



Universidad de Costa Rica
Escuela de Bibliotecología y Ciencias de la
Información

Software libre para la automatización de unidades de información

Ricardo Chinchilla Arley

Mynor Fernández Morales

Octubre, 2013



Temas a desarrollar

- ✦ El entorno y el problema
- ✦ Metodología
- ✦ Resultados preliminares
- ✦ Criterios para la definición de subniveles
- ✦ Matriz de evaluación de software
- ✦ Conclusiones



El entorno y el problema



✦ Realidad de nuestras unidades de información

- ✦ Bajo presupuesto

- ✦ Gran cantidad de software para automatización de procesos y servicios

✦ Se requiere evaluar las alternativas para determinar la mejor opción

✦ Vacíos en cuanto a la comparación y evaluación de software

✦ Proyecto de investigación *“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”*

Metodología



✦ Definición de niveles de automatización

- ◆ Automatización de catálogos
- ◆ Gestión de repositorios y bibliotecas virtuales
- ◆ Automatización integral
 - Pequeñas (centros de documentación especializados)
 - Medianas (biblioteca escolares)
 - Grandes (bibliotecas universitarias, sistemas de bibliotecas)

✦ Creación de un catálogo de software por nivel

✦ Diseño de una matriz de evaluación

✦ Instalación de las aplicaciones

Resultados preliminares

✦ Automatización de catálogos

- ✦ Orientados a la descripción bibliográfica
- ✦ Pequeños y versátiles
- ✦ No relacionales

✦ Catálogo

- ✦ WinIsis
- ✦ IsisMarc
- ✦ Clabel
- ✦ Catalis
- ✦ Malete (OpenIsis)

CATÁLOGO



BIBLIOTECA

Resultados preliminares

✦ Repositorios y bibliotecas virtuales

- ✦ Almacenamiento y recuperación de documentos digitales
- ✦ Servicios asociados

✦ Catálogo

- ✦ DSpace
- ✦ EPrints
- ✦ Greenstone
- ✦ iAH
- ✦ Potnia
- ✦ Fedora Commons
- ✦ VuFind
- ✦ Google Custom Search



Resultados preliminares

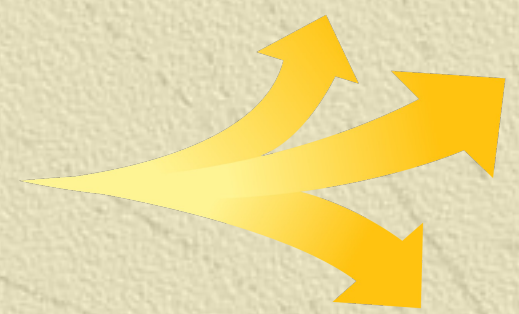
✦ Sistemas integrados de automatización

- ◆ Dirigidos a la gestión de todas las áreas de una U.I. (adquisiciones, catalogación, circulación, publicaciones periódicas, consultas, inventario y estadísticas)
- ◆ Sistemas complejos, modulares y utilizan bases de datos relacionales
- ◆ Basados en estándares internacionales (MARC, Dublin Core)
- ◆ Requieren programas adicionales (servidores web, motores de bases de datos, módulos Java, etc.)

✦ Se dividen en tres subniveles

- ◆ Pequeñas
- ◆ Medianas
- ◆ Grandes

Subniveles - criterios



✦ Definición de las **Unidades de información** en tamaño

- ✦ Número y tipo del público objetivo
- ✦ Tamaño y tipo de las colecciones
- ✦ Cantidad de personal y clasificación según funciones
- ✦ Volumen de trabajo
- ✦ Presupuesto
- ✦ Infraestructura tecnológica (servidor, red, etc.)
- ✦ Soporte técnico para la infraestructura tecnológica

Subniveles - criterios



✦ Definición del **software** en capacidad

- ✦ Nivel de robustez del motor de base de datos que soporta la aplicación
- ✦ Nivel de parametrización
- ✦ Plataforma donde se encuentra desarrollado
- ✦ Soporte técnico brindado
- ✦ Nivel de facilidad para el desarrollo de nuevos requerimientos
- ✦ Interoperabilidad del software
- ✦ Manejo de estándares internacionales para importar y exportar información

