

Factores para aumentar la visibilidad en repositorios digitales: metadatos, interoperabilidad, identificadores persistentes y optimización SEO-GEO

Danilo Reyes-Lillo

Universitat Pompeu Fabra, España

<https://orcid.org/0000-0002-0141-8324>

Cristòfol Rovira

Universitat Pompeu Fabra, España

<https://orcid.org/0000-0002-6463-3216>

Alejandro Morales-Vargas

Universidad de Chile, Chile

<https://orcid.org/0000-0002-5681-8683>

Reyes-Lillo, D., Rovira, C., & Morales-Vargas, A. (2025). Factores para aumentar la visibilidad en repositorios digitales: metadatos, interoperabilidad, identificadores persistentes y optimización SEO-GEO. En J. Guallar, M. Vállez, & A. Ventura-Cisquella (Coords). *Comunicación digital. Tendencias y buenas prácticas* (pp. 126-141). Ediciones Profesionales de la Información. <https://doi.org/10.3145/cuvicom.09.esp>

Resumen

En el contexto de la ciencia abierta y la comunicación académica, mejorar la visibilidad de los repositorios digitales es esencial para maximizar el alcance, el impacto y la capacidad de descubrimiento del contenido que albergan. Este capítulo de libro explora cinco estrategias clave para mejorar la visibilidad de los repositorios: optimización de la calidad de los metadatos, habilitación de protocolos de interoperabilidad, adopción de identificadores persistentes, implementación de acciones de optimización para motores de búsqueda (SEO) y adopción de la optimización para motores generativos (GEO). A través de un análisis detallado y recomendaciones prácticas, este trabajo destaca cómo los metadatos estandarizados, los vocabularios controlados y los identificadores persistentes, como DOI, Handle y ARK, contribuyen a mejorar la visibilidad. Este capítulo también enfatiza la importancia de alinear los repositorios con las tecnologías web en evolución y los motores impulsados por inteligencia artificial para garantizar que el contenido siga siendo accesible, rastreable e integrado en la experiencia de búsqueda de los usuarios.

Palabras clave

Repositorios digitales; Visibilidad; Calidad de metadatos; Interoperabilidad; Identificadores persistentes; Optimización para Motores de Búsqueda; SEO; Optimización para Motores Generativos; GEO.

I. Introducción

En la era del acceso abierto y la ciencia abierta, los repositorios digitales juegan un papel fundamental en la difusión del conocimiento. Incrementar su visibilidad no solo amplía el alcance de los contenidos depositados, sino que también fortalece el impacto institucional y contribuye a la democratización del acceso a la información. Diversos estudios han demostrado que los trabajos disponibles en acceso abierto –especialmente a través de repositorios institucionales– tienden a recibir más citas y resultan más accesibles que aquellos restringidos por barreras de pago (Piwowar et al., 2018; Swan, 2010).

Además, una mayor visibilidad en los motores de búsqueda académicos como Google Scholar o bases de datos como OpenAIRE y CORE permite que los repositorios cumplan con los requisitos de mandatos de acceso abierto establecidos por financiadores nacionales e internacionales (UNESCO, 2021). Estrategias como la correcta implementación de metadatos normalizados, el uso de identificadores persistentes –como DOIs y ORCID– y la interoperabilidad con otros sistemas a través de protocolos como OAI-PMH (Open Archive Initiative-Protocol for Metadata Harvesting) son claves para lograr una mayor exposición (OpenAIRE, 2020).

Por lo tanto, trabajar activamente en la optimización de la visibilidad de los repositorios no es solo una cuestión técnica (Reyes-Lillo et al., 2025), sino una estrategia institucional para asegurar que la producción intelectual cumpla su objetivo final: ser consultada, utilizada y citada por comunidades académicas y la sociedad en general.

A continuación, analizamos cinco técnicas de optimización que pueden ser utilizadas para mejorar la visibilidad del contenido de un repositorio digital:

1. Optimización de calidad de metadatos.
2. Habilitación de los protocolos de interoperabilidad.
3. Adopción de identificadores persistentes.
4. Optimización SEO del repositorio.
5. *Generative Engine Optimization* en repositorios.

2. Optimización de calidad de metadatos

La normalización de metadatos es crucial para garantizar la interoperabilidad, la visibilidad, el acceso y la reutilización de los contenidos depositados. Una estrategia de metadatos debe considerar tanto los aspectos técnicos como los organizativos y normativos del repositorio.

Si bien el modelo de metadatos puede estar estrechamente ligado a la plataforma en la que se sostiene el repositorio, es fundamental que este esquema se adapte a las necesidades locales sin romper la compatibilidad con los estándares internacionales.

