

OpenProdoc

Gestor Documental Open Source

Joaquín Hierro



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Introducción

- Esta presentación describe los orígenes, arquitectura, funcionamiento y evolución prevista de OpenProdoc.
- OpenProdoc es una herramienta de ECM, un gestor documental desarrollado bajo licencia de Código Abierto (Open Source).
- Incluye adicionalmente funciones para gestionar tesauros multilingües e integrarlos con los documentos.
- Como característica particular, dispone de una versión portable multiplataforma con las mismas funciones que la versión estándar, que un usuario aislado puede utilizar.
- Un gestor documental es una herramienta muy sofisticada, por lo que esta presentación es solo una breve introducción a algunos aspectos de OpenProdoc.



Introducción (Unificando lenguaje I)

- Qué no implica “Código Abierto”:
 - Productos “amateur”.
 - Producto de peor o mejor calidad (Ej. Linux, MySQL,)
 - Gratuidad (Ej. versiones “Community” y “Enterprise”)
 - Falta de una organización detrás (Apache, Oracle, IBM).
 - Peor soporte (Autores e Internet frente a centros de soporte de pago crecientemente reducidos).
- Qué sí implica:
 - Disponibilidad aunque cierre la empresa o sea comprada por otra.
 - Posibilidad de modificar de acuerdo a nuestras necesidades
 - Seguridad de que no hay nada “oculto”.
 - Posibilidad de aprender.
 - Posibilidad de detectar y corregir errores.



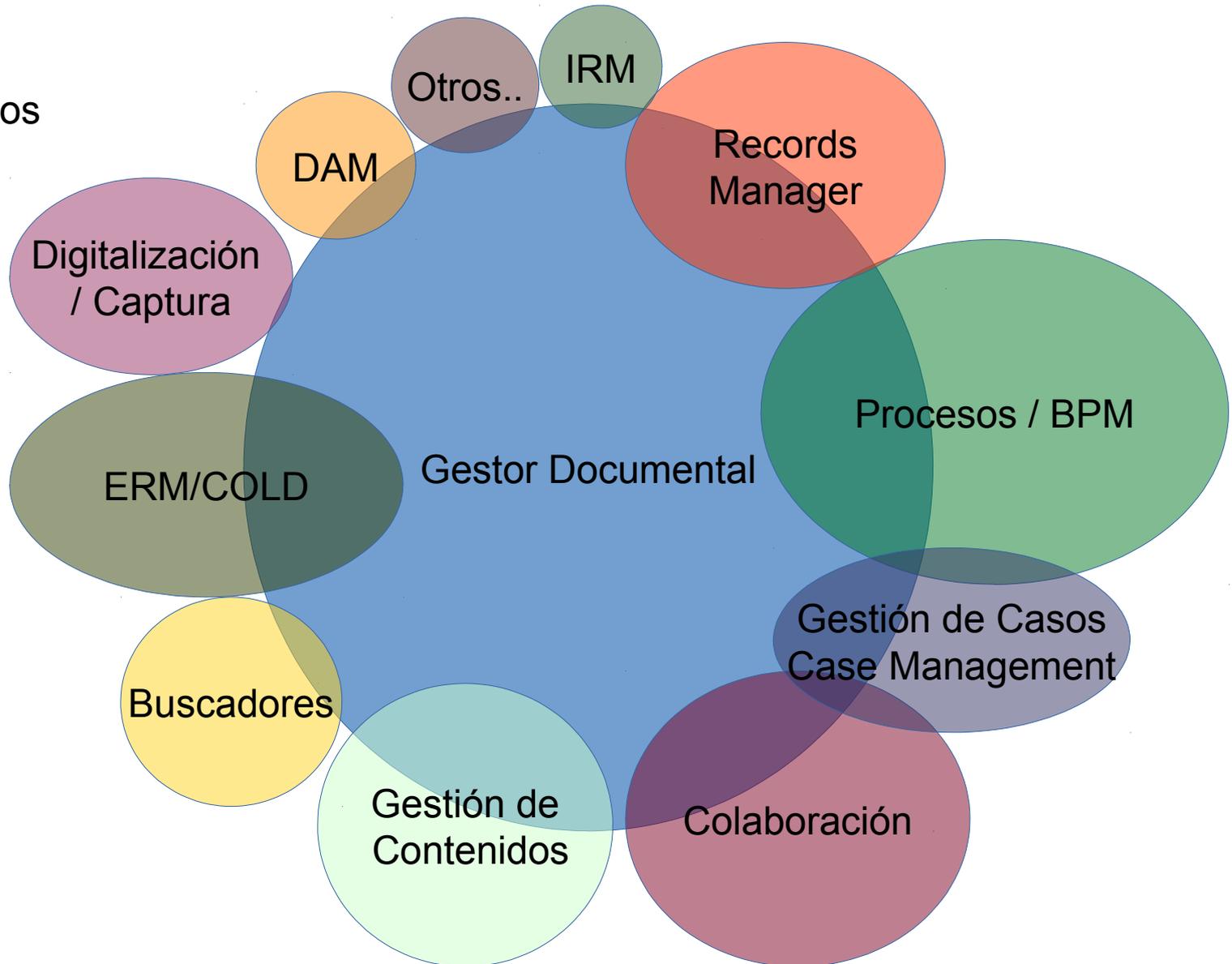
Introducción (Unificando lenguaje II)

