DOCS

Dryad Repository: High-level documentation

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56

info@mire.be
www.atmire.com

DOCS

1. DISCLAIMER

This documentation contains an overview of the implemented modules, with their respective additions to the existing configuration files, changes to the source code, implications of the changes and a schematic overview of these changes for a high-level overview.

2. INTRODUCTION

High-level documentation outlining the changes made to the DSpace source code and a schematic representation or diagram outlining.

The following general initial changes were made to the SVN repository once it was setup.

API Directory

Since changes to the API were required, we have included an api directory in the modules section on the SVN. We have also altered the module's pom.xml, the main dspace's pom.xml and the xmlui's pom.xml to automatically include the code changes from the api.

Removal of Unchanged Source Projects

The source trees for the following projects were excluded from the repository.

dspace-api, dspace-xmlui, dspace-jspui, dspace-lni, dspace-sword.

This alteration will ease and shorten the build process, eliminate duplicate code in both your repository and in general for DSpace, allow you to only need to maintain those sections of code you actually alter under "modules" within your SVN.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56

info@mire.be
www.atmire.com

DOCS

3. DATASET EMBARGO

Introduction

NESCent wishes to have the capability to embargo datasets in DSpace for a specific amount of time. A Dataset is a DSpace Item and associated Bitstreams. Only the bitstreams (and all the bitstreams) within this item will be protected, while the metadata remains public at all times. By default, the Curators Group and the DSpace administrators will retrieve permissions to access the embargoed files. The Curators Group is a single group which is defined uniquely in the DSpace repository. The search and browse functionality will still be available on the embargoed item's metadata and their full-text when indexed. The submitter will be allowed to choose the embargo from a predefined list of available ranges.

Changes required to the source code

UpdateEmbargoedBits

- Class created to be included as a daily cron job
  - This is a class which will be used to automatically update the permissions of the bitstreams when the embargo date passes. The embargoes of all the items in the repository will be checked. When the embargo has expired the anonymous group will be granted read permissions to the bitstreams.
  - It is advised to execute this class (without any arguments) on a daily basis, similar to the filter-media. The command to execute is given below.
  - bin/dsrun org.dspace.authorize.UpdateEmbargoedBits

InstallItem

- A method reference has been included to configure the correct bitstream permissions based on the date.embargoedUntil field of the corresponding item.
  - In the installItem methode, once the item is ready to be installed and archived, BitstreamUtil.setBitstreamEmbargos() is called to configure the correct permissions after the submission or during the import of a new item.

BitstreamUtil

- Utility class for configuring permissions based on the date.embargoedUntil field
  - All bitstreams of the given item will be updated according to the contents of the date.embargoedUntil field in the setBitstreamEmbargos method.
  - The setBitstreamEmbargos method is called in the UpdateEmbargoedBits cron job, the FlowItemUtils.processEditEmbargo() method, and the InstallItem.installItem() method.

FlowItemUtils

- method processEditEmbargo
  - This method has been included to process changes to the embargoes of an existing item. This will read the embargo information configured in the edit item embargo page, store it in the item metadata, and update the bitstream permissions based on the given information.
- method fleshOut
  - Helper method to convert a date to the YYYY-MM-DD format.

EditItemEmbargoForm

- New class to offer a form for configuring the embargo
  - This class offers a user interface for the user to enable/disable/edit the embargo of this item.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

EditItemBitstreamsForm

- Link to EditItemEmbargoForm
  - Includes a link to the EditItemEmbargoForm to allow the embargo to be edited.

EditItemMetadataForm

- Link to EditItemEmbargoForm
  - Includes a link to the EditItemEmbargoForm to allow the embargo to be edited.

EditItemStatusForm

- Link to EditItemEmbargoForm
  - Includes a link to the EditItemEmbargoForm to allow the embargo to be edited.

admistrative.js

- method doEditItem
  - Includes code to support the EditItemEmbargoForm link
- method doEditItemStatus
  - Includes code to support the EditItemEmbargoForm link
- method doEditItemBitstreams
  - Includes code to support the EditItemEmbargoForm link
- method doEditItemMetadata
  - Includes code to support the EditItemEmbargoForm link
- method doEditItemEmbargo
  - New method to display the EditItemEmbargoForm and process its' data after submitting the information

dspace/modules/xmlui/src/main/resources/aspects/Administrative/

sitemap.xmap

- Transformer
  - Includes the EditItemEmbargoForm transformer to support the URL which will be resolved in displaying the appropriate form.

messages.xml

- Keys
  - All displayed values are included in the messages.xml file with keys starting with xmlui.administrative.item.EditItemEmbargoForm.

