GeoNetwork User Manual

Release 2.10.4-0

GeoNetwork

July 06, 2015

Contents

1 Preface **3**
1.1 About this Project . . . 3
1.2 License Information . . . 3
1.3 Author Information . . . 4

2 Quick Start Guide **5**
2.1 Geographic Information Management for all . . . 5
2.2 Getting Started . . . 8
2.3 Viewing and Analysing the Data . . . 22
2.4 Adding a metadata record . . . 29
2.5 Uploading a New Record using the XML Metadata Insert Tool . . . 47
2.6 Metadata in Spatial Data Management . . . 48
2.7 New Features . . . 52
2.8 Installing the software . . . 56
2.9 Upgrading to a new Version . . . 64

3 Administration **65**
3.1 System configuration . . . 65
3.2 Authentication . . . 75
3.3 OGC CSW server configuration . . . 81
3.4 Advanced configuration . . . 86
3.5 User and Group Administration . . . 101
3.6 Localization . . . 106
3.7 System Monitoring . . . 108

4 Managing Metadata **111**
4.1 Templates . . . 111
4.2 Ownership and Privileges . . . 113
4.3 Import facilities . . . 116
4.4 Export facilities . . . 123
4.5 Status . . . 125
4.6 Versioning . . . 128
4.7 Harvesting . . . 134

4.8 Formatter . . . 188
4.9 Processing . . . 191
4.10 Fragments . . . 193
4.11 Schemas . . . 200

5 Features **203**
5.1 Multilingual search . . . 203
5.2 Search Statistics . . . 208
5.3 Thesaurus . . . 210
5.4 User Self-Registration Functions . . . 222

6 Glossary of Metadata Fields Description **227**

7 ISO Topic Categories **231**

8 Free and Open Source Software for Geospatial Information Systems **235**
8.1 Web Map Server software . . . 235
8.2 GIS Desktop software . . . 235
8.3 Web Map Viewer and Map Server Management . . . 235

9 Frequently Asked Questions **237**
9.1 HTTP Status 400 Bad request . . . 237
9.2 Metadata insert fails . . . 237
9.3 Thumbnail insert fails . . . 237
9.4 The data/tmp directory . . . 238
9.5 What/Where is the GeoNetwork data directory? . . . 239
9.6 The base maps are not visible . . . 239

10 Glossary **241**

Index **245**

Welcome to the GeoNetwork User Manual v2.10.4-0. The manual is a guide describing how to use the metadata catalog.

Other documents:

GeoNetwork Developer Manual

GeoNetwork User Manual (PDF)

CHAPTER 1

Preface

1.1 About this Project

This document provides guidelines to install, configure, use and customise the GeoNetwork opensource software.

The GeoNetwork project started out as a Spatial Data Catalogue System for the Food and Agriculture organisation of the United Nations (*FAO*), the United Nations World Food Programme (*WFP*) and the United Nations Environmental Programme (*UNEP*).

At present the project is widely used as the basis of Spatial Data Infrastructures all around the world.

The project is part of the Open Source Geospatial Foundation (*OSGeo*) and can be found at GeoNetwork opensource.

1.2 License Information

1.2.1 Software

The GeoNetwork opensource software is released under the *GPL* v2 license and can be used and modified free of charge.

1.2.2 Documentation

Documentation is released under a *Creative Commons* license with the following conditions.

You are free to Share (to copy, distribute and transmit) and to Remix (to adapt) the documentation under the following conditions:

- Attribution. You must attribute GeoNetwork opensource documentation to GeoNetwork opensource developers.
- Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.

With the understanding that:

- Any of the above conditions can be waived if you get permission from the copyright holder.
- Public Domain. Where the work or any of its elements is in the public domain under applicable law, that status is in no way affected by the license.

Other Rights. In no way are any of the following rights affected by the license:

- Your fair dealing or fair use rights, or other applicable copyright exceptions and limitations;
- The author's moral rights;
- Rights other persons may have either in the work itself or in how the work is used, such as publicity or privacy rights.

Notice: For any reuse or distribution, you must make clear to others the license terms of this work. The best way to do this is with a link to this web page.

You may obtain a copy of the License at Creative Commons Attribution-ShareAlike 3.0 Unported License

The document is written in reStructuredText format for consistency and portability.

1.3 Author Information

The documentation was written by the GeoNetwork opensource Developers and other community members. The basis for the reStructuredText based documentation is based on the work done by the GeoServer project and the Sphinx framework.

If you have questions, found a bug or have enhancements, please contact us through the GeoNetwork opensource Development Mailing list at geonetwork-devel@lists.sourceforge.net

CHAPTER 2

Quick Start Guide

2.1 Geographic Information Management for all

2.1.1 Introduction

What is GeoNetwork opensource

GeoNetwork opensource is a standard based and decentralised spatial information management system, designed to enable access to geo-referenced databases and cartographic products from a variety of data providers through descriptive metadata, enhancing the spatial information exchange and sharing between organisations and their audience, using the capacities and the power of the Internet. The system provides a broad community of users with easy and timely access to available spatial data and thematic maps from multidisciplinary sources, that may in the end support informed decision making. The main goal of the software is to increase collaboration within and between organisations for reducing duplication and enhancing information consistency and quality and to improve the accessibility of a wide variety of geographic information along with the associated information, organised and documented in a standard and consistent way.

Main Features

- Instant search on local and distributed geospatial catalogues
- Uploading and downloading of data, documents, PDF's and any other content
- An interactive Web map viewer that combines Web Map Services from distributed servers around the world
- Online map layout generation and export in PDF format
- Online editing of metadata with a powerful template system
- Scheduled harvesting and synchronisation of metadata between distributed catalogues
- Groups and users management
- Fine grained access control

Background and evolution

The prototype of the GeoNetwork catalogue was developed by the Food and Agriculture organisation of the United Nations (FAO) in 2001 to systematically archive and publish the geographic datasets produced within the organisation. The prototype was built on experiences within and outside the organisation. It used metadata content available from legacy systems that was transformed into what was then only a draft metadata standard, the ISO 19115. Later on, another UN agency, the World Food Programme (WFP) joined the project and with its contribution the first version of the software was released in 2003 and operational catalogues were established in FAO and WFP. The system was based on the ISO19115:DIS metadata standard and embedded the Web Map Client InterMap that supported Open Geospatial Consortium (OGC) compliant Web Map Services. Distributed searches were possible using the standard Z39.50 catalogue protocol. At that moment it was decided to develop the program as a Free and Open Source Software to allow the whole geospatial users community to benefit from the development results and to contribute to the further advancement of the software.

Jointly with the UN Environmental Programme (UNEP), FAO developed a second version in 2004. The new release allowed users to work with multiple metadata standards (ISO 19115, FGDC and Dublin Core) in a transparent manner. It also allowed metadata to be shared between catalogues through a caching mechanism, improving reliability when searching in multiple catalogues.

In 2006, the GeoNetwork team dedicated efforts to develop a DVD containing the GeoNetwork version 2.0.3 and the best free and open source software in the field of Geoinformatics. The DVD was produced and distributed in hard copy to over three thousand people. More recently, the OSGeo Live project has been developed with GeoNetwork and all the best Open Source Geospatial software available on a self-contained bootable DVD, USB thumb drive or Virtual Machine based on Xubuntu. The GeoNetwork community has been a part of this project and will continue to make sure the latest stable version of GeoNetwork is included. You can download the OSGeo-Live images from OSGeo Live website.

GeoNetwork opensource is the result of the collaborative development of many contributors. These include among others the Food and Agriculture organisation (FAO), the UN Office for the Coordination of Humanitarian Affairs (UNOCHA), the Consultative Group on International Agricultural Research (CSI-CGIAR), The UN Environmental Programme (UNEP), The European Space Agency (ESA) and many others. Support for the metadata standard ISO19115:2003 has been added by using the ISO19139:2007 implementation specification schema published in May 2007. The release also serves as the open source reference implementation of the OGC Catalogue Service for the Web (CSW 2.0.2) specification. Improvements to give users a more responsive and interactive experience have been substantial and include a new Web map viewer and a complete revision of search interface.

The use of International Standards

GeoNetwork has been developed following the principles of a Free and Open Source Software (FOSS) and based on International and Open Standards for services and protocols, like the ISO-TC211 and the Open Geospatial Consortium (OGC) specifications. The architecture is largely compatible with the OGC Portal Reference Architecture, i.e. the OGC guide for implementing standardised geospatial portals. Indeed the structure relies on the same three main modules identified by the OGC Portal Reference Architecture, that are focused on spatial data, metadata and interactive map visualisation. The system is also fully compliant with the OGC specifications for querying and retrieving information from Web catalogues (CSW). It supports the most common standards to specifically describe geographic data (ISO19139 and FGDC) and the international standard for general documents (Dublin Core). It uses standards (OGS WMS) also for visualising maps through the Internet.

Harvesting geospatial data in a shared environment

Within the geographic information environment, the increased collaboration between data providers and their efforts to reduce duplication have stimulated the development of tools and systems to significantly improve the information sharing and guarantee an easier and quicker access of data from a variety of sources without undermining the ownership of the information. The harvesting functionality in GeoNetwork is a mechanism of data collection in perfect accordance with both rights to data access and data ownership protection. Through the harvesting functionality it is possible to collect public information from the different GeoNetwork nodes installed around the world and to copy and store periodically this information locally. In this way a user from a single entry point can get information also from distributed catalogues. The logo posted on top each harvested record informs the user about the data source.

2.1.2 GeoNetwork and the Open Source Community Development

The community of users and developers of the GeoNetwork software has increased dramatically since the release of version 2.0 in December 2005 and the subsequent releases. At present, the user and developer mailing lists count well over 250 subscriptions each. Subscription to these lists is open to anyone interested. The archive of the mailing lists provides an important resource for users and can be freely browsed online. Members provide feedback within the community and provide translations, new functionalities, bug reports, fixes and instructions to the project as a whole. Building a self sustaining community of users and developers is one of the biggest challenges for the project. This community-building process relies on active participation and interaction of its members. It also relies on building trust and operating in a transparent manner, thereby agreeing on the overall objectives, prioritization and long term direction of the project. A number of actions have been taken by the project team to facilitate this process.

The foundation for the establishment of a GeoNetwork Advisory Board was laid at the 2006 workshop in Rome and membership criteria were defined.

A work plan is presented and discussed at the yearly GeoNetwork workshop; subsequently, the plan is maintained and updated throughout the year where needed. The project management team reports back to the advisory board about the reached developments and objectives during the annual workshops.

Two public Websites have been established. One focuses on the users of the software (http://geonetwork-opensource.org), while the other one is dedicated to the developers (http://trac.osgeo.org/geonetwork). Both can be updated and maintained online by trusted members of the community. They provide documentation, bug reporting and tracking, Wiki pages et cetera. A small part of the community connects through Internet Relay Chat (IRC) on a public `irc://irc.freenode.net/geonetwork` channel. But most interaction takes place on the user and the developer mailing lists.

During the 2006 workshop, the Project Advisory Board decided to propose the GeoNetwork opensource project as an incubator project to the newly founded Open Source Geospatial Foundation (OSGeo). This incubation process was successfully completed and the project websites were moved to servers accessible under the umbrella of the OSGeo foundation.

Source code is maintained in a publicly accessible code repository, hosted at an independent service provider, github.com that hosts thousands of FOSS projects. Developers and users have full access to all sections of the source code, while trusted developers can make changes in the repository itself. A special mailing list has been established to monitor changes in the code repository. This "commit mailing list" delivers change reports by email to its subscribers.

The documentation is written in reStructuredText format using the Sphinx framework to ensure versioning and support of multiple output formats (e.g. HTML and PDF).

2.2 Getting Started

Please make sure you have opened the home page of the GeoNetwork based catalogue.

If you installed the software on your local machine and started it, the default URL is http://localhost:8080/geonetwork

There are many different ways to search the catalogue for maps and other geographic data. This guide will introduce you to the most popular search methods: default, advanced and by category. Whichever search you choose, remember that you will see results based on your privileges and assigned work group (*Ownership and Privileges*).

Note: The term *data* in this application refers to datasets, maps, tables, documents, etc. that are linked to the metadata of a specific record.

2.2.1 Default Search

The default search allows you to search text within the entire record, such as keywords of the metadata and/or geographic location.

Free text search. *Type* a search term in the *What?* field. You can type anything here (free text). You can use quotes around text to find exact combinations of words.

Text and operators (and, or, not) are not case sensitive.

Geographic search. For the geographic search, two options are available for selecting a particular region to limit the search:

You can select a **region** from a predefined list;

You can select your own *area of interest* in a more interactive way. A small global map is shown on the screen from which you can drag and drop the frame of your location area. Just click on the button on the upper right of the map screen.

Perform search. Both types of search, free text search and geographic search can be combined to restrict the query further.

Click the *Search* button to proceed and show the results.

2.2.2 Searching by Categories

An additional way to search data within the GeoNetwork database, from the home page, is searching by **Category**. A list of categories is provided to the user to identify data at a more generic level: **Applications**, **Audio/Video**, **Case study and best practises**, **Conference proceedings**, **Datasets**, **Directories**, **Interactive resources**, **Maps and graphics**, **Other information resources**, **Photo**.

To search only for maps, click on **Maps and Graphics**. A list of maps will be displayed from which you may view details of every single map; just clicking on the **Metadata** button of the map you wish to review.

Figure 2.1: *The free text field.*

Figure 2.2: *The region field*

Figure 2.3: *Interactive Area Of Interest map*

Figure 2.4: *The Search button*

Figure 2.5: *Search by Category*

2.2.3 Advanced Search

The advanced search option works similarly to the default search. However, you can be more specific in your search criteria as it offers different elements to look for data, each of them focusing one of the following aspects: *What?, Where?, When?*

To perform an **advanced search**, from the home page **click Advanced** just below the search bottom.

In the **WHAT?** section the elements are all related to the data content. Through them, in addition to searching only free keywords in the entire metadata content, you can also search directly in the title or abstract fields and add more keywords to customise your search further. You can also specify the level of accuracy you wish to reach in performing your search.

- To search by **Title, Abstract, Free Text, or Keyword(s)** type any text into the respective field. You can enter information in one or multiple field(s). If you do not want to search by a given field, simply leave it blank;
- You can choose the **accuracy of your search**, in terms of spelling words, from **Precise** = 1 to **Imprecise** = 0.2, through 3 more consecutive steps which are equal to 0.8, 0.6, 0.4.

The **WHERE?** parameters, which are related to the spatial extent, allow you, as in the default search, either to select your own area of interest or to select a predefined region from the drop-down list. In this section you can also type the geographic coordinates of a specific location that is not available from the above list.

- To select **your own area of interest**, drag and drop the frame of your area on the global map using the appropriate tool on the bottom left of the map screen;
- To use **free coordinates**, type the lat-long geographic references in the appropriate fields around the map screen, without any limitation of decimal figures;
- To use the coordinates of a **predefined region**, select the region from the drop-down list.

Whatever type of geographic search you decide to perform, in the **Spatial search type** field, you can choose from different options: **is**, **overlaps**, **encloses**, **is fully outside of**. If you use this field, be cautious as this limits your output data as follows:

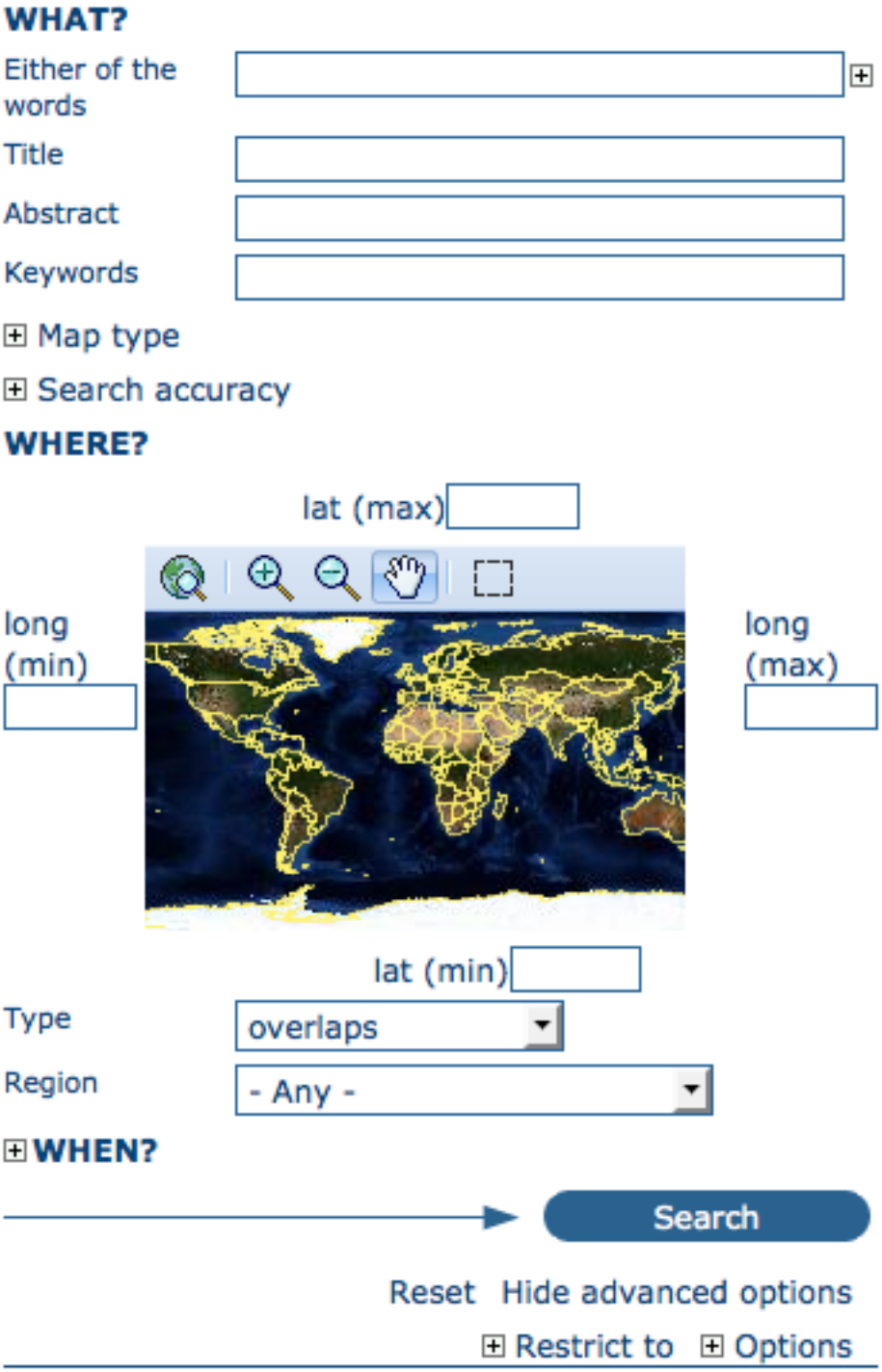


Figure 2.6: *Advanced search options*

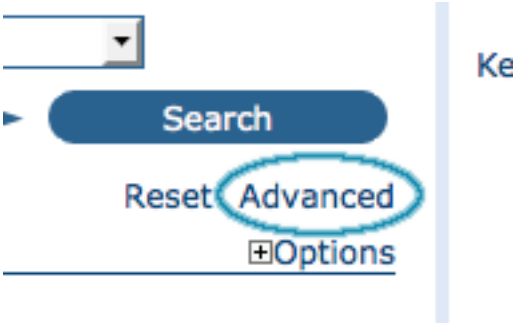


Figure 2.7: *Show advanced search options*

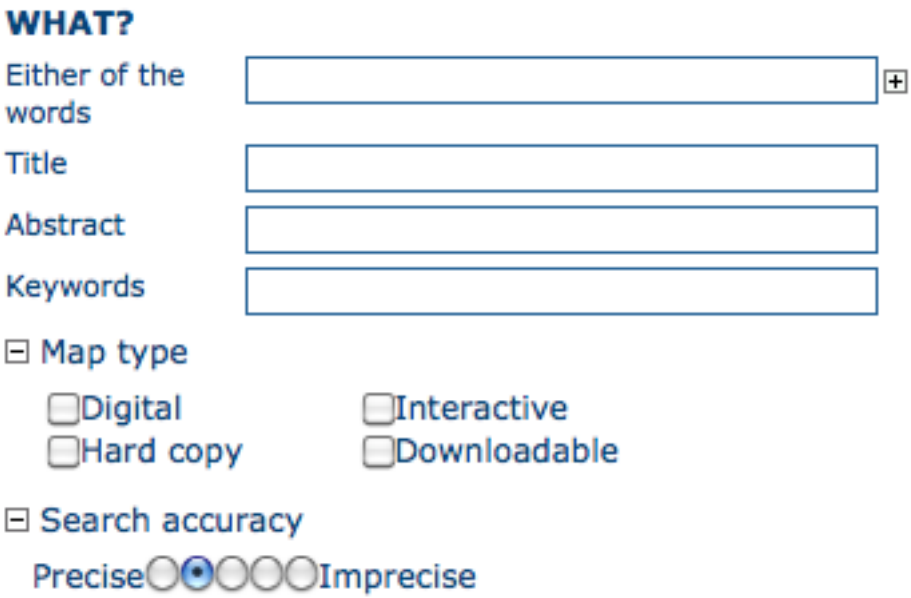


Figure 2.8: *"What" section in the Advanced search*

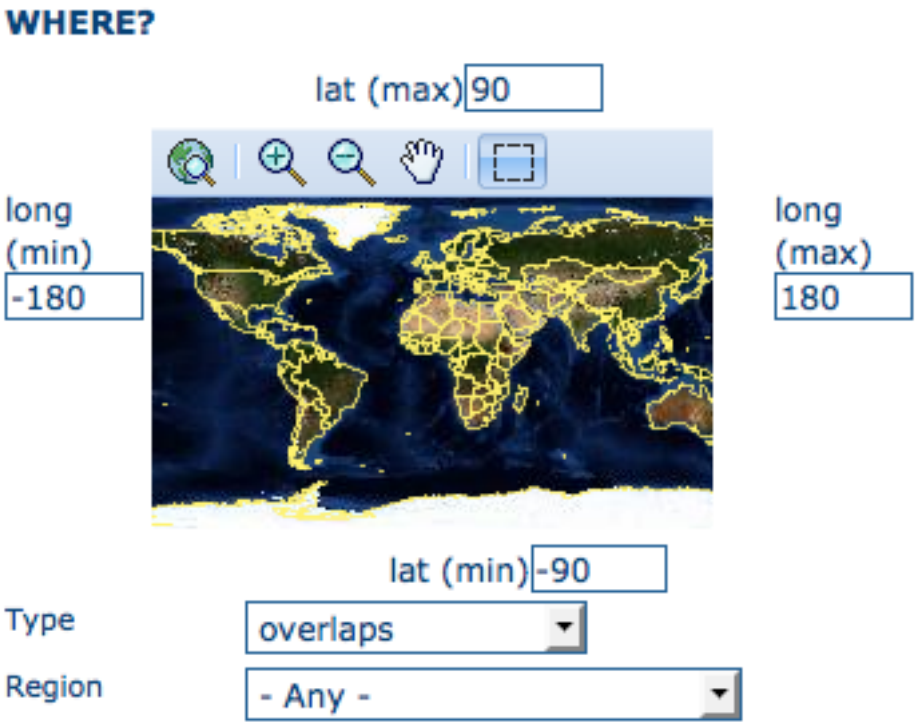


Figure 2.9: *"Where" section in the Advanced search*

- If you choose **Spatial search type** *is* "Country", only maps for the selected country will be displayed. In other words, a city map within that country will not show in the output results.
- If you choose **Spatial search type** *overlaps* "Country", all maps with the bounding box overlapping that country will be displayed in the results, i.e. the neighbouring countries, the continent of which that country is part of and the global maps.
- If you choose **Spatial search type** *encloses* "Country" you will get, in the output results, maps of that country first and then all maps within its bounding box.
- Similarly, if you choose **Spatial search type** *is fully outside of* a selected region, only maps that follow that exact criteria will show in the output results.

The **WHEN?** section gives you the possibility to restrict your search in terms of temporal extent, indicating a specific range of time referred to the data creation or publication date.

- To specify a range of time, click on the date selector button next to **From – To** fields. Make use of the symbols **>** and **>>** on top of the calendar to select the month and the year first and then click on the exact day; a complete date will be filled in using the following standard order: YY-MM-DD.
- To clean the time fields, simply click on the white cross on their right; the box **Any** will be automatically selected and the search will be performed without any restriction on the time period.

Finally, the advanced search allows you to apply further restrictions on the basis of additional parameters as data source, data categories and data format.

- To limit your queries to only one **Catalogue** out of those made available by the installation through the harvesting process, highlight the catalogue of preference or just keep **Any** selected to search all sites.
- To search for data organised by **Category**, such as Applications, Datasets, etc., simply highlight the category you wish to search in from the related drop-down list, otherwise we suggest to leave

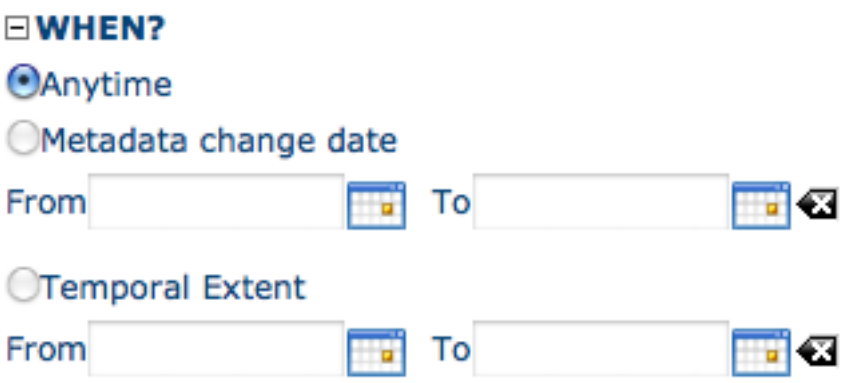


Figure 2.10: *"When" section in the Advanced search*

this field in **Any** Category.

