

ESCUELA COMPLUTENSE LATINOAMERICANA

*preservación
digital de
documentos*
archivos, bibliotecas y museos

Interfaces de un SGOD

**Dr. Gonzalo Luján
Villarreal**

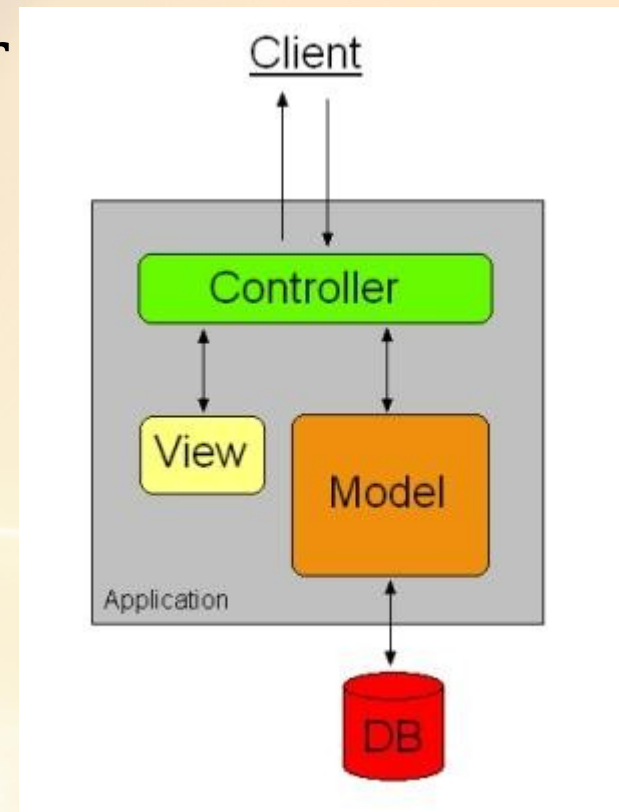
PREBI-SEDICI - Universidad
Nacional de La Plata

gonzalo@prebi.unlp.edu.ar

5 de marzo de 2015

Interfaces

- Las interfaces permiten acceder y manipular los datos de los sistemas de software
 - No debemos pensar las interfaces como elementos meramente estéticos
- El diseño de software busca separar los elementos del negocio de la presentación de los mismos
 - Ej. patrón de diseño Model-View-Controller (MVC)



Elementos del patrón MVC. Fuente: Wikimedia Commons

Interfaces

Las interfaces definen cómo los humanos interactuamos con los sistemas. Algunos ejemplos típicos:

- Interfaz web (HTML)
- Software de escritorio
- Interfaz de línea de comandos

La estrategia de separar la lógica de la aplicación de la presentación permite tener múltiples vistas para un mismo modelo.

Por ejemplo, DSpace brinda una completa interfaz web para usuarios y administradores, y una amplia variedad de funcionalidad accesible desde la línea de comandos: importación/exportación, tareas de curación, generación de reportes estadísticos

<https://wiki.duraspace.org/display/DSDOC4x/Command+Line+Operations>

Interfaces

En ciertos contextos, pueden incluso determinar si un sistema es usable o no para determinados usuarios:

- interfaces adaptables (ej. dispositivos móviles)
- interfaces internacionalizables (multi idioma)
- interfaces accesibles (accesibilidad web)

Interfaces

Vale la pena realmente analizar estos casos "especiales" ?

Accesibilidad

El estado Argentino promueve la adopción de normas de accesibilidad web (Ley 26.653)

Internacionalización y adaptabilidad

Las estadísticas de usuarios en SEDICI sugieren que es conveniente considerar los otros casos

Ley 26.653: <http://infoleg.mecon.gov.ar/infolegInternet/anexos/175000-179999/175694/norma.htm>

Interfaces

Países desde donde se han registrado visitas durante 2014



Si bien predominan las visitas desde países latinoamericanos, aproximadamente un 15% de las visitas provienen de otras regiones.

Interfaces

Accesos a SEDICI desde el 17 de enero 2015 al 16 de febrero de 2015

Sistema operativo	Sesiones	% Sesiones
1. Windows	49.156	79,42 %
2. Android	5.922	9,57 %
3. Macintosh	2.742	4,43 %
4. iOS	2.160	3,49 %
5. Linux	1.320	2,13 %
6. Windows Phone	208	0,34 %
7. BlackBerry	146	0,24 %
8. (not set)	96	0,16 %
9. Series40	80	0,13 %
10. Chrome OS	29	0,05 %

Más de 8500 sesiones desde dispositivos móviles (aprox. 14%.)

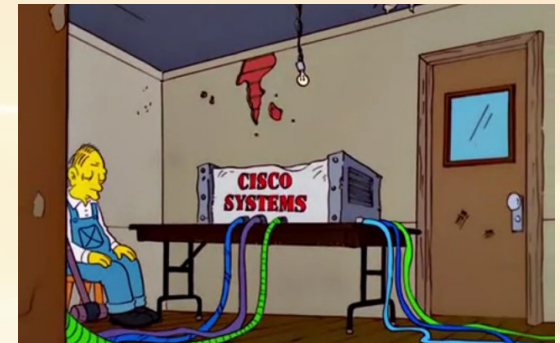
13% en octubre 2014
6,5% en octubre 2013
2,7% en octubre 2012
1,2% en octubre 2011

La tendencia parece evidente

Interfaces

Algunas ideas previas

- qué es un servidor?
 - y un servidor web?
 - y un contenedor web?
- qué es un request?
- qué es el HTTP?
- qué es el HTML?
 - estructura
 - separación de datos vs presentación (CSS)
 - XHTML => HTML "correcto"



Interfaces

Las interfaces no sólo consideran a la interacción del usuario "humano" con el software
Permiten también definir cómo se expondrán datos hacia otros sistemas: HTML, XML, JSON...

En el campo de los repositorios, puede afectar enormemente la difusión e interoperabilidad. Ej:

- Campos META para Google Scholar
- Exposición de registros en XML (ej. RSS, Open Search)
- Campos especiales para Open Graph (og:title, og:image , og:url)

Google Scholar. Incluso Guidelines for Webmasters

<https://scholar.google.com/intl/en/scholar/inclusion.html>

The Open Graph Protocol <http://ogp.me/>

Ejemplo de Interfaz JSON en Twitter REST API <https://dev.twitter.com/rest/public>

Interfaces

Tomemos un documento cualquiera, por ej:

<http://sedici.unlp.edu.ar/handle/10915/44162>

Analicemos (muy por arriba!) la estructura del documento HTML...

qué información se genera allí que le sirve a Google Scholar?

Comparemos con esta URL:

<http://sedici.unlp.edu.ar/handle/10915/2>

por qué no se genera toda esa información para Google Scholar? qué le interesa a GS?

