad de información (catalogación, circulación, adquisiciones, inventarios, etc.), no aplica para el primer y segundo nivel, ya que los sistemas que agrupan no cuentan con dichos módulos.

## 3. Resultados

Cómo se indicó en la sección anterior, los parámetros de la matriz serán clasificados en cuatro grupos. El primero de ellos se refiere a las generalidades que caracterizan al software, donde se incluyen 19 parámetros como nombre de la aplicación, versión, país de origen, idioma, creador, entre otros. El segundo grupo contempla 7 parámetros que se refieren a la gestión de estándares, que permitirán evaluar el grado de interoperabilidad del software a través de la importación y exportación de información, utilizando, o no, formatos estándar para el intercambio de información. En el tercer grupo se contemplan 13 parámetros relacionados con aspectos técnicos varios que caracterizan al software, tales como facilidad de auditoría, amigabilidad, eficiencia de la ejecución, facilidad de desarrollo de nuevos requerimientos y otros parámetros que permitirán tener una idea más concreta de las características técnicas de la aplicación. Finalmente, se incluyen en el último grupo 16 parámetros que evalúan la funcionalidad de la aplicación, como módulos disponibles y configuración de políticas generales que gobiernan el software (parametrización).

Aunque la matriz de evaluación incluye los principales parámetros para realizar una evaluación técnica del

software, esto no limita de ninguna forma la inclusión futura de nuevos parámetros que sean de valor técnico y decisorio para el grupo responsable del proceso de automatización de una unidad de información. Por cuanto, si se revisan las páginas Web de las múltiples aplicaciones de software libre para automatización de unidades de información, es fácil detectar que se está ante un campo en evolución continua y constante cambio, que sufre de múltiples transformaciones.

En el anexo 1 se presenta la matriz de evaluación propuesta, donde se observan las cuatro categorías y los parámetros técnicos que las integran.

Es importante resumir las principales características que forman cada una de las cuatro categorías, donde el primer grupo generalidades, corresponde a parámetros que identifican las aplicaciones y son útiles para que el usuario pueda establecer un reconocimiento técnico y conocer donde obtenerlo. El segundo grupo se refiere al manejo de estándares internacionales que permitirán

una interoperabilidad del software a través de la importación y exportación de archivos a otros sistemas. El tercer grupo comprende diversos aspectos técnicos que caracterizan al software y qué pueden tener poca o mucha relevancia para cubrir la necesidades del usuario, en su proyectos de automatización. Finalmente, la última categoría, denominada funcionalidad de la aplicación, cubre aspectos técnicos propios de la funcionalidad. Aquí es importante señalar que esta categoría se aplica al software del tercer nivel, o sea el orientado a la automatización integral de bibliotecas.

### 3.1 Primer grupo: Generalidades

En este primer grupo se incluirán 20 parámetros referentes a características generales que identifican la aplicación, este primer grupo pretende hacer una identificación general de la aplicación y de sus principales características.

El resumen detallado de este primer grupo se presenta en la tabla 1: Generalidades que se muestra a continuación:

Tabla 1: Generalidades

Número de parámetro	Nombre del parámetro	Descripción
1	Nombre de la aplicación	Se refiere al nombre de la aplicación
2	Versión	Se refiere a la versión de la aplicación (este es muy importante, ya que las aplicaciones se liberan para el uso público en una determinada fecha, y se identifican con un número de versión). Una misma aplicación pueda estar funcionando en diferentes organizaciones, pero con números de versiones diferentes, lo cual las convierte en aplicaciones distintas.
3	País de Origen	Corresponde al país de origen de la aplicación.
4	Idiomas que maneja	Es referente a los idiomas de la aplicación.
5	Creador	Contiene el creador de la aplicación.
6	Licencia	Es referente al tipo de licencia de la aplicación, entre las cuales tenemos: <ul style="list-style-type: none"> <li>• Common Public License v.1.0.</li> <li>• GNU General Public License v.2.0.</li> <li>• GNU General Public License v.3.0.</li> <li>• Eclipse Public License.</li> <li>• eCos License v.2.0</li> <li>• Sleepycat Software Product License.</li> <li>• Affero License v.1.0</li> <li>• Affero License v.2.0</li> <li>• OpenSSL License.</li> </ul>
7	Página de Web	Se tiene la dirección URL de la página Web de la aplicación
8	Página de Descarga	Indica la dirección URL de su página de descarga