- You can search for **Digital** or **Hard Copy** maps. To search in one or the other, simply check the box next to the one you wish to search. If no box is checked, all content will be searched.

At last, you can customise the number of output results per page in the *Hits Per Page* field. Simply highlight the number of records to be displayed or leave the field set on the default number (10).

- Click the **Search** button.

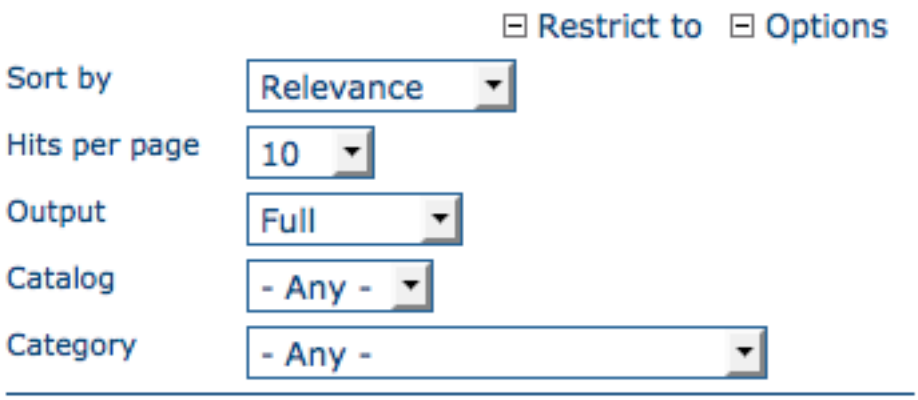


Figure 2.11: *Other options in the Advanced search*

Inspire

If INSPIRE Search panel is enable in Administration > System configuration page, an additional section is displayed to allow searching INSPIRE metadata in the catalog.

- *Annex*: Allows to search for metadata related to a specific Inspire annex. The Inspire annexes for a metadata are based on the Inspire theme keywords assigned to it.
- *Source type*: Allows to search for dataset or service metadata.
- *Service type*: Allows to search for service metadata using the service type values defined in INSPIRE metadata regulation (section 1.3.1).

INSPIRE SEARCH OPTIONS

- [ ] Only INSPIRE metadata

Annex

Source type

Services

Service type

View Service (view)

Classification of data services

Geographic human interacti

INSPIRE Theme

Annex I

- [ ] Geographical names
- [ ] Administrative units
- [ ] Addresses
- [ ] Cadastral parcels
- [ ] Transport networks
- [ ] Hydrography
- [ ] Protected sites

Annex II

- [ ] Elevation
- [ ] Land cover
- [ ] Orthoimagery
- [ ] Geology

Annex III

- [ ] Statistical units
- [ ] Buildings
- [ ] Soil
- [ ] Land use
- [ ] Human health and safety
- [ ] Utility and Government services
- [ ] Environmental monitoring facilities
- [ ] Production and industrial facilities
- [ ] Agriculture and aquaculture facilities
- [ ] Population distribution - demography
- [ ] Area management/restriction/regulation zones and reporting units
- [ ] Natural risk zones
- [ ] Atmospheric conditions
- [ ] Meteorological geographical features
- [ ] Oceanographic geographical features
- [ ] Sea regions
- [ ] Bio-geographical regions
- [ ] Habitats and biotopes
- [ ] Species distribution

- *Classification of data services*: Allows to search for metadata that have selected keyword from the *Inspire service taxonomy* thesaurus.
- *Inspire themes*: Allows to search for metadata that have selected keywords from the *Inspire themes* thesaurus.

2.2.4 Search Results

The output of a search provides you a list of the metadata records that should fit your request. For each record, the result page shows the title, an abstract and the keywords. According to the privileges that have been set for each metadata, a maximum of four sections can be consulted, as shown below.

Figure 2.13: *Search results*

1. **Metadata**: The metadata section describes the dataset (e.g.: citation, data owner, temporal/spatial/methodological information) and could contain links to other web sites that could provide further information about the dataset.
2. **Download**: Depending on the privileges that have been set for each record, when this button is present, the dataset is available and downloadable. The process for retrieving data is simple and quick by just clicking the download button or by using the proper link in the specific metadata section for distribution info in the full metadata view.
3. **Interactive Map**: The map service is also optional. When this button is shown, an interactive map for this layer is available and, by default, it will be displayed on the map screen of the simple

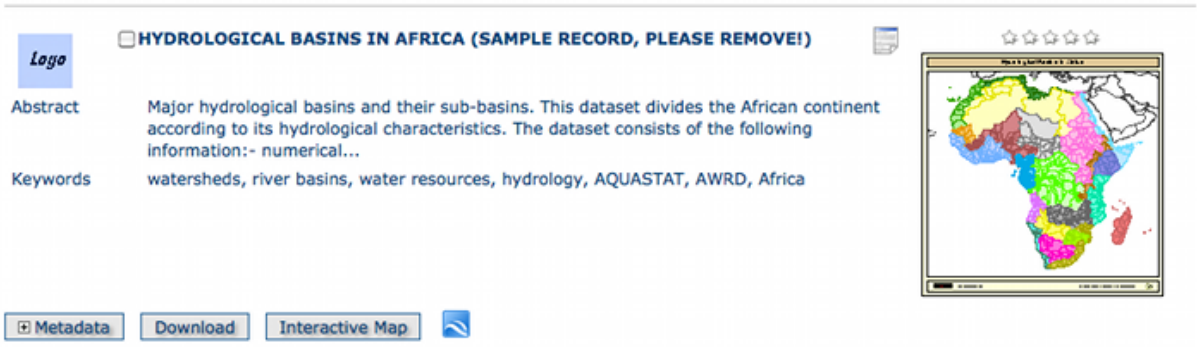


Figure 2.14: *A single search result*

Figure 2.15: *Available services related to the resource*

search. To better visualise the map through the map viewer, **click** on **Show Map** on the top of search results panel.

4. **Graphic Overviews**: There are small and large overviews of the map used to properly evaluate usefulness of the data, especially if the interactive map is not available. Simply click on the small image to enlarge it.

2.2.5 Privileges, roles and user groups

GeoNetwork uses a system of *Privileges*, *Roles* and *User groups*.

There are no restrictions for users to search and access **public information** in a GeoNetwork opensource based catalogue. To get access to **restricted information** or advanced functionality, an account to log in is required. This should be provided by the GeoNetwork administrator.

To log in, simply go to the home page and enter your username and password in the dedicated fields on the top right corner, then click the login button.

Privileges. Depending on the privileges set on a metadata record and on your role as an authenticated user, you will be able to read about a resource and download or interactively browse data related to that resource.

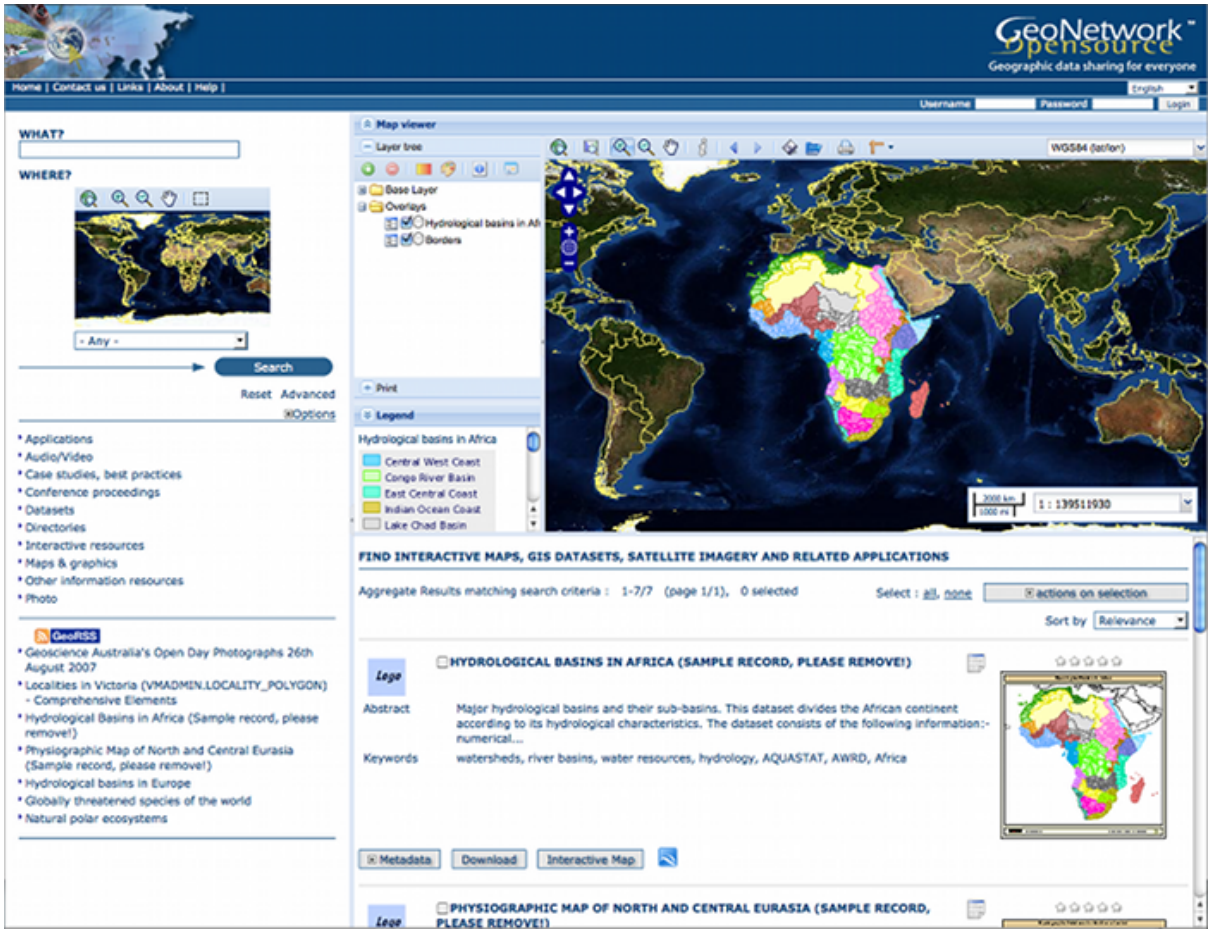


Figure 2.16: *The interactive map viewer*

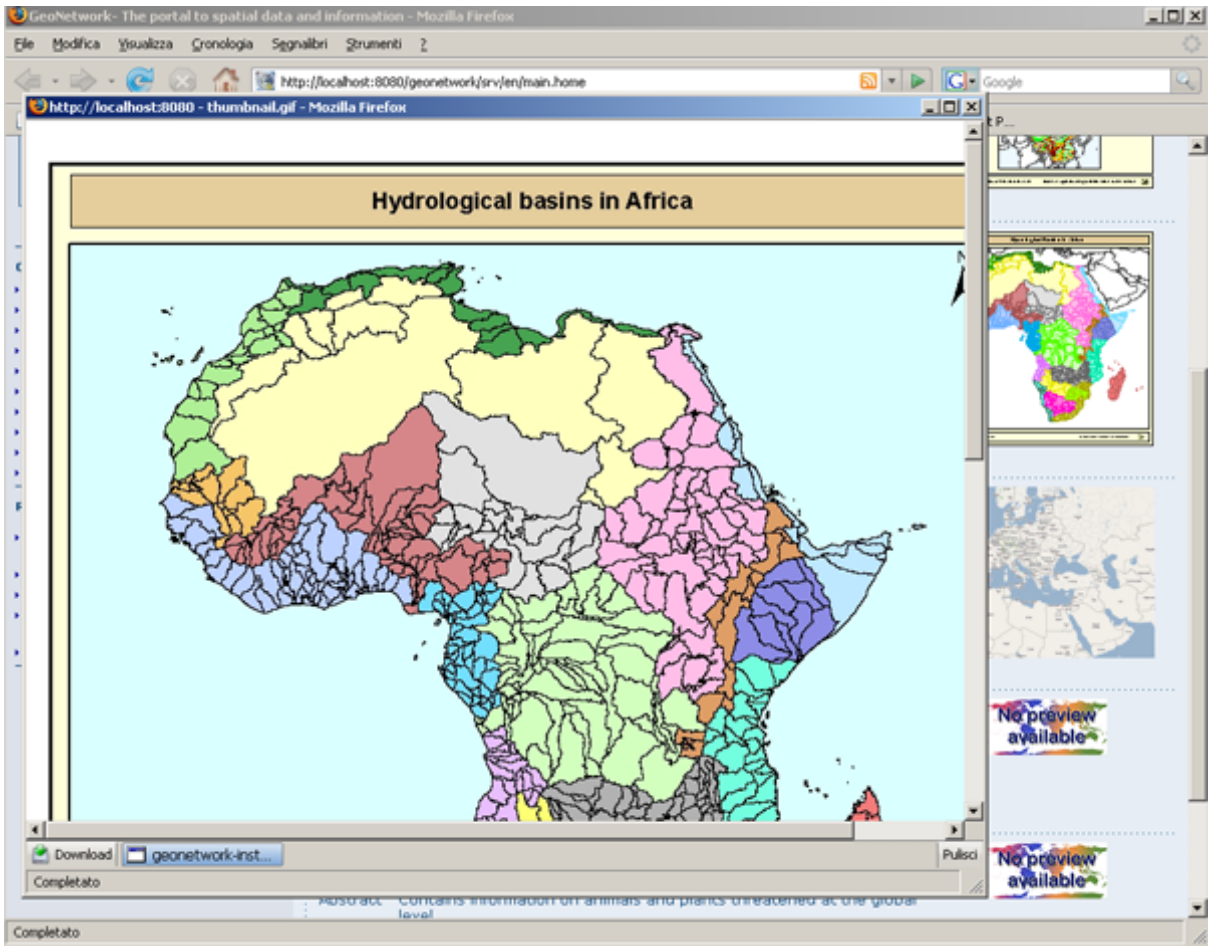


Figure 2.17: *Large preview image*

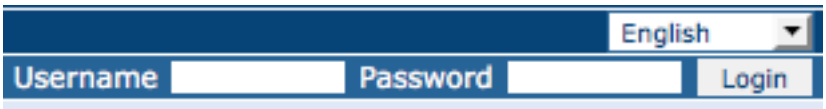


Figure 2.18: *Login*

Roles. Users with an *Editor* role can create, import and edit metadata records. They can also upload data and configure links to interactive map services.

User groups. Every authenticated user is assigned to a particular work group and is able to view data within that work group.

2.3 Viewing and Analysing the Data

Once you have completed your search, you view details of a particular record by clicking on the *Metadata* button.

The metadata profiles used by GeoNetwork opensource to present and describe geographic data and general documents stored in the catalogue are based on the *International Standard ISO 19115:2003*, encoded according to the *implementation schema 19139:2007*, the *FGDC* and the international standard *Dublin Core*.

In this guide the *ISO 19139* metadata implementation will be described in details since it is also suggested as profile for the creation of new metadata records.

2.3.1 Metadata Description

The metadata ISO 19139 profile used by GeoNetwork opensource to describe the geographic data and services is based on the ISO standard 19115:2003 and provides information related to the identification, the maintenance and constraints, the spatial and temporal extent, the spatial representation and reference, the quality and distribution of a geographic dataset.

The metadata profile is organised in sections and the most important, illustrated below, are the: *Identification Section, Distribution Section, Reference System Section, Data Quality Section and Metadata Section*. These sections are described here in details.

Identification Section

This section includes information on the citation of the resource (*title, date of creation or publication, edition, presentation form*), the *abstract*, the *purpose* and the present*status* of the resource that can be defined among the options: completed, historical archive, obsolete, ongoing, planned, required or under development.

This section also contains information about the person or organisation responsible for the data and who is considered to be a *point of contact* for the resource i.e. the dataset owner, originator, distributor, publisher, etc. and it provides information on *data maintenance* i.e. annually, monthly, daily, not planned, as needed, etc.

Elements for *keywords* and for describing restrictions on *data access* and *use* are also included in this section in addition to *spatial representation info* like data type (vector, raster, text table, etc.)

The identification section provides information about the *scale*, the *language* and *character set* used within the resource and the list of *ISO categories* through which your map could be classified.

Finally, the temporal and spatial extent are also defined in this section. The temporal extent is defined through the starting and ending date of data validation.