Interfaces en DSpace

ESCUELA COMPLUTENSE LATINOAMERICANA 2015

Preservación digital de documentos: archivos, bibliotecas y museos



UNIVERSIDAD
COMPLUTENSE
MADRID



UNIVERSIDAD
NACIONAL DE
LA PLATA

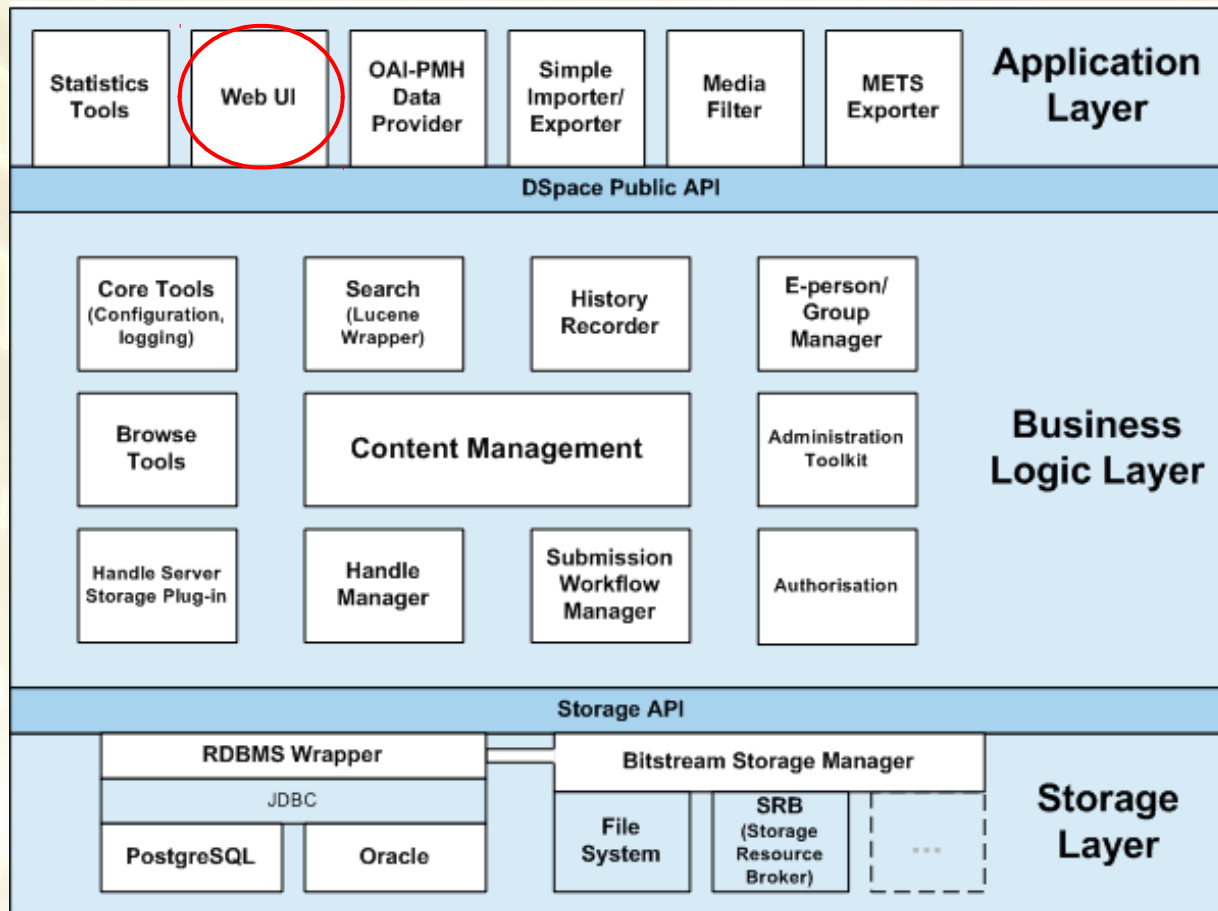
Interfaces en DSpace

DSpace sigue un diseño en capas

- Aplicación (Web UI, OAI PMH data provider, media filter)
- Lógica de negocios (core, handle, lucene search)
- Almacenamiento (RDMBS, Bitstream)

DSpace Architecture <https://wiki.duraspace.org/display/DSDOC5x/Architecture>

Interfaces en DSpace



Arquitectura, capas y componentes de DSpace. Fuente <https://wiki.duraspace.org/>

Interfaces en DSpace

Uno de los componentes esenciales de la capa de aplicación es la interfaz de usuario web (WEB UI)

Permite a los usuarios acceder a DSpace a través de los navegadores web

DSpace soporta dos tipos de interfaces: JSPUI y XMLUI

JSPUI

La interfaz JSPUI está implementada usando Java Servlets para gestionar la lógica de negocios, y JavaServer Pages (JSP) para producir el código HTML. Los documentos HTML se construyen ensamblando "porciones" generadas por archivos JSP, organizados en un layout común a todas las páginas.

<https://wiki.duraspace.org/display/DSDOC4x/Configuration+Reference>

JSPUI

Dentro del directorio `<dspace-src>/dspace/modules/dspace-jspui/src/main/webapp/` se encuentra: el archivo `styles.css.jsp` (hoja de estilos principal)

El directorio `layout` posee todos los archivos `.jsp` utilizados para construir los documentos HTML

JSPUI

The screenshot displays the DSpace web interface with several key components highlighted:

- (Header):** The top section featuring the DSpace logo, "About DSpace Software" link, and the text "(Header)".
- (Location Bar):** A navigation bar showing "DSpace at My University >" and the text "(Location Bar)".
- (Navigation Bar):** A sidebar on the left containing search options (Search DSpace, Advanced Search, Subject Search), home links (Home, Communities & Collections, Titles, Authors, Subjects, By Date), and sign-on options (Receive email updates, My DSpace authorized users, Edit Profile, Help, About DSpace). This area is labeled "(Navigation Bar)".
- (Browse Layout):** The main content area showing a "DSpace is Live" message, a search box, and a "Communities in DSpace" section. A secondary view shows a "Browse by Title" section with a letter navigation bar and a table of items. This area is labeled "(Browse Layout)".

The table in the "Browse by Title" section contains the following data:

Issue Date	Title	Author(s)
1996	The 1995 review and extension of the Nuclear Non-Proliferation Treaty	edited by Thomas Morgan, Jason Zych and Clifford E. Singer
1998	Aadhi raat ke baad : "After midnight"	S. Rashid Naim
2005	Anatomy of interlinking rivers in India : a decision in doubt	A.C. Shukla and Vandana Asthana; Asthana, Vandana
Feb-2004	Asymptotic theory of ignition and failure of self-sustained detonations	-
Jun-2001	Atomistic scale experimental observations and micromechanical/continuum models for the effect of hydrogen on the mechanical behavior of metals	-

XMLUI

La interfaz XMLUI se construye sobre el Framework Apache Cocoon

- Arquitectura basada en componentes
- Procesamiento en *pipeline*



XMLUI

Brinda modularidad, extensibilidad y una organización multicapa (tiers)

- Capa de estilos (style tier)
- Capa de temas (theme tier)
- Capa de aspectos (aspect tier)

XMLUI

style	utiliza CSS y XHTML, para "retocar" un tema ya existente
theme	utiliza XSLT, CSS y XHTML, y permite crear nuevos temas o realizar ajustes más avanzados
aspect	utiliza el framework Apache Cocoon y código Java (y XSLT), y permite definir nuevas características y generar nuevo contenido dentro de los DRI

XMLUI

DSpace viene "de fábrica" con un theme llamado **Manakin**

Posee una estructura muy flexible y personalizable
Incluye mucha funcionalidad, fácilmente adaptable mediante **archivos de configuración**

Configuración vs programación: ahorro de mucho tiempo y esfuerzo (para quién?)