Número de parámetro	Nombre del parámetro	Descripción
9	Documentación	Dice si existe documentación técnica
10	Interfaz	Es referente a la interfaz que presenta la aplicación, ya sea si es a través de un browser que se visualiza y opera, o si es a través de una interfaz propia.
11	Plataforma	Se refiere a la plataforma en que se encuentra desarrollado el software: 1 si es un sistema de archivos, 2 si es cliente/servidor, y 3 si es una aplicación WEB.
12	Lenguaje(s) de programación	Corresponde a los lenguajes de programación utilizados para su desarrollo.
13	Modificación del código fuente	Se refiere a si se permite la modificación del código fuente.
14	Costo del software	Trata sobre el costo. Aunque se trata de software libre y no debería en principio tener costo, se verifica la existencia o no de algún costo por obtener la aplicación.
15	Mantenimiento anual	Va relacionado con el anterior, pero aquí se verifica la existencia o no de algún costo de mantenimiento anual
16	Actualización periódica	Es referente a la actualización periódica, con el fin de reflejar si el software está en revisión y actualización periódica o si por el contrario no ofrece la posibilidad de actualizaciones.
17	Motor de bases de datos	Incluye los motores de bases de datos soportados.
18	Robustez del motor de base de datos	El parámetro 19 está estrechamente relacionado con el parámetro 18, y se relaciona con la robustez del motor de base.
19	Soporte multibiblioteca	Relacionado con la aplicación, ofrece la posibilidad de soportar múltiples bibliotecas. Este parámetro es importante ya que permite determinar si es posible el manejo de varias unidades de información de manera independiente con una sola instalación del software en un servidor.
20	Sistema Operativo	Está relacionado con los sistemas operativos en los cuales puede ejecutarse la aplicación.

### 3.2 Segundo grupo: Gestión de estándares

Este grupo de la matriz agrupa 7 parámetros técnicos relacionados con la estructura interna del formato de almacenamiento de los datos, donde, de acuerdo al estándar que utilice la aplicación, se determina cómo están organizados y almacenados los datos de la misma, además de que se infiere cuáles facilidades tiene el software para la importación y exportación de información a través de la utilización de estándares definidos y aceptados por la comunidad bibliotecológica. La

disponibilidad o no de un determinado estándar para importar o exportar información hace al software más o menos interoperable. Un sistema con una alta interoperabilidad permite el intercambio de información entre diferentes sistemas, mientras que un sistema con una baja interoperabilidad no permite el intercambio de información con otros sistemas (Córdoba, 2011).

El resumen detallado de este grupo se presenta en la tabla 2: Gestión de Estándares, que se muestra a continuación:

Tabla 2: Gestión de estándares

Número de parámetro	Nombre del parámetro	Descripción
1	ISO 2709	Es un estándar ISO para la descripción bibliográfica, aquí se indica si la aplicación utiliza o no este estándar.
2	Protocolo Z39.50	Se refiere al protocolo Z39.50, que es un protocolo cliente-servidor orientado a facilitar la búsqueda y recuperación de información en distintos sistemas a través de una misma interfaz. Igualmente se indica si la aplicación ofrece o no este protocolo.
3	Exportación TXT rotulado	Se refiere a si se ofrece o no la posibilidad de exportar los datos almacenados a un formato texto TXT rotulado.

Número de parámetro	Nombre del parámetro	Descripción
4	MARC21	Es una norma para el intercambio de información que permite estructurar e identificar los datos de tal forma que puedan ser reconocidos y manipulados por computadora. Este formato fue creado por un equipo de bibliotecarios de la Biblioteca del Congreso de los Estados Unidos. Aquí también se indica si la aplicación tiene la posibilidad o no de manejar este formato.
5	MARXML	En el año 2002, la Biblioteca del Congreso desarrolló el esquema MARC-XML como una estructura alternativa de registro, que permite que los registros MARC sean representados en XML. Igualmente, aquí se indica si la aplicación tiene la posibilidad o no de manejar este formato.
6	SRU,XML	Luego tenemos el parámetro SRU (Search/Retrieve vía URL), el cual es un estándar para búsqueda y recuperación de información en formato XML. Aquí se indica si la aplicación soporta o no este estándar
7	Dublin Core	Se refiere a si permite o no el uso del formato Dublin Core.