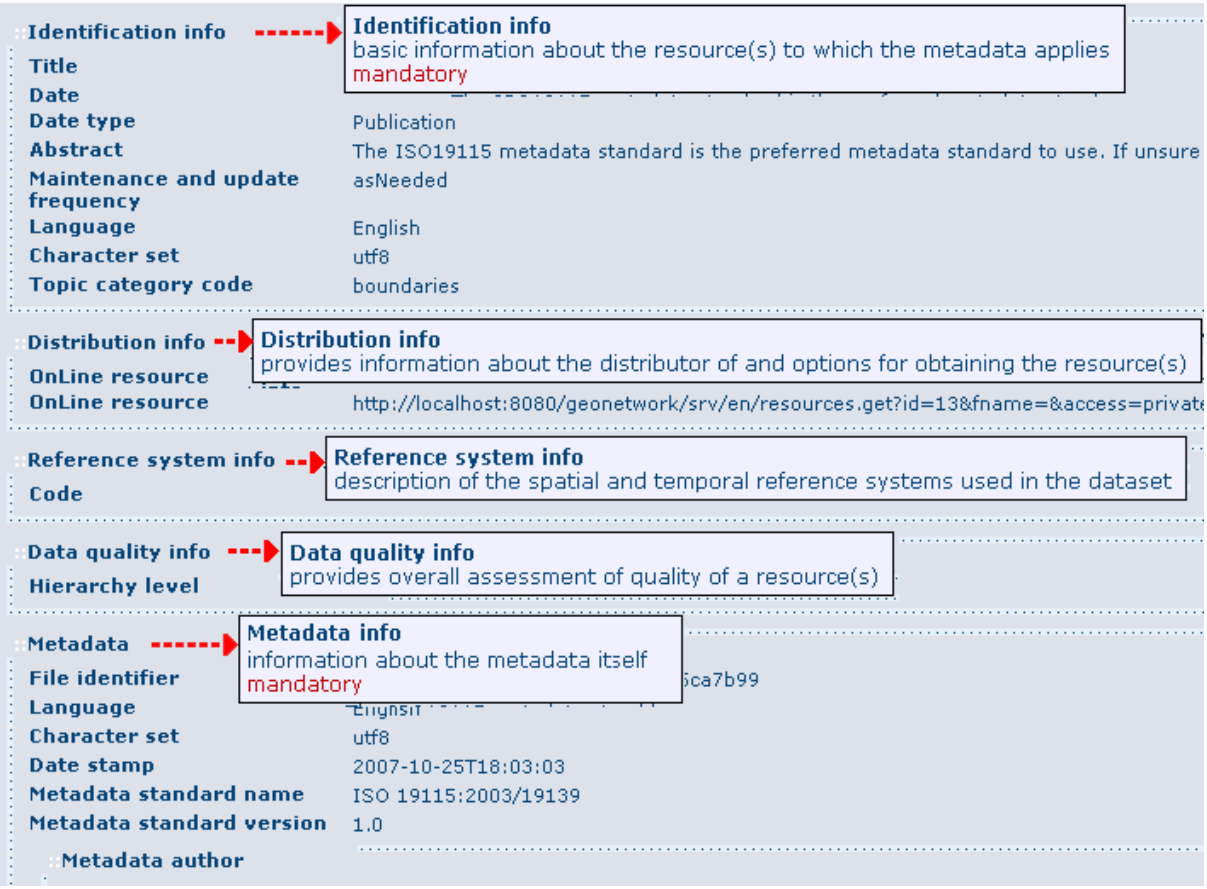


Figure 2.19: *Main metadata sections*

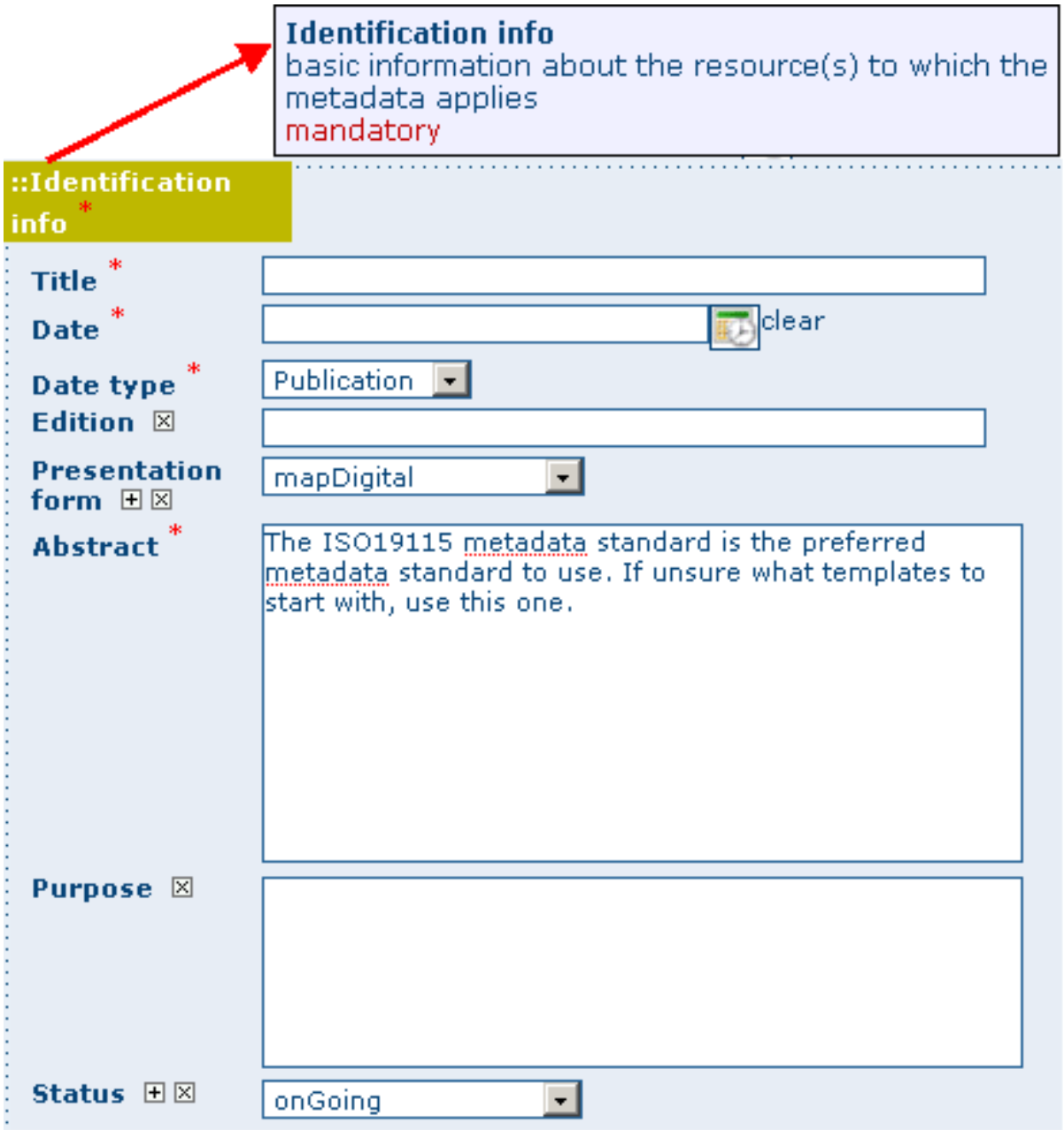


Figure 2.20: *Identification information*

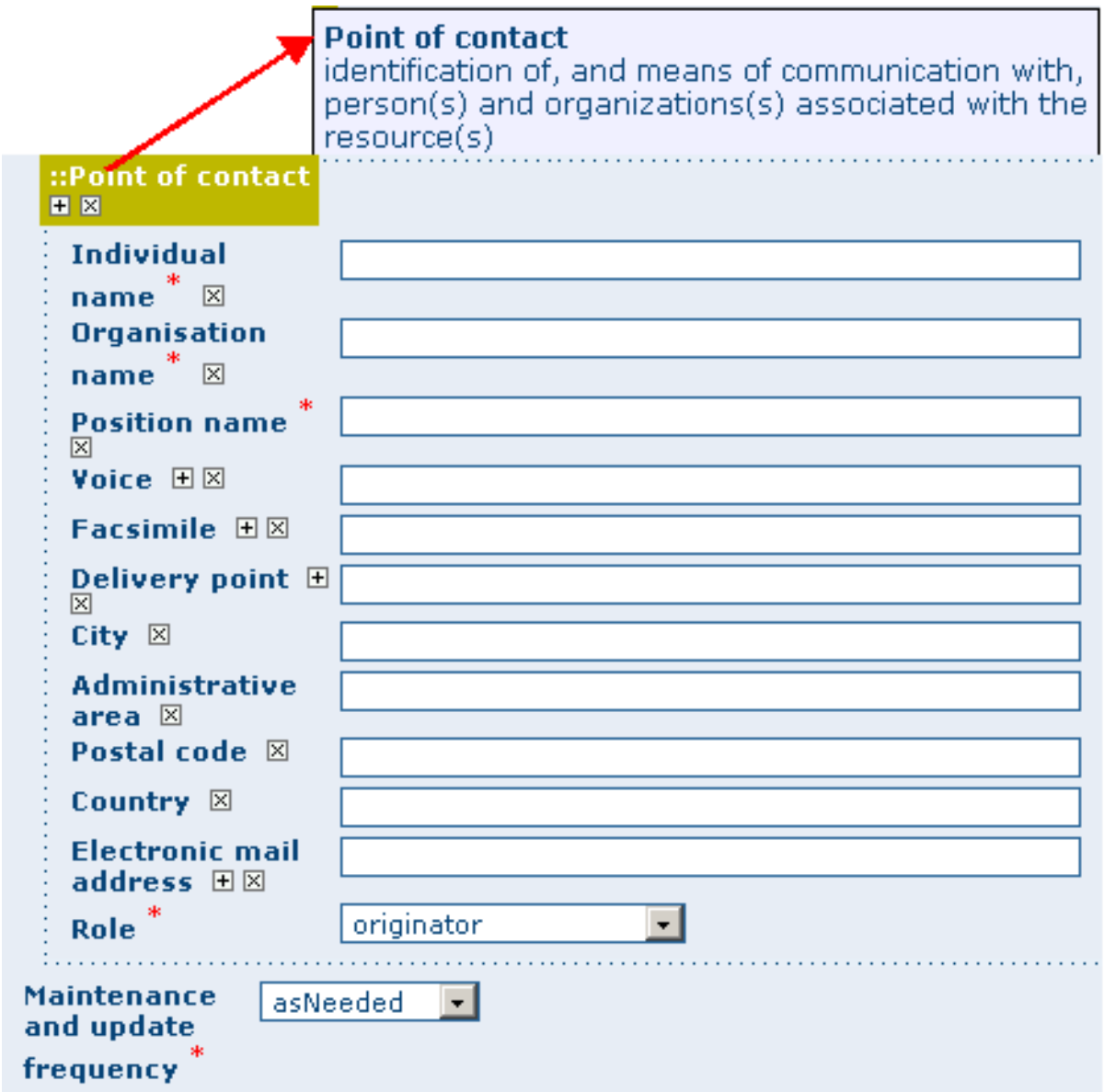


Figure 2.21: *Point of Contact*

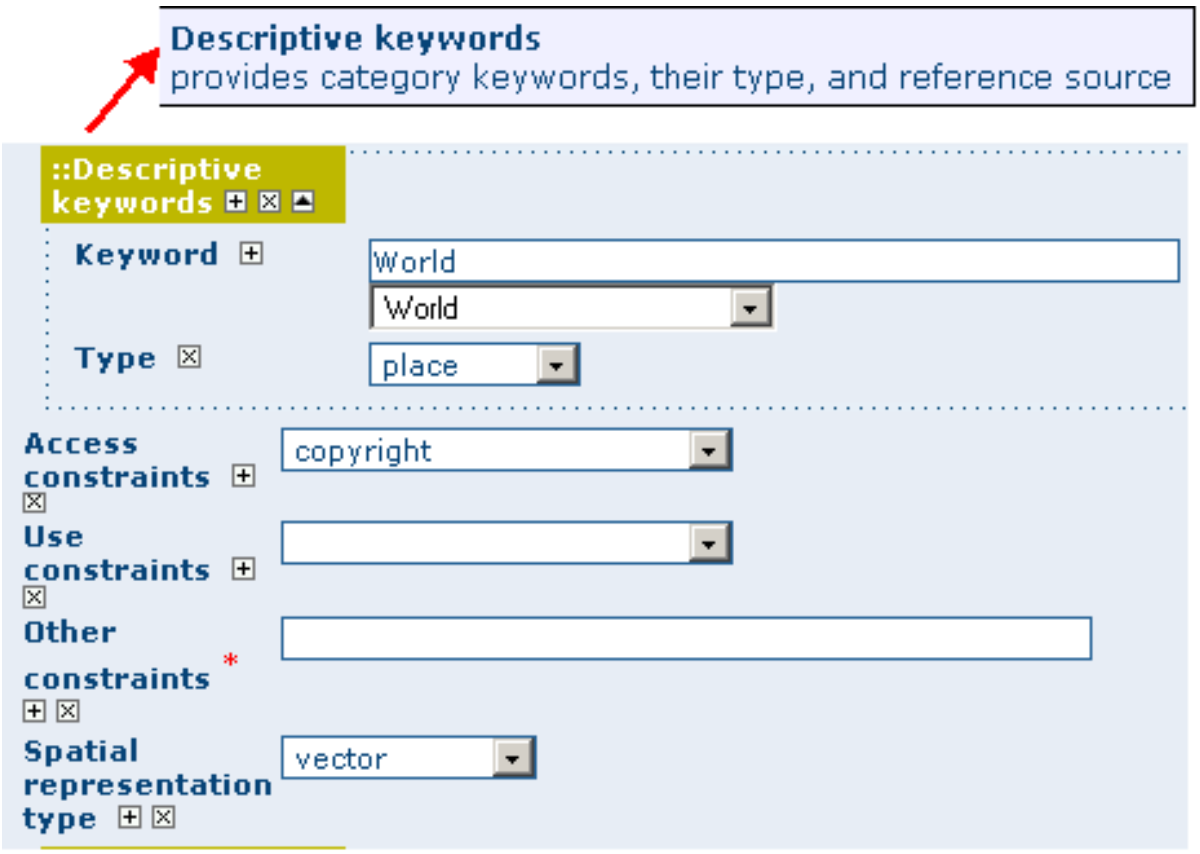


Figure 2.22: *Descriptive keywords*

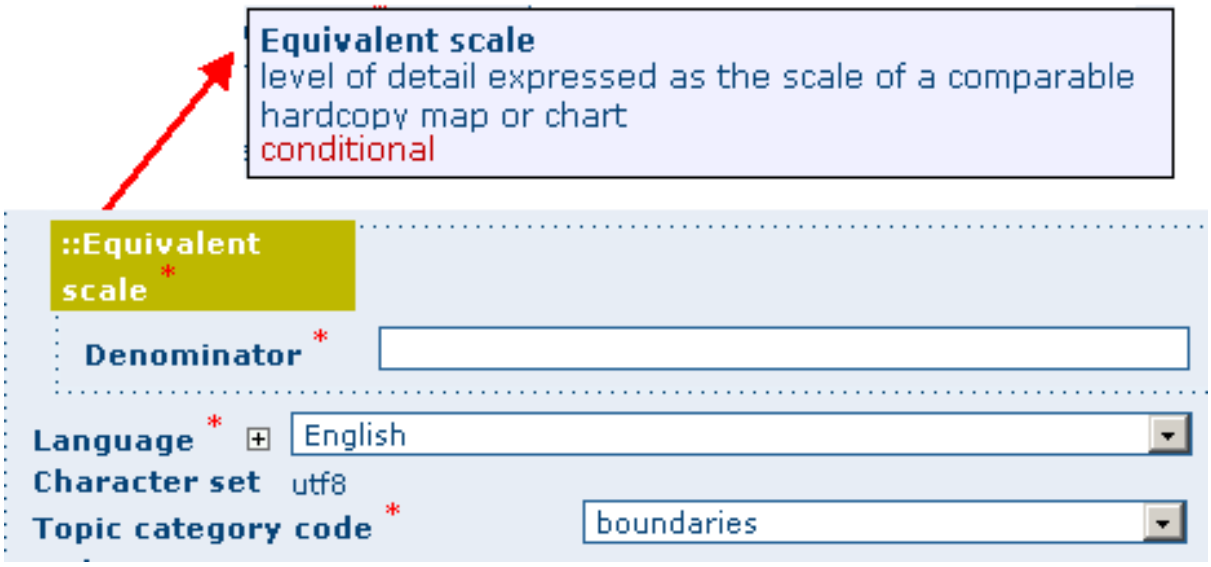


Figure 2.23: *Scale and other data properties*

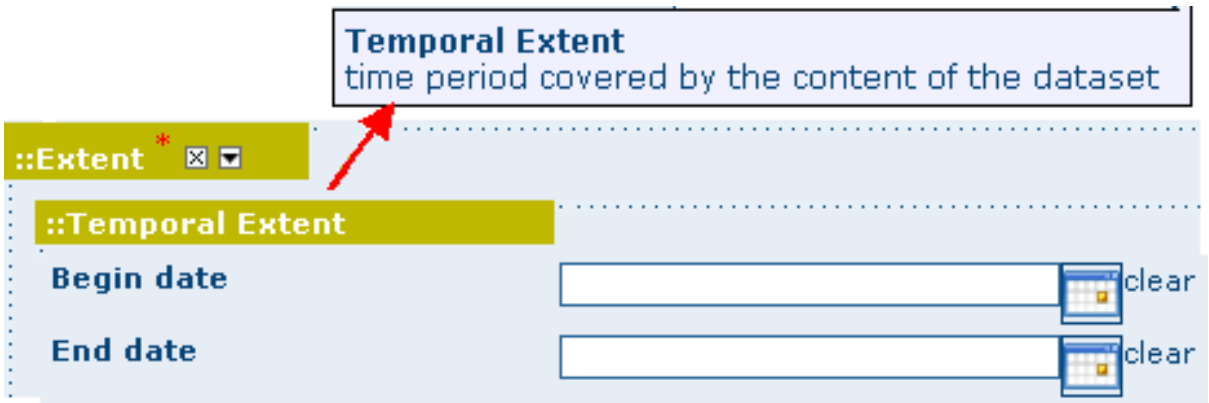


Figure 2.24: *Temporal extent*

The spatial extent of the interested area is defined through geographic coordinates or through the selection of a country or region from a predefined list. Free text *supplemental information* can be added to complete the data identification section.

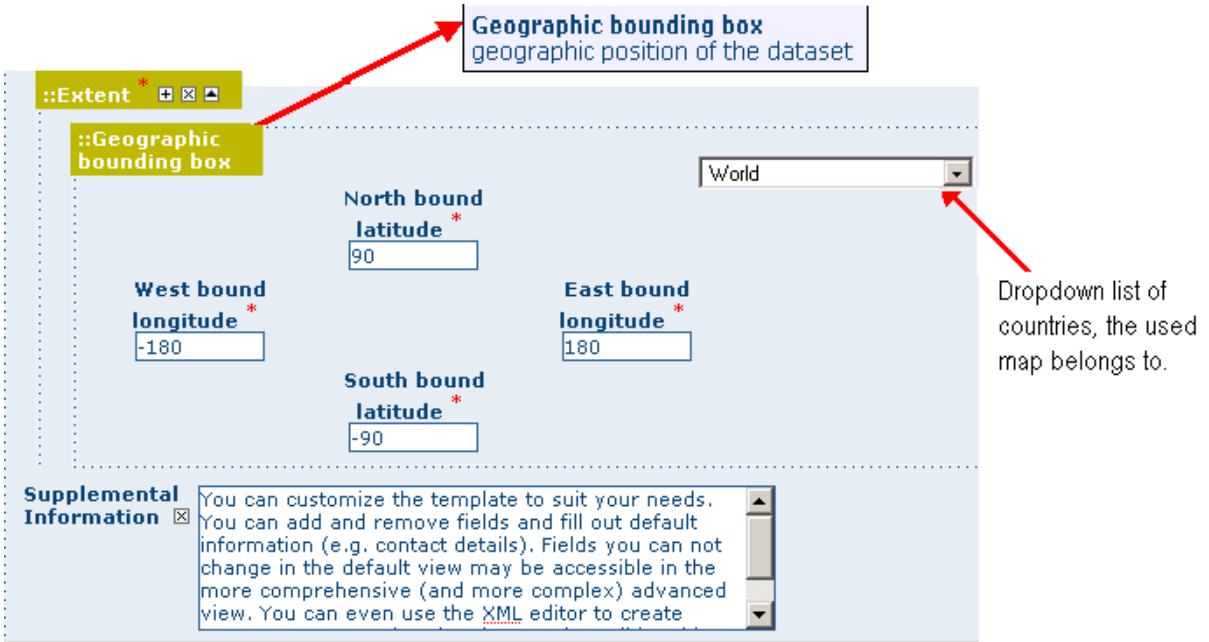


Figure 2.25: *Geographic bounding box*

Distribution Section

This section provides metadata elements for *accessing* other useful *on-line resources* available *through the web*. The distribution elements allow for on-line access using an URL address or similar addressing scheme and provide the protocol for the proper connection *for accessing geographic data* or any other types of digital documents *using the download function*. Furthermore, it is possible *to link* a metadata with a *predefined map service* through the *online resource* and see the map interactively.

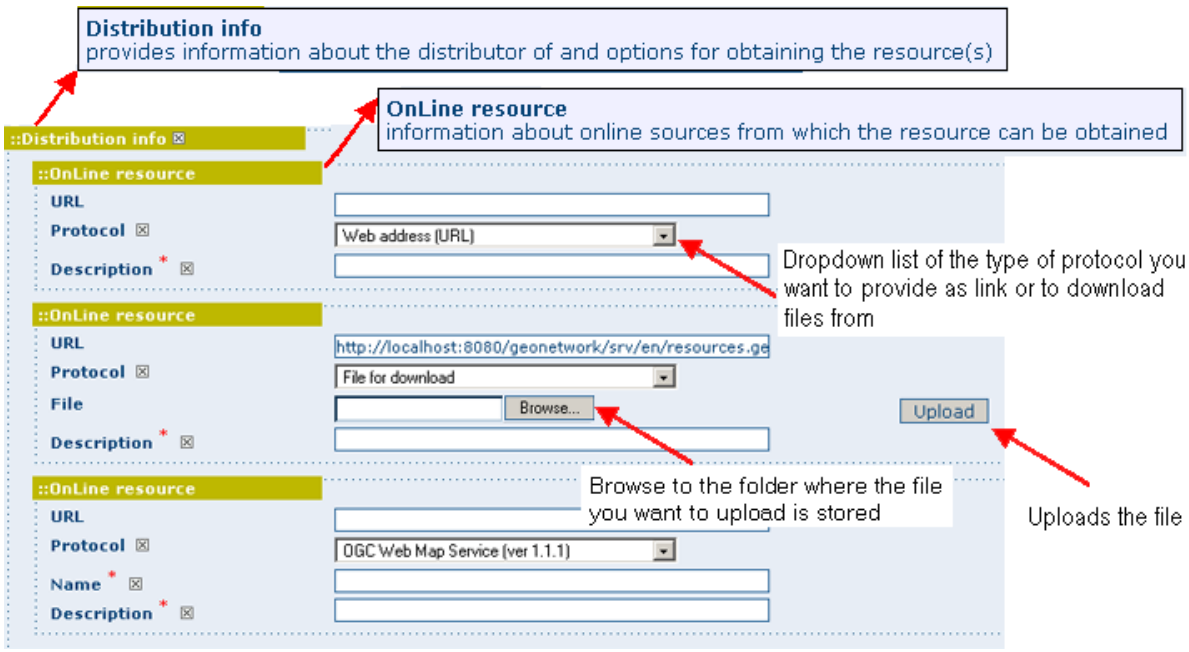


Figure 2.26: *Distribution information*

Reference System Section

The Spatial Reference System section *defines metadata* required *to describe* the *spatial reference system* of a dataset. It contains one element to identify the name of the reference system used. Using elements *from the advanced form*, this section may be modified to provide more *details* on *data projection*, *ellipsoid* and *datum*. Note that if this information is provided, a reference system identifier is not mandatory.

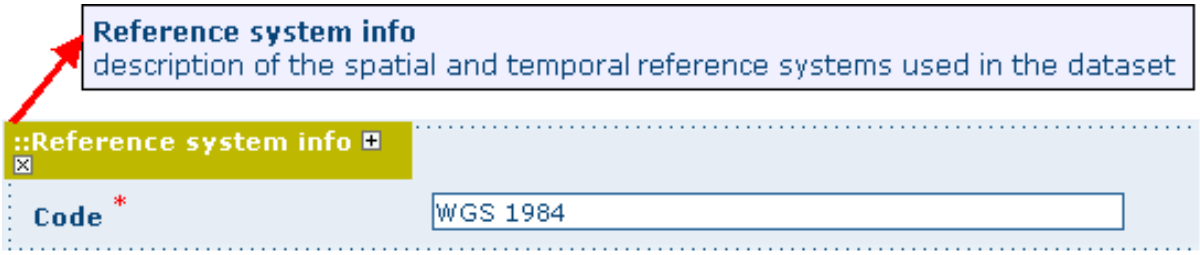


Figure 2.27: *Reference system*

Data Quality Section

The Data Quality section provides a *general assessment* of the *quality of the data*. It describes the*different hierarchical levels of data quality*, namely a *dataset series*, *dataset*, *features*, *attributes*, etc. This section also contains information about *sources of the input data*, and a *general explanation of the production processes* (lineage) used for creating the data.

Metadata Information Section

This section contains information about the metadata itself: the Universally Unique Identifier (UUID) assigned to the record (this is the 'File identifier'), *language* and *characterset* used, *date* of *last edit*

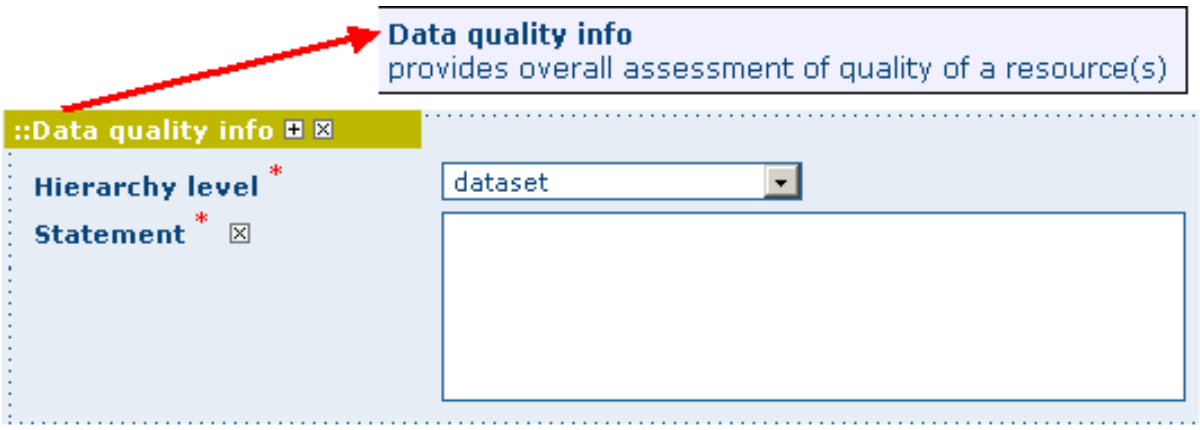


Figure 2.28: *Data quality*

('Date stamp') and the *metadata standard* and *version name* of the record. It also contains information on the *metadata author* responsible for the metadata record; this person can also be a point of contact for the resource described. Information on the Metadata author is mandatory.

2.4 Adding a metadata record

This section guides you through the process of adding new metadata records with associated data and/or services into the GeoNetwork catalog. You will use metadata template records, add thumbnails, upload data, link to services and set access privileges to the metadata and the data it describes.

To add or edit metadata, you must be **registered** as a user with an **Editor** profile or higher. That user should be a member of the User Group you want to add information for. Contact your administrator if you are not a registered Editor for your User Group.

For metadata creation using the online editor, GeoNetwork provides a set of simplified metadata templates based on the cited standards available in your GeoNetwork instance: typically ISO19139 (an implementation of ISO19115), FGDC and Dublin Core. The templates for describing vector or raster geographic data based on ISO19139 are preferred because they are devised in a way that hides much of the complexity of the ISO19115 standard in the default view. At the same time those templates are extensible with new elements to fit specialized needs through the advanced view.

To produce a good metadata record, always try to:

- gather as many details as possible on the resource that you want to describe taking into account the metadata elements that have been presented in the previous chapter
- develop and reuse the same terms or phrases to describe the concepts you want to capture. A record of these terms and phrases will be helpful for others in understanding your metadata.

The next step is to fill out properly the fields provided by the metadata templates, while at the same time avoiding duplication of information throughout the form.

The most important fields that may not be waived while compiling a standard based metadata record are the following: **Title, Date of Creation or Publication, Abstract, Language used for documenting data, Topic Category, Scale, Maintenance and Update Frequency, Metadata Author, Language Used for Documenting Metadata.**

In addition to the main mandatory fields, we recommend you to fill out these optional but critical fields (if information is available): **Purpose - Keywords - Presentation Form - Status - Spatial Representation**

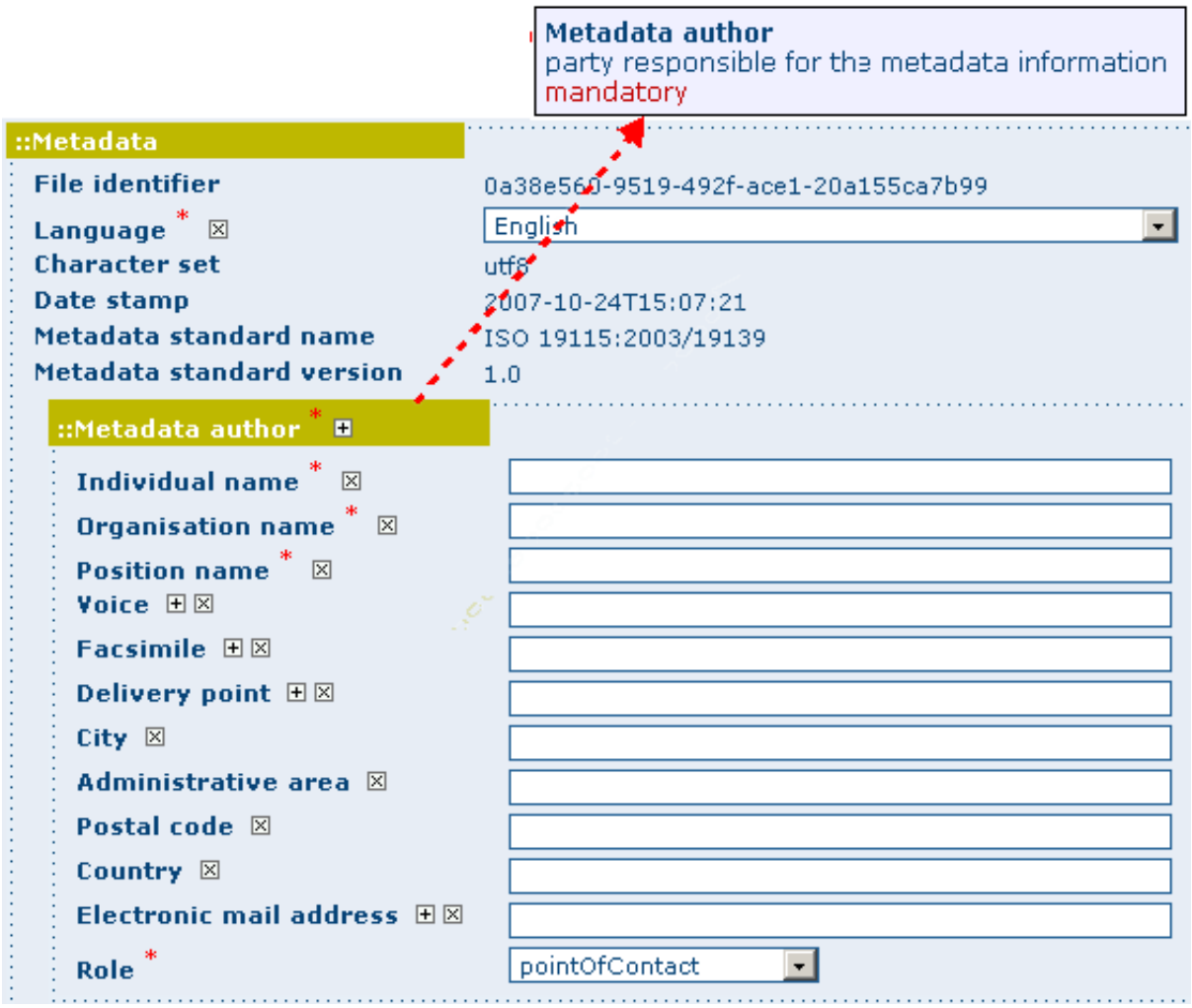


Figure 2.29: *Metadata properties*

Type - Geographic Location - Reference System Info - Temporal Extent - Data Quality Info - Access and Use Constraints - Point of Contact - Distribution Info: Online Resources.

You should also prepare an image of your data that is required to be displayed in search results as thumbnail.

Next section will guide you through the process of metadata creation using the online editor.

2.4.1 Creating a New Record using the Metadata Editor

1. In the home page, click on the **Administration Tab**.
2. Select **New Metadata** from the List of the admin page.
3. Select the metadata standard **Template**, if possible, using the preferred ones. GeoNetwork open-source comes by default with support for three metadata standards, ISO19139, FGDC and Dublin core. For the ISO standard, two templates have been developed; one for vector and one for raster data. Both contain a relevant set of elements to describe the respective types of data. More templates can be developed online.
4. Select the **Group** the metadata will belong to. These are the groups authorized to add metadata to by your administrator.
5. Click on **Create**.

2.4.2 The steps in more details

1. Enter your username and password and click on the login button. The system will identify you and assign the correct privileges to work with.

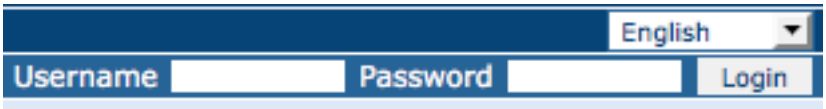


Figure 2.30: *Login*

2. Open the Administration page by clicking the Administration button in the banner and then click on the New metadata link.

3. From the metadata creation page, select the metadata standard to use from the dropdown list (Figure 4.3, "Template selection")

4. After selecting the correct template, you should identify which group of users the metadata will belong to and finally click on **Create**.