- Según AIIM (y Wikipedia):
 - *Enterprise Content Management (ECM) son las estrategias, métodos y herramientas utilizadas para capturar, gestionar, almacenar, preservar y entregar contenido y documentos relacionados con los procesos organizativos. Herramientas y estrategias de ECM permiten la gestión de la información no estructurada de una organización, donde quiera que exista esa información.*
- ECM cubriría herramientas y tecnologías como:
 - Gestores documentales
 - Gestión de contenidos
 - Herramientas de Colaboración (Wiki, blog, mensajería)
 - Gestión de Registros (Record Management)
 - Captura y digitalización
 - Gestión de Procesos (WorkFlow / BPM)
 - Búsqueda
 - DAM (Digital Assets Management)



Introducción (Unificando lenguaje III)

Familia
Productos
ECM





Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Construcción (Orígenes)

- A partir de la experiencia utilizando y administrando los principales productos del mercado, así como realizando procesos de análisis y selección de soft. documental, me planteé crear un gestor documental con las características que considero más interesantes de ellos.
- Hace unos años no había apenas gestores Open Source ni tenían demasiada calidad.
- El crear un gestor permite aprender mucho sobre el funcionamiento interno y los posibles problemas (de diseño, portabilidad, rendimiento,..), lo que permite analizar y utilizar otros productos con mucha mayor base.
- Por ultimo, como reto personal, ya que crear un producto de este tipo con la calidad, rendimiento y funciones adecuadas es un proyecto muy complejo.



Construcción (Diseño)

- Para su desarrollo era condición de partida utilizar un lenguaje orientado a objetos por la facilidad de evolución y mantenimiento que aporta. Se ha utilizado Java debido a las ventajas de portabilidad y estandarización de la arquitectura J2EE.
- Otro requisito importante era la independencia de plataforma, de forma que pudiera utilizarse con cualquier base de datos, servidor de aplicaciones y sistema operativo, para ello se ha cuidado el desarrollo y probado en diversos entornos.
- El modelado interno de los metadatos en BBDD ha sido uno de los puntos en los que he empleado más tiempo para asegurar flexibilidad y rendimiento.
- OpenProd se basa en conectores para acceder a otros sistemas, de forma que pueda expandirse.



Construcción (Desarrollo y pruebas)

- El desarrollo se ha realizado en su integridad utilizando el entorno de desarrollo java [Netbeans](#), con la ayuda de herramientas SQL como [Squirrel SQL](#).
- Para la depuración ha sido imprescindible la ayuda de [Firebug](#) y el uso de una herramienta de virtualización como [VirtualBox](#), para simular múltiples equipos y combinaciones de software.
- El servidor embebido de base de datos elegido para la versión portable ha sido [HSQLDB](#).
- Respecto a las pruebas, se han realizado de 3 tipos:
 - Pruebas de desarrollo para verificar la corrección de cada una de las funciones que se añadían.
 - Pruebas completas instalando el programa sobre una máquina limpia para asegurar la compatibilidad y el funcionamiento completo.
 - [Pruebas de carga](#) para estudiar el rendimiento.