Algunos ejemplos:

- assume logon: true/false (permite tomar el rol de otro usuario)
- login redirect: path (indica hacia donde se dirigen los usuarios al iniciar sesión)
- analytics key: clave de google analytics
- force ssl : true/false (para todas las conexiones autenticadas)

XMLUI

Más allá de los aspectos generales del repositorio, Manakin permite personalizar:

- comunidades
- colecciones
- ítems particulares

(esta es una gran ventaja en comparación con JSPUI)

Ejemplo:

- Radio Universidad en SEDICI <http://sedici.unlp.edu.ar/handle/10915/25224>
- Revistas UNLP <http://sedici.unlp.edu.ar/handle/10915/51>
- Unidades Académicas <http://sedici.unlp.edu.ar/handle/10915/1>

XMLUI

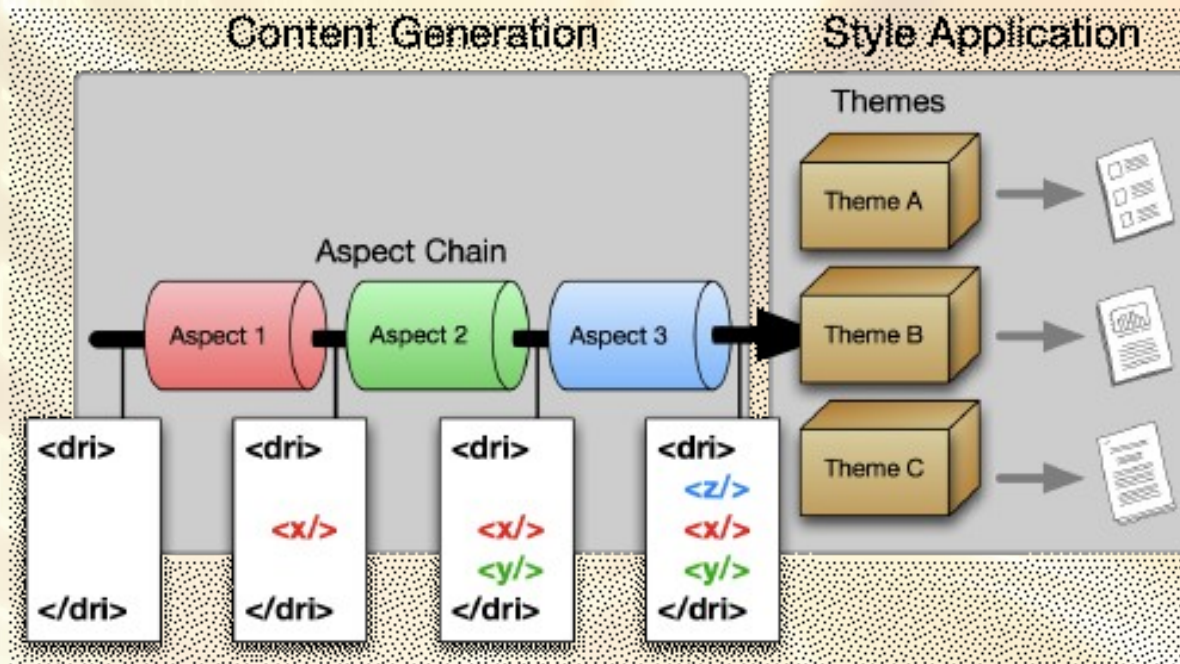
Además de personalizar aspectos estéticos, podemos brindar servicios, destacar colecciones, compartir información, y mucho más.

Algunos ejemplos interesantes:

- <http://dome.mit.edu/>
- <https://tdl-ir.tdl.org/tdl-ir/>
- <http://minds.wisconsin.edu/>
- <https://www.ideals.illinois.edu/>
- <http://sedici.unlp.edu.ar>

cómo se han personalizado estos repositorios?
Qué elementos consideran importantes? Qué servicios incorporan?

XMLUI Request



Manaking request processing. Fuente: Phillips, S.; Green, C. ; Maslov, A.; Mikeal,A. LeggetT, J. "Manakin. A New Face for DSpace" D-Lib Magazine, November/December 2007. Volume 13 Number 11/12

XMLUI Request

Algunas ideas previas...

DRI es

- un documento XML
- una representación abstracta de una página en el repositorio
- podemos verla agregando ?XML o &XML a la URL
 - ej <http://sedici.unlp.edu.ar/handle/10915/44112?XML>
qué elementos aparecen en ese documento XML?

XMLUI Aspects

Paquetes autocontenidos que incluyen:

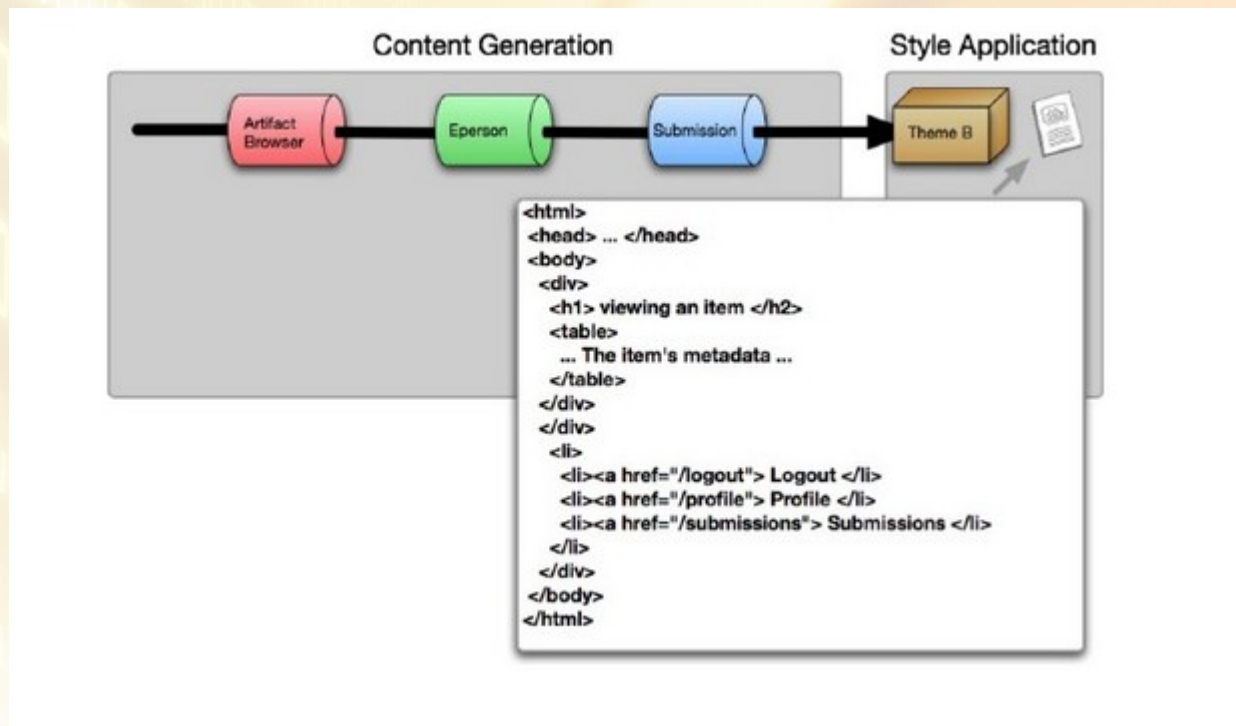
- código Java
- scripts Javascript
- configuración XML

Implementan funciones puntuales, y se organizan en una cadena (Aspect Chain, pipeline)

Requieren un DRI válido como entrada, y generan un nuevo DRI como salida

<https://wiki.duraspace.org/pages/viewpage.action?pageId=19006128>

XMLUI Aspects



Tim Donohue (2009), "Making DSpace XMLUI Your Own". Disponible en <http://www.slideshare.net/ttonohue/making-dspace-xmlui-your-own>