La tabla 1 muestra una comparativa entre algunos de los softwares de repositorios digitales y su esquema de metadatos de base:

Tabla 1
Softwares de repositorios digitales y su esquema de metadatos de base

Software	Esquema de metadatos de base	Información adicional
DSpace	Qualified Dublin Core https://github.com/DSpace/DSpace/blob/main/dspace/config/registries/dublin-core-types.xml	Usa un perfil propio de Dublin Core calificado; permite extensiones como METS/MODS. Desde DSpace 7, hay soporte para esquemas múltiples.
Fedora	No posee esquema fijo https://wiki.lyrasis.org/display/FEDORA6x/Data+Modeling	Usa modelos RDF/Linked Data; el esquema depende de la implementación (MODS, DC, PREMIS, etc.).
EPrints	Esquema de metadatos configurable y extensible https://wiki.eprints.org/w/Metadata#Metadata_Field_Types	Esquema basado en campos definidos por el administrador del repositorio. Puede interoperar con otros esquemas mediante exportaciones e importaciones.
InvenioRDM	JSON es una estructura alineada con DataCite https://inveniordm.docs.cern.ch/reference/metadata	El esquema se ajusta a DataCite's Metadata Schema v4.x con menores añadidos y modificaciones.
TIND	MARC21 https://www.tind.io	Utiliza como base MARC21 ya que está basado en INVENIO, desarrollado por el CERN. Sin embargo, tiene diversos módulos y se puede adaptar a otros formatos.
Digital Commons	No especifica esquema, pero puede ser mapeado a Dublin Core dimple o <i>qualified</i> https://digitalcommons.elsevier.com/en_US/organization-content-planning/metadata-options-in-digital-commons	Declara un esquema de metadatos flexible.
Dataverse	Utiliza diferentes metadatos compatibles con estándares https://guides.dataverse.org/en/latest/user/appendix.html	Garantiza la interoperabilidad y preservación de los metadatos mediante el uso de esquemas compatibles con estándares (como DDI, DataCite, Dublin Core, ISA-Tab y VOResource), lo que permite una fácil asignación y exportación en formatos estructurados.

Para iniciar una estrategia de optimización de metadatos, se recomienda iniciar con un diagnóstico inicial que incorpore al menos la evaluación de los siguientes elementos:

- Auditoría de metadatos existentes: el objetivo es revisar una muestra representativa de registros para detectar inconsistencias, errores de formato, campos vacíos o mal utilizados.
- Identificación de esquemas de metadatos utilizados: es esencial identificar el esquema de metadatos de base, pero también los diversos mapeos y posibilidades de exportación a otros esquemas.
- Revisión de vocabularios y autoridades: revisar y validar la implementación de vocabularios o listas controladas, como tesauros, autoridades e identificadores, así como listas de tipo de ítems o control de formato en los campos de lenguas y fechas.

Una vez realizado el diagnóstico, se recomiendan cuatro procesos fundamentales para optimizar los metadatos del repositorio, que se describen a continuación.

2.1. Limpieza y depuración de datos

Esta etapa resulta fundamental para optimizar la visibilidad de los contenidos almacenados en un repositorio. Los metadatos de alta calidad “should allow digital users to intuitively conduct the tasks such as identifying, describing, managing and searching data” (Ma et al., 2009, p. 1).

En ese sentido, la *data cleaning* (limpieza de datos) es un proceso de detección y corrección de errores, inconsistencias y campos incompletos orientado a mejorar la interoperabilidad y la experiencia del usuario (van Hooland y Verborgh, 2015; Westbrook et al., 2012). De esta forma, se fortalece la precisión y se mejora la recuperación de información a través de las herramientas de búsqueda de un sistema.

Para mejorar la calidad de los metadatos en repositorios, existen diversas herramientas, pero destaca ampliamente el uso de OpenRefine (<https://openrefine.org>). Esta es una herramienta de código abierto y gran potencia para limpiar, transformar y reconciliar datos tabulares desordenados. OpenRefine es especialmente útil para estandarizar y enriquecer metadatos en bibliotecas, archivos y conjuntos de datos de investigación.

Para utilizar OpenRefine en el proceso de *data cleaning* es fundamental: 1) exportar los metadatos en un formato compatible con OpenRefine, como CSV, TSV, Excel o JSON; y 2) que cada fila representa un recurso –por ejemplo, un documento o imagen–, y cada columna un elemento del esquema de metadatos. Si usáramos Dublin Core (DC), el ejemplo sería *dc:title*, *dc:creator*, *dc:subject*, entre otros.

De esta manera, es posible hacer una limpieza en lotes de registros que tengan inconsistencias, errores de tipeo o errores de formato. A continuación, se expone a modo de ejemplo una tabla de registros DC que potencialmente pueden ser mejorados con herramientas como OpenRefine (figura 1).

Figura 1

Ejemplo de registros DC con potencial de optimización a través de data cleaning.

dc:title	dc:creator	dc:subject	dc:date	dc:language
History of Science	Laura Gonzalez	Science; history	2020-06-10	eng
history of science	Laura gonzález	science; history	10/06/2020	en
Artificial Intelligence	María Gómez	Technology, AI; Machine Learning	2023	english
Big Data		Massive data; analytics	2021-03-14	en
Machine Learning	Carlos Ruiz	machine learning, artificial intelligence	2024-11-25	en
Cybersecurity	Ana Lopez	information security	2021/05/03	English

Cabe señalar que este proceso se puede efectuar sin importar el esquema de metadatos de base que tenga el sistema. Para sistematizar los metadatos en un formato aceptado por OpenRefine, es posible utilizar herramientas anexas como MarcEdit (<https://marcedit.reeset.net>), si el esquema de base es MARC21.

Una vez importados los registros, OpenRefine puede corregir inconsistencias como espacios innecesarios, mayúsculas desiguales, duplicados y formatos incorrectos –especialmente en fechas–, así como normalizar términos mediante vocabularios controlados. Se pueden dividir celdas, agrupar valores similares y aplicar transformaciones con expresiones GREL¹. Además, OpenRefine permite detectar valores faltantes, validar formatos y reconciliar datos con fuentes externas. Finalmente, los datos limpios se exportan en el formato deseado, listos para su reutilización o reintegración en el repositorio.