5. A new metadata form based on the selected template will be displayed for you to fill out.

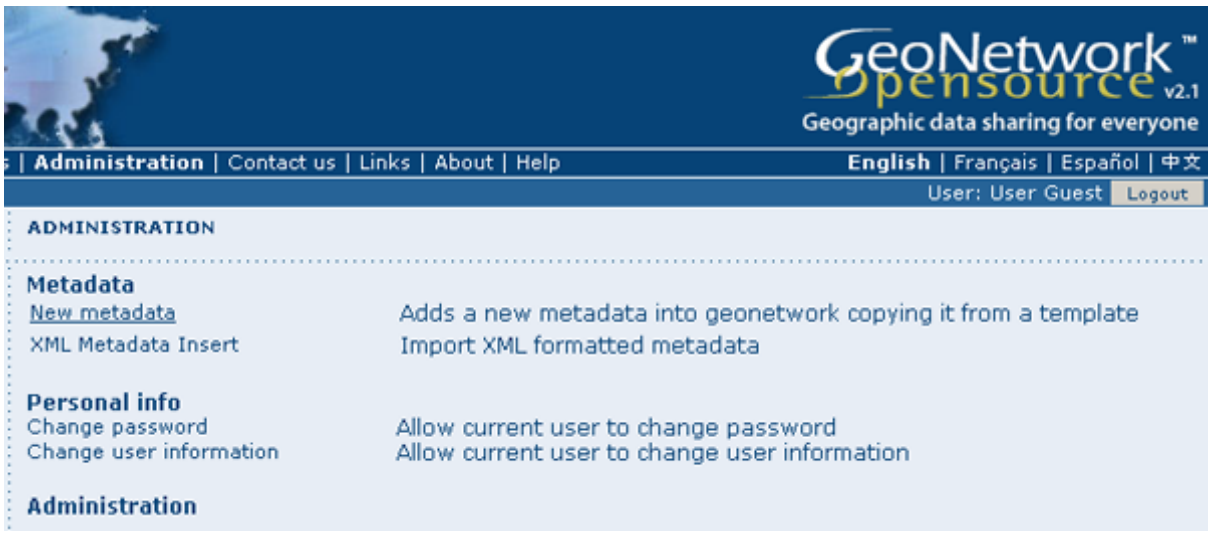


Figure 2.31: *Administration panel*

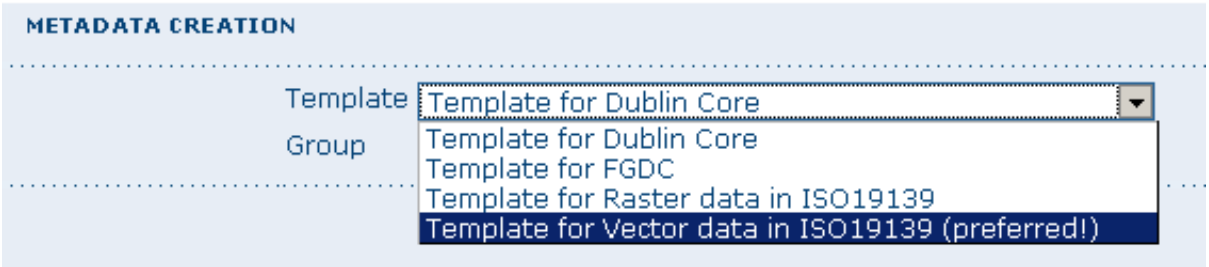


Figure 2.32: *Template selection*

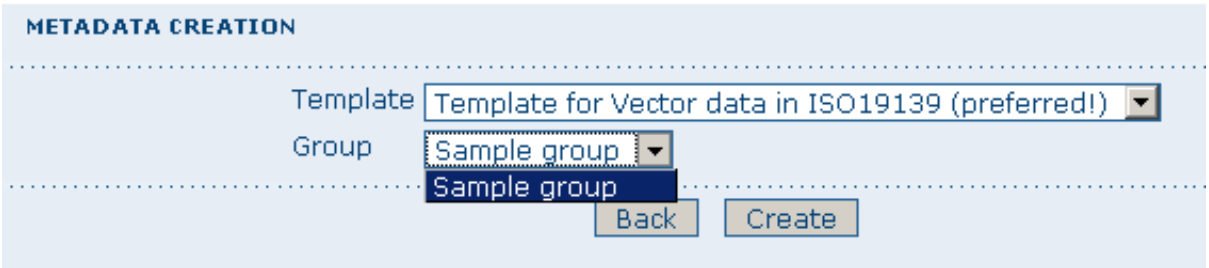


Figure 2.33: *Group selection*

2.4.3 Switching Editing Views from Default to Advanced to XML View

Once you create a new record, you can choose between **Default, Advanced** or **XML View**. To switch view, simply click on the view you want to switch to on the left column of the page. The view in **bold** is the view you are currently using.

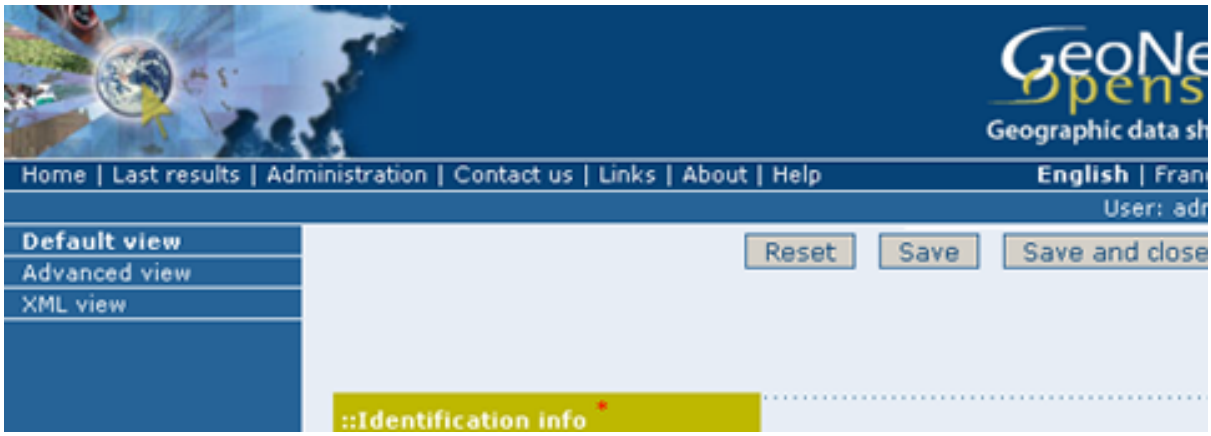


Figure 2.34: *Metadata view options*

In the previous chapter you have analyzed the metadata structure as it is presented in the **Default View**. A selection of the main fields from different categories of information is shown in one single view. The minimum set of metadata required to serve the full range of metadata applications (data discovery, determination of data fitness for use, data access, data transfer and use of digital data) is defined here, along with optional metadata elements to allow for a more extensive standard description of geographic data, if required. However, if should be there a need to add more metadata elements, you can switch to the advanced view at any time while editing.

In the **Advanced View**, the ISO profile offers the possibility to visualize and edit the entire metadata structure organized in sections accessible through tabs from the left column. You can use this view to write more advanced metadata descriptions or templates to fit specialized needs.

The **XML View** shows the entire content of the metadata in the original hierarchical structure; different colors allow to distinguish between an element's name and its value. The XML structure is composed of tags and to every tag must correspond a closing tag. The content is entirely contained withing the two, i.e.:

```
<gmd:language>
  <gco:CharacterString>eng</gco:CharacterString>
</gmd:language>
```

Nevertheless, the use of the XML view requires some knowledge of the XML language.

Both the **Default** and the **Advanced Views** are composed of `mandatory`, `conditional` and `optional` metadata fields. The meaning of mandatory and optional is fairly intuitive; the mandatory fields are required, like `Title` and `Abstract` for instance, whereas the optional fields can be provided but are not fundamental, depending on the metadata author. The conditional fields may be considered mandatory under certain circumstances: essentially a conditional requirement indicates that the presence of a specified data element is dependent on the value or presence of other data elements in the same section. For instance, the `Individual name` metadata element of the `Point of Contact`, which is a conditional element of the Identification section, becomes mandatory if another element of the same section, `Organization name` or `Position name` is not already defined.

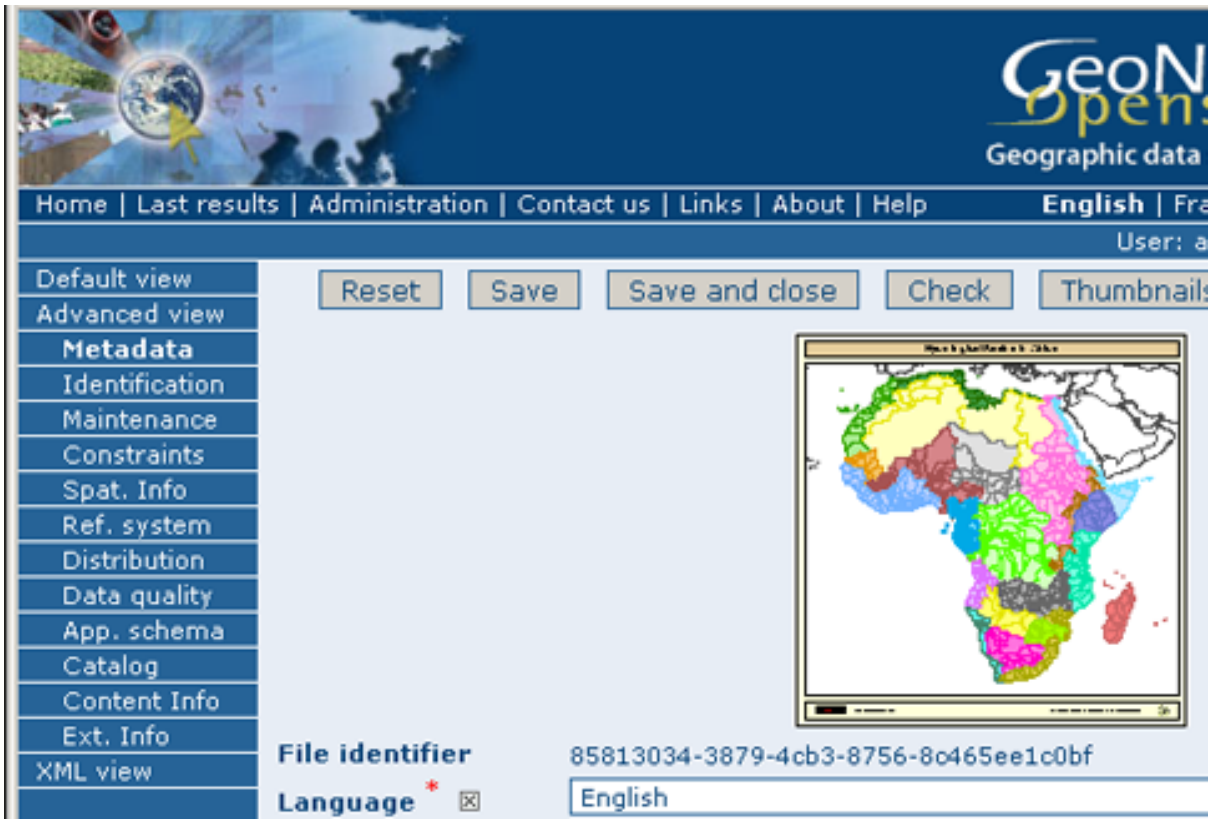


Figure 2.35: *Advanced view*

Figure 2.36: *XML view*

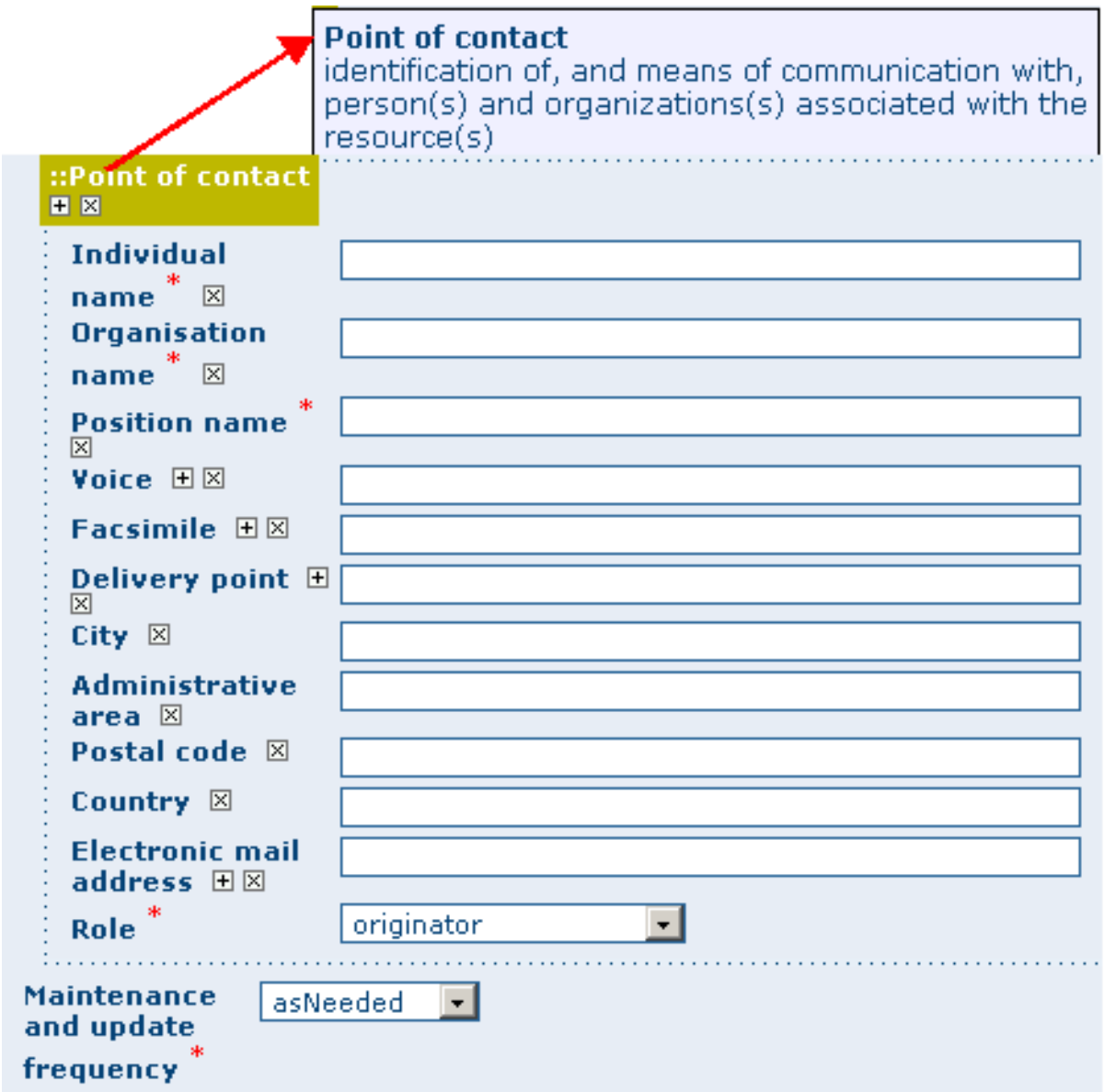


Figure 2.37: *Point of Contact*

The **mandatory fields** as well as those **highly recommended** are flagged with **red asterisk [*]**. The standard definition for each field can be read by passing the mouse on the element name.

The **Default View** is the preferred view as it provides a selection of the available metadata elements, facilitating both the user and the editor in reading and editing a metadata record, and at the same time it ensures that a geospatial data can be properly described, through :

- the minimum set of metadata required to serve the full range of metadata applications (data discovery, determination of data fitness for use, data access, data transfer, and use of digital data);
- optional metadata elements - to allow for a more extensive standard description of geographic data, if required;
- a method for extending metadata to fit specialized needs.

2.4.4 Using basic commands of the editor

Fields are either **free text fields** or **drop down lists**. Free text means you can type any text into that field. Drop down lists allow you to select only one option from the list. You can add multiple fields of the same kind by clicking on the **[+]** symbol next to the element. Every new field that you will add in the advanced view will then be visible in the default view. You can also delete existing fields by clicking on the **[x]** symbol next to the element. Clearly, mandatory fields cannot be deleted. One example of the need to add multiple fields can arise if the content of your dataset has some text written in two different languages.

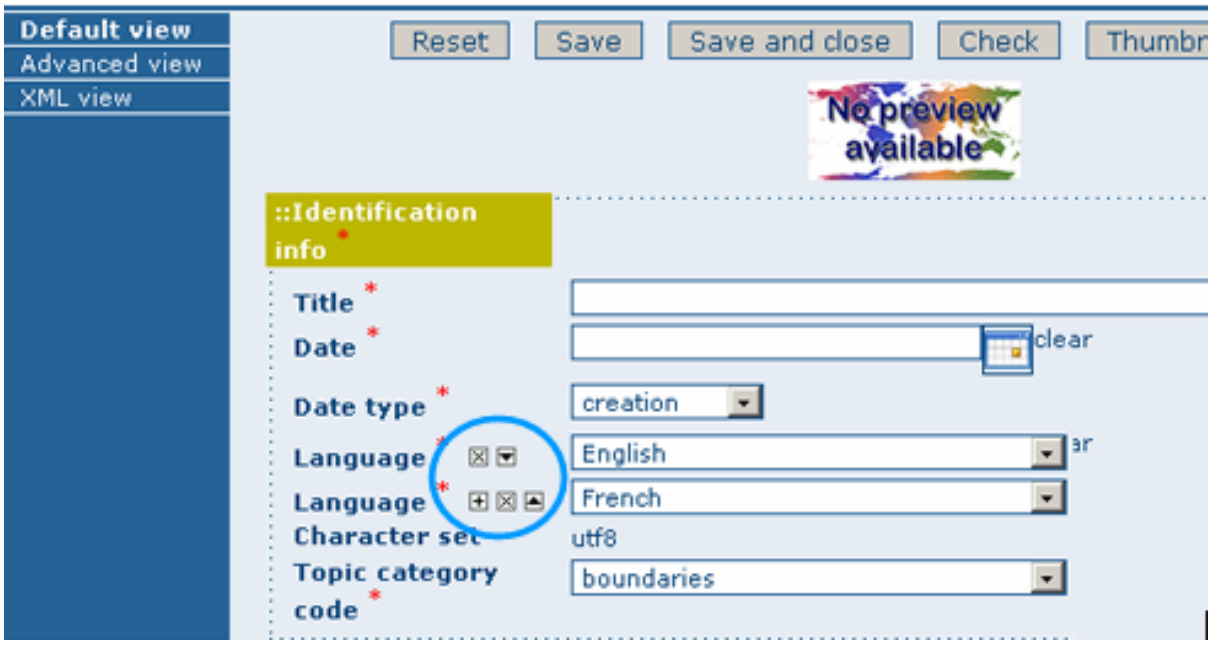


Figure 2.38: *Describing multilingual data*

2.4.5 Example: Entering metadata for a Thematic Map

As we mentioned in the introduction to this guide, GeoNetwork provides tools to describe any type of geographic data (vector layers, raster, tables, map services, etc.) as well as general documents like reports, projects, papers, etc. For the purpose of this Quick Start Guide, an example of required and useful metadata elements to properly describe a thematic map will be provided hereafter. You should

gather as much information as possible to identify and understand the map's resource and characteristics you want to describe. Use the default view to start. If necessary, you can always switch to advanced view or come back later and edit the record with the additional information collected.

Please follow these steps to enter your map's metadata. Note that we will only go through the fields that have been identified as compulsory (i.e. those fields marked with the asterix [*], mandatory or highly recommended).

Title *: Under the Identification Info field, give your map a name. There will be a default name of your data. Use free text to describe your map here.

Date *: Indicate the exact date of creation, publication or revision on your map.

Presentation Form: Specify the type of presentation, i.e. digital, hard copy, table, etc.

Abstract *: Enter some description of the map.

Purpose: Enter a short summary of the purposes for your map to be developed.

Status: Specify the status of your map within the following options: completed, historical archive, obsolete, ongoing, planned, required, under development.

Point of Contact: Enter all mandatory information and others you have at hand for the contact of the person(s) associated with this resources of the map. Note that some fields are only conditionally mandatory, such as Organization Name if Individual Name and Position are not entered.

Maintenance and update frequency * : Specify the frequency with which you expect to make changes and additions to your map after the initial version is completed. If any changes are scheduled you can leave As Needed selected from the drop-down list.

Descriptive Keywords: Enter keywords that describe your map. Also specify the type of keyword you are entering, i.e. place, theme, etc. Remember that you can add another keyword field if you need to add different types of keywords.

Access Constraints: Enter an access constraint here, such as a copyright, trademark, etc. to assure the protection of privacy and intellectual property.

User Constraints: Enter a user constraint here to assure the protection of privacy and intellectual property.

Other Constraints * : Enter other constraint here to assure the protection of privacy and intellectual property. Note that this field is conditionally mandatory if Access and Use constraints are not entered.

Spatial representation type: Select, from the drop-down list the method used to spatially represent your data. The options are: vector, grid, text table, stereo model, video.

Scale Denominator * : Enter the denominator for an equivalent scale of a hard copy of the map.

Language* : Select the language used within your map

Topic category * : Specify the main ISO category/ies through which your map could be classified (see Annex for the complete list of ISO topic categories).

Temporal Extent * : Enter the starting and ending date of the validity period.

Geographic Bounding Box * : Enter the longitude and latitude for the map or select a region from the predefined drop-down list. Make sure you use degrees for the unit of the geographic coordinates as they are the basis for the geographic searches.

Supplemental Information: Enter any other descriptive information about your map that can help the user to better understand its content.

Distribution Info: Enter information about the distributor and about options for obtaining your map.

Online Resource: Enter information about online resources for the map, such as where a user may download it, etc. This information should include a link, the link type (protocol) and a description of the resource.

Reference System Info: Enter information about the spatial reference system of your map. The default view contains one element to provide the alphanumeric value identifying the reference system used. GNos uses the EPSG codes which are numeric codes associated with coordinate system definitions. For instance, EPSG:4326 is Geographic lat-long WGS84, and EPSG:32611 is "UTM zone 11 North, WGS84". Using elements from the advanced view, you may add more details on data projection, ellipsoid and datum. Note that if this information is provided, a reference system identifier is not mandatory.

Data Quality: Specify the hierarchal level of the data (dataset series, dataset, features, attributes, etc.) and provide a general explanation on the production processes (lineage) used for creating the data. The statement element is mandatory if the hierarchical level element is equal to dataset or series. Detailed information on completeness, logical consistency and positional, thematic and temporal accuracy can be directly added into the advanced form.

Metadata Author * : Provide information about the author of the map, including the person's name, organization, position, role and any other contact information available.

After completion of this section, you may select the Type of document that you are going to save in the catalogue. You have three options: Metadata, Template, Sub-template. By default Metadata is set up.

When done, you may click `Save` or `Save and Close` to close the editing session.

2.4.6 Metadata validation

In editing mode, editors can validate the current metadata record against standard rules and recommendations.

For all standards, a first level of validation is made for XML metadata validation based on XML Schema (XSD). For ISO19139 records, other rules are checked:

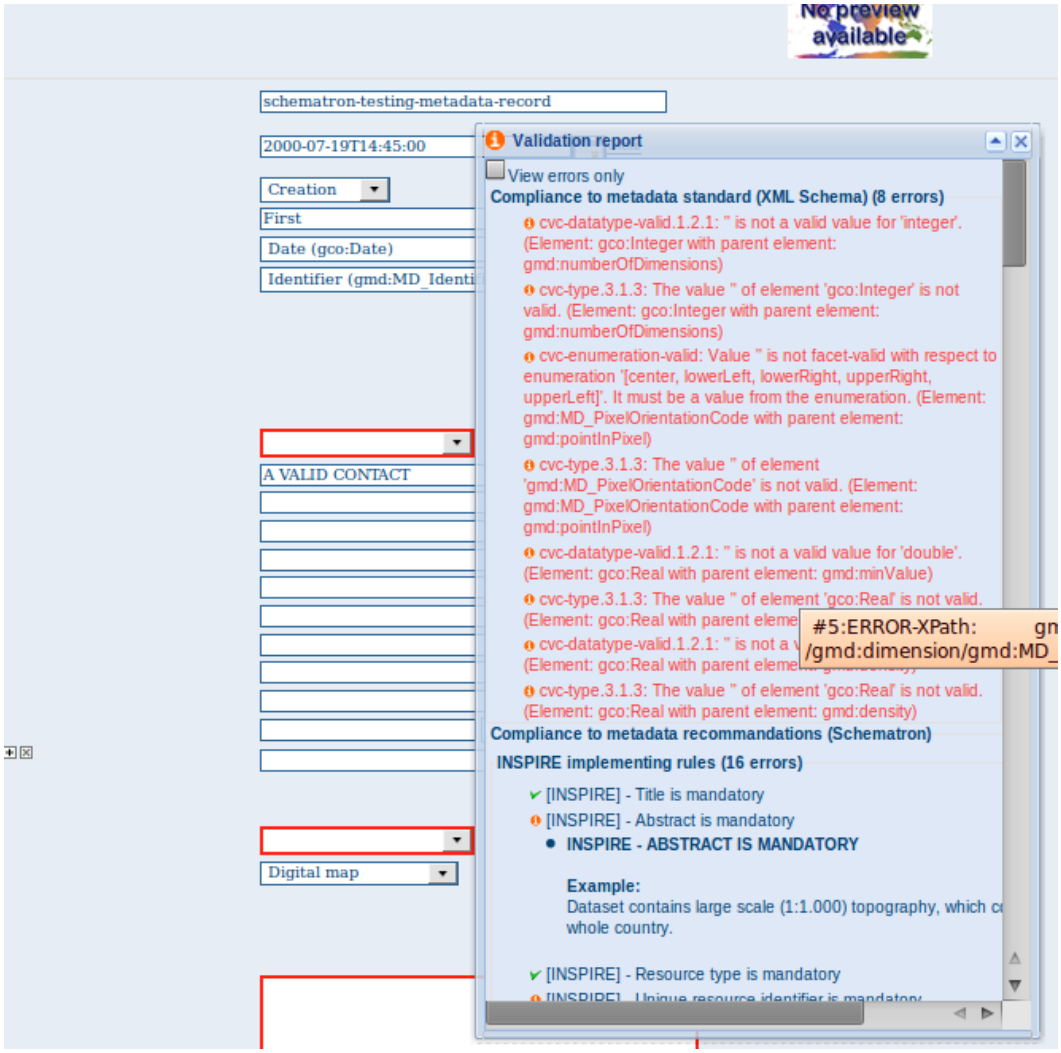
- ISO recommendations
- GeoNetwork recommendations
- (Optional and not available by default) INSPIRE recommendations

The validation report display the list of rules checked and their status (pass or failed). The top checkbox allows to display only errors or all.