XMLUI Aspects

Aspectos del core de Manakin:

- **ViewArtifacts** Responsable de mostrar metadatos de ítems individuales
- **BrowseArtifacts** Responsable de mostrar diferentes opciones de exploración
- **SearchArtifacts** Muestra los diferentes cuadros de búsqueda (conflicto con Discovery)
- **Administrative** Responsable de la administración de Dspace: creación, modificación y eliminación de comunidades, colecciones, e-persons , grupos...
- **E-Person** Responsable del inicio y cierre de sesiones, registro de nuevos usuarios, recuperación de contraseña...
- **Submission** Envío de items a DSpace, incluyendo determinación del workflow e ingesta de items
- **Statistics** Muestra información estadística
- **Workflow** Muestra tareas del flujo de trabajo (entra en conflicto con el workflow XMLWorkflow)

XMLUI Aspects

Además, hay algunos aspectos opcionales

- **XMLWorkflow** incorporado en DSpace 1.8, responsable del nuevo sistema de flujos de trabajo configurables
- **Discovery** Reemplaza la búsqueda y *búsqueda facetada** estándar. Incluye también opciones de búsqueda facetada
- **SwordClient** Muestra opciones para enviar contenido de Dspace hacia otro sistema con servidores SWORD
- **XMLTest** ayuda a los desarrolladores a crear themes, mostrando muchas opciones de debug

*búsqueda facetada: que usa *facets* . Más información en

http://en.wikipedia.org/wiki/Faceted_search

XMLUI Request

1. El request comienza en el archivo [xmlui]/sitemap.xmap, punto de entrada **de todos los requests**. Aquí se definen rutas para satisfacer los requests
 - a. Si el request actual coincide con un request de descarga, entonces se satisface inmediatamente
 - b. Caso contrario, desde sitemap.xmap se levanta el archivo que controla los temas ([xmlui]/themes/themes.xmapfile)
 - i. Se buscan el themes (en [dSPACE]/config/xmlui.xconf) que mejor coincide (match) con el request
 - ii. Se levanta el archivo sitemap.xmap del theme y se procesa:
 1. se genera el documento DRI para la página actual :
<map:generate type="file" src="cocoon://DRI/{1}"/>
 2. se cargan los archivos del theme (XSLT, CSS, etc.)
 3. se genera un nuevo request interno, que también es procesado por [xmlui]/sitemap.xmap

XMLUI Request

2. De vuelta en el sitemap base, se busca una ruta que coincida con la llamada DRI/** . Esto provoca que se levante el archivo [xmlui]/aspects/aspects.xmapfile , que controla todos los aspectos
 - a. Se cargan los aspectos en orden de aparición
 - b. Se ejecutan uno a uno (aspect chain), formando el documento DRI final. Cada aspecto puede agregar, quitar o modificar el contenido dentro del documento DRI
3. Al finalizar todos los aspectos, el flujo de control vuelve al archivo sitemap.xmap
4. El procesamiento continúa, por lo general aplicando transformaciones XSLT al documento DRI, resultando el un documento XHTML.
5. Finalmente, el documento XHTML y todos los *assets* necesarios (css, imágenes, js, etc.) son enviados al cliente

Difusión e Interoperabilidad

(porque no sólo los humanos necesitan interfaces)



Difusión e interoperabilidad

Los repositorios tienen la capacidad de interoperar para

- proveer otros mecanismos de ingesta (ej. sword)
- proveer otros mecanismos para difusión
 - RSS
 - Open Search
 - OAI PMH
 - otros?

Difusión e interoperabilidad

Feeds y RSS (Really Simple Syndication)



- originalmente pensado para publicar información de actualizaciones frecuentes
- genera un documento RSS con un formato XML estándar
- Cada documento representa un *canal* RSS
- Un sitio puede tener múltiples canales: global, categoría, sección...

Difusión e interoperabilidad

Dspace genera múltiples RSS <url> [/feed/atom_1.0/](#)
<handle>

- de todo el repositorio
- de una comunidad
- de una colección

Ejemplos:

- Portal de Revistas Científicas de la UNLP
- Blogs personales

También podemos "consumir" RSS para mostrar feeds propios o feeds externos interesantes para nuestra comunidad de usuarios

Difusión e interoperabilidad

Open Search

- Colección de formatos simples para compartir información de búsquedas
- Puede extenderse RSS o ATOM para proveer campos especiales
- Su integración es similar a RSS: transmisión mediante http, documentos XML, etc.

Difusión e interoperabilidad

Para qué usaríamos Open Search?

- para compartir con nuestros alumnos una lista de materiales de cierta temática
- para mostrar en el sitio web del laboratorio todas nuestras publicaciones
- para armar mi CV personal
-

Busqueda por handle (scope):

http://sedici.unlp.edu.ar/open-search/discover?rpp=100&format=atom&sort_by=2&order=desc&start=0&scope=10915/25293

http://sedici.unlp.edu.ar/open-search/discover?rpp=100&format=atom&sort_by=2&order=desc&start=0&query=sedici.subtype:%22Articulo%22%20OR%20sedici.subtype:%22Tesis%20de%20Doctorado%22%20OR%20sedici.subtype:%22Objeto%20de%20conferencia%22

Búsqueda por campo (sedici.creator.person)

http://sedici.unlp.edu.ar/open-search/discover?rpp=100&format=atom&sort_by=2&order=desc&start=0&query=sedici.creator.person:%22De%20Giusti,%20Marisa%20Raquel%22

Difusión e interoperabilidad

Cómo sería la URL de OpenSearch para...

- Todas las publicaciones de "Villarreal, Gonzalo" ?
- La tesis doctoral de "Villarreal, Gonzalo" ?
- Todas las publicaciones generadas en cualquier evento "Asamblea General ISTECS" ?
- sólo las de la "XX Asamblea General ISTECS" ?

Difusión e interoperabilidad

Ej: el laboratorio LISEA muestra en su portal las publicaciones alojadas en SEDICI. Todo esto se recupera mediante Open Search



Lisea

Laboratorio de Investigación de Sistemas Ecológicos y Ambientales

Publicaciones en SEDICI

Artículo

- Título:** [Biomasa y productividad aérea de bosques de *Austrocedrus chilensis* en Río Negro, Argentina](#)
Autor: [Ferrando, J. J.](#) - [Goya, Juan F.](#) - [Barrera, M. D.](#) - [Yapura, P. F.](#) - [Frangi, Jorge L.](#)
Resumen: El objetivo de este trabajo fue determinar la biomasa y la productividad de un bosque puro de *Austrocedrus chilensis* y analizar los efectos de las diferencias estructurales so...
- Título:** [Crecimiento individual y de rodales de *Austrocedrus chilensis* en El Bolsón, Río Negro, Argentina](#)
Autor: [Goya, Juan F.](#) - [Yapura, P. F.](#) - [Ferrando, J. J.](#) - [Barreda, M. D.](#) - [Arturi, M. A.](#)
Resumen: Se analizó el crecimiento en volumen y peso seco a nivel individual y de rodal en dos rodales de *Austrocedrus chilensis* (Don.) Florin et Boutleje (ciprés de la cordillera) c...