2.2. Implementación de vocabularios controlados

Para una mejora de metadatos con el objetivo de la optimización de visibilidad e interoperabilidad con otros sistemas, los vocabularios controlados juegan un rol relevante (Chipangila et al., 2024). Por una parte, mejoran la experiencia de búsqueda en el propio repositorio, controlando el vocabulario en campos clave como autores, idioma y palabras clave. Por otro lado, optimizan la integración con sistemas externos, a través del uso de identificadores, vocabularios o listas controladas, y formatos preestablecidos en campos clave para la interoperabilidad.

Para incorporar vocabularios controlados en repositorios digitales, el primer paso es identificar los campos de metadatos que se podrían beneficiar de la normalización. Por ejemplo, autores, fechas, idioma, palabras clave o temas, y tipo de recurso.

Posteriormente, es necesario seleccionar los vocabularios adecuados para controlar determinados campos. A continuación, listamos algunas opciones según categoría:

- Para controlar materias o palabras clave: algunas alternativas son las Library of Congress Subject Headings (LCSH), el tesauro de la UNESCO o el tesauro de Naciones Unidas (UNBIS). Luego de decidir el vocabulario que se quiere integrar, es necesario obtener el tesauro en un formato SKOS (Simple Knowledge Organization System), RDF (Resource Description Framework) o similar, e integrarlo a la plataforma. Dicha integración va a depender estrictamente de la plataforma y debe asegurar que los campos

1 GREL es un lenguaje de programación simple diseñado para facilitar la organización, transformación y consulta de datos en OpenRefine.

de materias o descriptores estén mapeados a los términos del tesauro. Por ejemplo, el campo dc.subject en Dublin Core o 650 en MARC21. Finalmente, es necesario verificar que los términos se autocompleten correctamente y que estos estén enlazados a sus URLs.

- Para controlar autoridades: es posible utilizar vocabularios controlados como VIAF (Virtual International Authority File) o identificadores persistentes como ORCID o ROR (Research Organization Registry). El procedimiento de integración de estos elementos va a depender también estrictamente de la plataforma. De igual manera, se debe asegurar que los campos del esquema de metadatos relacionados con autoridades estén adecuadamente mapeados al vocabulario o lista que se quiera integrar.
- Para controlar tipo de ítem: existen diversos vocabularios para controlar el tipo de ítem y fortalecer la interoperabilidad, por ejemplo, con sistemas como gestores de referencias. Entre los más destacados, podemos mencionar DCIM Type Vocabulary (Dublin Core), COAR Resource Type Vocabulary, MODS Resource Types, Schema.org / CreativeWork Types, OpenAIRE Guidelines Types, entre otros. Luego de seleccionar el vocabulario, se debe mapear los valores locales existentes al vocabulario elegido, ajustar formularios de ingreso de datos para que utilicen listas controladas y asegurar que los tipos sean consistentes en los registros. Además, se adapta la exportación de metadatos –por ejemplo, en Dublin Core o XML– para cumplir con estándares de interoperabilidad y facilitar la cosecha (harvest) por agregadores. El proceso puede incluir validación de datos, ajustes en la interfaz de usuario y pruebas para garantizar que los valores se reflejan correctamente tanto en la interfaz interna de administración (backend) como en la visualización pública del repositorio.
- Para controlar campos clave: la normalización de campos clave como el idioma o la fecha de publicación es fundamental para estandarizar estos metadatos. Una adecuada normalización favorece la experiencia de búsqueda en el propio repositorio y garantiza un adecuado intercambio de datos con servicios externos. En estos campos es recomendable adoptar la norma ISO-8601 para normalizar fechas y la norma ISO 639 para normalizar idiomas. Para lograr esta incorporación, se debe usar el formato YYYY-MM-DD para fechas y códigos de dos o tres letras para idiomas. Esto se obtiene mediante listas controladas, validación automática en formularios de ingreso y ajustes en las plantillas de metadatos del sistema –por ejemplo, en Dublin Core, MARC21 o MODS (Metadata Object Description Schema)–. Así se garantiza consistencia, interoperabilidad y compatibilidad con cosechadores externos.

2.3. Auditorías periódicas y ajuste según cambios de normativas

Para realizar auditorías periódicas de calidad de metadatos, se recomienda establecer una rutina regular, por ejemplo, trimestral o semestral, en la que se revisen muestras representativas de registros. Durante esta revisión, se debe verificar la coherencia, completitud, la utilización correcta de vocabularios controlados, formatos normalizados –como ISO 8601 e ISO 639-1– y la ausencia de errores tipográficos o duplicados. El uso de herramientas de validación automática y generación de informes de calidad ayuda a detectar problemas de forma eficiente.

Frente a cambios en normativas o estándares, es fundamental mantener un monitoreo activo de actualizaciones en esquemas como Dublin Core, COAR o OpenAIRE. Ante una modificación, se debe revisar el mapeo de metadatos en el repositorio, ajustar formularios, vocabu-

larios o plantillas de exportación, y capacitar al personal sobre los cambios. Documentar cada ajuste garantiza la trazabilidad y facilita las auditorías futuras.

3. Habilitación de los protocolos de interoperabilidad

Habilitar los protocolos de interoperabilidad en un repositorio digital es fundamental para aumentar su visibilidad y alcance. Estos protocolos permiten que servicios externos, como agregadores, motores de búsqueda académicos y portales nacionales o regionales, puedan cosechar automáticamente los metadatos y contenidos del repositorio, sin necesidad de intervención manual (Eells et al., 2024).