### 3.3 Tercer grupo: Aspectos técnicos varios

Este grupo de la matriz contiene 13 parámetros técnicos que caracterizan a la aplicación y que son importantes de evaluar a la hora de considerar el software. El primer parámetro de este grupo es sobre si dispone o no dispone de la facilidad de auditoría. Esta característica técnica se refiere a las pistas de auditoría que genera el programa cuando son ejecutadas las diferentes transacciones permitidas en sus módulos. El siguiente parámetro se indica si el sistema es o no amigable; este aspecto se refiere a qué tan intuitivo es para el usuario final. La

amigabilidad de un sistema es una característica muy importante en las aplicaciones de hoy en día, donde el grado de amigabilidad con el usuario se incrementa día con día. Un ejemplo muy práctico de amigabilidad lo constituyen los teléfonos inteligentes, donde la competencia por un software cada vez más intuitivo es una de las características de mayor importancia para ganar clientes.

El resumen detallado de este grupo se presenta en la tabla 3: Aspectos Técnicos varios, que se muestra a continuación:

Tabla 3: Aspectos técnicos varios

Número de parámetro	Nombre del parámetro	Descripción
1	Facilidad de auditoría.	Es sobre si dispone o no dispone de la facilidad de auditoría. Esta característica técnica se refiere a las pistas de auditoría que genera el programa cuando son ejecutadas las diferentes transacciones permitidas en sus módulos
2	Amigabilidad	Se indica si el sistema es o no amigable; este aspecto se refiere a que tan intuitivo es para el usuario final. La amigabilidad de un sistema es una característica muy importante en las aplicaciones de hoy en día, donde el grado de amigabilidad con el usuario se incrementa día con día. Un ejemplo muy práctico de amigabilidad lo constituyen los teléfonos inteligentes, donde la competencia por un software cada vez más intuitivo es una de las características de mayor importancia para ganar clientes.
3	Facilidad de desarrollo de nuevos requerimientos.	Esta característica va a depender en primer lugar del lenguaje asociado a la aplicación y la disponibilidad del código fuente.
4	Independencia del hardware.	Se referente a la independencia del hardware, donde se indica si la aplicación es independiente o no del hardware en que se encuentra instalada.
5	Seguridad lógica.	Se establece en una escala de 1 a 5, donde 1 es poco seguro y 5 es muy seguro.

Número de parámetro	Nombre del parámetro	Descripción
6	Autodocumentado.	Se refiere a si el software se encuentra autodocumentado, es decir, si en la propia aplicación a través de teclas rápidas o funciones establecidas ofrece documentación y ayuda al usuario.
7	Simplicidad.	Se refiere al nivel de simplicidad de la aplicación; aquí se establece una escala de 1 a 5, donde 1 es poco simple y 5 es muy simple.
8	Independencia del sistema software.	Se refiere a la independencia que presenta la aplicación del sistema de software en que se apoya, aquí se indica si es o no independiente.
9	Disponibilidad	Que tan fácil es descargarla de Internet. Igualmente se calificará con una escala de 1 a 5, donde 1 significa poco disponible y 5 significa muy disponible.
10	Mantenibilidad	Que es posible darle al sistema con una escala de 1 a 5, donde 1 significa poco mantenible y 5 significa muy mantenible.
11	Soporte en línea	Aquí se indica si dispone o no dispone de dicho soporte.
12	Foros	Se refiere a la disponibilidad de foros en que los usuarios puedan participar para exponer sus dudas y problemas. Igualmente se indica si existen o no dichos foros.
13	Comunidades	Se refiere a la disponibilidad o no disponibilidad de comunidades de usuarios. Estas comunidades son de suma importancia para formar equipos de usuarios vía Internet que se identifican con fortalezas, debilidades y nuevos aportes y desarrollos que presenta la aplicación.