Resultados preliminares

✦ Sistemas integrados de automatización

◆ Modelo de clasificación de software

Criterio de clasificación	Nivel		
Robustez del motor de Base de datos	Bajo	Medio	Alto
Plataforma de hardware y software	Archivos	Cliente/servidor	Web
Nivel de parametrización	Bajo	Medio	Alto
Facilidad de desarrollo de nuevos requerimientos	Bajo	Medio	Alto
Interoperabilidad del software	Bajo	Medio	Alto
Manejo de estándares internacionales	Bajo	Medio	Alto
Soporte técnico	Bajo	Medio	Alto

Resultados preliminares

✦ Catálogo de software

Software de primer subnivel:

OpenMarcoPolo

Biblio

CaMPI

BibliotecaPHP

WebLIS

PHL

GCBibliotecas

Aguapey

Software de segundo subnivel:

OpenBiblio

EspaBiblio

Emilda

GNUteca

Infocid

Biblioteca 2000

PhPMyLibrary

ABCD

Software de tercer subnivel

Koha / Kobli

Merán

PMB

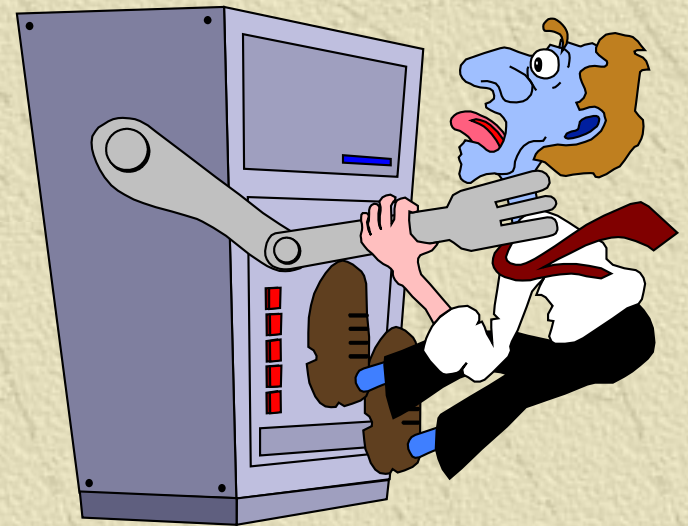


Matriz de evaluación

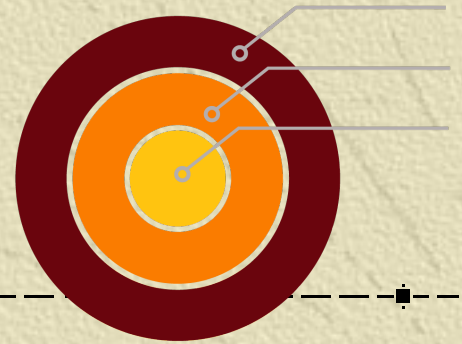
✦ Una vez definido el nivel de automatización, se evalúa el software

✦ Matriz de evaluación – secciones:

- ◆ Generalidades
- ◆ Gestión de estándares
- ◆ Aspectos técnicos
- ◆ Funcionalidad



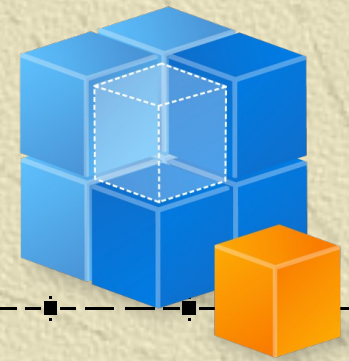
Matriz de evaluación - generalidades



✦ 20 parámetros que muestran las generalidades del software, como nombre de la aplicación, versión, licencia, idioma, actualización, creador, entre otros.