Gracias a esta interoperabilidad, los registros del repositorio pueden integrarse en plataformas como OpenAIRE, BASE, CORE, WorldCat, o incluso repositorios nacionales o regionales, lo que mejora su descubribilidad por parte de investigadores, estudiantes y el público en general. Además, facilita la integración con otros sistemas institucionales y garantiza que los contenidos cumplan con estándares abiertos y políticas de acceso abierto.

Existen diversas maneras de promover esta interoperabilidad, a través de diversos protocolos que enlistamos a continuación:

- Protocolo OAI-PMH: Es un protocolo basado en HTTP y XML que permite a otros sistemas cosechar metadatos de un repositorio de forma automática. Es ampliamente usado para que agregadores como OpenAIRE o BASE recojan información sobre los recursos disponibles.
- REST-API: Es una interfaz que permite a otros sistemas consultar, crear o modificar recursos en un repositorio mediante peticiones HTTP, como GET, POST, PUT, o DELETE. Es muy flexible y usada para integraciones con sistemas externos o aplicaciones personalizadas.
- SWORD: es un protocolo que permite depositar contenidos, como artículos, datasets o tesis, de manera remota en un repositorio. Facilita integraciones entre plataformas de publicación, sistemas institucionales y repositorios digitales.

La habilitación de los diversos protocolos para interoperar con otros sistemas va a depender estrictamente del software y la versión que se esté utilizando. Se deben utilizar las herramientas disponibles en cada repositorio para activar la API o SWORD con el objetivo de permitir la carga o actualización desde clientes externos.

Asimismo, cada repositorio permite la configuración de su protocolo OAI-PMH, donde se define la URL de base del servicio y se exponen los metadatos en un formato adecuado. En este ámbito, los metadatos son fundamentales y deben ser expuestos correctamente en formatos legibles para el protocolo como Dublin Core, MARC o DataCite.

La tabla 2 explica cómo activar el protocolo OAI-PMH según cada software:

Tabla 2
Activación de protocolo OAI según cada software.

Software	¿Cómo activar OAI-PMH?
DSpace https://wiki.lyrasis.org/display/DSDOC8x/OAI	Si bien OAI-PMH viene activado por defecto, es necesario verificar que en "local.cfg" o "dspace.cfg" estén "oai.enabled=true" y "oai.path=oai".
EPrints https://wiki.eprints.org/w/OAI	El soporte OAI viene activado por defecto. Es necesario configurar la URL de base ("oai.base_url"), "archive_id", "sets" y cualquier <i>stylesheet</i> XSL en "cfg/cfg.d". Es necesario mirar particularmente el archivo <i>oai.pl</i> .
Fedora https://github.com/saw-leipzig/foaipmh	Fedora 6 no incluye OAI-PMH nativo. Requiere implementar un endpoint externo (ej. Django + foaipmh) conectado con su REST API para operar.
InvenioRDM https://inveniordm.docs.cern.ch/reference/oai_pmh/	OAI-PMH activado por defecto en "/oai2d". Desde la interfaz de administración puedes definir sets y formatos ("oai_dc", "oai_datacite").
TIND	Desde el panel de administración, ir a "OAI Repository Admin", activar el proveedor y definir sets.
Digital Commons https://digitalcommons.elsevier.com/integration-preservation/digital-commons-and-oai-pmh	Soporte OAI habilitado por defecto. Se configuran los campos que se exponen a través del administrador de metadatos.
Dataverse https://guides.dataverse.org/en/latest/admin/harvestserver.html	Desde "Harvesting Server" en el Dashboard, se habilita el servicio OAI-PMH y se definen sets. El endpoint suele ser "/oai".

Para garantizar un correcto funcionamiento del protocolo OAI-PMH existen herramientas conocidas como validadores, que permiten evaluar la operatividad del protocolo. Una de las herramientas más conocidas (figura2) es el OAI-PMH validator (<https://validator.oaipmh.com>). Este validador permite verificar que los conjuntos (sets) estén bien definidos y que los registros exporten los campos correctamente: fechas, identificadores, tipos de ítem, idioma, etc. Asimismo, permite revisar las respuestas XML para detectar errores o formatos mal estructurados. Para analizar el funcionamiento del protocolo de un repositorio, solo se necesita ingresar la URL de base y el sistema realiza el análisis.

Figura 2
OAI-PMH Validator.

Para sostener un correcto funcionamiento del protocolo es necesario realizar actividades de mantenimiento y monitorear periódicamente el estado del protocolo. Asimismo, es importante adoptar los mapeos de metadatos si cambian las recomendaciones de COAR, OpenAIRE u otros agregadores. Asimismo, para el seguimiento, es relevante documentar los *end-points*² y mantener coordinación con otros sistemas que dependan del repositorio.

4. Adopción de identificadores persistentes

Un identificador persistente es una referencia digital única, duradera y resoluble a un objeto concreto, por ejemplo, un artículo, *dataset*, software, persona u organización. Estos identificadores están diseñados para mantener su validez y accesibilidad a lo largo del tiempo, incluso si cambia su ubicación física o su servidor (Meadows et al., 2019).

Normalmente, un identificador persistente tiene tres componentes esenciales: 1) unicidad global, lo que implica que incluye una sintaxis controlada y espacio de nombres gobernados por autoridades claramente definidas; 2) persistencia, lo que garantiza enlaces y funciones de resolución estables, así como esquemas y objetos referidos persistentes, y 3) resolubilidad, lo que implica que son accesibles tanto para humanos como para máquinas, proporcionando información sobre cómo encontrar, acceder o usar el objeto referido (de Castro et al., 2023).