### 3.4 Cuarto grupo: Funcionalidad de la aplicación

Este último grupo de la matriz abarca 16 parámetros que se refieren a la funcionalidad del software a evaluar. Su importancia radica en que a través de ellos se obtiene una visión de cuáles son los módulos que la aplicación contempla, así como el nivel de completitud

de cada uno de ellos. En esta categoría se calificará cada uno de los parámetros con una nota de 0 a 5, donde 0 indica que no está presente el módulo, mientras 1 indica que es poco completo, y que 5 que muy completo.

El resumen detallado de este grupo se presenta en la tabla 4: Funcionalidad de la Aplicación, que se presenta a continuación:

Tabla 4: Funcionalidad de la aplicación

Número de parámetro	Nombre del parámetro	Descripción
1	Módulo Administrador	Se refiere a si la aplicación tiene un módulo de administración. El módulo Administrador es el que se encarga de todas las funciones administrativas y de configuración y parametrización de la aplicación.
2	Selección y Adquisiciones	Se refiere al módulo de Selección y Adquisiciones, responsable por facilitar y registrar los procesos de intercambios, compras y donaciones en una unidad de información.
3	Módulo Catalogación	Es referente con el módulo de Catalogación, que incluye las funciones relativas a los procesos técnicos relacionados con el desarrollo y mantenimiento de los catálogos bibliográficos.
4	Gestión de Autoridades	Es relativo a la Gestión de Autoridades, las cuales facilitan la utilización de una forma normalizada de los puntos de acceso de los catálogos automatizados, y muestra además las relaciones entre los distintos puntos de acceso.

Número de parámetro	Nombre del parámetro	Descripción
5	Módulo de Circulación	Se refiere al módulo de Circulación, el cual es el responsable de todos los procesos relativos al préstamo y devoluciones de materiales por parte de los usuarios.
6	Gestión de Publicaciones Periódicas	Se relaciona con la Gestión de Publicaciones Periódicas y el control de analíticas.
7	Catálogo Público en Línea (OPAC)	Es sobre si se tiene un Catálogo Público en línea, llamado también OPAC, para la consulta de información referencial por los usuarios desde sitios remotos a través de la Web.
8	Módulo Estadísticas	Es sobre el módulo de Estadísticas, donde se refleja el uso de los distintos servicios ofrecidos en la unidad de información, lo cual permite hacer estudios cuantitativos de usabilidad con miras a impulsar acciones para su mejoramiento.
9	Gestión de Tesauro	Es referente a la Gestión de Tesoros, los cuales permiten una normalización terminológica para mejorar el canal de acceso y comunicación entre los usuarios y la unidad de información.
10	Módulo de seguridad	Se refiere al módulo de Seguridad, que es el responsable de los procesos relativos con la protección y confiabilidad de la información que se administra.
11	Módulo de Inventario	Es el responsable de la gestión y mantenimiento del inventario de materiales de la unidad de información.
12	Servicio de Información y Referencia	Se refiere al Servicio de Información y Referencia, módulo que gestiona y administra las solicitudes de información que realiza el usuario y que requieren de apoyo bibliotecológico directo para su atención.
13	Servicio de Difusión Selectiva de la Información	Se relaciona con los servicios de Difusión Selectiva de la Información que se ofrecen a los usuarios para atender sus necesidades programadas de información.
14	Indización y Resumen	Se enfoca en el módulo de Indización y Resumen, el cual facilita el análisis documental de documentos digitales para facilitar su acceso.
15	Signaturización y Etiquetado	Se relaciona con la Signaturización y Etiquetado, para la generación de firmas y etiquetas de código de barras para identificar la colección de la unidad de información y facilitar los procesos de préstamo y devolución.
16	Variedad formatos visualización	Se relaciona con la Variedad de formatos que presenta el software para visualización de la información, y qué tan flexible es la aplicación para cambiar dichos formatos.

#### 4. Discusión

La matriz propuesta pretende facilitar al responsable de una unidad de información la selección del software apropiado para su proyecto. Con la implementación de los grupos de parámetros se tendrían cuatro filtros por los cuales se realizaría una selección cada vez más especializada, para determinar así realmente cuál es la aplicación que se requiere de acuerdo a los intereses y necesidad reales de la unidad de información.