✦ *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.*

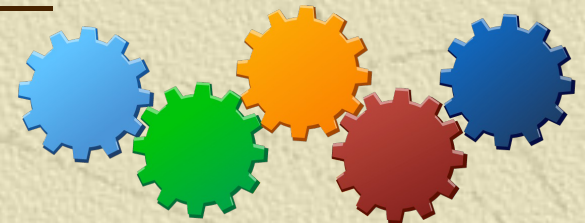
Matriz de evaluación – gestión de estándares



✦ 7 parámetros referentes a la gestión de estándares que permiten evaluar el grado de interoperabilidad a través de la importación y exportación de información.

✦ *Representa un filtro para determinar los estándares de gestión e intercambio. Al conocer el formato en el cual es estructurada la información se determina tanto la compatibilidad con otras aplicaciones así como la posibilidad de migración e intercambio de datos.*

Matriz de evaluación – aspectos técnicos



-
- ✦ 13 parámetros relacionados con aspectos técnicos varios
 - ✦ Determinan elementos relativos a la facilidad de uso de la aplicación, así como escalabilidad, auditoría, seguridad, mantenimiento y soporte.
 - ✦ Es posible evaluar que tan amigable, intuitivo y robusto es el sistema.
 - ✦ Es posible seleccionar las aplicaciones que mejor se adapten a las expectativas técnicas y las capacidades de la unidad de información

Matriz de evaluación – funcionalidad



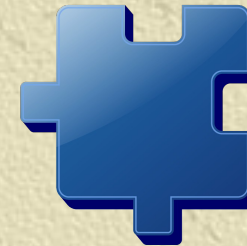
-
- ✦ 16 parámetros que evalúan la funcionalidad de la aplicación, como módulos disponibles y configuración de políticas generales que gobiernan el software (parametrización).
 - ✦ *Orientados principalmente al tercer nivel, permiten considerar o descartar software de acuerdo a sus funciones.*

Ejemplo



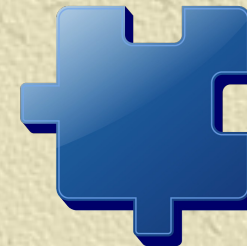
GENERALIDADES

1 Nombre de la aplicación	OpenBiblio
2 Versión	Release 0.7.1 – 18 mar. 2012
3 País de Origen	Estados Unidos
4 Idiomas que maneja	Inglés-Español
5 Creador	Dave Stevens
6 Licencia	GPL
7 Página de Web	http://obiblio.sourceforge.net/
8 Página de Descarga	http://sourceforge.net/projects/obiblio/files/OpenBiblio/0.7.1/
9 Documentación: Sistema Operativo:	Ubuntu versión 11.10, Windows Server, Memoria de 1.9 GiB, Procesador 0: intel ® Core (TM)2 dual CPU E6550 @ 2.33 GHz Procesador 1: intel ® Core™ 2 dual CPU E6550 @ 2.33 GHz Estado disponible en disco de 128.8 GiB
	S:SI S
	N:No
10 Interfaz	1
	1. Browser
	2. Aplicación propia



Ejemplo

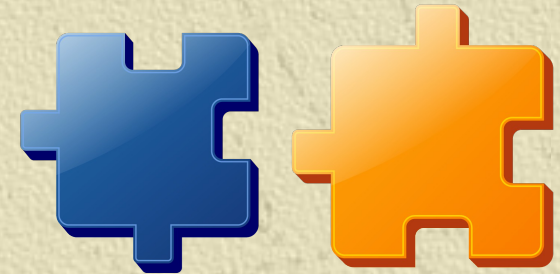
11 Plataforma	3
1.Sistema de archivos	
2.Cliente/Servidor	
3.Multicapa	
12 Lenguaje(s) de programación	PHP, CGI-PERL
13 Modificación del código fuente	S
S: SI	
N:No	
14 Costo del software	No tiene
15 Mantenimiento anual	No tiene
16 Actualización periódica	S
S: SI	
N:No	
17 Motor de bases de datos	MySQL
18 Robustez del motor de base de datos	2
(1. Poco robusto 2. Robustez media 3. Alta Robustez)	
19 Soporte multibiblioteca	N
S: SI	
N:No	
20 Sistema Operativo	Multiplataforma