Los identificadores persistentes (tabla 3) son relevantes para optimizar la visibilidad de las publicaciones y la citabilidad, ya que facilitan que buscadores, repositorios académicos y herramientas de análisis encuentren y enlacen automáticamente los documentos, sin depender de URLs fluctuantes. Asimismo, entregan estabilidad a largo plazo, promoviendo la accesibilidad de recursos, dado que combaten el "*link rot*"³ y el "*content drift*"⁴: incluso si el objeto se mueve, su identificador persistente seguirá resolviendo correctamente. Finalmente, si los identificadores persistentes están asociados a metadatos estructurados y respaldados por infraestructuras robustas, mejoran la confianza y reputación del contenido.

Entre los identificadores persistentes más reconocidos, encontramos el Digital Object Identifier (DOI). Es el esquema más empleado para artículos, libros, *datasets* y software. Este sistema combina un identificador permanente con metadatos obligatorios y resolución garantizada. Al resolver un DOI, este conduce a una *landing page* con metadatos, promoviendo la visibilidad y el seguimiento de citaciones. Los DOIs son asignados por agencias registradores como CrossRef y DataCite y, en líneas generales, resulta más costoso que otros identificadores.

Por otro lado, también existe el sistema Handle, que es un identificador no-comercial usado desde 1995. Su principal objetivo es proveer servicios de identificación persistente y resolución, operados de manera centralizada por la Corporation for National Research Initiatives (CNRI). Un identificador Handle consiste en un prefijo que identifica la autoridad, en conjunto con un sufijo, que se refiere al objeto al cual hace referencia. Handle es la base técnica de DOI y resulta más económico en su implementación que DOI. Algunos sistemas como DSpace

2 Un *endpoint* es una dirección o URL específica que permite a un sistema externo interactuar con el repositorio y acceder a sus servicios o datos.

3 *Link rot* ocurre cuando un hipervínculo ya no conduce al contenido previsto porque la página ha sido movida, eliminada o el dominio ya no está activo.

4 *Content drift* ocurre cuando el contenido en una URL determinada cambia con el tiempo, por lo que ya no refleja lo que originalmente se citó o se pretendía, aunque el enlace aún funcione.

integran el sistema Handle por defecto, pero necesita ser adquirido y configurado para que resuelva correctamente los recursos a través del identificador (<https://wiki.lyrasis.org/display/DSDOC8x/Handle.Net+Registry+Support>).

Otra alternativa es el identificador persistente ARK (Archival Resource Key), un sistema diseñado para proporcionar enlaces duraderos y confiables a objetos digitales, particularmente útil en bibliotecas, archivos y museos. A diferencia de otros identificadores como el DOI o Handle, ARK es más barato, descentralizado y altamente flexible (<https://arks.org>), lo que permite a las instituciones generar y gestionar sus propios identificadores sin depender de una entidad registradora central. Su formato típico es ark:/NAAN/identificador, donde el NAAN identifica a la organización emisora. Una característica distintiva de ARK es su capacidad de ofrecer acceso no solo al objeto digital, sino también a sus metadatos y a una declaración de compromiso de persistencia, lo que refuerza la transparencia y la confianza en la conservación a largo plazo. Este sistema se ha adoptado ampliamente en instituciones como la Biblioteca del Congreso de EE. UU. y la California Digital Library, favoreciendo la visibilidad y trazabilidad de recursos culturales y académicos.

Tabla 3
Comparación entre identificadores persistentes.

	DOI	Handle	ARK
Gestión	Centralizada (DataCite, Crossref)	Distribuida (CNRI)	Descentralizada (institucional)
Estructura	10.1234/abc123	20.5000/xyz456	ark:/12345/x6789
Resolución	Sí, mediante https://doi.org/	Sí, mediante https://hdl.handle.net/	Sí, mediante https://n2t.net/ o local
Acceso a metadatos	Sí (<i>landing page</i> obligatoria)	Sí (dependiendo de uso)	Sí (mediante inflexión, es decir, un carácter modificador de búsqueda)
Persistencia garantizada	Alta (por contrato)	Alta (depende de repositorio)	Variable (según política institucional)
Coste	Requiere membresía de pago y puede incluir coste adicional por cada asignación de DOIs	Cuota anual de 50 USD	No requiere pago, ni membresía con costo asociado.
Usos típicos	Artículos, <i>datasets</i> , software	Objetos en repositorios (DSpace, Fedora)	Archivos, bibliotecas, museos digitales

Para implementar un identificador persistente, lo primero es seleccionar cuál se va a implementar. Cabe señalar que los identificadores no son excluyentes entre sí y se pueden combinar. Por ejemplo, añadir un DOI y un ARK a un recurso.

A continuación, es necesario registrarse con un proveedor de identificadores persistentes. Para obtener DOIs, se puede hacer a través de una de las agencias registradoras, como Crossref o DataCite, aunque ambos requieren una membresía de pago y puede conllevar un costo asociado por acuñar cada DOI.

Para implementar el sistema Handle, se puede adquirir con la CNRI, pagando alrededor de 50 USD y vinculándolo con un sistema de repositorios compatible (<https://www.handle.net/payment.html>). Una vez configurado, cada objeto depositado recibe un identificador con un prefijo autorizado, asignado por CNRI, y un sufijo único. Estos identificadores se resuelven a través de <https://hdl.handle.net/>, lo que garantiza su accesibilidad a largo plazo, incluso si el recurso cambia de ubicación física.