2.4.7 Creating a Thumbnail

To help the user identify a metadata record of interest, you can create a graphic overview (or thumbnail) in the form of an image and attach it to the metadata record. For example, if your metadata record describes some geographic dataset then the the graphic overview could be an image of the map with legend produced by an OGC Web Map Service.

You can associate two thumbnails with a record: a small thumbnail, which will be displayed in search results and a large thumbnail with more details in case the user is interested in more information. The large thumbnail will be displayed when the user clicks on the small thumbnail.


No preview available
schematron-testing-metadata-record
2000-07-19T14:45:00
Creation
First
Date (gco:Date)
Identifier (gmd:MD_Identi
A VALID CONTACT
Digital map
Validation report
View errors only
Compliance to metadata standard (XML Schema) (8 errors)
cvc-datatype-valid.1.2.1: '' is not a valid value for 'integer'. (Element: gco:Integer with parent element: gmd:numberOfDimensions)
cvc-type.3.1.3: The value '' of element 'gco:Integer' is not valid. (Element: gco:Integer with parent element: gmd:numberOfDimensions)
cvc-enumeration-valid: Value '' is not facet-valid with respect to enumeration '[center, lowerLeft, lowerRight, upperRight, upperLeft]'. It must be a value from the enumeration. (Element: gmd:MD_PixelOrientationCode with parent element: gmd:pointInPixel)
cvc-type.3.1.3: The value '' of element 'gmd:MD_PixelOrientationCode' is not valid. (Element: gmd:MD_PixelOrientationCode with parent element: gmd:pointInPixel)
cvc-datatype-valid.1.2.1: '' is not a valid value for 'double'. (Element: gco:Real with parent element: gmd:minValue)
cvc-type.3.1.3: The value '' of element 'gco:Real' is not valid.
#5:ERROR-XPath:
/gmd:dimension/gmd:MD_
cvc-type.3.1.3: The value '' of element 'gco:Real' is not valid. (Element: gco:Real with parent element: gmd:density)
Compliance to metadata recommandations (Schematron)
INSPIRE implementing rules (16 errors)
[INSPIRE] - Title is mandatory
[INSPIRE] - Abstract is mandatory
INSPIRE - ABSTRACT IS MANDATORY
Example:
Dataset contains large scale (1:1.000) topography, which c
whole country.
[INSPIRE] - Resource type is mandatory

To create a thumbnail, go to the editing menu. If you are no longer in editing mode, retrieve the metadata record using one of the search options then click on Edit. Then follow these simple steps:

From the editing menu, click on the Thumbnails button on the top or bottom of the page.

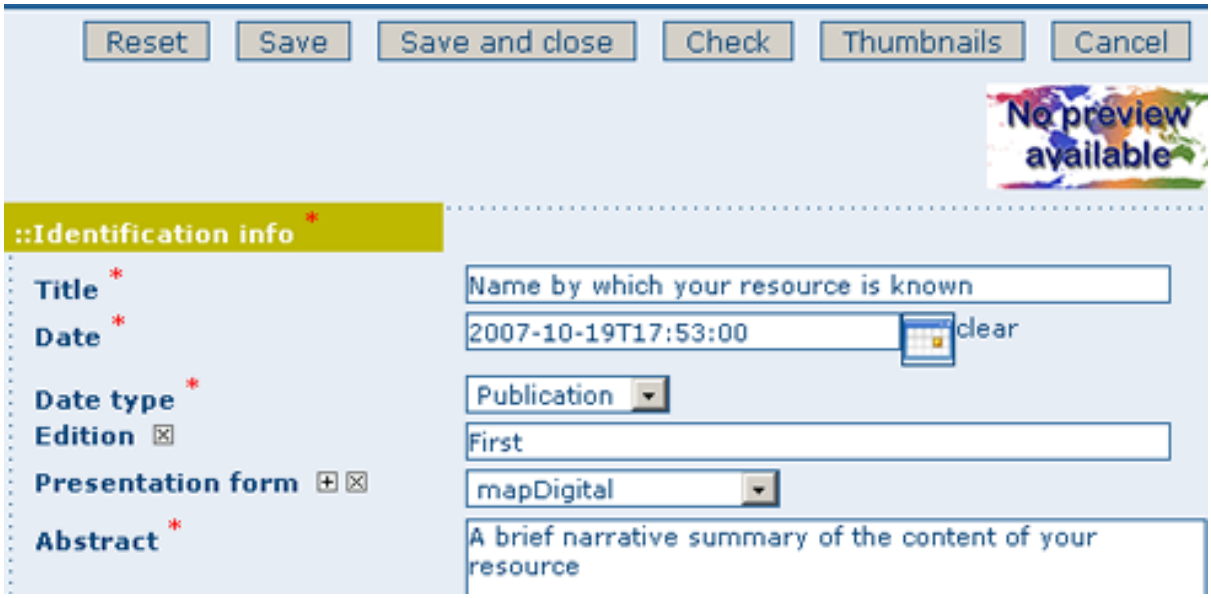


Figure 2.39: *The thumbnail wizard button*

- You will be taken to the Thumbnail Management wizard.
- To create a small or large thumbnail, click on the Browse button next to either one. It is recommended that you use 180 pixels for small thumbnails and 800x600 for large thumbnails. Using the 'Large thumbnail' option allows you to create both a small and large thumbnail in one go.
- You can use GIF, PNG and JPEG images as input for the thumbnails.
- A pop up window will appear allowing you to browse your files on your computer. Select the file you wish to create a thumbnail with by double-clicking on it.
- Click on `Add`.
- Your thumbnail will be added and displayed on the following page.
- You can then click on Back to Editing and save your record.

2.4.8 Compute bounding box from keywords

Editor can add extent information based on keyword analysis.

- For each keywords
- Search for the keyword in thesaurus
- If keyword in the thesaurus has an extent
- Add an extent with a description and a bounding box to the metadata record.

The process could be run in 2 modes :

- Add : Keep existing extent elements and add the new one at the end. Editor could clean the section after processing.

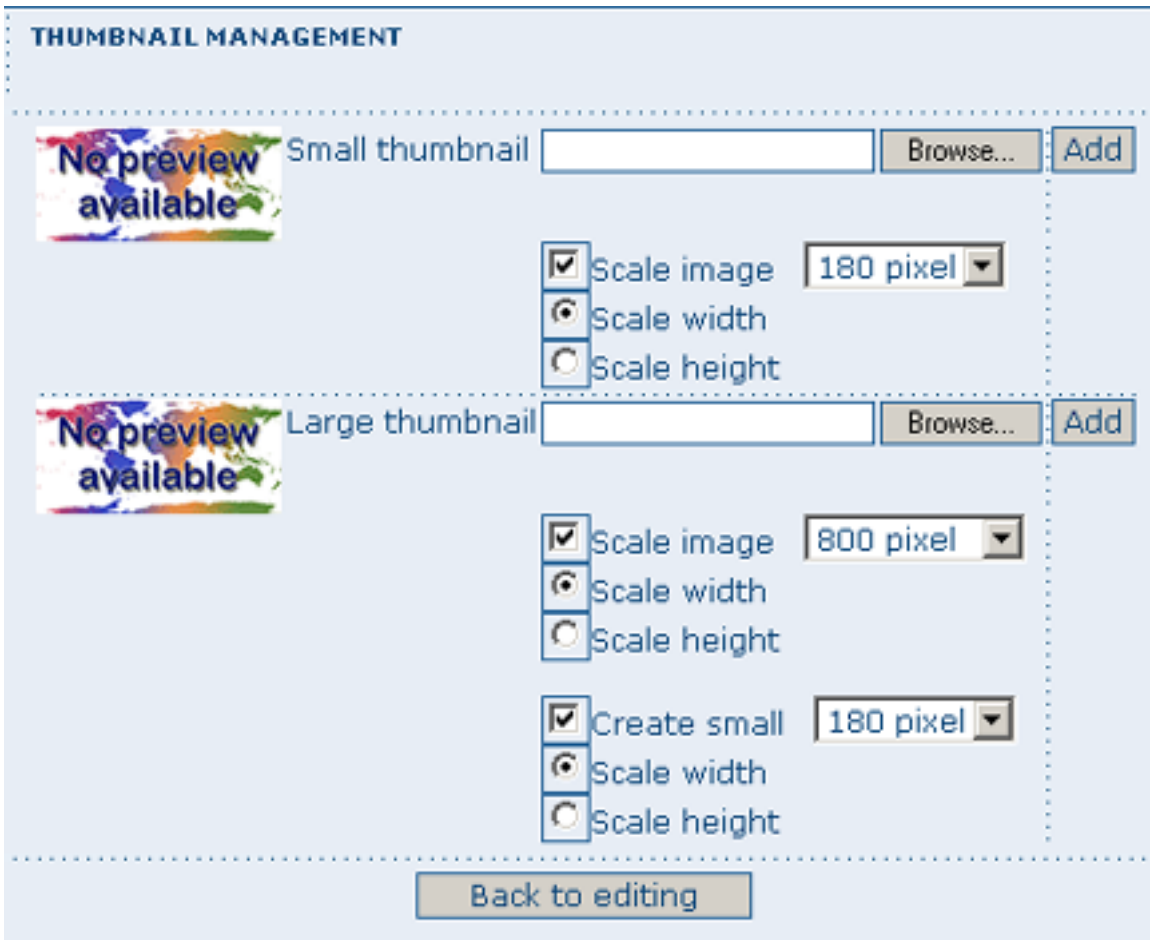


Figure 2.40: *Thumbnail wizard*

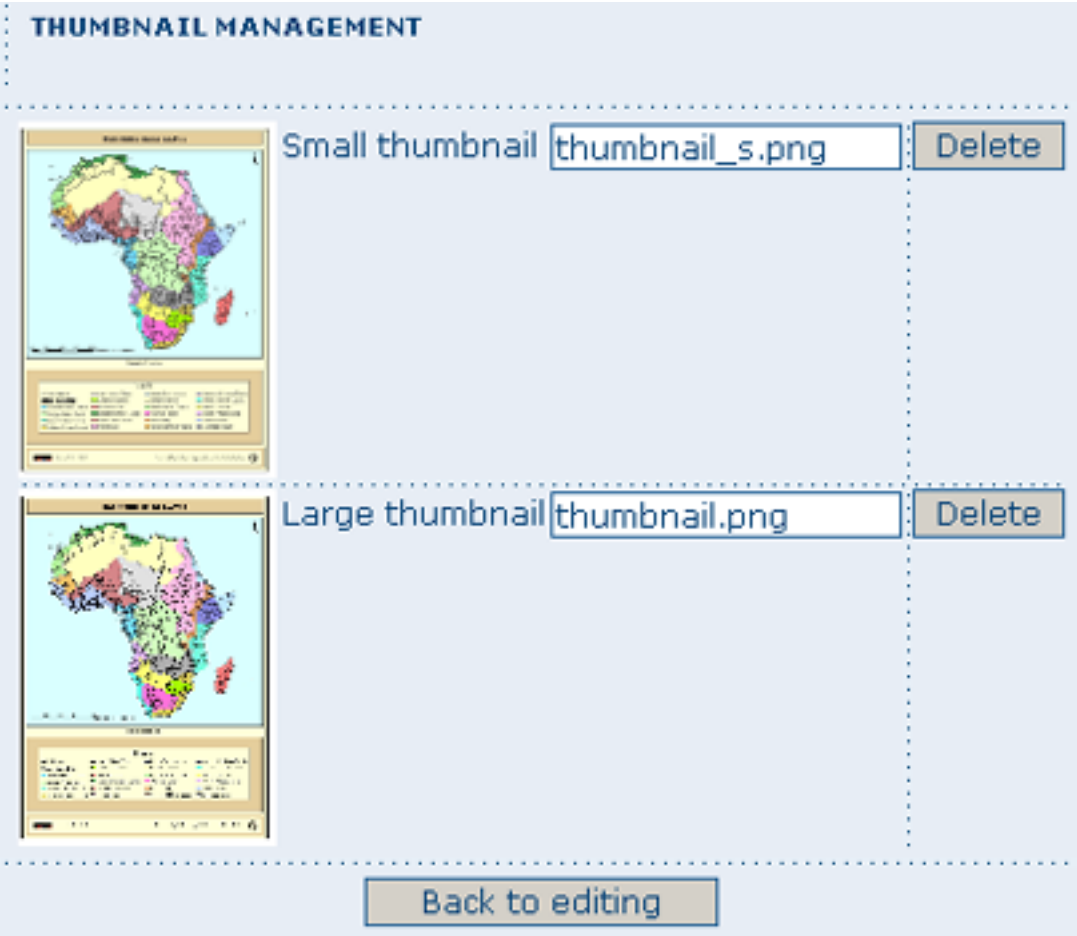
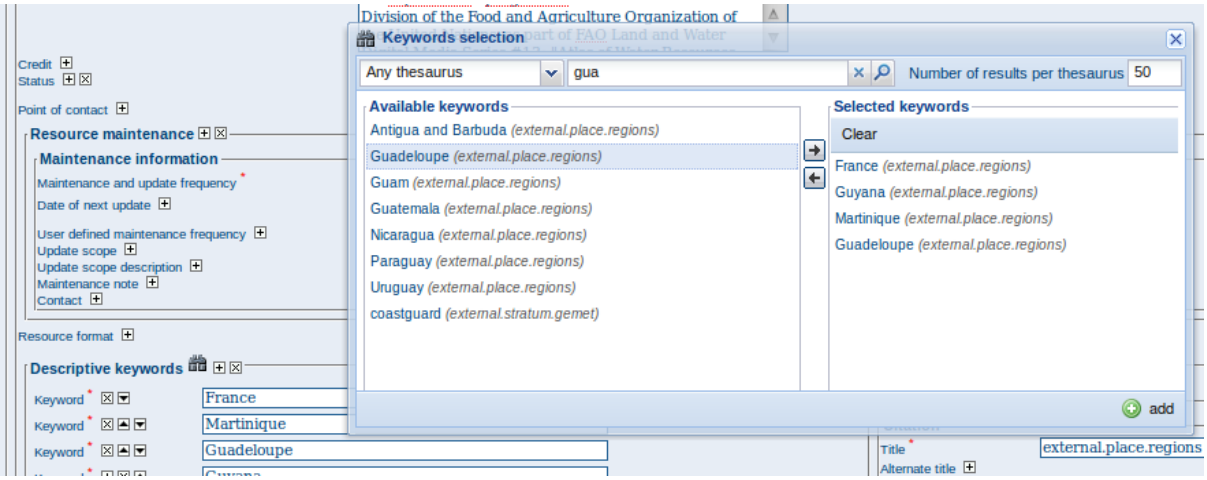


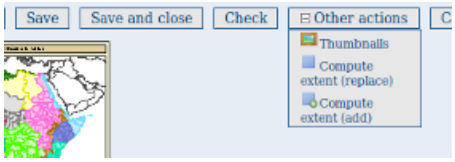
Figure 2.41: *Completed thumbnail wizard*

- Replace : Remove all extent having only a bounding box (temporal, vertical and bounding polygon are not removed), and add the new one at the end.

Editor need to select keyword from a thesaurus with spatial information. The name is added to the extent description field.



Then in the other actions menu, the compute boundinx box menus are available:



The metadata is saved during the process and one extent is added for each keywords.

If user manually add keywords just before computing bounding box, then it's recommended to save your metadata record before launching the action in order to have latest keywords taken into account.

2.4.9 Assigning Privileges

To assign privileges to your metadata record and any attached data you will need to identify User Groups and the privileges you want to assign to users in these groups. eg. View the metadata, download the data attached to the record, etc.

For instance, you can specify that the metadata and related services are visible to all (Internet users) or just to internal users only (Intranet). Privileges are assigned on a per group basis. Depending on the user profile (Guest, Registered User, Editor, Admin etc.) access to these functions may differ on a per user basis.

To assign privileges, follow these steps:

- Find your metadata record by using the search option. Whether you have multiple or single results from the search, on top of the individual record or next to the record you will always see a row of buttons including a Privileges button.
- Click on the Privileges button. A drop down menu will appear from which you can assign certain privileges to specific groups using checkboxes. Simply click on the small box next to the privilege to place or remove a checkmark. Set All and Clear All buttons allow you to place and remove the checkmarks all at once.

Figure 2.42: *The editing toolbar with Privileges button*

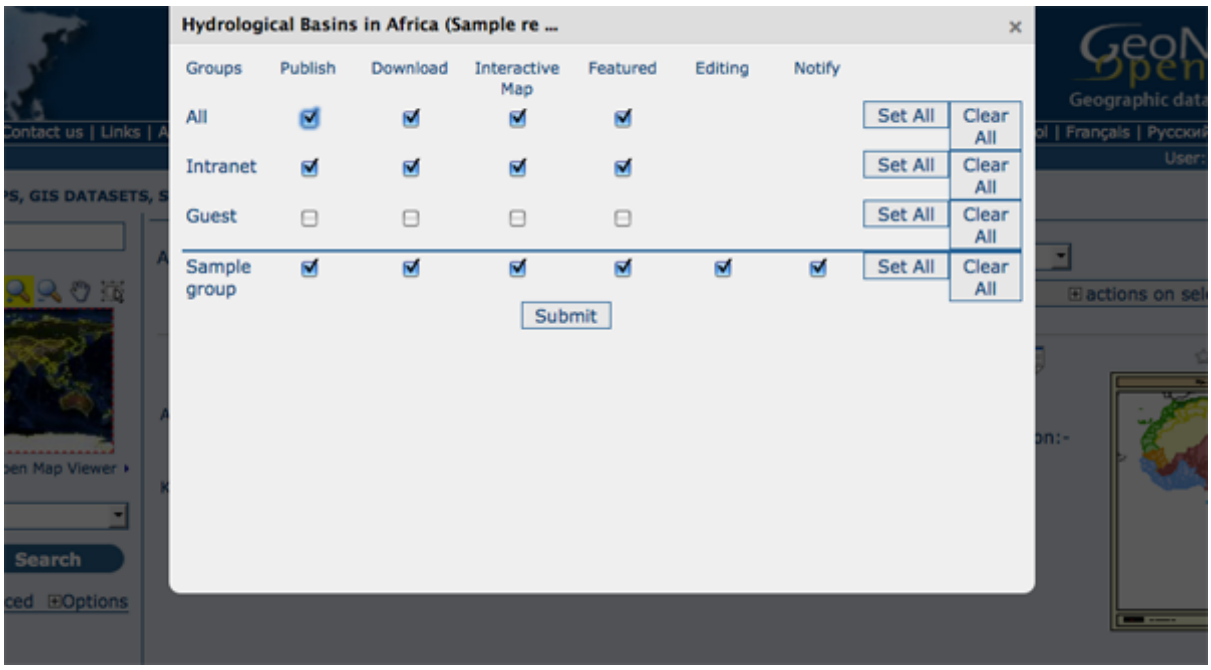


Figure 2.43: *Privileges settings*

Below is a brief description for each privilege to help you identify which ones you should assign to which group(s).

Publish: Users in the specified group/s are able to view the metadata eg. if it matches search criteria entered by such a user.

Download: Users in the specified group/s are able to download the data.

Interactive Map: Users in the specified group/s are able to get an interactive map. The interactive map has to be created separately using a Web Map Server such as GeoServer, which is distributed with GeoNetwork.

Featured: When randomly selected by GeoNetwork, the metadata record can appear in the *Featured* section of the GeoNetwork home page.

Notify: Users in the specified group receive notification if data attached to the metadata record is downloaded.

2.4.10 Assigning Categories

Each GeoNetwork site has a set of local categories that can be used to classify metadata records for easy searching. To assign categories to a metadata record, follow these steps:

- Find your metadata record using the search option. Whether you have just one or many results from your search, you will always see a row of buttons including a **Categories** button.
- Click on the **Categories** button. A drop down menu will appear from which you can assign one or more categories using checkboxes. Simply click on the small box next to the category to place or remove a checkmark.

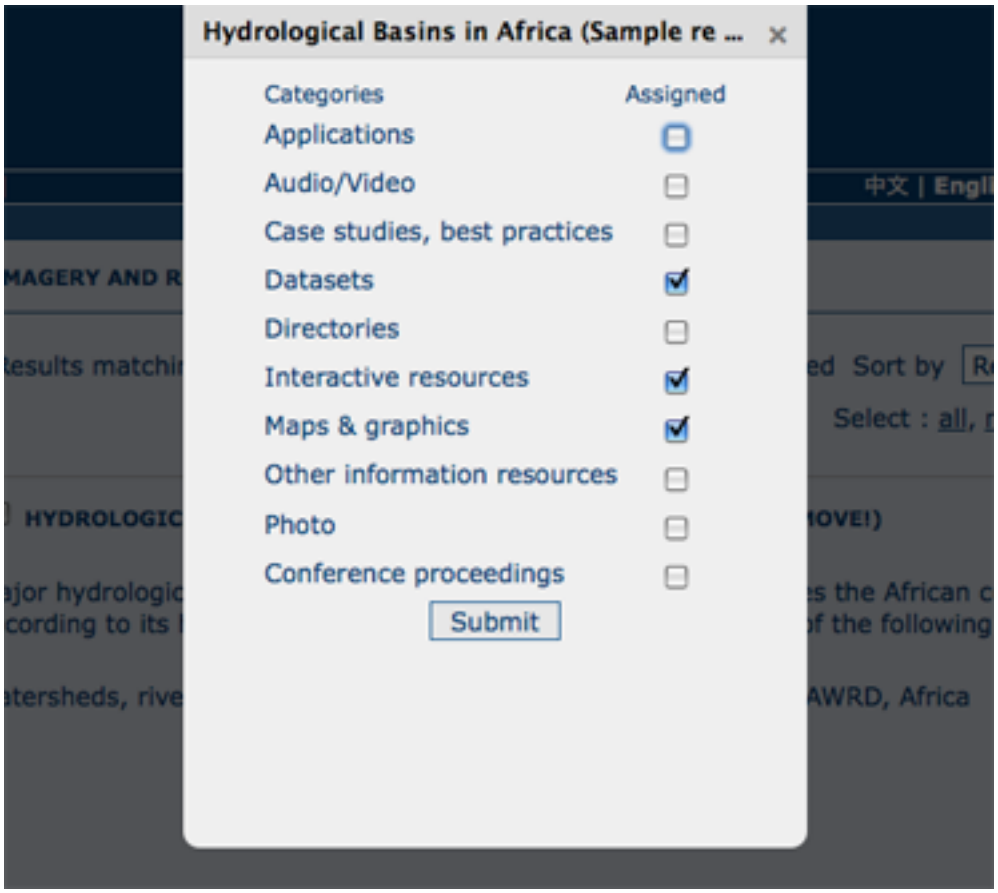


Figure 2.44: *Category management*

2.4.11 Multilingual metadata in ISO19139

Editors can create multilingual metadata for ISO19139. A default template is provided but user could add translation to an existing record.

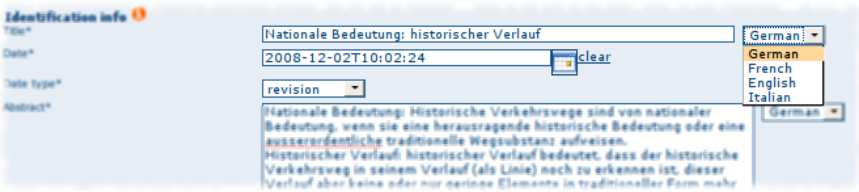
To declare a new language in a metadata record:

- First check, the main language is defined in the metadata section
- then add one or more languages in the other language in the metadata section.

In editing mode, each multilingual elements are composed of:

- text input
- language selection list (language declared on the other language section are listed here)

By default, the selected language is the GUI language if language is defined in the metadata.



Alternatively, Google translation service could be used. Translation could be suggested to the editor using the small icon right to the language selector. The translation convert the default metadata character string in the current selected language.

In view mode, according to GUI language : if GUI language is available in the metadata, the element is displayed in this language else the element is displayed in metadata default language. This behaviour is also applied to dublin core output for CSW services.

2.5 Uploading a New Record using the XML Metadata Insert Tool

A more advanced procedure to upload a new metadata record in the GeoNetwork system is using an XML document. This procedure is particularly useful for users who already have metadata in XML format, for instance created by some GIS application. To this regard, it has to be noted that the metadata must be in one of the standards used by GeoNetwork: ISO19115, FGDC and Dublin Core.

To start the metadata uploading process through the **XML Metadata Insert** tool, you should log in and select the appropriate option from the Administration page.