[Ir a SEDICI](#)

Informe tecnico

- Título:** [Plan de manejo del Campo de Deportes del Centro Basko Euzko Etxea](#)
Autor: [Goya, Juan F.](#) - [Presutti, Miriam E.](#) - [Burns, Sarah Lilian](#) - [Sandoval, Martín](#)
Resumen: El presente informe es el resultado del Acuerdo firmado entre la Facultad de Ciencias Agrarias y Forestales de la Universidad Nacional de La Plata y el Centro Basko Euzko Etxe...

[Ir a SEDICI](#)

Libro

- Título:** [Ecología y ambiente](#)



NOVEDADES FACULTAD DE CIENCIAS AGRARIAS Y FORESTALES

Estrategia de Ingreso 2015. Exámenes de Nivelación. 1° Turno marzo. Química

Estrategia de Ingreso 2015. Exámenes de Nivelación. 1° Turno marzo. Físicomatemática

Se reunió la Comisión de Seguridad e Higiene

Autoridades de la Facultad se reunieron con el Ministro Casamiquela

Fisiología y resistencia a herbicidas

NOVEDADES FACULTAD DE CIENCIAS NATURALES Y MUSEO

Curso de Formación de Guías

Postergación de la mesa de examen final de la Asignatura SENSORES REMOTOS

INSCRIPCIÓN A EXAMENES FINALES PARA EL PRIMER LLAMADO DE MARZO

Postergación mesa de examen final de la Asignatura Mastozoología

Abierta la inscripción a los Curso de Cerámica con técnicas aborígenes y Lengua Quechua



Difusión e interoperabilidad

Open Archives Initiative - Protocol for Metadata Harvesting (OAI PMH)

- provee un mecanismo muy simple y eficiente para la diseminación de metadatos
- service providers y data providers
- ampliamente utilizado en el ámbito de los repositorios
 - repositorios que exponen registros de otros repositorios
 - agregadores (OPAC ISTECA, SNRD, LA Referencia)
 - directorios internacionales (Open Doar, ROAR)

Difusión e interoperabilidad

OAI PMH: 6 verbos: [Identify](#), [ListSets](#), [ListMetadataFormats](#), [ListRecords](#), [ListIdentifiers](#), [ListIdentifiers](#)

No sólo es utilizado por los repositorios. Muchos sistemas que buscan la difusión de su producción buscarán ser OAI-compliant.

Ejemplos:

- OAI en SEDICI: http://sedici.unlp.edu.ar/oxoi/request?verb=ListIdentifiers&metadataPrefix=oai_dc

<http://sedici.unlp.edu.ar/oxoi/driver>

[?verb=ListRecords&metadataPrefix=oai_dc](http://sedici.unlp.edu.ar/oxoi/request?verb=ListRecords&metadataPrefix=oai_dc)

<http://sedici.unlp.edu.ar/oxoi/openaire>

[?verb=ListRecords&metadataPrefix=oai_dc](http://sedici.unlp.edu.ar/oxoi/request?verb=ListRecords&metadataPrefix=oai_dc)

<http://sedici.unlp.edu.ar/oxoi/snrd>

[?verb=ListRecords&metadataPrefix=oai_dc](http://sedici.unlp.edu.ar/oxoi/request?verb=ListRecords&metadataPrefix=oai_dc)

- OJS en el Portal de Revistas de la UNLP (hay vida más allá de DSpace) <http://revistas.unlp.edu.ar/domus/oai>

Difusión e interoperabilidad

El harvesting de registros puede servir

- para exponerlos
 - <http://www.openoar.org/>
 - <http://opac-istec.prebi.unlp.edu.ar/>
 - <http://repositoriosdigitales.mincyt.gob.ar:8380/dnet-web-generic/>
 - <http://www.lareferencia.info/vufind>
- para brindar servicios (ej. estadísticas, actividad del repositorio)
 - <http://roar.eprints.org/>
 - <http://opac-istec.prebi.unlp.edu.ar/>

Proceso de envío

(submission process)



Submission process

Uno de los elementos centrales de todo repositorio es la ingesta de ítems

La calidad de los datos del repositorio se determinan en gran medida durante los procesos de envío por los que atraviesan

Los envíos pueden realizarse desde la administración, desde otros sistemas, por la comunidad de usuarios...

Es necesario definir con mucha precisión cómo se realiza esta ingesta: actores involucrados, etapas, restricciones, destino de los ítems ingresados, estado con que el que quedan

Submission process

DSpace permite personalizar con un alto nivel de detalle cómo se realizan los envíos de documentos hacia el repositorio

Es posible definir múltiples flujos para los envíos (no confundir con *workflow*)

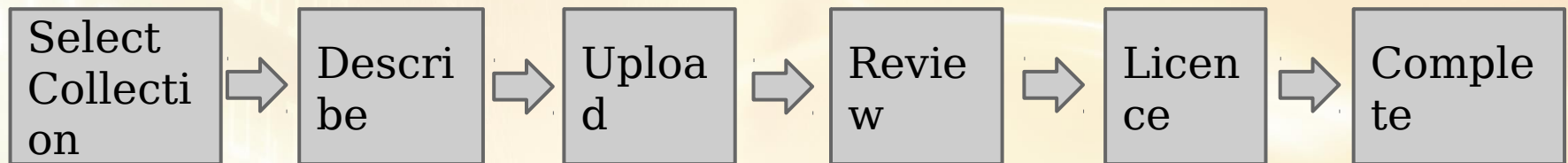
Las configuraciones de los envíos se encuentran en el archivo `[dspace]/config/item-submission.xml`

Esto es independiente de la interfaz que se utiliza (XML-UI o JSP-UI)

<https://wiki.duraspace.org/display/DSDOC3x/Submission+User+Interface#SubmissionUserInterface-UnderstandingtheSubmissionConfigurationFile>

Submission process

Los procesos de envíos de DSpace consisten en una serie de pasos (*steps*)
Cada paso se corresponde con una o más páginas de la interfaz de usuario