Para implementar ARK, es necesario solicitar un Name Assigning Authority Number (NAAN) en [arks.org](https://arks.org/about/getting-started-implementing-arks) (<https://arks.org/about/getting-started-implementing-arks>). Un NAAN es un número único que identifica a la institución emisora de ARKs dentro del sistema. Funciona como un prefijo oficial que garantiza que cada organización que crea identificadores ARK tenga un espacio de nombres propio y exclusivo.

Posteriormente, es necesario configurar el repositorio para emitir y mantener identificadores persistentes. Esto va a depender de cada software que se utilice para la gestión del repositorio institucional, y se puede realizar mediante las agencias registradoras, el uso de *plugins*, la integración en el propio sistema o el manejo de archivos de configuración clave en el sistema.

Finalmente, es necesario integrar al flujo de trabajo de la ingesta de nuevos recursos la asignación de identificadores persistentes en el campo "*identifier*", junto con asegurar que el identificador se resuelva adecuadamente en una *landing page* del objeto con sus metadatos y acceso al recurso.

De esta manera, los identificadores persistentes como DOI, Handle y ARK son herramientas fundamentales para fortalecer la visibilidad de los documentos en un repositorio digital. Al proporcionar enlaces estables, únicos y duraderos, permiten que los recursos sean fácilmente localizables, accesibles y citables, incluso cuando su ubicación técnica cambia con el tiempo. Además, al estar integrados en infraestructuras globales de resolución y asociados a metadatos estructurados, estos identificadores facilitan el descubrimiento por motores de búsqueda, recolectores, sistemas de citación académica y redes de datos abiertos. En síntesis, estos identificadores garantizan que los documentos no solo sean preservados, sino también difundidos y reconocidos en los entornos digitales actuales.

5. Optimización SEO del repositorio

La optimización para motores de búsqueda o SEO (Search Engine Optimization) en repositorios digitales es una estrategia clave para aumentar la visibilidad, el acceso y el impacto de los contenidos académicos, científicos y culturales que albergan. A pesar de contar con metadatos estructurados y estándares de preservación, muchos repositorios no logran posicionarse adecuadamente en buscadores como Google o Bing, lo que limita que los usuarios descubran sus recursos de forma orgánica.

Implementar buenas prácticas SEO, como el uso correcto de etiquetas HTML, la exposición de metadatos Dublin Core en formato schema.org, la generación de URLs amigables, el uso de *sitemaps* y la indexación automática de nuevos contenidos, permite que los documentos sean mejor comprendidos, clasificados y priorizados por los motores de búsqueda.

Además, la combinación de SEO con identificadores persistentes (como DOI o Handle) refuerza la estabilidad y la trazabilidad de los recursos en la web.

En un entorno digital donde la atención es limitada y la competencia por visibilidad es alta, optimizar el SEO de un repositorio no es solo una mejora técnica, sino una acción estratégica para garantizar que los recursos depositados cumplan su misión de ser encontrado, utilizado y citado.

Para optimizar el SEO en un repositorio, se recomienda tener en consideración los siguientes elementos:

5.1. Uso correcto de etiquetado semántico HTML

Utilizar etiquetas semánticas como <title>, <meta name="description">, <meta name="citation_doi">, <meta name="citation_author">, <h1>, <h2>, <article>, <section>, entre otras, para ayudar a los motores de búsqueda a comprender la estructura del contenido. Todo esto incluso ayuda a que herramientas como Altmeter puedan rastrear de mejor manera las métricas de un recurso en particular (<https://help.altmetric.com/support/solutions/articles/6000240582-required-metadata-for-content-tracking>) (Reyes-Lillo y Pastor-Ramon, 2024).

Hay que asegurar que cada página de recurso, como un documento, tenga un <title> único y descriptivo. Adicionalmente, es recomendable incluir metadatos enriquecidos usando schema.org o Dublin Core embebido en etiquetas <meta> o en JSON-LD.

A continuación, es posible distinguir un uso adecuado del etiquetado semántico en Dublin Core (figura 3) o en JSON-LD (figura 4):

Figura 3

Ejemplo de Dublin Core embebido en <meta>.

```
<head>
  <meta name="DC.title" content="How to improve your metadata" />
  <meta name="DC.creator" content="Laura Martinez" />
  <meta name="DC.date" content="2023-11-15" />
  <meta name="DC.identifier" content="http://doi.org/10.1234/test-doi.2025.001" />
  <meta name="DC.language" content="es" />
  <meta name="DC.type" content="Book" />
</head>
```

Figura 4

Ejemplo de JSON-LD.

```
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@type": "Book",
  "name": "How to improve your metadata",
  "author": {
    "@type": "Person",
    "name": "Laura Martinez"
  },
  "datePublished": "2023-11-15",
  "identifier": {
    "@type": "PropertyValue",
    "propertyID": "Handle",
    "value": "http://doi.org/10.1234/test-doi.2025.001"
  },
  "inLanguage": "es",
  "license": "https://creativecommons.org/licenses/by/4.0/",
  "publisher": {
    "@type": "Organization",
    "name": "Universidad Nacional"
  }
}
</script>
```

5.2. Crear y mantener un *sitemap* XML

La creación y mantención de un *sitemap* XML en un repositorio digital es una práctica esencial para mejorar la indexación y visibilidad de los contenidos por parte de los motores de búsqueda como Google o Bing. Un *sitemap* actúa como un mapa estructurado que lista todas las páginas relevantes del repositorio, permitiendo a los buscadores descubrir rápidamente nuevos documentos, actualizaciones o recursos depositados.