Implications of the changes

The edit item page has been extended to offer a user-interface for configuring the embargo of the corresponding files. An example overview of this configuration page is displayed below.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

The submission of a new item using the default DSpace submission or the default DSpace Item Import or any other process using the InstallItem class for creating new items will automatically process information stored in the date.embargoedUntil field to define the required permissions. This date.embargoedUntil field should contain a date in the YYYY-MM-DD format. This format can be achieved in the submission with the standard date input type. An example of such a date is 2008-12-31 or 2009-01-01.

Every time the UpdateEmbargoedBits class is executed all items are checked to update the bitstreams' permissions according to the actual embargo information. This implies that the script must be executed daily to automatically update the permissions on the actual day the embargo has passed, but if for some reason a day has been skipped, the file permissions will still automatically be updated for all embargo's with a date in the past.

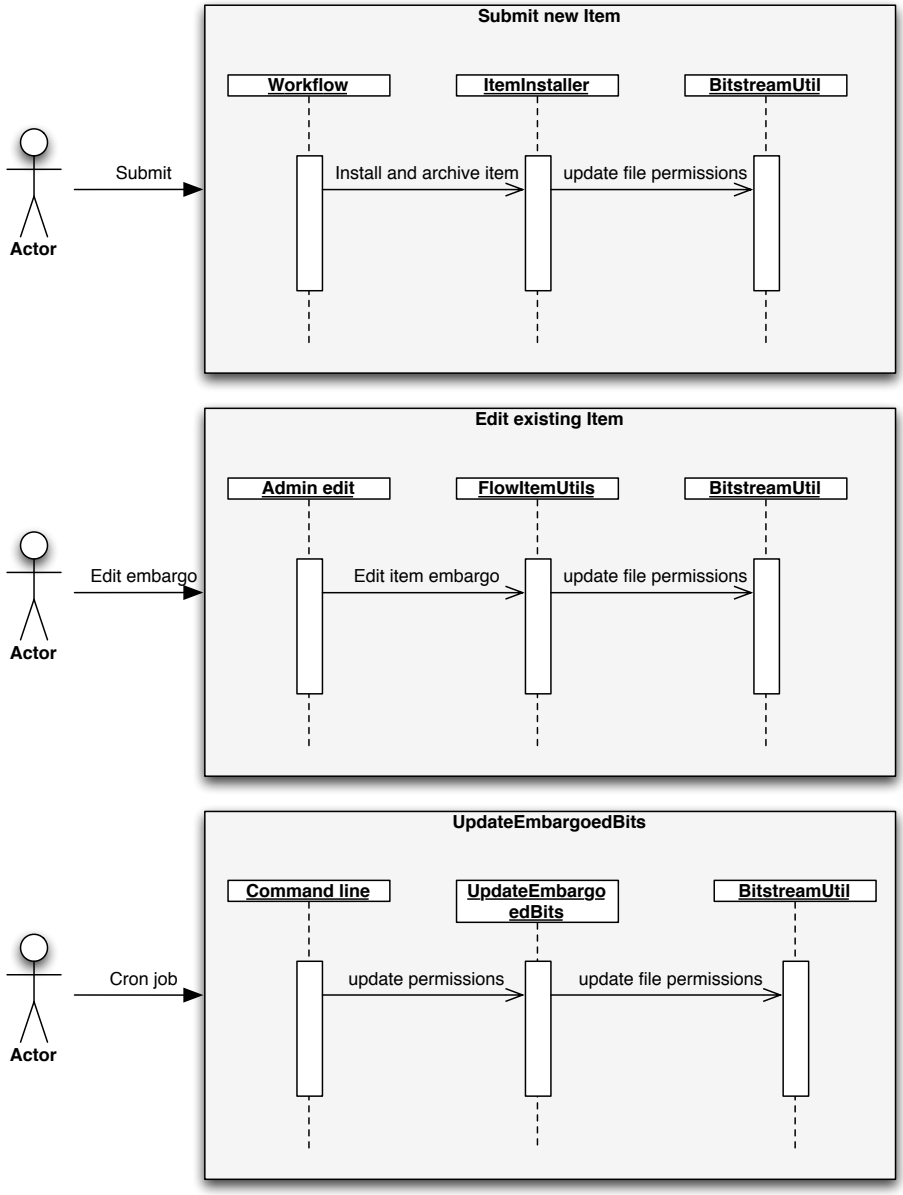
If the permissions of the bitstreams are updated, the actual bitstream will become searchable or no longer searchable according to "9. Full-text Indexing" since the update of the search-index for the item is executed after the item has been updated.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56

info@mire.be
www.atmire.com

DOCS

Schematic representation



4. DATASET SECURITY

Introduction

NESCent requires the capability in DSpace to secure discovery and viewing of dark copies of bitstreams. Dark bitstreams are bitstreams with an embargo applied to them as stated in “Dataset Embargo” or bitstreams which are not available to the current user. All metadata and links to datasets will still be displayed, but the links to the bitstreams are removed unless the current user has permissions to access the bitstreams. These changes do not reflect the browse or search pages since no direct links to the bitstreams will be offered there.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