The main part of the page **Import XML Formatted Metadata** that is displayed is the **Metadata** text area, where the user can paste the XML metadata to import. Below this, there is the **Type** choice, which allows you select the type of record that you are going to create (Metadata, Template and Subtemplate). Then you can apply a stylesheet to convert your metadata input from ArcCatalog8 to ISO1915 or from ISO19115 to ISO19139, if required. Otherwise you can just leave none selected. The **Destination schema** list provides you with four options to choose the final standard layout for your metadata (ISO19115, ISO19139, FGDC and Dublin Core). Finally you should select the **Group** as main group in charge of the metadata and the **Category** that you want to assign to your metadata. By clicking the **Insert** button the metadata is imported into the system; please note that all links to external files, for instance to thumbnails or data for download, have to be removed from the metadata input, to avoid any conflict within the data repository.

Figure 2.45: *Administration panel*

If your metadata is already in ISO19115 format, the main actions to be performed are the following:

1. Paste the XML file that contains the metadata information in the **Metadata** text area;
2. Select Metadata as **type** of record that you are going to create
3. Select the metadata schema ISO19139 that will be the final **destination schema**;
4. Select the **validate** check box if you want your metadata to be validated according to the related schema.
5. Select the **group** in charge of the metadata from the drop down list;
6. Select **Maps and Graphics** from the list of categories;
7. Click the **Insert** button and the metadata will be imported into the system.

2.6 Metadata in Spatial Data Management

2.6.1 What is Metadata?

Metadata, commonly defined as "data about data" or "information about data", is a structured set of information which describes data (including both digital and non-digital datasets) stored in administrative systems. Metadata may provide a short summary about the content, purpose, quality, location of the data as well as information related to its creation.

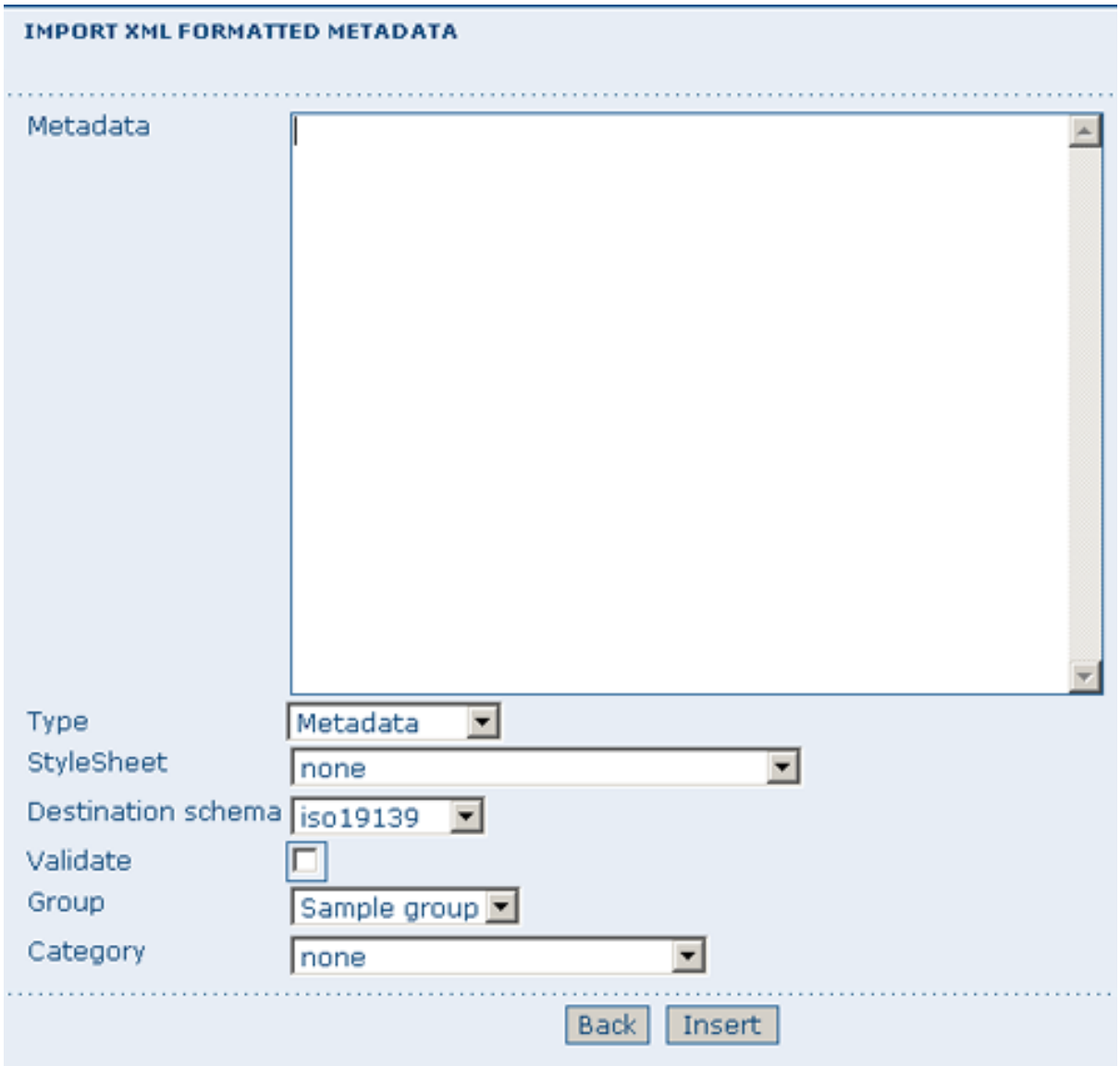


Figure 2.46: *XML metadata import tool*

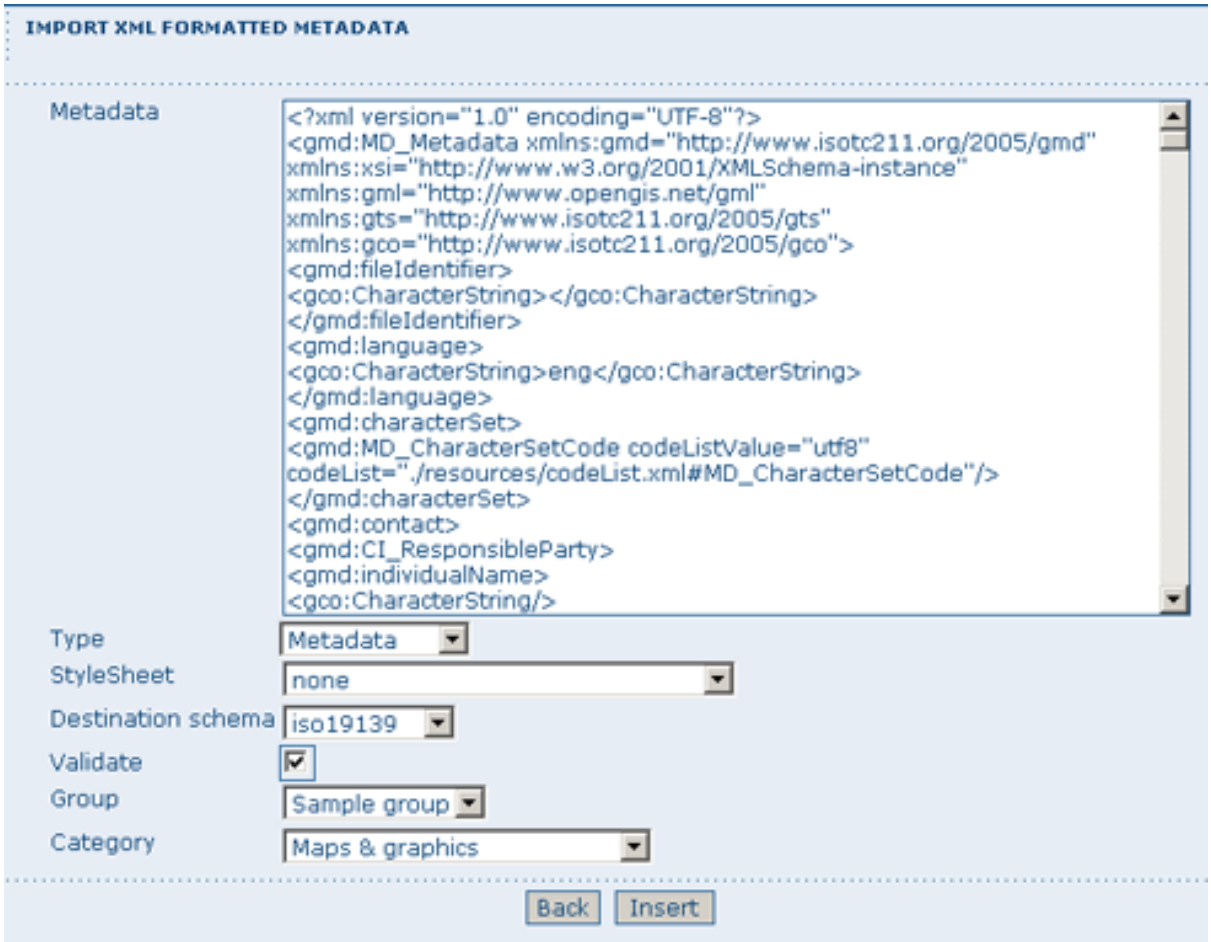


Figure 2.47: *XML metadata import 2*

2.6.2 What are Metadata Standards?

Metadata standards provide data producers with the format and content for properly describing their data, allowing users to evaluate the usefulness of the data in addressing their specific needs.

The standards provide a documented, common set of terms and definitions that are presented in a structured format.

2.6.3 Why do we need Standardised Metadata?

Standardised metadata support users in effectively and efficiently accessing data by using a common set of terminology and metadata elements that allow for a quick means of data discovery and retrieval from metadata clearinghouses. The metadata based on standards ensure information consistency and quality and avoid that important parts of data knowledge are lost.

2.6.4 Geographic Information Metadata Standard

Geographic data, which can be defined as any data with a geographic component, is often produced by one individual or organisation, and may address the needs of various users, including information system analysts, programme planners, developers of geographic information or policy makers. Proper standard documentation on geographic data enable different users to better evaluate the appropriateness of data to be used for data production, storage, update.

The metadata standards supported by GeoNetwork opensource are the **ISO 19115:2003** - approved by the international community in April 2003 as a tool to define metadata in the field of geographic information - and the **FGDC** - the metadata standard adopted in the United States by the Federal Geographic Data Committee. In addition, GeoNetwork opensource supports also the international standard **Dublin Core** for the description of general documents.

This ISO Standard precisely defines how geographic information and related services should be described, providing mandatory and conditional metadata sections, metadata entities and metadata elements. This standard applies to data series, independent datasets, individual geographic features and feature properties. Despite ISO 19115:2003 was designed for digital data, its principles can be extended to many other forms of geographic data such as maps, charts, and textual documents as well as non-geographic data.

The underlying format of an ISO19115:2003 compliant metadata is XML. GeoNetwork uses the *ISO Technical Specification 19139 Geographic information - Metadata - XML schema implementation* for the encoding of this XML.

2.6.5 Metadata profiles

A metadata profile is an adaptation of a metadata standard to suit the needs of a community. For example, the ANZLIC profile is an adaptation of the ISO19115/19139 metadata standard for Australian and New Zealand communities. A metadata profile could be implemented as:

- a specific metadata *template* that restricts the fields/elements a user can see with a set of validation rules to check compliance
- all of the above plus new fields/elements to capture concepts that aren't in the basic metadata standard

Building a metadata profile is described in the Schema Plugins section of the GeoNetwork Developers Manual. Using this guide and the GeoNetwork schema plugin capability, a profile can be built by an experienced XML/XSL software engineer.

2.6.6 Transition between metadata standards

With the ISO19115:2003 Metadata standard for Geographic Information now being the preferred common standard, many have a need to migrate legacy metadata into the new standard.

GeoNetwork provides import (and export) functionality and has a number of transformers in place. It is an easy process for a system administrator to install custom transformers based on XSLT.

2.7 New Features

The new GeoNetwork opensource comes with substantial upgrades of different components.

2.7.1 2.10 release

Search

- **Faceted search**: Narrow your search by easily selecting new filter
- **Data Catalog Vocabulary and RDF services**: Increase discoverability and enable applications easily to consume metadata using W3C DCAT format
- **Javascript widget user interface** : A 3rd flavour of home page based on HTML5 is also available

Metadata

- **Metadata on maps**: Add template for making metadata on static or interactive maps. Add a search criteria for easily found maps. Web Map Context could be loaded into the map viewer.
- **Metadata linked data**: Easier metadata relation configuration and new support for source dataset and siblings. See *metadata_link*)
- **Hide part of metadata**: Provide a method to hide portions of metadata using withHeld ISO attribute
- **Wiki markup in metadata**: Allow users to enter markup text in metadata elements and have results shown with rendered html
- **WFS data downloader**: Simple component to download WFS data

Administration

- **User profile**: Setup user belonging to multiple groups with different profiles
- **Virtual CSW** configuration interface to add new end points

Others

- **Security layer**: New security layer based on Spring Security adding support for CAS and much more flexible LDAP configuration
- **Xlink**: Add local:// as a protocal for xlink links
- Provide basic functionnalities (ie. search and view) when **database is in readonly**

2.7.2 2.8 release

User interface

- **Javascript widget user interface:** A new user interface using one of the latest Javascript widget libraries (extJS) has been added to support searching, editing and viewing metadata records. The user interface is now much easier for Javascript developers to reorganize and customize. GeoNetwork comes with two flavours of home page: one has the sidebar search similar to the old interface and the other uses a tabbed search layout. The 2.6.x user interface is still available as the default and has been updated.

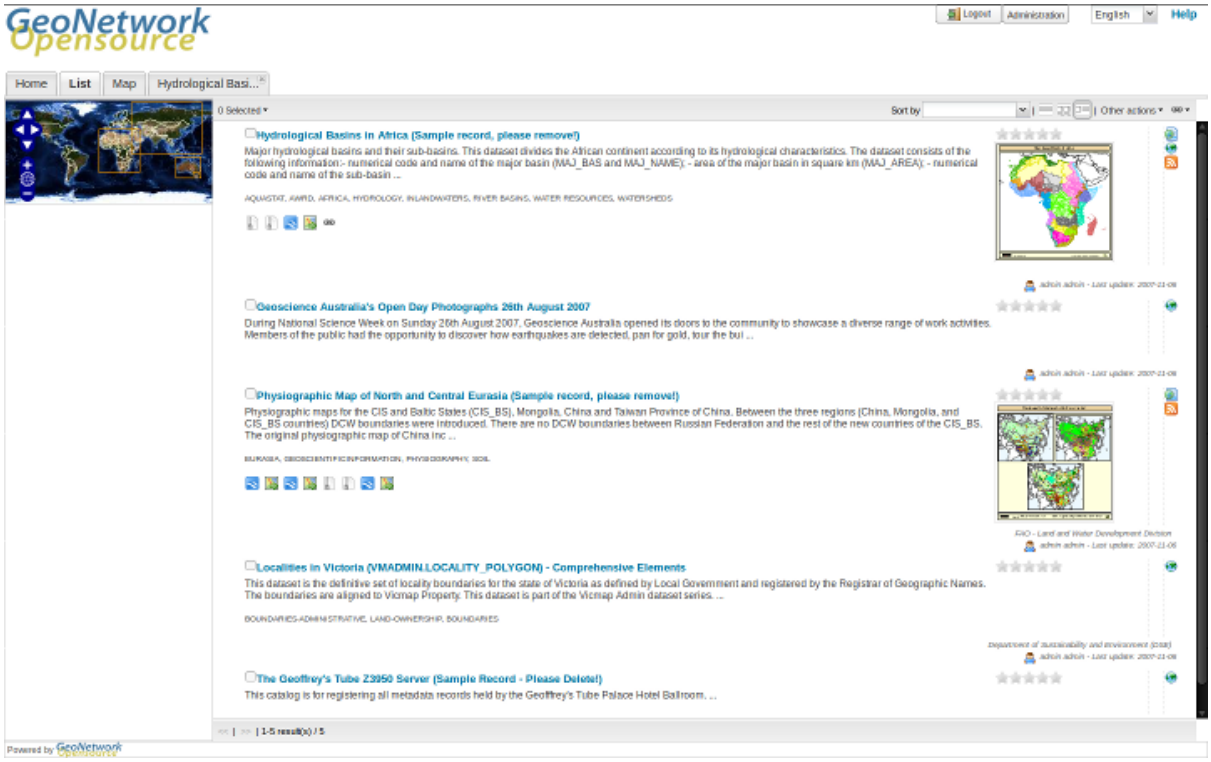


Figure 2.48: *New home page of GeoNetwork opensource using JavaScript Widgets - tab layout*

Administration

- **Search Statistics:** Captures and displays statistics on searches carried out in GeoNetwork. The statistics can be summarized in tables or in charts using JFreeChart. There is an extensible interface that you can use to display your own statistics. See *Search Statistics*.
- **New Harvesters:** OGC Harvesting: Sensor Observation Service, Z3950 harvesting, Web Accessible Folder (WAF), GeoPortal 9.3.x via REST API See *Harvesting*.

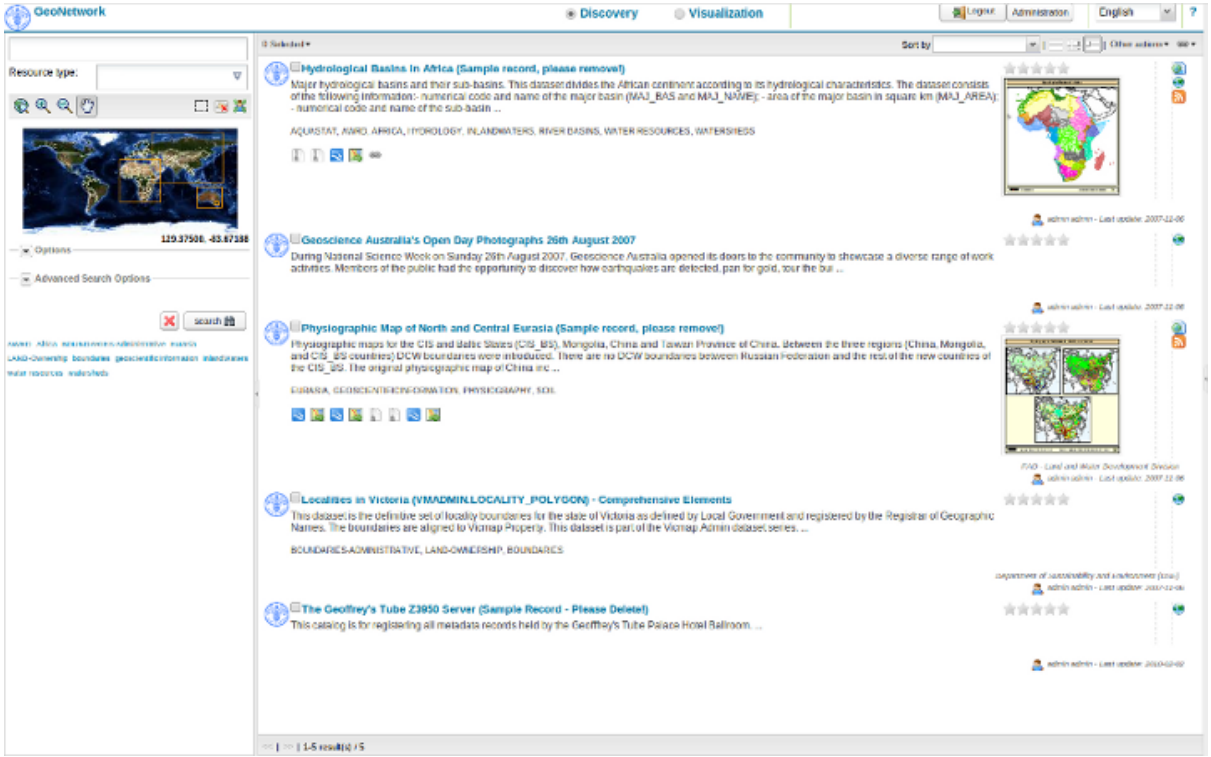


Figure 2.49: *New home page of GeoNetwork opensource using JavaScript Widgets- sidebar layout*

- **Harvest History and Scheduling:** Harvesting events are now recorded in the database for review at any time. See *Harvest History*. Harvester scheduling is now much more flexible, you can start a harvest at any time of the day and at almost any interval (weekly etc).
- **Extended Metadata Exchange Format (MEF):** More than one metadata file can be present in a MEF Zip archive. This is MEF version 2. See *Export facilities*.
- **System Monitoring:** Automatically monitoring the health of a Geonetwork web application. See *System Monitoring*.

Metadata

- **Metadata Status:** Allows finer control of the metadata workflow. Records can be assigned a status that reflects where they are in the metadata workflow: draft, approved, retired, submitted, rejected. When the status changes the relevant user is informed via email. eg. when an editor changes the status to 'submitted', the content reviewer receives an email requesting review. See *Status*.
- **Metadata Versioning:** Captures changes to metadata records and metadata properties (status, privileges, categories) and records them as versions in a subversion respository. See *Versioning*.
- **Publishing data to GeoServer from GeoNetwork:** You can now publish geospatial information in the form of GeoTIFF, shapefile or spatial table in a database to GeoServer from GeoNetwork. See *GeoPublisher*.
- **Custom Metadata Formatters:** You can now create your own XSLT to format metadata to suit your needs, zip it up and plug it in to GeoNetwork. See *Formatter*.
- **Assembling Metadata Records from Reusable Components:** Metadata records can now be as-

sembled from reusable components (eg. contact information). The components can be present in the local catalog or brought in from a remote catalog (with caching to speed up access). A component directory interface is available for editing and viewing the components. See *Fragments*.

- **Editor Improvements:** Picking terms from a thesaurus using a search widget, selecting reusable metadata components for inclusion in the record, user defined suggestions or picklists to control content, context sensitive help, creating relationships between records.
- **Plug in metadata schemas:** You can define your own metadata schema and plug it into GeoNetwork on demand. Documentation to help you do this and example plug in schemas can be found in the Developers Manual. Some of the most common community plug in schemas can be downloaded from the GeoNetwork source code repository. See *Schemas*.
- **Multilingual Indexing:** If you have to cope with metadata in different languages, GeoNetwork can now index each language and search all across language indexes by translating your search terms. See *Multilingual search*.
- **Enhanced Thesaurus support:** Thesauri can be loaded from ISO19135 register records and SKOS files. Keywords in ISO records are anchored to the definition of the concept in the thesaurus. See *Thesaurus*.

CSW service

- **Virtual CSW Endpoints:** Now you can define a custom CSW service that works with a set of metadata records that you define. See *Virtual CSW server entry points*.

INSPIRE Directive

- **Support for the INSPIRE Directive:** Indexing and user interface extensions to support those who need to implement the INSPIRE metadata directive (EU).
- **Installer package to enable INSPIRE options:** An optional new package in the installer enables GeoNetwork INSPIRE features if selected, avoiding manual steps to enable INSPIRE support.

Other

- **Improved Database Connection Handling and Pooling:** Replacement of the Jeeves based database connection pool with the widely used and more robust Apache Database Connection Pool (DBCP). Addition of JNDI or container based database connection support. See *Database configuration*.
- **Configuration Overrides:** Now you can add your own configuration options to GeoNetwork, keep them in one file and maintain them independently from GeoNetwork. See *Configuration override*.
- **Many other improvements:** charset detection and conversion on import, batch application of an XSLT to a selected set of metadata records (see *Processing*), remote notification of metadata changes, automatic integration tests to improve development and reduce regression and, of course, many bug fixes.

2.8 Installing the software

2.8.1 Where do I get the installer?

The software is distributed through the SourceForge.net Website at http://sourceforge.net/projects/geonetwork.

Use the platform independent installer (.jar) for all platforms except Windows. Windows has a .exe file installer.

2.8.2 System requirements

GeoNetwork can run either on **MS Windows** , **Linux** or **Mac OS X** .

Some general system requirements for the software to run without problems are listed below:

Processor : 1 GHz or higher

Memory (RAM) : 1 GB or higher

Disk Space : Minimum of 512MB of free disk space. Additional space is required depending on the amount of spatial data that you expect to upload.

Other Software requirements : A Java Runtime Environment (JRE 1.6.0). For server installations, Apache Tomcat and a dedicated JDBC compliant DBMS (MySQL, Postgresql, Oracle) can be used instead of Jetty and H2.

Additional Software

The software listed here is not required to run GeoNetwork, but can be used for custom installations.

1. MySQL DBMS v5.5+ (All) [1]
2. Postgresql DBMS v7+ (All) [1]
3. Apache Tomcat v5.5+ (All) [1]

Supported browsers

GeoNetwork should work normally with the following browsers:

1. Firefox v1.5+ (All) [1]
2. Internet Explorer v8+ (Windows)
3. Safari v3+ (Mac OS X Leopard)

2.8.3 How do I install GeoNetwork opensource?

Before running the GeoNetwork installer, make sure that all system requirements are satisfied, and in particular that the Java Runtime Environment version 1.6.0 is set up on your machine.