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Gestión de Documentos (I)

- Openprod doc incluye las funciones habituales de un gestor documental:
 - Definición de tipos documentales y de contenedores (carpetas/expedientes).
 - Organización de la información en estructuras jerárquicas.
 - Almacenamiento de los metadatos y archivos en contenedores.
 - Gestión de versiones
 - Gestión de usuarios y grupos
 - Seguridad basada en ACL (Listas de Control de Acceso)
 - Búsquedas por los metadatos
 - Gestión del ciclo de vida
 - Posibilidades de integración con otros sistemas.
- A ello añade:
 - Gestión de múltiples tesauros multilingües
 - Gestión de referencias
 - Funciones (limitadas) de DAM y DSI.
 - Búsqueda por texto libre (en breve)



Gestión de Documentos (II)

- Como la mayor parte de los gestores instalados en la actualidad, está pensado para que pueda utilizarse tanto de forma aislada como preferentemente integrado con la gestión de documentos y procesos de la institución en que se encuentra.
- En este escenario el objetivo suele ser almacenar la información internamente a la institución y optimizar el proceso, sin intercambiar información con otras entidades.
- Esto implica mucha flexibilidad para crear estructuras de información muy variadas y generalmente no normalizada (de acuerdo a normas externas).
- En este escenario, las posibilidades del modelado y de la integración son críticos.

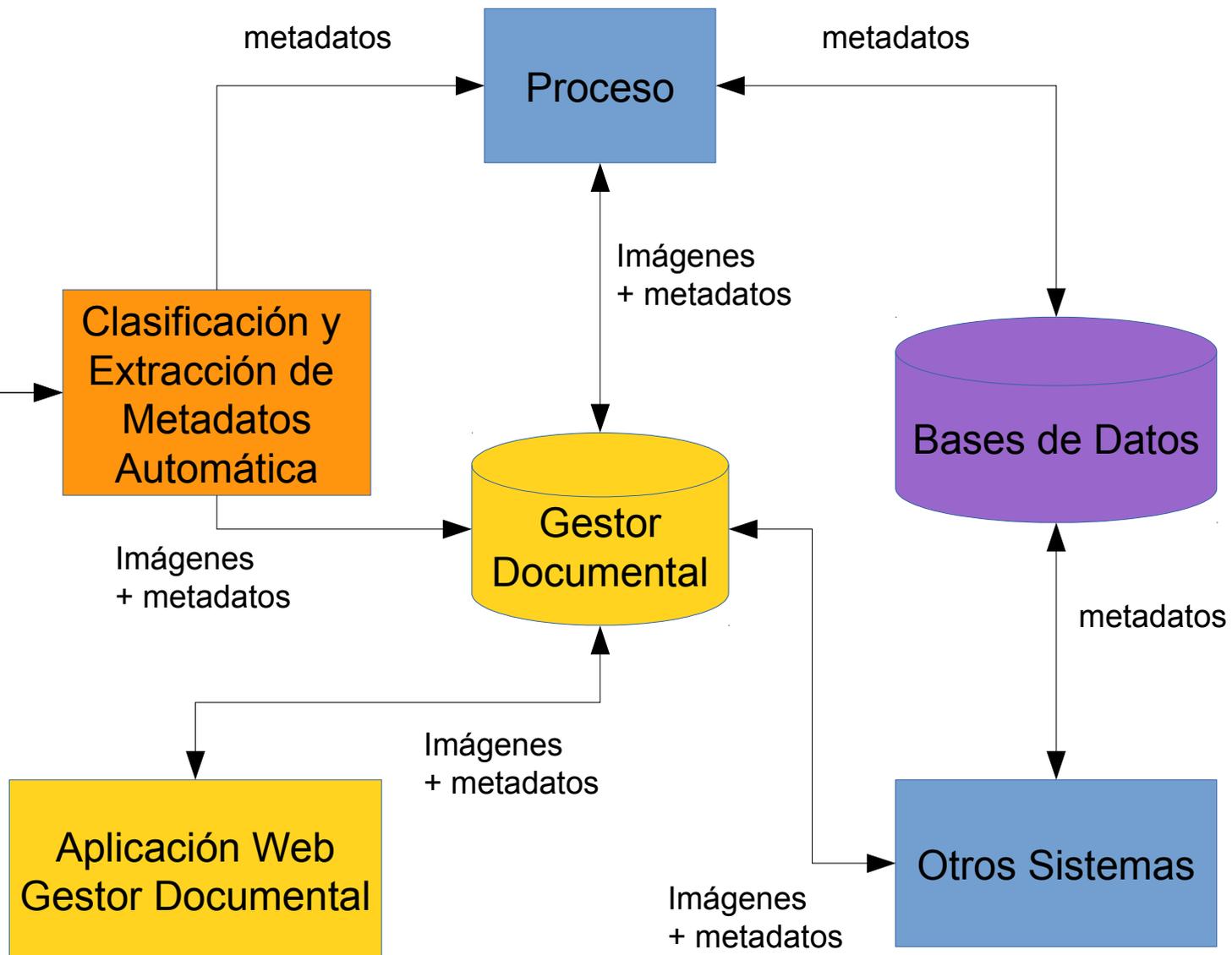
Gestión de Documentos (III)