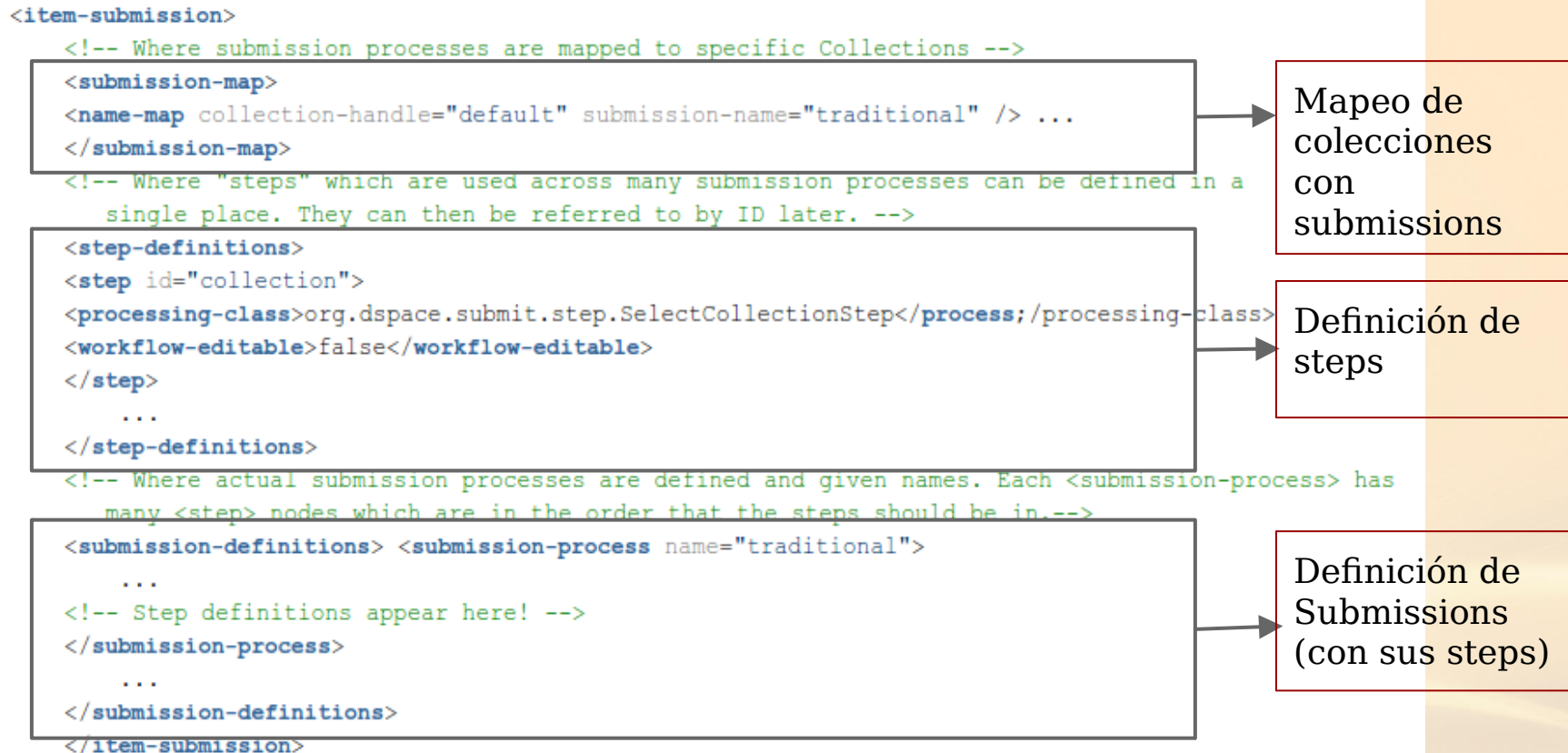


Submission process

Por un lado tenemos *Submission Processes*, que determinan el **orden** y los **pasos** que deben ocurrir en cada "item submission".

Existe un *process-map*, que permite asociar colecciones de DSpace (identificadas por su ID) con un submission process particular. Así, podemos tener distintas interfaces de envíos para distintas colecciones.

Submission process



Archivo item-submission.xml. Fuente <https://wiki.duraspace.org/display/DSDOC3x/Submission+User+Interface>

Submission process

DSpace tiene un process-map, con handle "default" (no es una colección), que asocia un submission-process por default cuando la colección no se especifica en un submission process particular.

Así, por ejemplo, podemos tener un proceso para el Autoarchivo, con ciertos campos obligatorios que deben completar los usuarios.

En el caso de SEDICI, en el submission process "Autoarchivo" no se muestra al usuario la colección (se selecciona automáticamente), y se saltea directamente este paso

Submission process

Cada *submission process* se compone de *steps* (pasos). Cada *step* posee un identificador único, que permite referenciarlo en múltiples *submission processes* (así, se promueve la reutilización).

Además del ID, cada *step* debe contar con una *processing-class* : La clase JAVA encargada de procesar la información en este paso. Debe ser subclase de `org.dspace.submit.AbstractProcessingStep` o alguna de las clases definidas en `org.dspace.submit.step.*`)

Submission process

Elementos opcionales para cada step:

- **heading** (encabezado): mensaje (código del mensaje internacionalizado) que se muestra en el encabezado de la barra de progreso. El prefijo del código del mensaje depende de la interfaz que se esté usando: `xmlui.Submission.<codigo>` para XML-UI , y `jsp.<codigo>` para JSP-UI
- **binding**: clase JAVA que genera la interfaz de usuario para este *step*. Esta propiedad depende de la interfaz en uso (`<xmlui-binding>` o `<jspui-binding>`)
- **workflow-editable**: verdadero o falso. Indica si este *step* aparecerá durante el workflow de Edición de Metadatos (o sea, si los metadatos involucrados en este step se mostrarán al editar los items).

Submission process

Existen algunos *steps* especiales predefinidos:
Collection *step* (id=Collection): se coloca al principio de *submission process*, asegura que SIEMPRE se seleccione una colección cuando se envíe un ítem. Si el usuario ya había seleccionado una colección previamente, entonces este paso se saltea.

Este paso se agrega automáticamente en cada *submission process*.

Complete *step*: (id=complete) se asigna al final de cada *submission process*, y se encarga de "cerrar" los envíos, ejecutando todo el procesamiento del **backend que ocurre en esta etapa**.

Submission process - Ejemplos de Steps

Selección de la licencia CC:

```
<step id="sediciCCLicense">
  <heading>submit.progressBar.CCLicense</heading>
  <processing-
class>org.dspace.submit.step.SediciCCLicenseStep</pro-
cessing-class>
  <jspui-
binding>org.dspace.app.webui.submit.step.JSPCCLicense
Step</jspui-binding>    <xmlui-
binding>org.dspace.app.xmlui.aspect.submission.SediciC
CCLicenseStep</xmlui-binding>
    <workflow-editable>>true</workflow-editable>
  </step>
```

Submission Process - Ejemplos de Steps

Verificación de datos del envío:

```
<step id="sediciVerifyItem">  
  <heading>submit.progressBar.verify</heading>  
  <processing-class> org.dspace.submit.step.VerifyStep  
  </processing-class>  
  <jspui-binding>  
  org.dspace.app.webui.submit.step.JSPVerifyStep</jspui-  
  binding> <xmlui-binding>  
  org.dspace.app.xmlui.aspect.submission.submit.ReviewSt  
  ep </xmlui-binding>  
  <workflow-editable>>true</workflow-editable>  
</step>
```

Submission Process

Hasta aquí contamos con muchos *steps* que nos permiten conformar distintos procesos de envíos. Para definir procesos de envíos concretos, debemos definir la secuencia de pasos dentro cada submission-process (dentro de la sección submission-definitions)

```
<submission-definitions>  
  <submission-process name="example-process">  
    <step id="xxx" />  
    <step id="yyy" />  
    <step id="zzz" />  
  </submission-process>  
</submission-definitions>
```

Submission process - Ejemplos

```
<submission-process name="autoarchiveSubmission">
  <step>
    <processing-
class>org.dspace.submit.step.SkipInitialQuestionsStep</process
ing-class>
    </step>
    <step id="sediciDescribeItem"/>
    <step id="sediciUploadItem"/>
    <step id="sediciVerifyItem"/>
    <step id="sediciCCLicense"/>
    <step id="sediciSignOffLicense"/>
    <step id="sediciWorkflowEditedControl"/> <!-- setea el
flag indicando que se completó el proceso de edición desde el
workflow. Sin interface. -->
  </submission-process>
```