Para que funcione correctamente, es fundamental que el *sitemap* incluya únicamente URLs públicas, permanentes y accesibles (como las que contienen identificadores persistentes tipo DOI o Handle), y que se actualice automáticamente cada vez que se agrega o modifica contenido. Además, debe estar correctamente referenciado en el archivo robots.txt y ser enviado a herramientas como Google Search Console para maximizar su eficacia.

Una buena implementación del *sitemap* no solo agiliza la indexación, sino que también mejora el posicionamiento SEO de los recursos, aumentando su alcance y uso dentro del ecosistema académico y científico.

5.3. Configuración adecuada del archivo robots.txt

El archivo robots.txt desempeña un papel clave en la optimización SEO de repositorios digitales, ya que controla cómo los motores de búsqueda acceden, rastrean e indexan sus contenidos. Este archivo, ubicado en la raíz del sitio web, indica a los robots, como Googlebot o Bingbot, qué partes del repositorio pueden ser exploradas y cuáles deben ser excluidas.

Una configuración adecuada permite que los buscadores accedan a las páginas de los objetos digitales, como las *landing pages* con identificadores persistentes. Por otra parte, se puede bloquear rutas irrelevantes o sensibles, como áreas administrativas o filtros de navegación que podrían generar contenido duplicado. Por esta razón, es indispensable que el archivo no bloquee rutas relevantes vinculadas a identificadores persistentes, como por ejemplo, /handle/ o /ark:/.

Un ejemplo simple (figura 5) es el siguiente:

Figura 5

Ejemplo de elementos a considerar en el archivo robots.txt.

```
User-agent: *  
Allow: /handle/  
Disallow: /admin/  
Disallow: /private/  
Sitemap: https://repository.org/sitemap.xml
```

Si no se configura correctamente, el robots.txt puede impedir accidentalmente la indexación de recursos importantes, afectando negativamente su visibilidad y descubrimiento en los resultados de búsqueda. Además, debe incluir una referencia al archivo sitemap.xml, facilitando a los buscadores el rastreo estructurado del contenido.

5.4. Aplicar SSR (Server-Side Rendering) o renderizado híbrido

Aplicar SSR (Server-Side Rendering) o un enfoque de renderizado híbrido en un repositorio digital es una estrategia técnica que contribuye a mejorar la visibilidad y rendimiento del sitio, especialmente en un contexto web cada vez más orientado a la inteligencia artificial y la indexación automatizada.

A diferencia del CSR (Client-Side Rendering) o renderizado puramente en el cliente, donde el contenido se genera dinámicamente en el navegador, el SSR permite que las páginas se generen en el servidor antes de ser enviadas al usuario o al robot de búsqueda. Esto tiene múltiples beneficios: en primer lugar, mejora el SEO, ya que los motores de búsqueda pueden acceder inmediatamente al contenido estructurado sin depender de JavaScript para renderizarlo. En segundo lugar, acelera los tiempos de carga inicial, lo que mejora la experiencia del usuario y favorece la navegación desde dispositivos móviles o redes lentas.

Además, al implementar un renderizado híbrido (SSR combinado con CSR), se logra un equilibrio óptimo entre rendimiento, interactividad y visibilidad, permitiendo que el repositorio sea eficaz tanto para humanos como para *bots* de indexación y motores generativos.

Existen herramientas como Next.js, Nuxt o Rendertron, que permiten adaptar sitios a un SEO amigable y orientado a estos renderizados.

6. Generative Engine Optimization en repositorios: un factor a considerar

Generative Engine Optimization (GEO) es un concepto emergente que se refiere a la optimización de contenidos digitales para motores de búsqueda generativos, como ChatGPT, Google Search Generative Experience (SGE) o Perplexity, que usan inteligencia artificial (IA) para responder con texto generado directamente, en lugar de solo mostrar enlaces como lo hace el SEO tradicional (Daniels, 2025).

GEO busca adaptar la forma en que los contenidos digitales están estructurados y etiquetados para que modelos de lenguajes (LLMs) los interpreten correctamente, los referencien en sus respuestas y sean integrados en el contenido generado por inteligencia artificial (Aggarwal et al., 2024).

De esta forma, así como el SEO optimiza los contenidos para que sean más visibles en Google, GEO optimiza los contenidos para ser comprendidos, citados y usados por los motores de generación de respuestas sobre la base de la IA.

En este ámbito, los repositorios también pueden tomar acciones específicas para optimizar su contenido basadas en estrategias GEO. Por ejemplo, estructurar los metadatos utilizando schema.org, JSON-LD, Open Graph, entre otras tecnologías es altamente recomendable para contribuir a la incorporación de su contenido en respuestas generadas por herramientas de IA.

Por otro lado, es necesario proporcionar contenido claro y accesible, evitando el texto oculto o técnico difícil de procesar para modelos LLMs. Asimismo, es aconsejable indicar claramente licencias de propiedad intelectual flexibles como Creative Commons para facilitar la reutilización de contenido.

Finalmente indicar que también son factores clave para mejorar el contenido para ser procesado por LLMs: el uso de *landing pages* descriptivas, la incorporación de identificadores persistentes para garantizar trazabilidad, proporcionar ficheros RDF o JSON y facilitar el acceso a la API.