Escenario
Integración
Gestor
Documental



Documentos
(correos,
anexos,
mensajes).

Imágenes
digitalizadas





Gestión de Documentos (IV)

- Para escenarios de este tipo, OpenProdDoc permite la definición de tipos documentales y contenedores orientada a objetos.
- Cada tipo documental se define como subtipo de uno ya existente. Por defecto existe un tipo básico de carpeta y un tipo básico de documento con un mínimo de metadatos. Todos los tipos documentales creados serán subtipos del tipo básico o de otro subtipo.
- Los metadatos y comportamiento se heredan; es decir un tipo documental tiene la suma de todos los atributos de cada uno de sus antecesores.
- A cada tipo documental puede añadirse metadatos, indicando el nombre y tipo (cadena, fecha, entero, booleano, controlado por un tesauro) , si es obligatorio, si es multivaluado y si es un valor único.



Gestión de Documentos (y V)

- La orientación a objetos tiene varias ventajas:
 - Permite una evolución gradual del modelo. Es muy complejo definir una estructura de tipos documentales para una organización grande con diversos departamentos. Esperar a tener un cuadro de clasificación completo retrasaría el arranque mucho tiempo. Adicionalmente es habitual una evolución del modelo por causas organizativas o legales.
 - Es más potente, ya que puede buscarse por un tipo documental o por un tipo y sus subtipos (Ej. Buscar por unos criterios en “Informes” o en “Informes” y todos sus subtipos). Esto aplica a también a otras operaciones definidas.
 - Es más sencilla y evita tener que definir una “estructura plana”, se realiza un análisis gradual que genera una estructura jerárquica. Los metadatos o propiedades definidas para un tipo se heredan para los subtipos.
 - Es una representación más “real”, ya que las definiciones y tipologías documentales no son totalmente independientes.



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Gestión de Tesauros (I)

- Puede crearse múltiples tesauros multilingües estructurados de acuerdo a las necesidades.
- Cada “tesauro” puede utilizarse como un tesauro completo (NT, BT, RT, UF) o simplemente como una lista de materias o una lista de términos, que puede ser jerárquica.
- Los metadatos de los tipos documentales pueden asociarse a tesauros concretos, de forma que se elija el valor del metadato dentro de los disponibles en el tesauro y se controle la integridad. Por ejemplo puede así definirse un metadato “País” que se valide contra una lista de países o un metadato “Tema del informe” que se valide contra un tesauro de materias.



Gestión de Tesauros (y II)

- Para importar y exportar se utiliza el estándar SKOS-RDF.
- OpenProdoc no necesita crear microtesauros. Las búsquedas pueden acotarse a partir de un término a todos los términos específicos del mismo, restringiendo la búsqueda a una rama del árbol.
- OPD no maneja tesauros poli-jerárquicos, de forma que la importación de un tesauro que incluya relaciones de ese tipo generará un aviso y guardará una sola relación.



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Seguridad (I) Perfiles.