Ejemplo

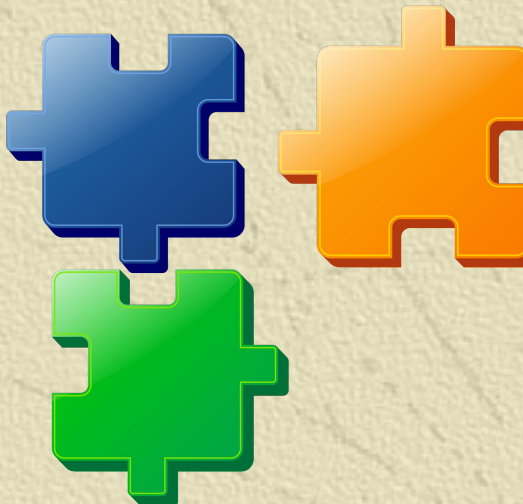
GESTIÓN DE ESTÁNDARES

1	ISO 2709	N
	S: SI	
	N:No	
2	Protocolo Z39.50	N
	S: SI	
	N:No	
3	Exportación TXT rotulado	N
	S: SI	
	N:No	
4	MARC21	S
	S: SI	
	N:No	
5	MARCXML	N
	S: SI	
	N:No	
6	SRU,XML	N
	S: SI	
	N:No	
7	Dublin Core	N
	S:SI	
	N:No	



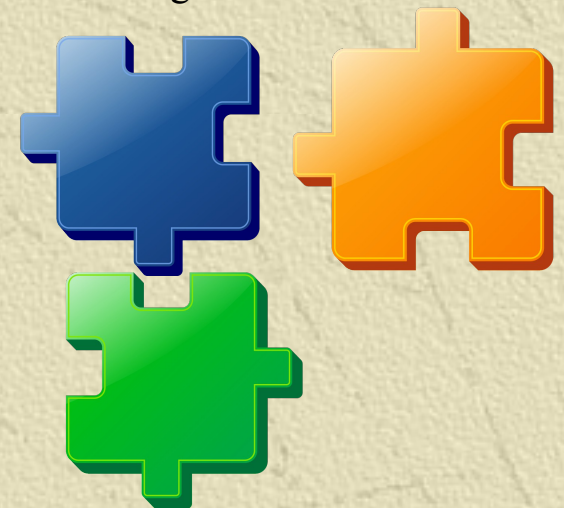
Ejemplo

ASPECTOS TÉCNICOS VARIOS

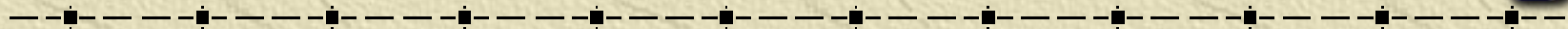
1	Facilidad de auditoría. S: SI N:No	S	
2	Amigabilidad S: SI N:No	S	
3	Facilidad de desarrollo de nuevos requerimientos. S: SI N:No	S	
4	Independencia del hardware. S: SI N:No	S	
5	Seguridad lógica. (La seguridad lógica se calificará con un rango de 1 a 5 donde 1 es poco seguro y 5 es muy seguro).	3	
6	Autodocumentado. S: SI N:No	S	

Ejemplo

7	<p>Simplicidad.</p> <p>(La simplicidad de uso se calificará con un rango de 1 a 5 donde 1 es poco simple y 5 es muy simple).</p>	5
8	<p>Independencia del sistema software.</p> <p>S: SI N:No</p>	S
9	<p>Disponibilidad</p> <p>(La disponiblidad del software se calificará con un rango de 1 a 5 donde 1 es poco disponible y 5 es muy disponible).</p>	5
10	<p>La facilidad de mantenimiento</p> <p>(La facilidad de mantenimiento del software se calificará con un rango de 1 a 5 donde 1 es poco mantenible y 5 es muy mantenible).</p>	3
11	<p>Soporte en linea</p> <p>S: SI N:No</p>	N
12	<p>Foros especializados</p> <p>S: SI N:No</p>	N
13	<p>Comunidades</p> <p>S: SI N:No</p>	N