Changes required to the source code

dspace.cfg

- useDarkBits
  - a parameter “useDarkBits” has been added to the dspace.cfg to enable or disable hiding the Bitstreams.
  - “useDarkBits = true” has to be added to the dspace.cfg file in order for this module to be enabled.

ItemAdapter

- Included the DSpace context in the ItemAdapter
  - The context is required to define whether the current user has permissions to download a specific file.
  - A new constructor has been added to include the context in the ItemAdapter.
  - If the previous constructor (without the context) is used, all bitstreams will be displayed. All references to this class have been updated to use the new constructor however.
- renderFileSection method
  - The renderFileSection will only include a bitstream in the METS file when either “useDarkBits” is not included in dspace.cfg, or “useDarkBits” is set to “false” in dspace.cfg, or the current user has permissions to download the specific file.
  - If the user doesn’t have access to any of the files in a bundle, the bundle will not be included at all in the generated METS file to avoid incorrect file sections in the user interface.

DSpaceMETSGenerator

- resolveAdapter method
  - The context is included as a parameter in the ItemAdapter constructor.

FlowAuthorizationUtils

- processEditPolicy method
  - The context is included as a parameter in the ItemAdapter constructor.

Implications of the changes

Each item display page is rendered from a METS document. This METS document contains the item’s metadata and files. An example of a file section for an item is given here:

```
<mets:fileSec>
<mets:fileGrp USE="CONTENT">
 <mets:file SIZE="153074" GROUP_ID="group_file_324" CHECKSUM="91be273cd97dc4d2774de16d0d139122"
MIMETYPE="image/jpeg" CHECKSUMTYPE="MD5" ID="file_324">
   <mets:FLocat LOCTYPE="URL" xlink:href="/dspace/bitstream/handle/10255/dryad.161/Peli_occiP0535_L.jpg?
sequence=1" xlink:title="Peli_occiP0535_L.jpg" xlink:type="locator"/>
 </mets:file>
</mets:fileGrp>
<mets:fileGrp USE="LICENSE">
  <mets:file SIZE="790" GROUP_ID="group_file_327" CHECKSUM="e0983a7878404d9913981a323b01b50b"
MIMETYPE="text/plain" CHECKSUMTYPE="MD5" ID="file_327">
      <mets:FLocat LOCTYPE="URL" xlink:href="/dspace/bitstream/handle/10255/dryad.161/license.txt?
sequence=2" xlink:title="license.txt" xlink:type="locator"/>
 </mets:file>
</mets:fileGrp>
</mets:fileSec>
```

If the current user doesn’t have any permissions to view the file, the file section will become:

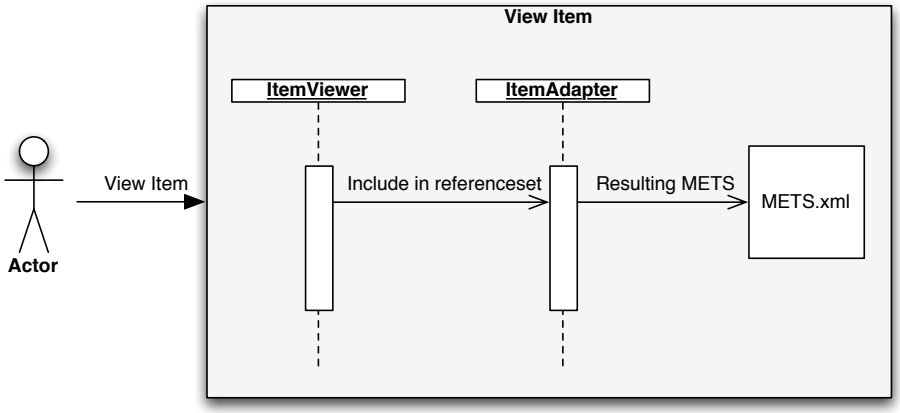
@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

```
<mets:fileSec>
<mets:fileGrp USE="LICENSE">
  <mets:file  SIZE="790"  GROUP_ID="group_file_327"  CHECKSUM="e0983a7878404d9913981a323b01b50b"
MIMETYPE="text/plain" CHECKSUMTYPE="MD5" ID="file_327">
      <mets:FLocat  LOCTYPE="URL"  xlink:href="/dspace/bitstream/handle/10255/dryad.161/license.txt?
sequence=2" xlink:title="license.txt" xlink:type="locator"/>
 </mets:file>
</mets:fileGrp>
</mets:fileSec>
```

The private files will therefor be removed from the view.