- La seguridad en OpenProd se cubre de varias formas.
- La primera y básica son los perfiles. Los perfiles indican los tipos de operaciones que puede realizar un usuario (por ejemplo, administrar usuarios, dar de alta documentos, administrar definiciones de documentos, administrar grupos, administrar los tesauros,..)
- Puede crearse perfiles con la combinaciones de permisos que se desee, sin necesidad de un único administrador “todopoderoso”.
- Puede crearse distintos tipos/perfiles de administrador, por ejemplo Administrador de Seguridad (encargado de usuarios y grupos), Administrador Documental (Encargado de definiciones de documentos y tesauros), Administrador tecnológico,...



Seguridad (II) ACLs

- En segundo lugar, y más importante está la seguridad aplicable a cada objeto (carpetas/expedientes o documentos).
- En OpenProd se utiliza seguridad basada en ACL (Listas de Control de Acceso) como es habitual en los principales gestores.
- Un ACL contiene un nombre y una lista de usuarios y grupos con los permisos asignados a ellos.
- Ej:
 - ACL: DocEconomicos
 - GrupoContabilidad: Lectura-Escritura-Borrado
 - GrupoDireccion: Lectura
- Si un grupo no aparece no tiene permisos, y no podrá acceder a los elementos con ese ACL.



Seguridad (III) ACLs

- ¿Porqué un modelo de ACLs y no un sistema más elemental?
- Por varios motivos.
 - Los sistemas basados en permisos de lectura-escritura o niveles de acceso (0-9) no reflejan la complejidad de un gestor ni una empresa. Un usuario puede poder escribir en un expediente y solo leer en otro dependiendo del área o tipo de expediente.
 - Las listas de usuarios asignados a cada elemento son inmanejables y si hay cambios obligan a cambiar miles de documentos o expedientes. Esos cambios, o deben hacerse manualmente o de forma jerárquica, lo que obliga a una estructura irreal.
 - Además el documento o expediente puede sufrir cambios durante su vida y tramitación.



Seguridad (y IV) ACLs

- Con los ACL, podemos cambiar la definición de seguridad y al momento millones de documentos y expedientes cambian sus permisos.
- Los ACL reflejan la seguridad del elemento y muestran el sentido mejor que una larga lista de nombres. Ej:
 - “DocumentosConfidenciales”,
 - “ExpedientesTramitados”,
 - “PendientesRevisionDireccion”
- Además puede cambiarse en los distintos estados de la vida. Ej
 - “Cuando un expediente se recibe su ACL será 'RegEntradaDepartamentoX' tras su revisión se asignará a los departamentos A o B con ACL 'ExpedientesA' o 'ExpedientesB' “



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Tareas (I)

- Para la gestión del ciclo de vida y la automatización, OpenProdDoc permite definir tareas.
- Las tareas pueden programarse en fechas y horas concretas (“Los sábados por la noche”, “todos los días a las 23h”, “Cada hora”,..) o bien cuando se produce un suceso (insertar, borrar o modificar un documento o carpeta/expediente) .
- Puede definirse a que estructura del árbol de carpetas afecta cada tarea (“Al fondo X”, “A documentos de la Dirección de RRHH”,..) y a que tipo de elementos (“Expedientes Médicos”, “Contratos”, “Informes”..).
- Puede definirse diferentes tareas del mismo tipo con distintos parámetros de acuerdo la estructura o el tipo de documento.



Tareas (II)

- Las operaciones posibles son:
- Programadas:
 - Borrado de carpetas/expedientes con una antigüedad.
 - Borrado de documentos (papelera) con una antigüedad.
 - Expurgo de documentos de la papelera.
 - Importación automática de documentos.
 - Exportación automática de documentos.
 - Informe automático de Carpetas/expedientes
 - Informe automático de documentos (para DSI o expurgo)
- Asociadas a eventos
 - Actualizar Metadatos Carpetas
 - Actualizar Metadatos Documentos
 - Copia de Carpetas
 - Copia de Documentos
 - Exportación Carpetas
 - Exportación Documentos
 - Conversión formato Documentos