[1] All = Windows, Linux and Mac OS X

On Windows

If you use Windows, the following steps will guide you to complete the installation (other FOSS will follow):

> **Warning:** Avoid installing in a directory containing spaces. Best is to install in `c:\programs` and not in `c:\program files`

1. Double click on **geonetwork-install-2.10.x.exe** to start the GeoNetwork opensource desktop installer
2. Follow the instructions on screen. You can choose to install the embedded map server (based on GeoServer and the European Union Inspire Directive configuration pack. Developers may be interested in installing the source code and installer building tools. Full source code can be found in the GeoNetwork github code repository at http://github.com/geonetwork.
3. After completion of the installation process, a 'GeoNetwork desktop' menu will be added to your Windows Start menu under 'Programs'
4. Click Start>Programs>GeoNetwork desktop>Start server to start the Geonetwork opensource Web server. The first time you do this, the system will require about 1 minute to complete startup.
5. Click Start>Programs>Geonetwork desktop>Open GeoNetwork opensource to start using GeoNetwork opensource, or connect your Web browser to http://localhost:8080/geonetwork/

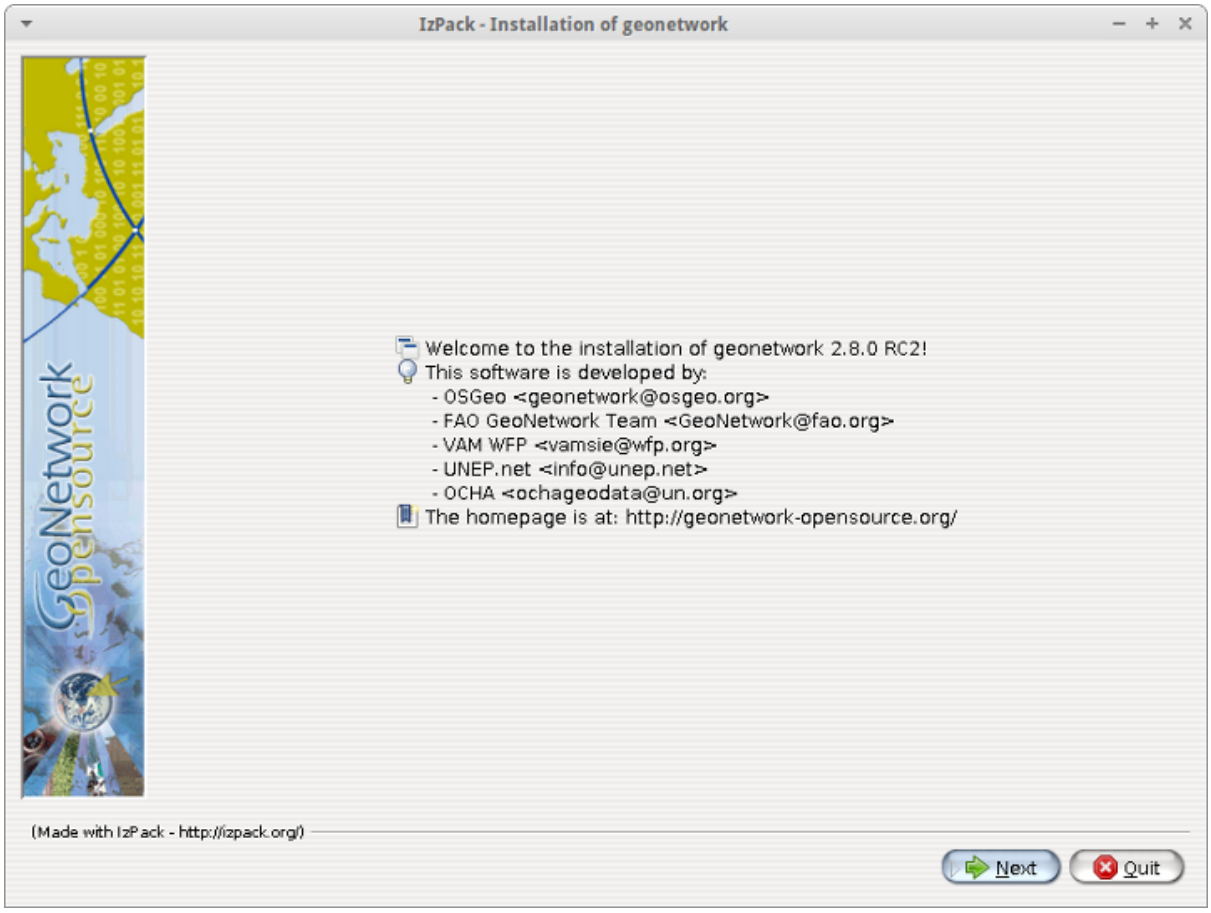


Figure 2.50: *Installer*

The installer allows to install these additional packages:

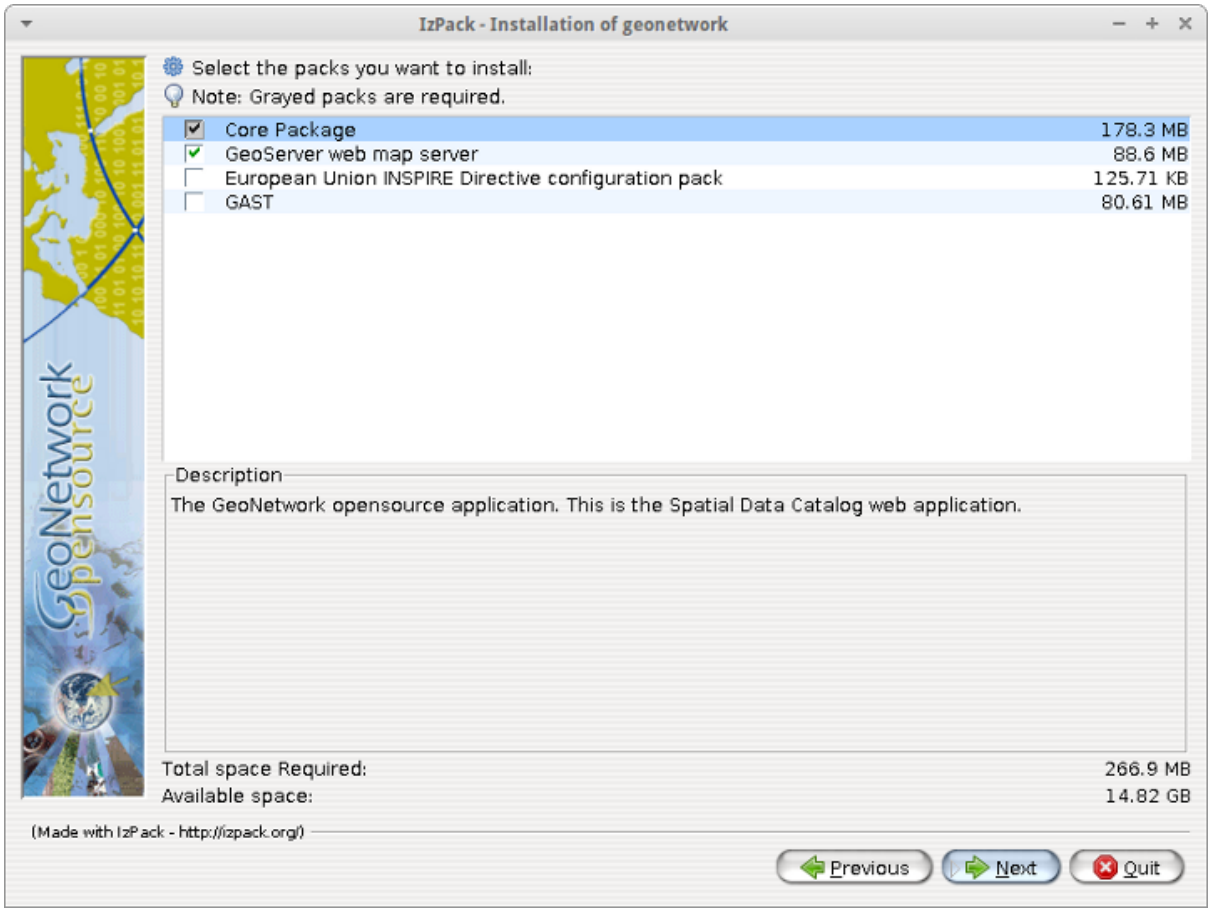


Figure 2.51: *Packages to be installed*

1. GeoNetwork User Interface: Experimental UI for GeoNetwork using javascript components based on ExtJs library.
2. GeoServer: Web Map Server that provides default base layers for the GeoNetwork map viewer.
3. European Union INSPIRE Directive configuration pack: Enables INSPIRE support in GeoNetwork.
   - INSPIRE validation rules.
   - Thesaurus files (GEMET, Inspire themes).
   - INSPIRE search panel.
   - INSPIRE metadata view.
4. GAST: Installs GeoNetwork's Administrator Survival Tool. See *gast*.

Installation using the platform independent installer

If you downloaded the platform independent installer (a .jar file), you can in most cases start the installer by simply double clicking on it.

Follow the instructions on screen (see also the section called On Windows).

At the end of the installation process you can choose to save the installation script.

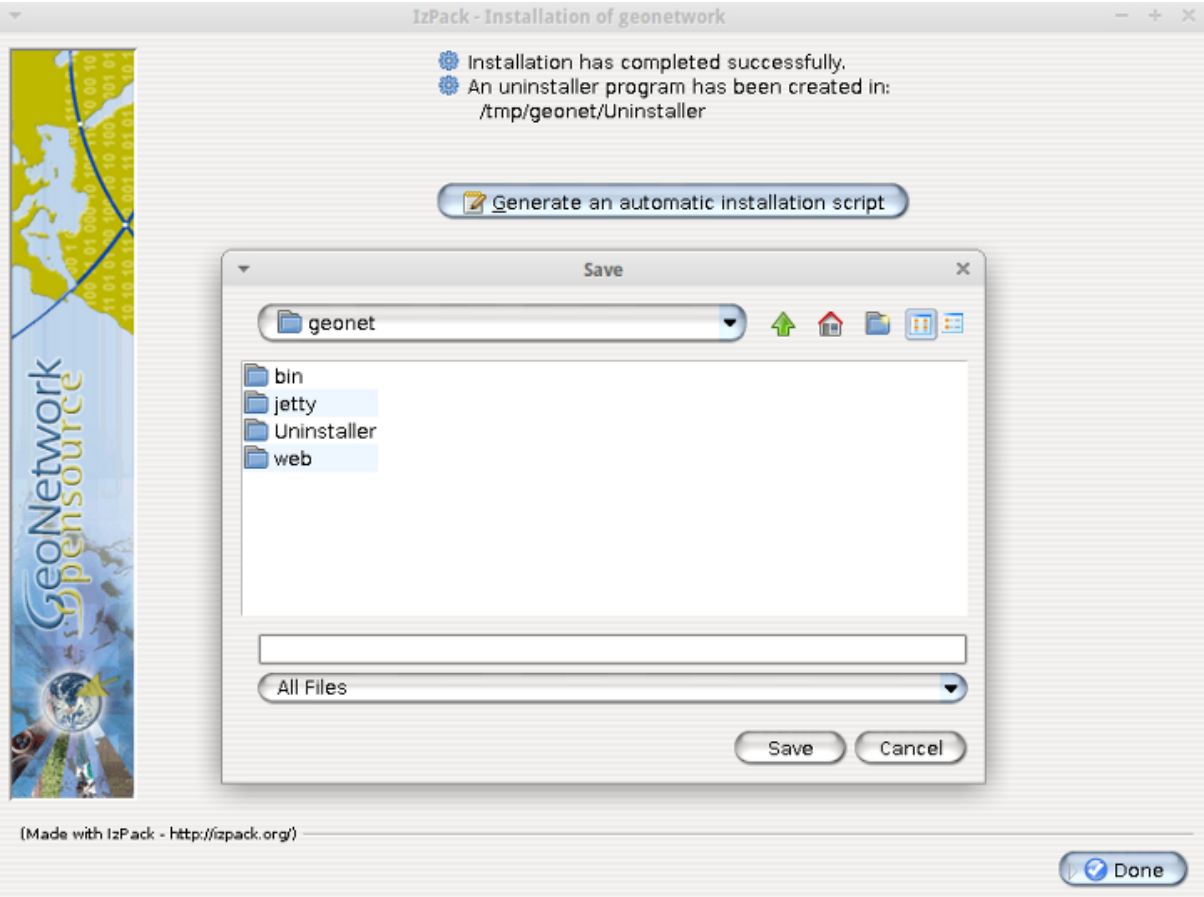


Figure 2.52: *Save the installation script for commandline installations*

Commandline installation

If you downloaded the platform independent installer (a .jar file), you can perform commandline installations on computers without a graphical interface. You first need to generate an install script (see Figure Save the installation script for commandline installations). This install script can be edited in a text editor to change some installation parameters.

To run the installation from the commandline, issue the following command in a terminal window and hit enter to start:

```
java -jar geonetwork-install-2.10.0.jar install.xml
[ Starting automated installation ]
Read pack list from xml definition.
Try to add to selection [Name: Core and Index: 0]
Try to add to selection [Name: GeoServer and Index: 1]
Try to add to selection [Name: European Union INSPIRE Directive configuration pack and I
Try to add to selection [Name: GAST and Index: 3]
Modify pack selection.
Pack [Name: European Union INSPIRE Directive configuration pack and Index: 2] added to s
Pack [Name: GAST and Index: 3] added to selection.
[ Starting to unpack ]
[ Processing package: Core (1/4) ]
[ Processing package: GeoServer (2/4) ]
[ Processing package: European Union INSPIRE Directive configuration pack (3/4) ]
[ Processing package: GAST (4/4) ]
[ Unpacking finished ]
[ Creating shortcuts ..... done. ]
[ Add shortcuts to uninstaller  done. ]
[ Writing the uninstaller data ... ]
[ Automated installation done ]
```

You can also run the installation with lots of debug output. To do so run the installer with the flag *-DTRACE=true*:

```
java -DTRACE=true -jar geonetwork-install-2.10.0.jar
```

2.8.4 User interface configuration

As mentioned above, GeoNetwork now provides two user interfaces:

- **Default** user interface is the old user interface from 2.6.x and earlier
- **Javascript Widgets** user interface is the new user interface for searching, editing and viewing metadata records in 2.8.x

The catalog administrator can configure which interface to use in *WEB-INF/config-gui.xml* as follows.

Configuring the Default user interface

WEB-INF/config-gui.xml is used to define which home page to use. To configure the Default user interface use:

```
<client type="redirect"
  widget="false"
  url="main.home"
  parameters=""
```

```
  stateId=""
  createParameter=""/>
```

Configuring the Javascript Widgets user interface

Widgets can be used to build custom interfaces. GeoNetwork provides a Javascript Widgets interface for searching, viewing and editing metadata records.

This interface can be configured using the following attributes:

- **parameter** is used to define custom application properties like default map extent for example or change the default language to be loaded
- **createParameter** is appended to URL when the application is called from the administration > New metadata menu (usually "#create").
- **stateId** is the identifier of the search form (usually "s") in the application. It is used to build quick links section in the administration and permalinks.

Sample configuration:

```
<!-- Widget client application with a tab based layout -->
<client type="redirect"
  widget="true"
  url="../../apps/tabsearch/"
  createParameter="#create"
  stateId="s"/>
```

Configuring the user interface with configuration overrides

Instead of changing config-gui.xml file, the catalog administrator could use the configuration overrides mechanism to create a custom configuration (See *Configuration override*). By default, no overrides are set and the Default user interface is loaded.

To configure which user interface to load, add the following line in WEB-INF/config-overrides.xml in order to load the Widgets based user interface:

```
<override>/WEB-INF/config-overrides-widgettab.xml</override>
```

2.8.5 XSLT processor configuration

The file `INSTALL_DIR/web/geonetwork/WEB-INF/classes/META-INF/javax.xml.transform.Tra` defines the XSLT processor to use in GeoNetwork. The allowed values are:

1. `de.fzi.dbs.xml.transform.CachingTransformerFactory`: This is the Saxon XSLT processor with caching (recommended value for production use). However, when caching is on, any updates you make to stylesheets may be ignored in favour of the cached stylesheets.
2. `net.sf.saxon.TransformerFactoryImpl`: This is the Saxon XSLT processor *without* caching. If you plan to make changes to any XSLT stylesheets you should use this setting until you are ready to move to production.

GeoNetwork sets the XSLT processor configuration using Java system properties for an instant in order to obtain its TransformerFactory implementation, then resets it to the original value, to minimize affect the XSL processor configuration for other applications that may be running in the same container.

2.8.6 Database configuration

Geonetwork uses the H2 database engine as default. The following additional database backends are supported (listed in alphabetical order):

- DB2
- H2
- Mckoi
- MS SqlServer 2008
- MySQL
- Oracle
- PostgreSQL (or PostGIS)

To configure one of these databases for use by GeoNetwork, three steps are required.

Choose a Database Connection Pool

To manage connections with the database efficiently, a database connection pool is used. GeoNetwork uses the Apache Database Connection Pool (DBCP). This connection pool can be configured directly in the config.xml file described below or in Jetty/tomcat through the Java Naming and Directory Interface (JNDI).

- **ApacheDBCPool**: This pool is recommended for smaller catalogs (less than 10,000 records).
- **JNDIPool**: This pool is configured in Jetty or Tomcat. It is recommended for larger catalogs (especially those with more than approx 30,000 records).

More details about the DBCP configuration parameters that can be used here are in the advanced configuration section of this manual (See *Database configuration*).

Download and install JDBC Drivers

For the Apache DBCP pool, JDBC database driver jar files should be in **INSTALL_DIR/WEB-INF/lib**. For Open Source databases, like MySQL and PostgreSQL, the jar files are already installed. For commercial databases like Oracle, the jar files must be downloaded and installed manually. This is due to licensing issues.

- DB2 JDBC driver download
- MS Sql Server JDBC driver download
- Oracle JDBC driver download

Specify configuration in GeoNetwork

GAST provides a graphical user interface to make database configuration easy. You can find out how to do this in the GAST section of the manual: *gast*.

Alternatively you can manually configure the database by editing **INSTALL_DIR/WEB-INF/config.xml**. In the resources element of this file, you will find a resource element for each database

that GeoNetwork supports. Only one of these resource elements can be enabled. The following is an example for the default H2 database used by GeoNetwork:

```
<resource enabled="true">
  <name>main-db</name>
  <provider>jeeves.resources.dbms.ApacheDBCPool</provider>
  <config>
    <user>admin</user>
    <password>gnos</password>
    <driver>org.h2.Driver</driver>
    <url>jdbc:h2:geonetwork;MVCC=TRUE</url>
    <poolSize>33</poolSize>
    <validationQuery>SELECT 1</validationQuery>
  </config>
</resource>
```

If you want to use a different database, then you need to set the enabled attribute on your choice to "true" and set the enabled attribute on the H2 database to "false". **NOTE:** If two resources are enabled, GeoNetwork will **not** start.

As a minimum, the **<user>** , **<password>** and **<url>** for your database need to be changed. Here is an example for the DB2 database:

```
<resource enabled="true">
  <name>main-db</name>
  <provider>jeeves.resources.dbms.ApacheDBCPool</provider>
  <config>
    <user>db2inst1</user>
    <password>mypassword</password>
    <driver>com.ibm.db2.jcc.DB2Driver</driver>
    <url>jdbc:db2:geonet</url>
    <poolSize>10</poolSize>
    <validationQuery>SELECT 1 FROM SYSIBM.SYSDUMMY1</validationQuery>
  </config>
</resource>
```

2.8.7 Starting up GeoNetwork with a new database

At startup, GeoNetwork checks if the database tables it needs are present in the currently configured database. If not, the tables are created and filled with initial data.

If the database tables are present but were created with an earlier version of GeoNetwork, then a migration script is run.

An alternative to running these scripts automatically is to execute them manually. This is preferable for those that would like to examine and monitor the changes being made to their database tables.

- The scripts for initial setup are located in **INSTALL_DIR/WEB-INF/classes/setup/sql/create/**
- The scripts for inserting initial data are located in **INSTALL_DIR/WEB-INF/classes/setup/sql/data/**
- The scripts for migrating are located in **INSTALL_DIR/WEB-INF/classes/setup/sql/migrate/**

2.8.8 Issues or exceptions with databases

If you run into problems when you start GeoNetwork with a particular database, you may find a solution in the *Specific Database Issues* section of this manual.

2.9 Upgrading to a new Version

The upgrade process from one version to another is typically a fairly simple process. Following the normal setup instructions, should result in Geonetwork successfully upgrading the internal datastructures from the old version to the new version. The exceptions to this rule are:

- Migration to Geonetwork 2.8 will reset all harvesters to run every 2 hours. This is because the underlying harvester scheduler has been changed and the old schedules are not longer supported. In this case one must review all the harvesters and define new schedules for them.

CHAPTER 3

Administration

3.1 System configuration

Many GeoNetwork System configuration parameters can be changed using the web interface. Database parameters can be changed using the GAST application.

Important: Configuration of these parameters is critically important for for a GeoNetwork catalogue in an operational context. Misunderstanding these settings may result in a system that does not function as expected. For example, downloads may fail to be correctly processed, or metadata harvesting from other servers may not work.

To get to the System configuration parameters, you must be logged on as administrator first. Open the Administration page and select System configuration (The link is inside the red ellipse).

Important: New installations of GeoNetwork use admin for both username and password. It is important to change the password using the links in the Administration page the first time you log on!

Clicking the link bring up the system configuration menu. A detailed description of these parameters follows.

Note: at the bottom of the page (you will need to scroll down) there are three buttons with the following purpose:

- **Back** Simply returns to the main administration page, ignoring any changes you may have made.
- **Save** Saves the current options. If some options are invalid, the system will show a dialogue with the wrong parameter and will focus its text field on the page. Once the configuration is saved a success dialogue will be shown.
- **Refresh** Reads the settings from the database again and refreshes the options with those values.

3.1.1 Site parameters

Catalogue identifier A universally unique identifier (uuid) that distinguishes your catalogue from any other catalogue. This a unique identifier for your catalogue and its best to leave it as a uuid.

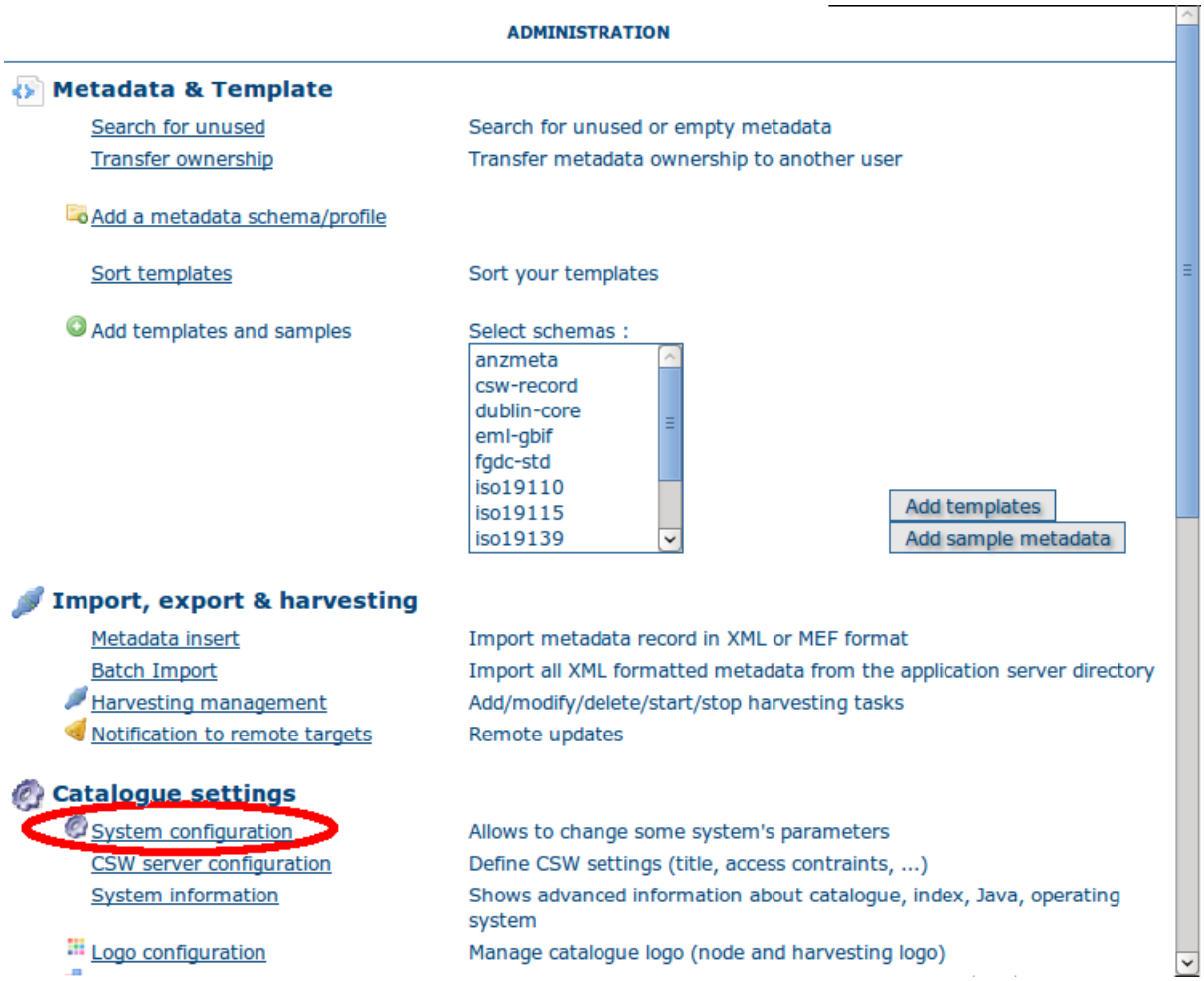


Figure 3.1: *The link to the System configuration page*

SYSTEM CONFIGURATION

SITE

Catalogue identifier: 783a30f7-2ab0-4bc4-b906-9c74b6c7d7

Name My GeoNetwork catalogue

Organization My organization

SERVER

Protocol http

Host localhost

Port 8080

INTRANET

Network 127.0.0.1

Netmask 255.0.0.0

METADATA SEARCH RESULTS

Maximum selected records 1000

MULTI-THREADED INDEXING

Number of processing threads : 1 (Database not tested or doesn't support)