Schematic representation



5. SEARCH SYSTEM CUSTOMIZATIONS

Introduction

NESCent requires that the capabilities of the DSpace search system should be extended.

1. Search results should show “Publication Items” as first level, with the related authors/description and then “Dataset Items” listed at a second level within the relevant publication.
2. Advanced Search Form Customization and Index configuration:
   A. Publications should be searchable via DOI.
   B. Advanced search should also contain other fields relevant to Dryad metadata.
3. Publications should be searchable via OpenURL.
4. Full-text indexing

6. PUBLICATION ITEMS

Introduction

Project will provide Dryad Grouping requirements by customizing XMLUI theming layer. The Grouping will be accomplished by evaluating and parsing METS/DIM metadata for dc.relation.hasPart references with Publication METS/DIM metadata. Publication/Dataset Items must adhere to following requirements:

A. dc.relation.hasPart and dc.relation.isPartOf references to the handles of the Publications/Datasets.

B. dc.type Identifies specific and local Dryad “Publication” and “Dataset” types.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

Changes required to the source code

dspace.cfg

- search.index.13 = kind:dc.type
  - An extra searchindex to be able to search a specific kind of document has been added to the dspace.cfg
  - This parameter is required to be able to filter the document type automatically.
  - This parameter has to be called “kind” and no other search parameter can be given the same name. Other search parameters referring to the type can be present however.

DSIndexer

- Included the boolean “reindex”
  - The boolean “reindex” has been included in some of the methods. This boolean is used to differentiate between a command-line full reindex or a regular update of a single or multiple items through the userinterface. This difference is required to ensure fast reindexes and avoid reindexing the same item multiple times due to the changes explained below.
  - The boolean has been included in a few methods, the original method is kept and a duplicate of the method including the extra parameter is added.
  - The effected methods are:
    - public static void indexContent(Context context, DSpaceObject dso, boolean force)
    - public static void updateIndex(Context context, boolean force)
    - private static void buildDocument(Context context, Item item, Term t)
- buildDocument method
  - The buildDocument method normally creates a lucene document containing all indexed field of the given item. This has been extended in case the given item is a “Publication”. All “Datastream” items referenced through the relation.haspart metadata fields will be included in the lucene document of the “Publication”. Determining whether the current item is a “Publication” is based upon the metadata field dc.type which should contain the value “Article” (as the first metadata value in this field). The entire contents of the Datastream’s lucene document will be included in the Publication’s lucene document except for the metadata fields which will be indexed in the “kind” field added as a separate index field. These have been excluded to be able to search specifically for all Publications without any Datastreams as explained in the DSQuery part below.
  - If the “reindex” (see above) parameter is set to false, and the current item is a Datastream, the corresponding Publication will be reindexed as well. This is required since the Publication contains the Dataset’s lucene document, and therefor the Publication’s lucene document needs to be updated whenever the Dataset’s lucene document needs to be updated. The Publication is determined by the first occurrence in the relation.ispartof metadata field.
- buildDoc method
  - This method is created to avoid duplication of code. The code available here was originally part of the buildDocument method. It iterates over the metadata fields to be indexed, and includes the item’s metadata in the lucene document.
  - If the boolean skipType is set to true, the item’s metadata fields will not be included in the “kind” field of the lucene document (see above).

DSQuery

- doQuery method

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

- This method will explicitly change every search query in the back-end. All queries will include the “kind:article” query. This implies only Publications will be retrieved. Due to the fact that the Datastream’s metadata is included in the Publication’s lucene document, this adapted query still retrieves the Datastreams as well. But the datastreams will be retrieved through the corresponding Publication to allow for the Publication to be displayed as the first item and the Datastreams to be displayed as ‘children’ of the Publication.
- querystring = "(" + querystring + ") AND kind:article";

AbstractSearch

- buildSearchResultsDivision method
  - In order to display the Datastreams in the search-results, the Datastream items are resolved from the relation.haspart metadata field. These items are included in the referenceSet as a separate reference to be handled in the Dryad.xsl layout file.