El primer grupo permitirá hacer una selección inicial de un conjunto de aplicaciones que cumplan una serie de características generales antes de realizar una evaluación más profunda del software. Así por ejemplo, si se están buscando aplicaciones de software libre, que no

contemplan ningún costo para su adquisición ni para su mantenimiento, así como que la interface sea con un browser común, que esté documentada y presente un nivel de actualización periódico, con este primer grupo se puede hacer un descarte rápido de aquellos paquetes que no cumplan estas características. Este método de evaluación de software, utilizando como primer filtro el primer grupo de parámetros técnicos establecidos, tiene la finalidad de facilitar el descarte de programas que no cumplen con los requerimientos generales de la solución buscada.

Este segundo grupo representa un filtro que pretende determinar los estándares de gestión e intercambio de información. De esta forma es posible determinar el formato en el cual es estructurada la información almace-

nada en la aplicación, lo cual determina tanto la compatibilidad con otras aplicaciones así como la posibilidad de migración e intercambio de datos. Así por ejemplo, si lo que se desea es la implementación de un repositorio de información basado en Dublin Core, solamente se evaluarán las aplicaciones que utilicen dicho formato y se dejarán de lado las que no cumplan con éste parámetro.

El tercer grupo de parámetros se refiere a las características técnicas de cada una de las aplicaciones. Con ello se determinarán elementos relativos a la facilidad de uso de la aplicación, así como su escalabilidad, facilidades de auditoría, procedimientos de seguridad, facilidad de mantenimiento y soporte. Con esto es posible evaluar que tan amigable, intuitivo y robusto es el sistema. De esta forma sería posible seleccionar las aplicaciones que mejor se adapten a las expectativas técnicas y las capacidades de la unidad de información.

El cuarto grupo constituye el último filtro, orientado principalmente al tercer nivel de automatización y que permitiría descartar o considerar software de acuerdo a sus funciones. Así por ejemplo, si un proyecto de automatización pretende automatizar el área de adquisiciones, esto desearía de inmediato el OpenBiblio, el cual carece de este módulo.

Es importante indicar que el análisis de este grupo va ligado a la complejidad de la unidad de información, tal y como lo indica Fernández (2013, p.3)

En el desarrollo e implantación de proyectos de sistemas, existe una relación directa entre el tamaño de organización y la complejidad del software, siendo completamente válido afirmar que una organización pequeña generalmente es soportada por software sencillo, mientras que una organización grande es soportada por software más complejo. Relación que es válida con totalidad, ya que un software complejo, por lo general, requiere de mayor cantidad de recursos técnicos y humanos, que ofrezca mayor soporte técnico y mayor esfuerzo operativo para cumplir, en forma satisfactoria, con una funcionalidad más amplia.

Con este grupo de parámetros es posible entonces determinar tanto los módulos con que cuenta la aplicación como su funcionalidad. De esta forma se podrían seleccionar los sistemas que cuenten con los módulos que realmente serán implementados o que cumplan con

los requerimientos previamente definidos, de acuerdo a la complejidad de la unidad de información.

Como elemento adicional a la matriz de evaluación es importante tener claro que es necesario evaluar la misma unidad de información para determinar el nivel de automatización que realmente persigue, y, si se encuentra en el tercer nivel, el subnivel adecuado para su proyecto de automatización (Fernández, 2013). De esta manera se podrá seleccionar el tipo de software que requiere, adecuado a sus capacidades técnicas y financieras.

## 5. Conclusiones

La matriz de evaluación propuesta es un instrumento técnico que facilitará la evaluación del software libre. La lista de parámetros que se propone para integrar esta matriz podría sufrir modificaciones requeridas para un determinado proyecto de automatización. Sin embargo, esta lista de parámetros se convierte en una herramienta básica con los aspectos técnicos importantes a considerar en la evaluación de software.

La matriz de evaluación presentada en este artículo es una propuesta técnica resultado de un proyecto de investigación vigente en la Universidad de Costa Rica. Esta propuesta no pretende ser un patrón fijo de evaluación, sino que puede ser modificada, a través de la agregación de nuevos parámetros o eliminación de algunos de los propuestos, dependiendo de las necesidades de automatización de los evaluadores, tales como fecha de evaluación del software, persona que evalúa el software y cualquier otro parámetro que se requiera. Además, por ser una propuesta técnica, la implementación de la misma podrá realizarse sobre cualquier programa que el evaluador decida, tal como una hoja de cálculo, una base de datos o el desarrollo de una aplicación.