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución

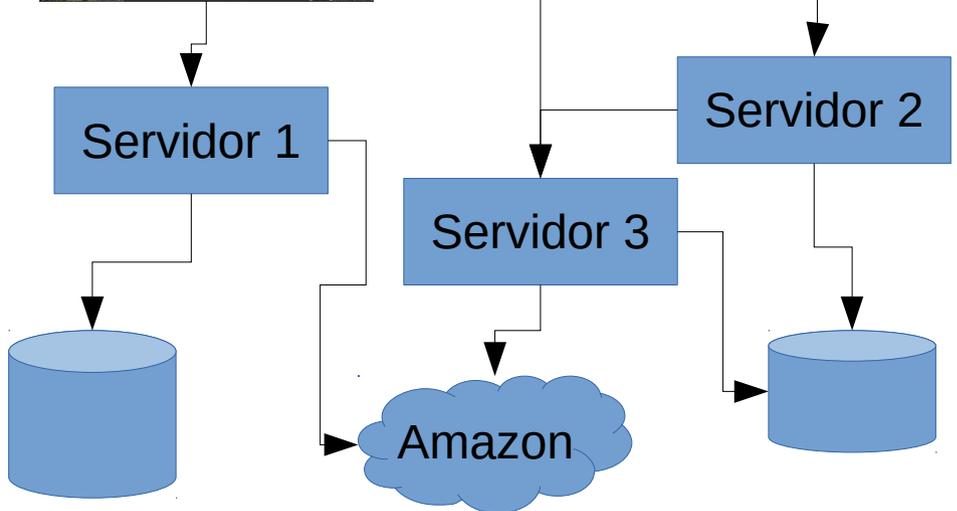
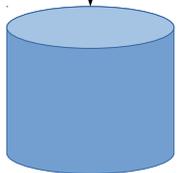


Arquitectura (I)

- Durante muchos años, las aplicaciones han sido monolíticas, con una estructura cerrada y con pocas posibilidades de interconexión.
- Gradualmente se ha evolucionado a modelos distribuidos donde múltiples sistemas se conectan entre sí, y donde las aplicaciones deben ser capaces de adaptarse y crecer.
- La forma en que se construye y estructura una aplicación condiciona sus posibilidades de admitir una carga mayor de usuarios y operaciones (escalabilidad) como su evolución e integración.
- Por eso la Arquitectura del Software tiene cada vez más peso y es importante conocerla. No basta que un programa realice bien sus funciones, tiene que estar internamente bien construido.