Dryad.xsl

- itemSummaryList-DIM block
  - In order to display the Datastreams in the search-results, the itemSummaryList-DIM block has been included in the Dryad.xsl file. This xsl will check the contents of the type metadata field to determine the layout of the displayed record. We have currently included the title for each item, and the author, publisher and publication date if the item is a Publication. The class of the record is also dependent on the item’s type. We have automatically assigned the class “publication” and “dataset”. Publications will also receive the class “odd” while datasets receive the class “even”.

style.css

- div.publication
  - This block currently doesn’t contain any special style information.
- div.dataset
  - This block currently contains “margin-left:50px;” to display the difference between the dataset and the publication

Implications of the changes

All searches will now only retrieve Publications. But since both the metadata of the Datasets is included in the Publication’s lucene document, and the Datasets are displayed as part of the corresponding Publication, the Datasets will still be included in the search results. This methods avoids any issues of duplicate results which might be included in the results set.

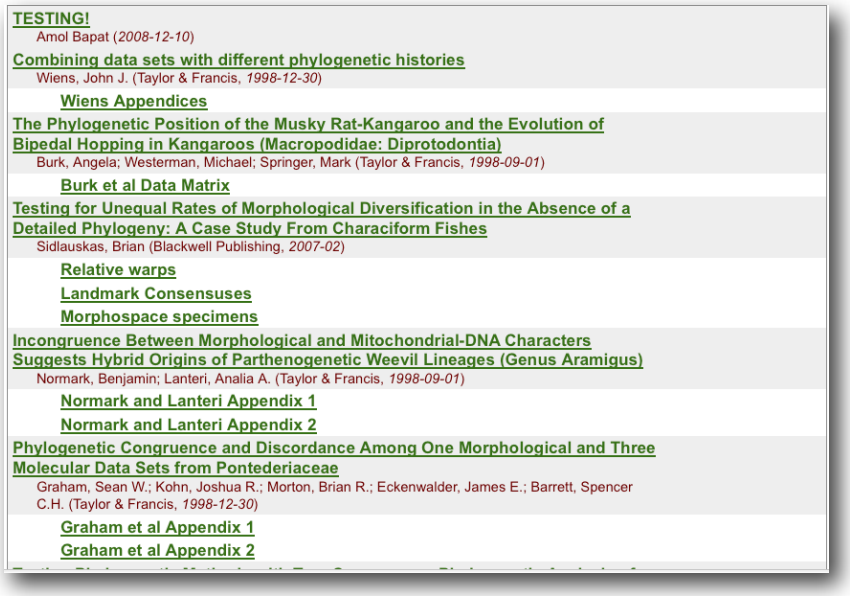
This approach does require each Dataset to have a corresponding Publication present in the repository, and with correct interlinking between the Dataset and the Publication. We currently define whether an item is either a Publication or a Dataset based on whether the type metadata field contains “article” or not. This might need to be changed later on if either the query should be performed through searching for multiple values of the type field, or by searching for keywords which may not occur in the type field.

The resulting search listing is displayed below:

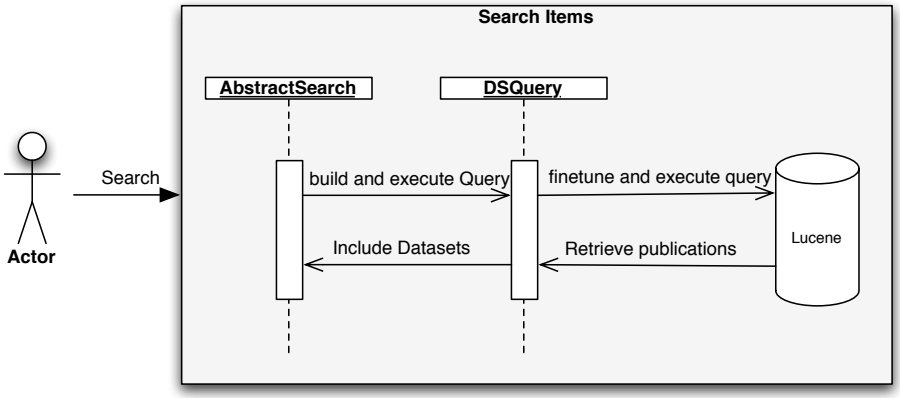
@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56