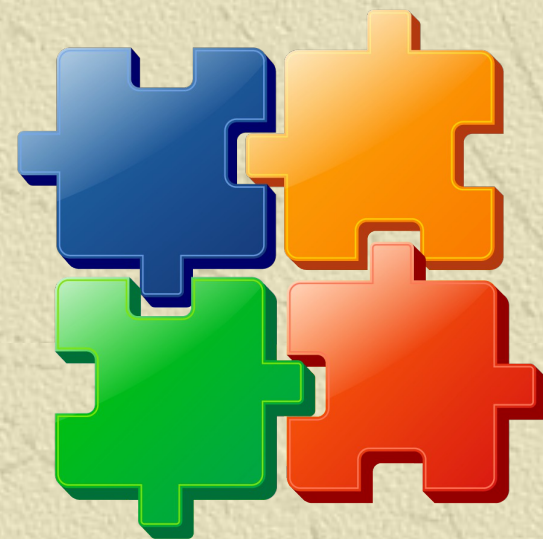
Ejemplo



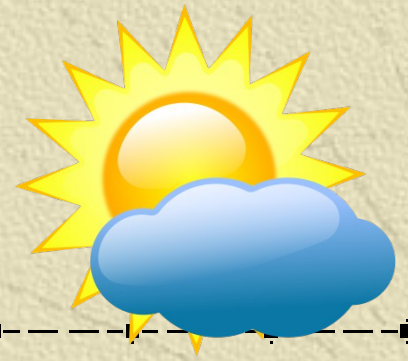
FUNCIONALIDAD

(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	4
2	Selección y Adquisiciones	0
3	Módulo Catalogación	4
4	Gestión de Autoridades	4
5	Módulo de Circulación	5
6	Gestión de Publicaciones Periódicas	0
7	Catálogo Público en Línea (OPAC)	5
8	Módulo Estadísticas	3
9	Gestión de Tesauro	0
10	Módulo de seguridad	4
11	Módulo de Inventario	0
12	Servicio de Información y Referencia	0
13	Servicio de Difusión Selectiva de la Información	0
14	Indización y Resumen	0
15	Signaturización y Etiquetado	4
16	Variedad formatos visualización	4

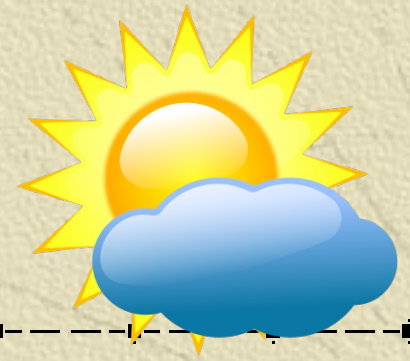


Conclusiones



- ✦ Los niveles son una herramienta para clasificar las aplicaciones y facilita el estudio y una evaluación de software comparable.
- ✦ El catálogo no pretende ser un listado único y exhaustivo del software.
- ✦ La subclasificación del tercer nivel es una recomendación técnica, no una limitación.
- ✦ La propuesta se basa en las bondades que ofrece el software libre (acceso al código fuente, posibilidad de modificación y adaptación) pero esto no implica que no se incurrirá en costos asociados.

Conclusiones



-
- ✦ Sin embargo es innegable el cuantioso ahorro obtenido respecto a la obtención de licencias, tanto en la aplicación como en los motores de base de datos, lenguajes de programación, software de servidores y sistemas operativos.
 - ✦ La matriz de evaluación es un instrumento técnico de evaluación. La lista de parámetros puede ser modificada para un determinado proyecto de automatización.
 - ✦ La matriz es una herramienta útil, pero su aplicación debe ser complementaria a los respectivos estudios de factibilidad técnica, económica y operacional relativos a cualquier proyecto de automatización.

Muchas gracias