Cada una de las cuatro categorías de la lista de parámetros, que integran esta matriz de evaluación, se convierte en un filtro para considerar o desestimar una determinada aplicación. Así por ejemplo, si se requiere una aplicación que tenga una alta portabilidad a través del manejo de protocolos estándar para la importación y exportación de información, la categoría 2 de la matriz podría ser aplicada a un grupo de aplicaciones candidatas, para seleccionar aquellas que cumplen las características técnicas requeridas para este fin. Así sucesiva-

mente se podrían ir aplicando las siguientes categorías de esta matriz como filtros de selección. Esta forma de selección de software permitirá tener un mecanismo más eficiente, y así disminuir el desgaste de estar considerando una gran cantidad de aplicaciones que no cumplen con determinadas características técnicas requeridas en un determinado proceso de automatización de unidades de información.

Los parámetros propuestos son útiles para la selección de software libre para la automatización de unidades de información. Estos parámetros constituyen un punto de partida para los responsables de los proyectos de automatización, quienes deben decidir el software libre a utilizar, a partir de la amplia gama de paquetes existentes. Además, con la propuesta de estos parámetros, en cuatro categorías, los responsables del proyecto de automatización tienen que dedicar menos tiempo para decidir qué variables considerar en la selección de software. Así, con esta matriz de evaluación, se tiene una guía técnica para facilitar la selección de software para un proceso de automatización en una unidad de información, a través de la consideración o desestimación de un programa a partir de un grupo de aplicaciones candidatas y la evaluación de los parámetros propuestos. Pero, esto no implica de ninguna forma eliminar los procesos técnicos relativos a la prueba del software, las cuales deben realizarse necesariamente en cualquier organización para designar en definitiva las aplicaciones que se utilizarán en el proceso de automatización. El proceso de pruebas se puede utilizar también, en caso de que varios de los programas candidatos superen cada uno de los filtros de selección de cada una de las cuatro categorías propuestas en la matriz.

## 6. Referencias

1. ALVITE DÍEZ, M. L. (2009). Interfaces y funcionalidades de bibliotecas digitales. *Anales de documentación*. (12), 7-23.
2. BERTOT, J. C. (2004) Assessing Digital Library Services: Approaches, Issues, and Considerations. *International*

*Symposium on Digital Libraries and Knowledge Communities in Networked Information Society, DLKC'04*. Disponible en: <http://www.kc.tsukuba.ac.jp/dllkc/e-proceedings/papers/dllkc04pp72.pdf>

3. CHINCHILLA-ARLEY, R. (2011). El software libre: una opción para la automatización de unidades de Información. *Bibliotecas*. 29 (1). Disponible en: <http://www.revistas.una.ac.cr/index.php/bibliotecas/article/view/1557>
4. CÓRDOBA, S. (2011). Los repositorios institucionales y de acceso abierto en Costa Rica. *E-Colabora: Revista de ciencia, educación, innovación y cultura apoyadas por redes de tecnología avanzada*. 1(2). Disponible en: <http://publicaciones.renata.edu.co/index.php/RCEC/article/view/45>
5. DAMSGARD, J. & KARLSBJERG, J. (2010). Seven principles for selecting software packages. *Communications of the ACM CACM Homepage archive*, Volume 53 Issue 8, August 2010, 63-71.
6. FERNÁNDEZ MORALES, M. (2013). Clasificación del software libre orientado a la automatización integral de bibliotecas según el nivel de complejidad de la biblioteca: bibliotecas simples, bibliotecas de mediana complejidad y bibliotecas de alta complejidad. *E-Ciencias de la Información*, 3(1). Disponible en: <http://revistaebci.ucr.ac.cr/volumenes/3/3-1/3-1-1/3-1-1.html>
7. MOREIRO, J. A., SÁNCHEZ-CUADRADO, S., PALACIOS, V., BARRA, E. (2011). Evaluación de software libre para la gestión de archivos administrativo. *El profesional de la información*. 20(2).
8. SIFUENTES ARROYO, A. (2005). *Criterios de evaluación de sistemas integrados para bibliotecas durante la fase de selección en un proyecto de automatización*. II Jornadas Nacionales de Bibliotecas Universitarias. 26 y 27 de agosto del 2005 - Trujillo, Perú
9. VILELA DEL ÁGUILA, S. (2005). *Proceso de evaluación de software bibliográfico para la Biblioteca de la Universidad de Piura*. II Jornadas Nacionales de Bibliotecas Universitarias. 26 y 27 de agosto del 2005 - Trujillo, Perú.