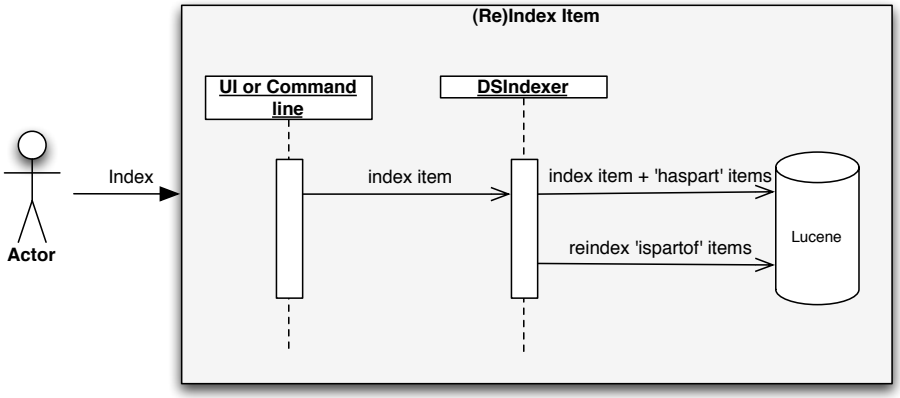
info@mire.be
www.atmire.com

DOCS



Schematic representation





@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

7. ADVANCED SEARCH FORM CUSTOMIZATION AND INDEX CONFIGURATION

Introduction

Project will provide customized XMLUI Advanced Search for specified terms to support Dryad required metadata fields (including DOI provided in dc.identifier metadata fields). This will require clarification of:

A. Appropriate DC metadata fields to be indexed on specific Lucene terms.

B. A detailed specification of the fields and their corresponding formatting.

Changes required to the source code

dspace.cfg

- search.advanced.display.1 = title
    - Extra fields are added to the dspace.cfg file to define which metadata fields should be displayed in the advanced search. All fields included here will be displayed, and this is independent of the search.index.x fields. The value of each of these fields is the search field's name. Internationalization of the fields displayed in the advanced search is still performed through the "xmlui.ArtifactBrowser.AdvancedSearch.type_author" field where "author" can be replaced with a different field name to include new fields.
    - The fields should start with search.advanced.display.1 and continue with search.advanced.display.2 etc. The actual order of the displayed fields is determined based on the numbers in the configuration ("Full text" is the first field displayed in the searhcch, search.advanced.display.1 is the second field, search.advanced.display.2 is the third field, …)
    - If no search.advanced.display field is available, all search fields in search.index.x will be displayed.

AdvancedSearch

- buildTypeField method
    - The buildTypeField method reads the search.advanced.display.x fields from the dspace.cfg file. If these fields are not present, the search.index.x fields will be used instead.
    - The "ANY" term will be included automatically and cannot be disabled.

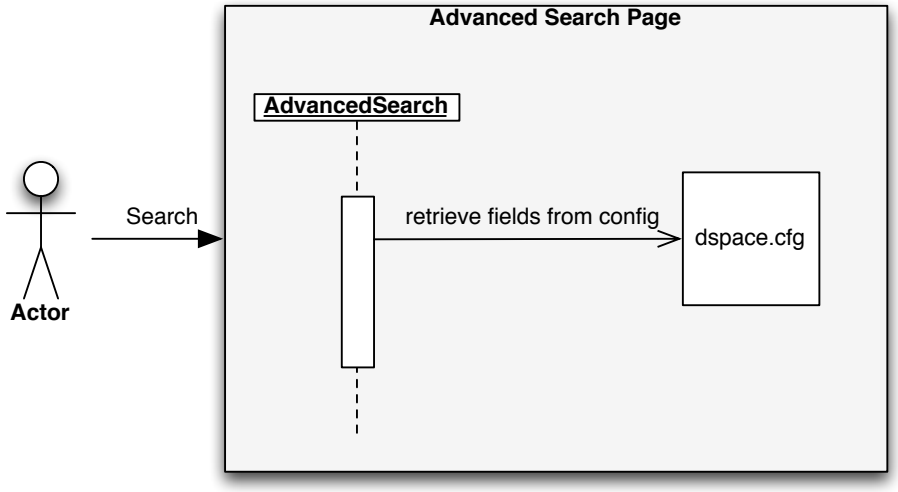
Implications of the changes

The displayed fields in the advanced search page can be configured explicitly through the dspace.cfg file. The names used in the "search.advanced.display.x" fields should be a subset of the names used in the "search.index.x" fields. If names in the "search.advanced.display.x" fields are used which do not correspond with a name in the "search.index.x" fields, search queries will be generated which will resolved in an empty result set.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

Schematic representation



8. OPENURL

Introduction

The OpenURL implementation in DSpace 1.5.x JSPUI is specific to the initial OpenURL 0.1 implementation. OpenURL COinS require support for OpenURL 1.0. Initial implementation of 1.0 is primarily the introduction of new http parameters to represent Context Objects within the request.