Formularios

ESCUELA COMPLUTENSE LATINOAMERICANA 2015

Preservación digital de documentos: archivos, bibliotecas y museos



UNIVERSIDAD
COMPLUTENSE
MADRID



UNIVERSIDAD
NACIONAL DE
LA PLATA

Formularios

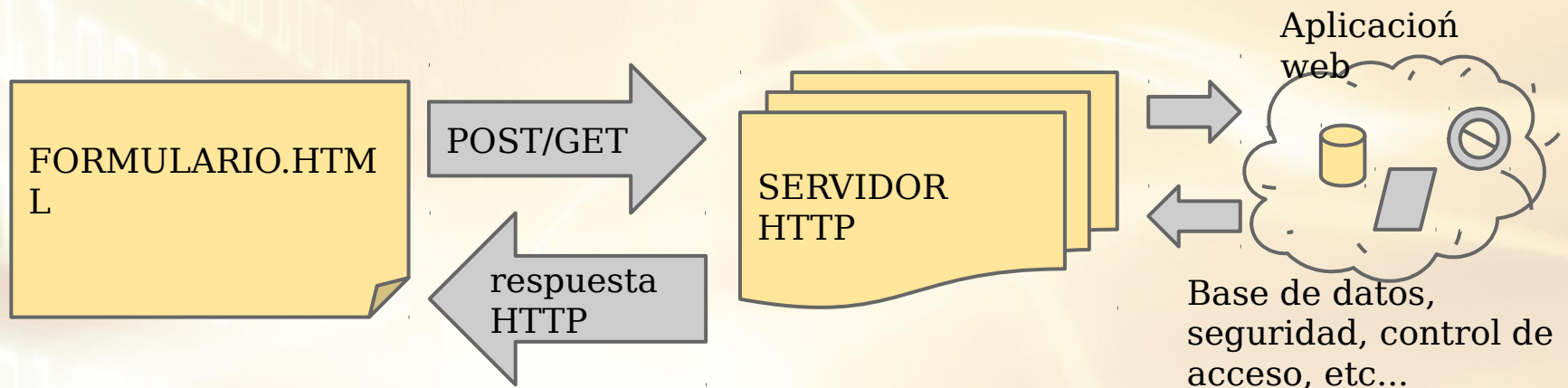
Hasta aquí hemos visto cómo armar interfaces de usuario, cómo preparar procesos de envíos para nuestras colecciones, como configurar cada paso en estos procesos...

Podemos entrar un poco más en el detalle, y trabajar directamente sobre los metadatos y campos de formularios que presentamos a nuestros usuarios

Formularios

Primero lo primero... qué es un formulario web?

Dicho simple, son componentes de las páginas web que permiten capturar datos ingresados por el usuario y enviarlos hacia un servidor



Formularios

Los formularios web se componen de campos de entrada (input fields)

Cada componente define (como mínimo):

- cómo se ingresan los datos (texto, seleccionar, etc.)
- el dominio de los datos (números, fechas, boolean)
- la obligatoriedad del campo

SEDICI
REPOSITORIO INSTITUCIONAL DE LA UNLP

Inicio Buscar material Subir material Institucional Preg

Iniciar sesión en SEDICI

Correo electrónico:
gonzalo@prebi.unlp.edu.ar

Clave:
.....

¿Olvidó su contraseña?

✓ Iniciar sesión

Formularios

HTML define varios campos básicos:

- text, password, select, radio, checkbox, textarea, button

Con algo de programación, es posible crear nuestros propios componentes:

- áreas de texto enriquecido
- calendarios
- campos con búsqueda online (suggest)
- puntos en mapas

Formularios

Más allá del *widget* utilizado, el dominio del dato, y la obligatoriedad, los campos de los formularios deben validarse:

- en el cliente (HTML5, Javascript)
- en el servidor
 - desde el controlador?
 - desde el modelo o la base de datos?

Estas (y otras) alternativas presentan sus ventajas y sus desventajas.

A veces conviene contar con más de una...

Formularios en DSpace

Los Web Forms de DSpace son usados por los administradores y usuarios (submitters) para ingresar y modificar los metadatos de los ítems que ingresan

Los metadatos de los web forms son controlados por el *step* Describe .

Sin embargo, son también configurables utilizando su propio archivo XML (input-forms.xml)

Las cuestiones estéticas de los campos (fuentes, tamaños, colores, fondos, etc.) son controladas por estilos CSS

Nuevamente, separamos el dato de la vista

Formularios en DSpace

DSpace trae una configuración de ejemplo (config/input-forms.xml) que implementa los formularios tradicionales para ingreso de metadatos

Pero también es posible definir nuestros propios conjuntos de formularios personalizados

En el contexto de DSpace, un form es un conjunto de páginas a través de las cuales los usuarios ingresan sus metadatos

- Sin embargo, **un form de DSpace se compone de múltiples forms HTML**
- Dentro de la estructura del XML, cada form se descompone en una serie de páginas, y cada una de ellas representa una página web separada para recolectar metadatos

Formularios en DSpace

Unidades académicas →

Envío de ítems

Describir → Describir → Adjuntar → Revisar → Licencia CC → Licencia → Finalizar

(*) Campos Obligatorios

Tipo de documento (*):
Seleccione el Tipo de Documento que desea cargar
- Seleccione un Tipo - ▾

Autor (*):
Autores de la obra
 + Agregar Otro

Título (*):
El título principal de la obra

Fecha de Publicación:
Fecha en la que la obra fue publicada en una revista, libro, etc. No debe confundirse con la fecha de entrega o defensa de una tesis, que debe cargarse en el campo Fecha de Presentación. Los valores posibles para este campo son días/mes/año o día/mes/año.
 ▾

Gonzalo Villarreal Salir
Editar perfil
Mis documentos

Submission progress bar

Selector simple (select)

Texto de una línea (text)

Texto multi línea (textarea)

Calendario

Formularios en DSpace

Fecha de Publicación:

Fecha en la que la obra fue publicada en una revista, libro, etc. No debe confundirse con la fecha de entrega o defensa de una tesis, que debe cargarse en el campo Fecha de Presentación. Los valores posibles para este campo son día/mes, mes/año o día/mes/año.

3 marzo 99
Día Mes Año

Entidad de origen (*):

Institución donde el documento fue creado

Universi|

Asociación de Radios Universitarias Nacionales Argentina (ARUNA) - Instituciones::Asociación de Radios Universitarias Nacionales Argentina (ARUNA)

Asociación de Universidades Grupo Montevideo - Instituciones::Asociación de Universidades Grupo Montevideo

Editorial de la Universidad Nacional de La Plata (EDULP) - Instituciones::Universidad Nacional de La Plata::Rectorado::Editorial de la Universidad Nacional de La Plata (EDULP)

Radio Universidad Nacional de La Plata - Instituciones::Universidad Nacional de La Plata::Rectorado::Radio Universidad Nacional de La Plata

Radio Universidad Nacional del Sur - Instituciones::Universidad Nacional del Sur::Radio Universidad Nacional del Sur

Red de Universidades con Carreras en Informática (RedUNCI) - Instituciones::Red de Universidades con Carreras en Informática (RedUNCI)

Universidad Nacional de La Plata - Instituciones::Universidad Nacional de La Plata

Universidad Nacional del Sur - Instituciones::Universidad Nacional del Sur

texto con sugerencia (suggest)

+ Agregar Otro

Palabras clave:

Palabras o terminos extraídos del documento

Formularios en DSpace

```
<input-forms>
```

```
<!-- Map of Collections to Form Sets -->  
<form-map>  
  <name-map collection-handle="default" form-name="traditional" />  
  ...  
</form-map>
```

Mapeo de formularios con colecciones

```
<!-- Form Set Definitions -->  
<form-definitions>  
  <form name="traditional">  
    ...  
  </form>  
  ...  
</form-definitions>
```

Definición de formularios

```
<!-- Name/Value Pairs used within Multiple Choice Widgets -->  
<form-value-pairs>  
  <value-pairs value-pairs-name="common_iso_languages" dc-term="language_iso">  
    ...  
  </value-pairs>  
  ...  
</form-value-pairs>
```

Pares de valores para campos de selección múltiple

```
</input-forms>
```

Formularios en DSpace

Cada name-map dentro de los form-map permite asociar un formulario a una colección particular

Como siempre, existe una colección default (para el caso donde no se especifique una colección)

```
<?xml version="1.0"?>
<!DOCTYPE input-forms SYSTEM "input-forms.dtd">

<input-forms>

  <form-map>
    <name-map collection-handle="default" form-name="traditional" />
  </form-map>

  <form-definitions>
    <form name="traditional">
      <page number="1">
        <field>
```

Formularios en DSpace

Para definir un formulario, debe especificarse una etiqueta *form* dentro de *form-definitions*, y asignarle un nombre único

Cada *form* define un conjunto de *pages*, y cada *page* posee un número (*number*) y define un conjunto de campos (*fields*)



Formularios en DSpace

De qué se compone un **campo** de un formulario?