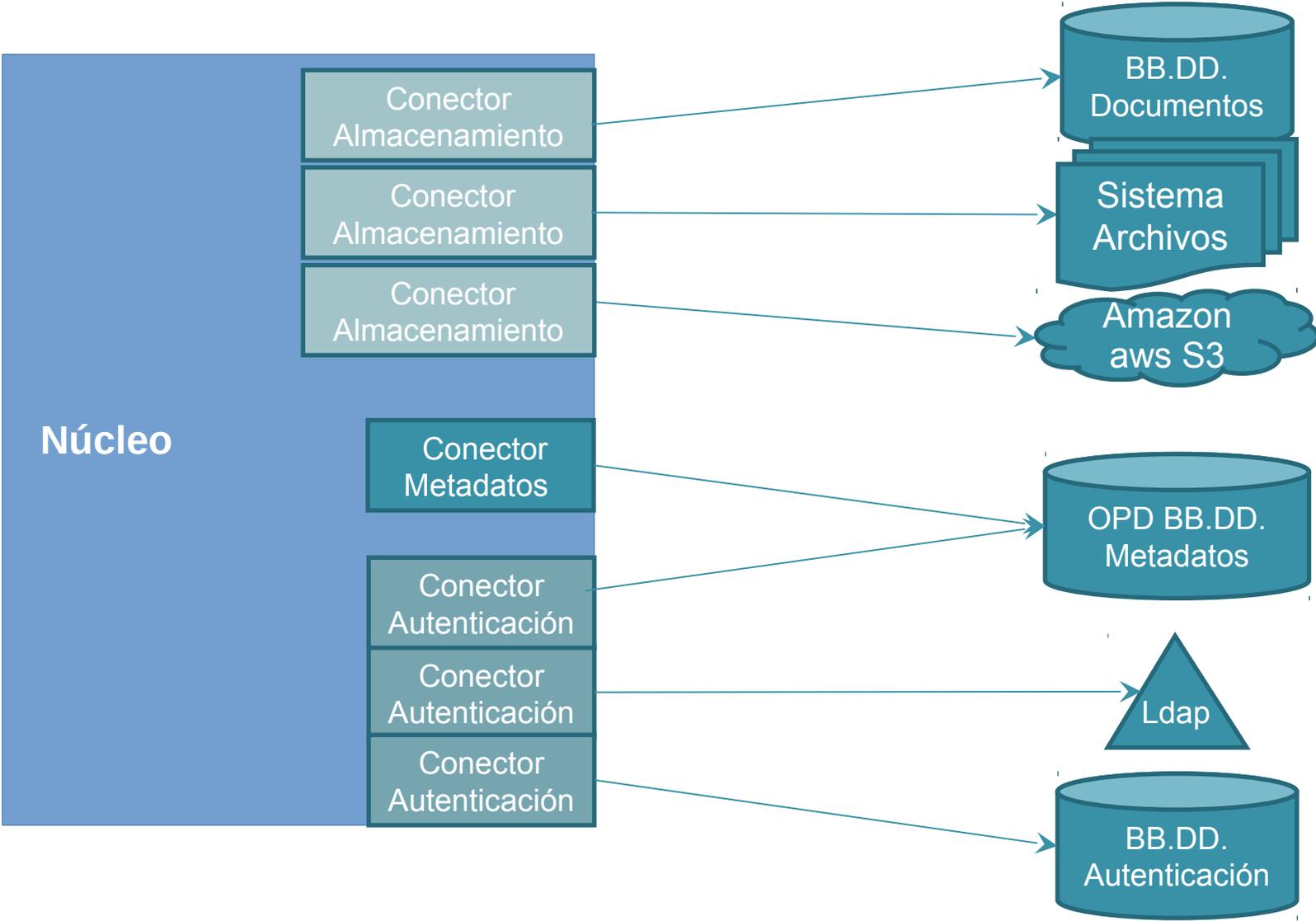


Arquitectura (II)