The OpenURL feature in DSpace translates the URL to a Lucene query to be able to retrieve items by its OpenURL. Exposing OpenURL in the XMLUI is comprised of three sub-projects:

A. Migrating OpenURL servlet into XMLUI cocoon Generator, mature to support

B. Adding customizations to theme to expose the same SFX OpenURL link and COinS.

C. Configuring Dryad to local SFX resolver services at NESCent/UNC.

Changes required to the source code

OpenURLReader

Current changes in the source-code involve the addition on a new class (*org.dspace.app.xmlui.cocoon.OpenURLReader)* under *[dspace-source]/dspace/modules/xmlui/src/main/java* and the addition of a copy of the *[dspace-source]/dspace/modules/xmlui/src/main/webapp/WEB-INF/sitemap.xml* to include the Reader into the default pipeline for processing requests.

The new class includes a port of the original OpenURL 0.1 codebase and the addition of new (but partial) support for the Dublin Core Community Profile of OpenURL 1.0.

Implications of the changes

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

Rather than dealing with various metadata formats possible within the OpenURL community, it is initially chosen to support the Dublin Core profile because it can remain orthogonal to the type of the Item being stored within DSpace.

rft_val_fmt=info:ofi/fmt:kev:mtx:dc

More can be seen in the Dublin Core OpenURL Profile Citation Guidelines:

http://dublincore.org/documents/dc-citation-guidelines/

And the Profile registered with the OpenURL community:

http://alcme.oclc.org/openurl/servlet/OAIHandler/extension?verb=GetMetadata&metadataPrefix=mtx&identifier=info:ofi/fmt:kev:mtx:dc

COinS within the DSpace XMLUI theme are implemented as in the following example:

<span class="Z3988" title="ctx_ver=Z39.88-2004&url_ctx_fmt=info:ofi/fmt:kev:mtx:ctx&rft_val_fmt=info:ofi/fmt:kev:mtx:dc&rft_id=10255/dryad.111&rft:title=Testing+Phylogenetic+Methods+with+Tree+Congruence%3A+Phylogenetic+Analysis+of+Polymorphic+Morphological+Characters+in+Phrynosomatid+Lizards&...">...

Within DSpace, the resolution of this COinS in the OpenURL service adheres to the following protocol:

- If a rft_id is present it will be used to resolve the DSpace Object directly via the HandleManager.
- Likewise, If a rft.identifier is present, it will be used to resolve the DSpace Object directly via the HandleManager.
- Finally, if neither above strategies are successful, the same full text search strategy applied within the legacy OpenURL resolver will be applied to the Item in a last ditch effort to resolve the resource. Thus, in DSpace *rft_id* are local stripped down versions of handle identifiers like the following

  rft_id=10255/dryad.111

Global identifiers will be any other valid dc.identifier present within that field. Thus:

dc.identifier.uri http://dx.doi.org/10.1080/106351598260806
dc.identifier.uri http://hdl.handle.net/10255/dryad.111

will lead to

rft.identifier=http%3A%2F%2Fdx.doi.org%2F10.1080%2F106351598260806
rft.identifier=http%3A%2F%2Fhdl.handle.net%2F10255%2Fdryad.111

And Thus be resolvable as well, other Dublin Core metadata will be likewise expressed inline as corresponding rtf.<element> query parameters.

Changes allow the use of OpenURL COinS within Dryad Item, Search Result and Browse Result pages to designate an appropriate query to resolve the item within an OpenURL resolver. Queries are based on OpenURL 1.0 Syntax and have the following structure:

http://dryaddata.org/openURL?ctx_ver=Z39.88-2004&url_ctx_fmt=info:ofi/fmt:kev:mtx:ctx&rft_val_fmt=info:ofi/fmt:kev:mtx:dc&rft_id=10255/dryad.111&rft:title=Testing+Phylogenetic+Methods+with+Tree+Congruence%3A+Phylogenetic+Analysis+of+Polymorphic+Morphological+Characters+in+Phrynosomatid+Lizards&...