Campos obligatorios

- dc-schema: Esquema empleado (ej. Dublin Core, **dc**)
- dc-element: Nombre del elemento Dublin Core (ej. contributor)
- label, hint
- input type: onebox (text), twobox (repeatable text), textarea, name (Last name, First name), date (calendar), series, qualdrop_value (qualified value), list

Formularios en DSpace

Campos opcionales

- repeatable (true/false): para permitir múltiples valores para el campo
- dc-qualifier: calificador del elemento Dublic Core especificado (ej. contributor.advisor). Si no se especifica este campo, quiere decir que se trata de un elemento sin calificador
- required : el campo es obligatorio `<required>mensaje</required>`
- visibility: indica el *scope* en el cual se mostrará (si no se especifica, se mostrará en todos los *scopes*)
 - workflow, ideal para ocultar (por ej.) campos complejos a los usuarios
 - submit: sólo en el envío inicial (y no en el workflow). Aquí podemos también definir el comportamiento: read-only, full-hidden
`<visibility otherwise="readonly">workflow</visibility>`

Más información en

<https://wiki.duraspace.org/display/DSDOC3x/Submission+User+Interface#SubmissionUserInterface-TheStructureofinput-forms.xml>

Formularios en DSpace

```
<form name="traditional">

  <page number="1">
    <!-- Tipo de documento: determina la carga de otros metadatos -->
    <field>
      <dc-schema>dc</dc-schema>
      <dc-element>type</dc-element>
      <label>Tipo de documento (*)</label>
      <input-type value-pairs-name="common_types">dropdown</input-type>
      <hint>Seleccione el Tipo de Documento que desea cargar</hint>
      <required>Debe seleccionar un tipo de documento, ya que el proceso de
    </field>
    <field>
      <dc-schema>sedici</dc-schema>
      <dc-element>subtype</dc-element>
      <label>Especifique el tipo de Artículo (*)</label>
      <input-type value-pairs-name="common_subtypes_articulo">dropdown</input-type>
      <hint>Seleccione el tipo de articulo que desea cargar</hint>
      <required>Debe seleccionar un subtipo de documento</required>
      <visibility>workflow</visibility>
      <type-bind>Articulo</type-bind>
    </field>
    <field>
      <dc-schema>sedici</dc-schema>
      <dc-element>subtype</dc-element>
      <label>Especifique el tipo de Tesis (*)</label>
      <input-type value-pairs-name="common_subtypes_tesis">dropdown</input-type>
      <hint>Seleccione el tipo de tesis que desea cargar</hint>
      <required>Debe seleccionar un subtipo de documento</required>
      <visibility>workflow</visibility>
      <type-bind>Tesis</type-bind>
    </field>
  </page>
</form>
```

Formularios en DSpace

DSpace (XMLUI) permite configurar campos dependientes del tipo de ítem (valor de dc.type)

Para ello, existe un elemento dentro de field, llamado type-bind.

```
<field>
  <dc-schema>sedici</dc-schema>
  <dc-element> identifier</dc-
element>
  <dc-qualifier>issn</dc-qualifier>
  <label>ISSN</label>
  <input-type>onebox</input-type>
  <hint>ISSN de la revista en la que
el artículo fué publicado</hint>
  <type-bind>Libro,
Artículo,Objeto de
conferencia</type-bind>
</field>
```

```
<field>
  <dc-schema>sedici</dc-schema>
  <dc-element>relation</dc-element>
  <dc-qualifier>journalTitle</dc-
qualifier>
  <label>Título de la serie</label>
  <input-type>onebox</input-type>
  <hint>Título de la serie en la que el
documento fué publicado</hint>
  <type-bind>Articulo</type-bind>
</field>
```

Formularios en DSpace

Pares de valores: permiten especificar una lista de valores, para que el usuario seleccione uno

```
<!--  
<file  
  <value-pairs value-pairs-name="common_types" dc-term="type">  
    <pair>  
      <displayed-value>- Seleccione un Tipo -</displayed-value>  
      <stored-value></stored-value>  
    </pair>  
    <pair>  
      <displayed-value>Articulo</displayed-value>  
      <stored-value>Articulo</stored-value>  
    </pair>  
    <pair>  
      <displayed-value>Tesis</displayed-value>  
      <stored-value>Tesis</stored-value>  
    </pair>  
    <pair>  
      <displayed-value>Libro</displayed-value>  
      <stored-value>Libro</stored-value>  
    </pair>  
    <pair>  
      <displayed-value>Objeto Físico</displayed-value>  
      <stored-value>Objeto Físico</stored-value>  
    </pair>  
    <pair>  
      <displayed-value>Objeto de aprendizaje</displayed-value>  
      <stored-value>Objeto de aprendizaje</stored-value>  
    </pair>  
  </value-pairs>  
</file>
```

o de ítems

Describir

→ Describir

→ A

Tipos Obligatorios

Selecione un documento (*):

Selecione el Tipo de Documento que desea

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

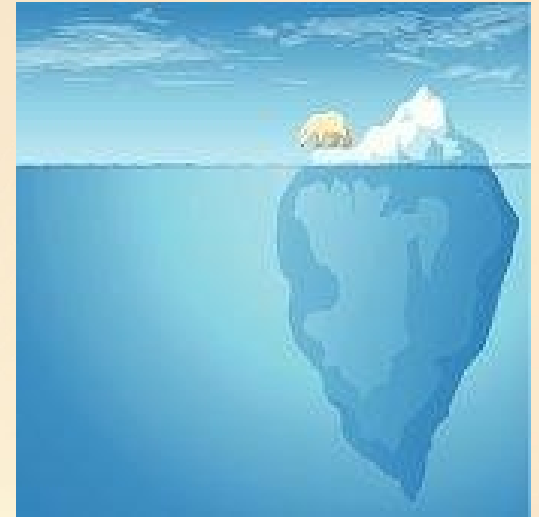
Selecione un Tipo -

Selecione un Tipo -

Selecione un Tipo -

Formularios en DSpace

- Esto es sólo la punta del iceberg
- No es nuestro objetivo conocer todos los parámetros de configuración de cada campo de un formulario, o de cada *step* de un *submission process*
 - (además de ser mucha información, evoluciona constantemente... no parece ser una buena estrategia)
- Pero necesitamos saber qué posibilidades nos brinda la herramienta, y cuál es el mejor lugar para introducir una modificación o una nueva característica



Fin...

Preguntas?



Gracias

<http://sedici.unlp.edu.ar>