En síntesis, GEO representa un nuevo paradigma de visibilidad para los repositorios digitales, donde ya no basta con aparecer en Google, sino que el contenido debe ser estructurado, accesible y comprensible por modelos de lenguaje. Implementar estrategias GEO no solo aumenta el alcance de los recursos, sino que prepara al repositorio para integrarse en el ecosistema de búsqueda y descubrimiento basado en inteligencia artificial que está definiendo el futuro del acceso al conocimiento.

7. Conclusión

La optimización de visibilidad en repositorios digitales requiere un enfoque integral que combine aspectos técnicos, normativos y estratégicos. La calidad de los metadatos garantiza que los recursos sean comprensibles y reutilizables tanto por humanos como por máquinas; la interoperabilidad permite su integración fluida en redes globales de información; y los identificadores persistentes aseguran su trazabilidad y acceso duradero. Al mismo tiempo, una sólida optimización SEO mejora el posicionamiento en buscadores, mientras que incorporar una perspectiva GEO amplía el alcance de los contenidos hacia agentes de inteligencia artificial generativa. En conjunto, estos elementos fortalecen la visibilidad, el impacto y la circulación del conocimiento alojado en los repositorios, alineándolos con los principios de la ciencia abierta y el acceso equitativo a la información.

8. Financiación

Este trabajo forma parte del proyecto “Parámetros y estrategias para incrementar la relevancia de los medios y la comunicación digital en la sociedad: curación, visualización y visibilidad (CUVICOM)”. Ayuda PID2021-123579OB-I00 financiada por MICIU/AEI/10.13039/501100011033 y por FEDER, UE.

9. Referencias

- Aggarwal, P., Murahari, V., Rajpurohit, T., Kalyan, A., Narasimhan, K., & Deshpande, A. (2024). *GEO: Generative Engine Optimization* (arXiv:2311.09735). arXiv. <https://doi.org/10.48550/arXiv.2311.09735>
- Chipangila, B., Liswaniso, E., Mawila, A., Mwanza, P., Nawila, D., M'sendo, R., Nyirenda, M., & Phiri, L. (2024). Controlled vocabularies in digital libraries: Challenges and solutions for increased discoverability of digital objects. *International Journal on Digital Libraries*, 25(2), 139–155. <https://doi.org/10.1007/s00799-023-00374-1>
- Daniels, C. (2025). *From SEO to GEO: How agencies are navigating LLM-driven search*. Campaign Asia. <https://www.campaignasia.com/article/from-seo-to-geo-how-agencies-are-navigating-llm-driven-search/501593>
- De Castro, P., Herb, U., Rothfritz, L., & Schöpfel, J. (2023). *Building the plane as we fly it: The promise of Persistent Identifiers*. Zenodo. <https://doi.org/10.5281/zenodo.7258286>

- Eells, L., Kelly, J., & Farrell, S. (2024). Repository (R)evolution: Metadata, interoperability, and sustainability. *Journal of Librarianship and Scholarly Communication*, 12(1). <https://doi.org/10.31274/jlsc.16890>
- Ma, S., Lu, C., Lin, X., & Galloway, M. (2009). Evaluating the metadata quality of the IPL. *Proceedings of the American Society for Information Science and Technology*, 46(1), 1–17. <https://doi.org/10.1002/meet.2009.1450460249>
- Meadows, A., Haak, L. L., & Brown, J. (2019). Persistent identifiers: The building blocks of the research information infrastructure. *Insights*, 32(1). <https://doi.org/10.1629/uksg.457>
- OpenAIRE. (2020). OpenAIRE Guidelines—OpenAIRE Guidelines documentation. Retrieved from: <https://guidelines.openaire.eu/en/latest>
- Piowar, H., Priem, J., Larivière, V., Alperin, J. P., Matthias, L., Norlander, B., Farley, A., West, J., & Haustein, S. (2018). The state of OA: A large-scale analysis of the prevalence and impact of Open Access articles. *PeerJ*, 6, e4375. <https://doi.org/10.7717/peerj.4375>
- Reyes-Lillo, D., Morales-Vargas, A. & Rovira, C. (2025). Visibility, discoverability, findability, Search Engine Optimization (SEO) and Academic SEO in digital repositories: A scoping review. *BiD: textos universitaris de biblioteconomia i documentació*, (54). <https://doi.org/10.1344/BID2025.54.0>
- Reyes-Lillo, D., & Pastor-Ramon, E. (2024). Use of handle in institutional repositories and its relationship with alternative metrics: A case study in Spanish-speaking America. *Hipertext. Net*, 29. <https://doi.org/10.31009/hipertext.net.2024.i29.17>
- Swan, A. (2010, Febrero). *The open access citation advantage: Studies and results to date*. <https://eprints.soton.ac.uk/268516/>
- UNESCO. (2021). *UNESCO recommendation on open science*. <https://unesdoc.unesco.org/ark:/48223/pf0000379949>
- Van Hooland, S., & Verborgh, R. (2015). *Linked data for libraries, archives and museums: How to clean, link and publish your metadata*. Facet. <https://doi.org/10.29085/9781783300389>
- Westbrook, R. N., Johnson, D., Carter, K., & Lockwood, A. (2012). Metadata clean sweep: A digital library audit project. *D-Lib magazine*, 18(5/6). <https://doi.org/10.1045/may2012-westbrook>