The following table is a breakdown of the OpenURL 1.0 fields and their description.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56

info@mire.be
www.atmire.com

DOCS

Fixed Parameters (same across all repository Items)		
Parameter	**Value**	**Description**
ctx_ver	Z39.88-2004	The version of the CO (this is fixed)
ctx_enc	info:ofi/enc:UTF-8	The encoding URI (this is fixed)
url_ctx_fmt	info:ofi/fmt:kev:mtx:ctx	Format of the ContextObject.
rft_val_fmt	info:ofi/fmt:kev:mtx:dc	The encoding of the metadata values.
rfr_id	info:sid/dyraddata.org:dryad	Referrer Identifier (fixed for Dryad)
Item Specific Parameters (varies across repository Items)		
Parameter	**Value**	**Description**
rft_id	[Local Item Id]	The id of this Item (local handle)
rft.title	The+Items+Title.	Dublin Core title
rft.creator	...	Dublin Core creator
rft.[element]	...	Other Dublin Core Fields ...

Caveats: As the DSpace Lucene indexes used to search on specific fields in DSpace is not explicitly Dublin Core, the mapping between Lucene fields and Dublin Core fields is not explicit and perfect. An eventual solution to this may be to either implement separate fields within the Lucene Document that map directly to the Dublin Core set used in the Community Profile, or to maintain an entirely separate set of search indexes for OpenURL support.

9. FULL-TEXT INDEXING

Introduction

Also for bitstreams that are dark or under embargo (see 1. Dataset Embargo & 2. Dataset security) full-text indexing will be disabled, even if full-text indexing is enabled in the DSpace configuration file.

Changes required to the source code

dspace.cfg

- useDarkBits
  - a parameter “useDarkBits” has been added to the dspace.cfg to enable or disable hiding the Bitstreams.
  - “useDarkBits = true” has to be added to the dspace.cfg file in order for this module to be enabled.

DSIndexer

- buildDocument method
  - The buildDocument method iterates over all bitstreams in the bundle “TEXT” included in the item. This iteration has been altered to only include the bitstream in the search index when the original bitstream, from which the “TEXT” version is the derivative, has anonymous read access. If the “useDarkBits” parameter is not present in dspace.cfg or useDarkBits is set to “false”, the bitstream will be included as well.
- findOriginalBitstream method
  - This method is included to define which bitstream is the original from which the given “TEXT” version is the derivative.

@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com

DOCS

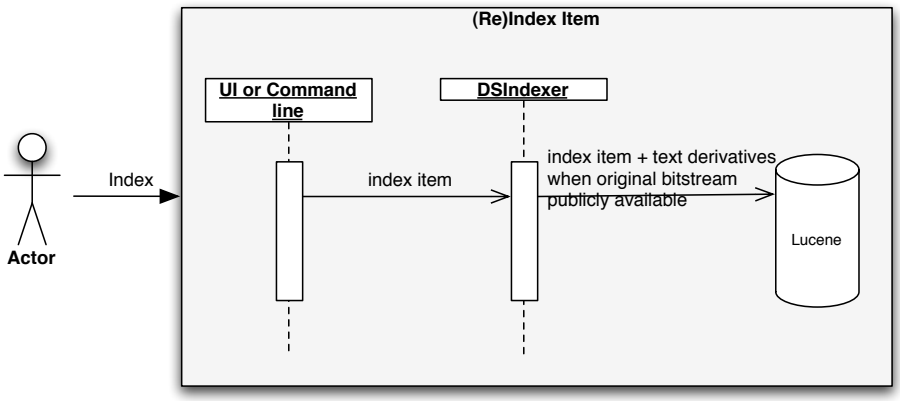
FlowAuthorizationUtils

- processEditPolicy method
  - The processEditPolicy method is executed when the policy of a DSpace object is altered. If the object is a Bitstream, the item must be reindexed to ensure the full text's availability is changed according to the new access permissions. This is achieved in the reindexDarkBitstream method.
- processDeletePolicies method
  - The processDeletePolicies method is executed when the policy of a DSpace object is deleted. If the object is a Bitstream, the item must be reindexed to ensure the full text's availability is changed according to the new access permissions. This is achieved in the reindexDarkBitstream method.
- reindexDarkBitstream method
  - The reindexDarkBitstream will reindex the given bitstream's item.

Implications of the changes

A private bitstreams plain text derivative will still be generated through the filter-media process, but the plain text will no longer be included in the item's lucene document. Since all indexed metadata from the dataset is included in the corresponding publication (see "Search System Customizations: Publication Items"), the publication's lucene document will automatically be updated to only include the bitstreams full-text when the bitstream is publicly available.

Schematic representation



@mire nv
Romeinse straat 18
B-3001 Heverlee
Belgium
☎ +32 2 888 29 56
info@mire.be
www.atmire.com