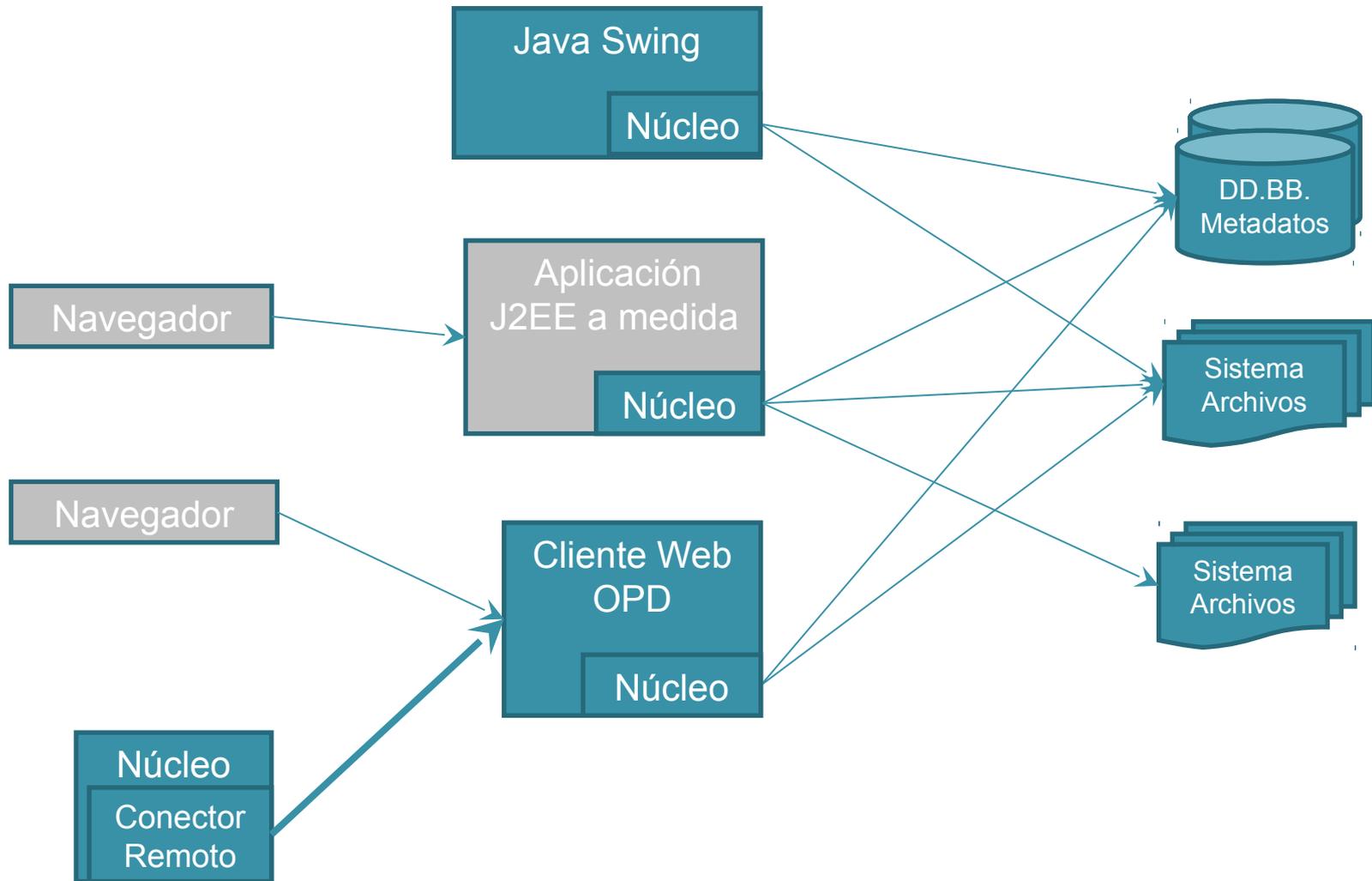




Arquitectura (III)



Arquitectura (IV)





Arquitectura (y V)

- Entre las distintas posibilidades de arquitectura, se ha incluido una versión portable multiplataforma, con todos los elementos y funciones embebidos en un solo punto.
- El objetivo es que pueda ser utilizado en escenarios como los siguientes:
 - Profesionales independientes que no disponen de una infraestructura corporativa con un gestor documental.
 - Como herramienta de formación, para que los alumnos puedan realizar ejercicios reales sobre un gestor documental.
 - Como herramienta de modelado de estructuras documentales, archivo o tesoro, antes de traspasarlas a un gestor documental corporativo.
 - Para poder llevar en modo desconectado parte de la información del gestor documental de la institución.



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Integración (I)

- Openprodoc tiene diversas formas de integrarse con otros sistemas:
 - Puede leer documentos, y sus metadatos, procesados con Kofax Capture o con Abby FlexyCapture.
 - Importar y exportar tesauros en formato SKOS
 - Importar y exportar documentos y metadatos a ficheros
 - Intercambiar definiciones y elementos de OpenProdoc con otras instalaciones.
 - Puede desarrollarse conectores (para almacenar documentos, autenticar o almacenar metadatos) para integrar con otros sistemas.
 - Puede desarrollarse nuevas tareas.
- La principal forma de integración, y la más completa, es por medio de Java, incluyendo el núcleo en cualquier aplicación.
- En un futuro, incluirá conexión por estándar CMIS.



Índice

Introducción

Construcción

Gestión de Documentos

Gestión de Tesoros

Seguridad

Tareas

Arquitectura

Integración

Evolución



Evolución

- La versión 1.2 (verano 2015) incluirá:
 - Generación de informes y archivos de exportación a medida.
 - Soporte de formato RIS para las referencias
 - Búsqueda por texto libre.
- La versión 1.3 (primavera 2016) incluirá:
 - Clasificación automática de los documentos en base a su contenido.
- Por otra parte, se puede contribuir al proyecto:
 - Creando tesauros
 - Creando paquetes (tipos, informes, tareas, ACL, grupos,..) sectoriales por ejemplo para abogados o colegios.
 - Definiendo nuevas hojas de estilo (CSS)
 - Creando informes.
 - Integrándolo con otros proyectos o aplicaciones.
- Las aportaciones se publicarían en la web de OpenProd



OpenProdoc

Joaquín Hierro

openprodoc@gmail.com

http://jhierrot.github.io/openprodoc/index_ES.html

<http://www.dokumentalistas.com/articulos/openprodoc-creando-un-gestor-documental-en-solitario/>

<http://www.biblogtecarios.es/author/joaquinhierro/>