# ANEXO

## MATRIZ DE EVALUACION

Figura 1

GENERALIDADES				
1	Nombre de la aplicación			
2	Versión			
3	País de Origen			
4	Idiomas que maneja			
5	Creador			
6	Licencia			
7	Página de Web			
8	Página de Descarga			
9	Documentación			
	S: Sí			
	N: No			
10	Interfaz			
	1. Browser			
	2. Aplicación propia			
11	Plataforma			
	1. Sistema de archivos			
	2. Cliente/Servidor			
	3. Web			
12	Lenguaje(s) de programación			
13	Modificación del código fuente			
	S: Sí			
	N: No			
14	Mantenimiento anual			
15	Actualización periódica			
	S: Sí			
	N: No			
16	Motor de bases de datos			
17	Soporte multibiblioteca			
	S: Sí			
	N: No			
18	Sistema Operativo			
	<b>GESTION DE ESTÁNDARES</b>			
1	ISO 2709			
	S: Sí			
	N: No			

[Automatización de unidades de información: Matriz técnica para la evaluación de software libre]

2	Protocolo Z39.50			
	S: Sí			
	N: No			
3	Exportación TXT rotulado			
	S: Sí			
	N: No			
4	MARC21			
	S: Sí			
	N: No			
5	MARCXML			
	S: Sí			
	N: No			
6	SRU, XML			
	S: Sí			
	N: No			
7	Dublin Core			
	S: Sí			
	N: No			
<b>ASPECTOS TECNICOS VARIOS</b>				
1	Facilidad de auditoría.			
	S: Sí			
	N: No			
2	Amigabilidad			
	S: Sí			
	N: No			
3	Facilidad de desarrollo de nuevos requerimientos.			
	S: Sí			
	N: No			
4	Independencia del hardware.			
	S: Sí			
	N: No			
5	Seguridad lógica.			
(La seguridad lógica se calificará con un rango de 1 a 5, donde 1 es poco seguro y 5 es muy seguro).				
6	Autodocumentado.			
	S: Sí			
	N: No			
7	Simplicidad.			
(La simplicidad de uso se calificará con un rango de 1 a 5, donde 1 es poco simple y 5 es muy simple).				
8	Independencia del sistema software.			
	S: Sí			
	N: No			

9	Disponibilidad			
(La disponibilidad del software se calificará con un rango de 1 a 5, donde 1 es poco disponible y 5 es muy disponible).				
10	Mantenibilidad			
(La mantenibilidad del software se calificará con un rango de 1 a 5, donde 1 es poco mantenible y 5 es muy mantenible).				
11	Soporte en línea			
	S: Sí			
	N: No			
12	Foros			
	S: Sí			
	N: No			
13	Comunidades			
	S: Sí			
	N: No			
	FUNCIONALIDAD			
(La completitud en la funcionalidad de cada módulo se calificará con una nota de 0 a 5, donde 0 significa que no tiene el módulo, mientras que el rango de 1 a 5, 1 significa poco completo y 5 significa que es muy completo).				
1	Módulo Administrador			
2	Selección y Adquisiciones			
3	Módulo Catalogación			
4	Gestión de Autoridades			
5	Módulo de Circulación			
6	Gestión de Publicaciones Periódicas			
7	Catálogo Público en Línea			
8	Módulo Estadísticas			
9	Gestión de Tesauro			
10	Módulo de seguridad			
11	Módulo de Inventario			
12	Servicio de Información y Referencia			
13	Servicio de Difusión Selectiva de la Información			
14	Indización y Resumen			
15	Signaturización y Etiquetado			
16	Variedad formatos visualización			