LUCENE INDEX OPTIMIZER

Enable

Run at 0 : 0 (hour of day : minute eg. 23:15)

Will run again every 1 day

Z39.50 SERVER

Enable

Port 2100

OAI PROVIDER

Datesearch temporal extent

ResumptionToken Timeout 3600

Cachesize 60

Figure 3.2: *The configuration options - part 1*

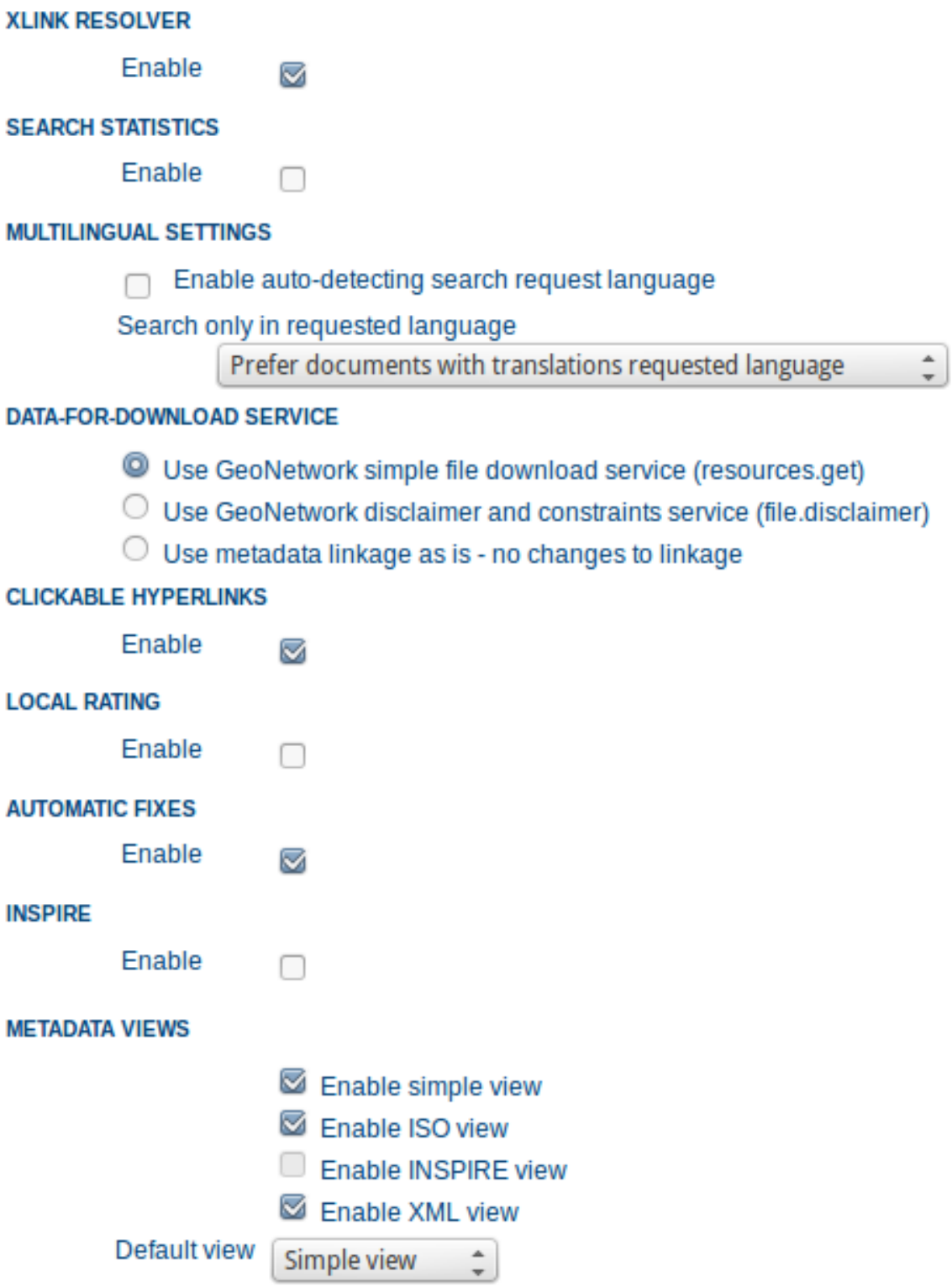


Figure 3.3: *The configuration options - part 2*

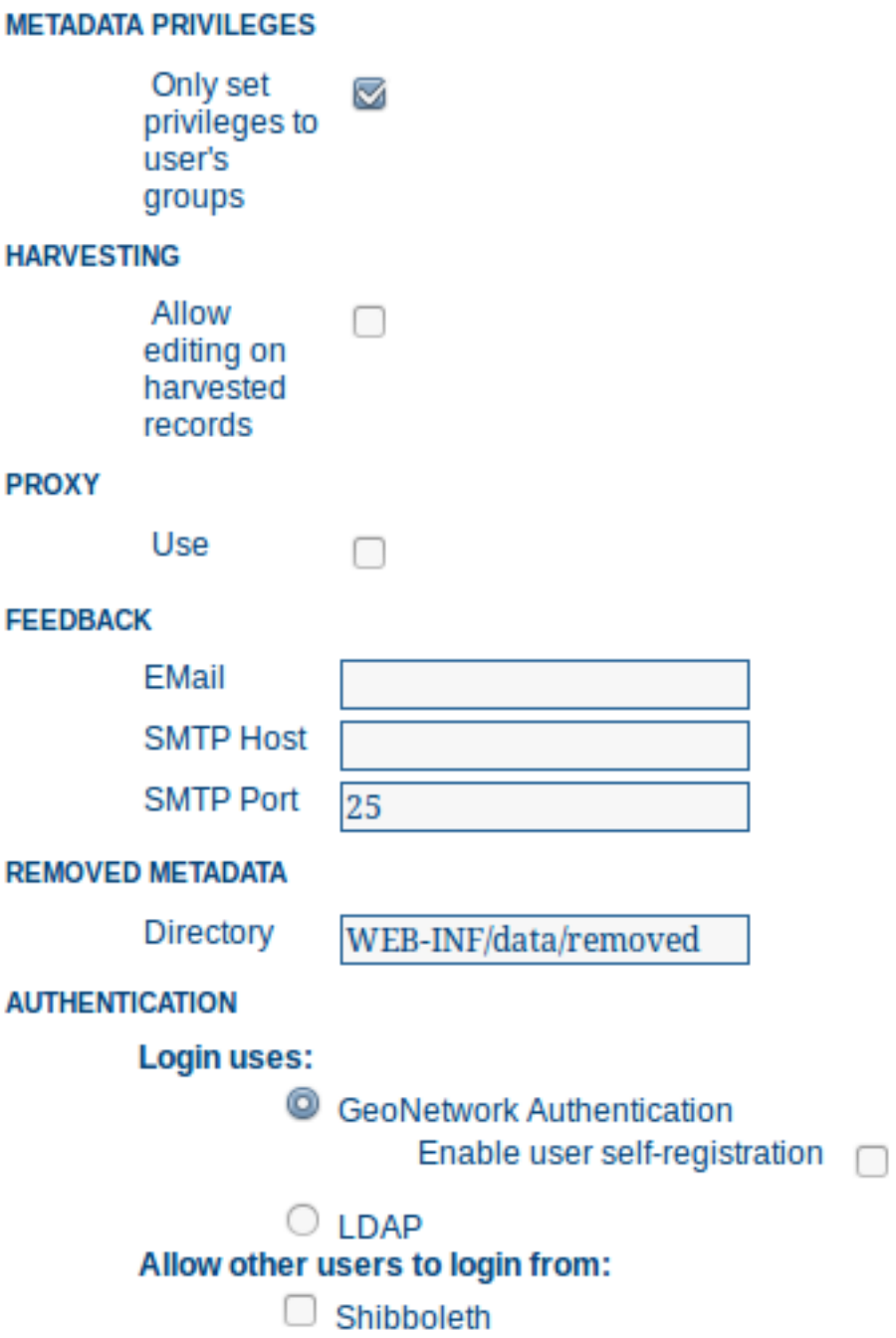


Figure 3.4: *The configuration options - part 3*

Name The name of the GeoNetwork node. Information that helps identify the catalogue to a human user.

Organization The organization the node belongs to. Again, this is information that helps identify the catalogue to a human user.

3.1.2 Server parameters

Here you have to enter the details of the web address of your GeoNetwork node. This address is important because it will be used to build addresses that access services and data on the GeoNetwork node. In particular:

1. building links to data file uploaded with a metadata record in the editor.
2. when the OGC CSW server is asked to describe its capabilities. The GetCapabilities operation returns an XML document with HTTP links to the CSW services provided by the server. These links are dynamically built using the host and port values.

Protocol The HTTP protocol used to access the server. Choosing http means that all communication with GeoNetwork will be visible to anyone listening to the protocol. Since this includes usernames and passwords this is not secure. Choosing https means that all communication with GeoNetwork will be encrypted and thus much harder for a listener to decode.

Host The node's address or IP number. If your node is publicly accessible from the Internet, you have to use the domain name. If your node is hidden inside your private network and you have a firewall or web server that redirects incoming requests to the node, you have to enter the public address of the firewall or web server. A typical configuration is to have an Apache web server on address A that is publicly accessible and redirects the requests to a Tomcat server on a private address B. In this case you have to enter A in the host parameter.

Port The node's port (usually 80 or 8080). If the node is hidden, you have to enter the port on the public firewall or web server.

3.1.3 Intranet Parameters

A common need for an organisation is to automatically discriminate between anonymous internal users that access the node from within an organisation (Intranet) and anonymous external users from the Internet. GeoNetwork defines anonymous users from inside the organisation as belonging to the group *Intranet*, while anonymous users from outside the organisation are defined by the group *All*. To automatically distinguish users that belong to the Intranet group you need to tell GeoNetwork the intranet IP address and netmask.

Network The intranet address in IP form (eg. 147.109.100.0).

Netmask The intranet netmask (eg. 255.255.255.0).

3.1.4 Metadata Search Results

Configuration settings in this group determine what the limits are on user interaction with the search results.

Maximum Selected Records The maximum number of search results that a user can select and process with the batch operations eg. Set Privileges, Categories etc.

3.1.5 Multi-Threaded Indexing

Configuration settings in this group determine how many processor threads are allocated to indexing tasks in GeoNetwork. If your machine has many processor cores, you can now determine how many to allocate to GeoNetwork indexing tasks. This can bring dramatic speed improvements on large indexing tasks (eg. changing the privileges on 20,000 records) because GeoNetwork can split the indexing task into a number of pieces and assign them to different processor cores.

Number of processing threads The maximum number of processing threads that can be allocated to an indexing task.

Note: this option is only available for databases that have been tested. Those databases are PostGIS and Oracle. You should also carefully consider how many connections to the database you allocate in the database configuration as each thread could tie up one database connection for the duration of a long indexing session (for example). See *Advanced configuration* for more details of how to configure the number of connections in the database connection pool.

3.1.6 Lucene Index Optimizer

Configuration settings in this group determine when the Lucene Index Optimizer is run. By default, this takes place at midnight each day. With recent upgrades to Lucene, particularly Lucene 3.6.1, the optimizer is becoming less useful, so this configuration group will very likely be removed in future versions.

3.1.7 Z39.50 configuration

GeoNetwork can act as a Z39.50 server. Z39.50 is the name of an older communication protocol used for distributed searching across metadata catalogs.

Enable: Check this option to enable the Z39.50 server, uncheck it to disable the Z39.50 server.

Port: This is the port on which GeoNetwork will be listening for incoming Z39.50 requests. Z3950 servers can run on any port, but 210 (not recommended), 2100 and 6668 are common choices. If you have multiple GeoNetwork nodes running on the same machine then you need to make sure each one has a different port number.

GeoNetwork must be restarted to put any changes to these values into use.

3.1.8 OAI Provider

Options in this group control the way in which the OAI Server in GeoNetwork responds to OAIPMH harvest requests from remote sites.

Datesearch: OAI Harvesters may request records from GeoNetwork in a date range. GeoNetwork can use one of two date fields from the metadata to check for a match with this date range. The default choice is *Temporal extent*, which is the temporal extent from the metadata record. The other option, *Modification date*, uses the modification date of the metadata record in the GeoNetwork database. The modification date is the last time the metadata record was updated in or harvested by GeoNetwork.

Resumption Token Timeout: Metadata records that match an OAI harvest search request are usually returned to the harvester in groups with a fixed size (eg. in groups of 10 records). With each group a resumption token is included so that the harvester can request the next group of records. The resumption token timeout is the time (in seconds) that GeoNetwork OAI server will wait for a resumption token to

be used. If the timeout is exceeded GeoNetwork OAI server will drop the search results and refuse to recognize the resumption token. The aim of this feature is to ensure that resources in the GeoNetwork OAI server are released.

Cache size: The maximum number of concurrent OAI harvests that the GeoNetwork OAI server can support.

GeoNetwork must be restarted to put any changes to the resumption token timeout and the Datesearch options into use.

3.1.9 XLink resolver

The XLink resolver replaces the content of elements with an attribute @xlink:href (except for srv:operatesOn element) with the content obtained from the URL content of @xlink:href. The XLink resolver should be enabled if you want to harvest metadata fragments or reuse fragments of metadata in your metadata records.

Enable: Enables/disables the XLink resolver.

Note: to improve performance GeoNetwork will cache content that is not in the local catalog.

3.1.10 Search Statistics

Enables/disables search statistics capture. Search statistics are stored in the database and can be queried using the Search Statistics interface on the Administration page. There is very little compute overhead involved in storing search statistics as they are written to the database in a background thread. However database storage for a very busy site must be carefully planned.

3.1.11 Multilingual Settings

Options in this group determine how GeoNetwork will search metadata in multiple languages.

Enable auto-detecting search request language: If this option is selected, Geonetwork will analyse the search query and attempt to detect the language that is used before defaulting to the GUI language.

Search results in requested language sorted on top: If this option is selected, a sort clause will be added to each query to ensure that results in the current language are always sorted on top. This is different from increasing priority of the language in that it overrides the relevance of the result. For example, if a german result has very high relevance but the search language is french then the french results will all come before the german result.

Search only in requested language The options in this section determines how documents are sorted/prioritised relative to the language in the document compared to the search language.

- *All documents in all languages (No preferences)* - The search language is ignored and will have no effect on the ordering of the results
- *Prefer documents with translations requested language* - Documents with a translation in the search language (anywhere in the document) will be prioritized over documents without any elements in the search language
- *Prefer documents whose language is the requested language* - Documents that are the same language as the search language (ie. the documents that are specified as being in the same language as the search language) are prioritized over documents that are not.

- *Translations in requested language* - The search results will only contain documents that have some translations in the search language.
- *Document language is the requested language* - The search results will contain documents whose metadata language is specified as being the in search language

3.1.12 Data-For-Download Service

GeoNetwork editor supports uploading one or more files that can be stored with the metadata record. When such a record is displayed in the search results, a 'Download' button is provided which will allow the user to select which file they want to download. This option group determines how that download will occur.

Use GeoNetwork simple file download service: Clicking on any file stored with the metadata record will deliver that file directly to the user via the browser.

Use GeoNetwork disclaimer and constraints service: Clicking on any file stored with the metadata record will deliver a zip archive to the user (via the browser) that contains the data file, the metadata record itself and a summary of the resource constraint metadata as an html document. In addition, the user will need to provide some details (name, organisation, email and optional comment) and view the resource constraints before they can download the zip archive.

3.1.13 Clickable hyperlinks

Enables/disables hyperlinks in metadata content. If a URL is present in the metadata content, GeoNetwork will detect this and make it into a clickable hyperlink when it displays the metadata content.

3.1.14 Local rating

Enables/disables local rating of metadata records.

3.1.15 Automatic fixes

For each metadata schema, GeoNetwork has an XSLT that it can apply to a metadata record belonging to that schema. This XSLT is called update-fixed-info.xsl and the aim of this XSLT is to allow fixed schema, site and GeoNetwork information to be applied to a metadata record every time the metadata record is saved in the editor. As an example, GeoNetwork uses this XSLT to build and store the URL of any files uploaded and stored with the metadata record in the editor.

Enable: Enabled by default. It is recommended you do not use the GeoNetwork default or advanced editor when auto-fixing is disabled. See http://trac.osgeo.org/geonetwork/ticket/368 for more details.

3.1.16 INSPIRE

Enables/disables the INSPIRE support:

- CSW GetCapabilities includes the INSPIRE section (ie. ExtendedCapabilities) that administrator can customize in xml/csw/capabilities_inspire.xml and response support language extensions. The language provided defines:

- Natural language fields are returned in the language requested (see *OGC CSW server configuration*)
- The end-points are returned for the language requested
- INSPIRE themes are indexed (check that INSPIRE themes thesaurus is available and reindex the catalog)
- Enables/disables INSPIRE search panel: Add INSPIRE criteria in the advanced search panel (eg. Annex, INSPIRE theme)

3.1.17 Metadata Views

Options in this section enable/disable metadata element groups in the metadata editor/viewer.

Enable simple view: The simple view in the metadata editor/viewer: - removes much of the hierarchy from nested metadata records (such as ISO19115/19139) - will not let the user add metadata elements that are not already in the metadata record It is intended to provide a flat, simple view of the metadata record. A disadvantage of the simple view is that some of the context information supplied by the nesting in the metadata record is lost. *Enable ISO view*: The ISO19115/19139 metadata standard defines three groups of elements: - Minimum: those elements that are mandatory - Core: the elements that should be present in any metadata record describing a geographic dataset - All: all the elements *Enable INSPIRE view*: Enables the metadata element groups defined in the EU INSPIRE directive. *Enable XML view*: This is a raw text edit view of the XML record. You can disable this if (for example), you don't want inexperienced users to be confused by the XML presentation provided by this view.

3.1.18 Metadata Privileges

Only set privileges to user's groups: If enabled then only the groups that the user belongs to will be displayed in the metadata privileges page (unless the user is an Administrator). At the moment this option cannot be disabled and is likely to be deprecated in the next version of GeoNetwork.

3.1.19 Harvesting

Allow editing on harvested records: Enables/Disables editing of harvested records in the catalogue. By default, harvested records cannot be edited.

3.1.20 Proxy

For some functions (eg. harvesting) GeoNetwork must be able to connect to remote sites. This may not be possible if an organisation uses proxy servers. If your organisation uses a proxy server then GeoNetwork must be configured to use the proxy server in order to correctly route outgoing requests to remote sites.

Use: Checking this box will display the proxy configuration options panel.

Host: The proxy server name or address to use (usually an IP address).

Port: The proxy server port to use.

Username (optional): a username should be provided if the proxy server requires authentication.

Password (optional): a password should be provided if the proxy server requires authentication.

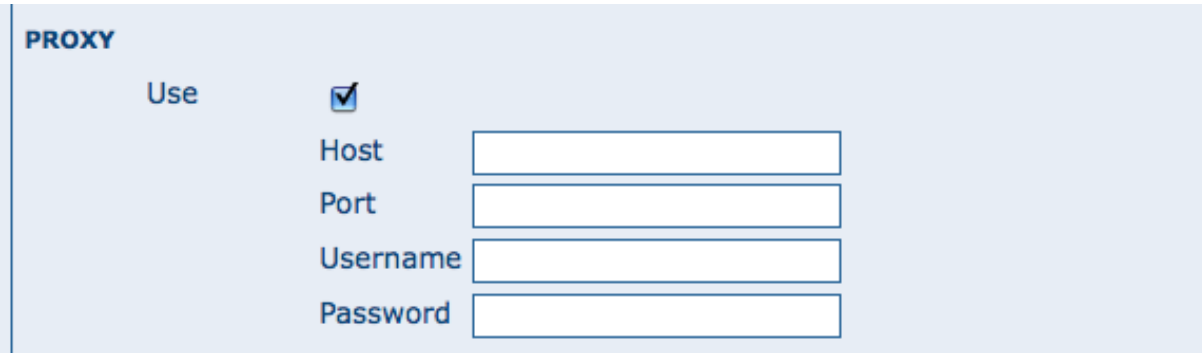


Figure 3.5: *The proxy configuration options*

3.1.21 Feedback

GeoNetwork needs to send email if:

- you are using the User Self-registration system or the Metadata Status workflow
- a file uploaded with a metadata record is downloaded
- a user provides feedback using the online form.

You have to configure the mail server GeoNetwork should use in order to enable it to send these emails.

Email: This is the email address that will be used to send the email (the From address).

SMTP host: the mail server address to use when sending email.

SMTP port: the mail server SMTP port (usually 25).

3.1.22 Removed metadata

Defines the directory used to store a backup of metadata and data after a delete action. This directory is used as a backup directory to allow system administrators to recover metadata and possibly related data after erroneous deletion. By default the removed directory is created in the GeoNetwork data folder.

3.2 Authentication

In this section you define the source against which the catlaog will authenticate users and passwords.

By default, users are authenticated against info held in the catalog database. When the catalog database is used as the authentication source, the user self-registration function can be enabled. A later section (see *User Self-Registration*) discusses user self-registration and the configuration options it requires.

You may choose to authenticate logins against either the catalog database tables or LDAP (the lightweight directory access protocol) or both. The next section describes how to authenticate against the different authentication providers:

- LDAP
- CAS
- Shibboleth authentication
- User Self-Registration

In addition to either of these options, you may also configure other authentication sources. At present, Shibboleth is one additional authentication source that can be configured. Shibboleth is typically used for national access federations such as the Australian Access Federation. Configuring shibboleth authentication in the catalog to use such a federation would allow not only users from a local database or LDAP directory to use your installation, but any user from such a federation.

Authentication is using Spring Security framework and could support multiple authentication providers.

Authentication configuration is defined in WEB-INF/config-security.properties file. When making a change to that configuration file, the catalog need to be restarted to take parameters into account.

3.2.1 LDAP

Connection Settings

To enable LDAP support:

1. add the LDAP base URL property in config-security.properties:

   ```
   # LDAP security properties
   ldap.base.provider.url=ldap://localhost:389
   ldap.base.dn=dc=fao,dc=org
   ldap.security.principal=cn=admin,dc=fao,dc=org
   ldap.security.credentials=ldap
   ```

   - ldap.base.provider.url: This tells the portal where the LDAP server is located. Make sure that the computer with the catalog can hit the computer with the LDAP server. Check to make sure that the appropriate ports are opened, etc.
   - ldap.base.dn=dc=fao,dc=org: this will usually look something like: "dc=organizationnamehere,dc=org"
   - ldap.security.principal & ldap.security.credentials: Define LDAP administrator user to use to bind to LDAP. If not define, an anonymous bind is made. Principal is the username and credentials property the password.
   - To verify that you have the correct settings, try to connect to the LDAP server using an LDAP browser application.

2. define where to find users in LDAP structure for authentication:

   ```
   ldap.base.search.base=ou=people
   ldap.base.dn.pattern=uid={0},${ldap.base.search.base}
   #ldap.base.dn.pattern=mail={0},${ldap.base.search.base}
   ```

   - ldap.base.search.base: this is where the catalog will look for users (for authentication)
   - ldap.base.dn.pattern: this is the distinguished name for the user to bind with. {0} is replaced by the user name typed in the sign in screen.

3. add the following import to config-security.xml:

```
<import resource="config-security-ldap.xml"/>
```

Authorization Settings

When using LDAP, user information and privileges could be defined from the LDAP attributes.

User details

All user informations could be retrieved from the LDAP as defined in the config-security-overrides.properties. This property file defined for each user attribute in the catalog database which LDAP attributes match. If the attribute is empty or not defined, a default value could be defined. The configuration is the following:

```
# Map user information to LDAP attributes and default values
# ldapUserContextMapper.mapping[name]=ldap_attribute,default_value
ldapUserContextMapper.mapping[name]=cn,
ldapUserContextMapper.mapping[surname]=givenName,
ldapUserContextMapper.mapping[mail]=mail,data@myorganization.org
ldapUserContextMapper.mapping[organisation]=,myorganization
ldapUserContextMapper.mapping[kind]=,
ldapUserContextMapper.mapping[address]=,
ldapUserContextMapper.mapping[zip]=,
ldapUserContextMapper.mapping[state]=,
ldapUserContextMapper.mapping[city]=,
ldapUserContextMapper.mapping[country]=,
```

Privileges configuration

When using LDAP, user groups and user profiles could be set from LDAP information or not. To manage user privileges from the local database, set the ldap.privilege.import property in config-security.properties to false:

```
ldap.privilege.import=false
```

If LDAP information should be used to define user privileges, set it to true:

```
ldap.privilege.import=true
```

When importing privileges from LDAP, the catalog administrator could decide to create groups defined in the LDAP and not defined in local database. For this set the following property to true:

```
ldap.privilege.create.nonexisting.groups=false
```

Simple privileges configuration In order to define which groups the user is member of and which profile is the user:

```
ldapUserContextMapper.mapping[privilege]=groups,sample
# If not set, the default profile is RegisteredUser
# Valid profiles are http://geonetwork-opensource.org/manuals/trunk/eng/developer/apidoc
ldapUserContextMapper.mapping[profile]=privileges,RegisteredUser
```

Attributes configuration:

- privilege attribute contains the group this user is member of. More than one group is allowed.
- profile attribute contains the profile of the user

User valid profiles are:

- Administrator
- UserAdmin
- Reviewer
- Editor
- RegisteredUser
- Guest

Profile mapping configuration If LDAP attribute containing profiles does not match the catalog profile list, a mapping could be defined in config-security-overrides.properties:

```
# Map LDAP custom profiles to catalog profiles. Not used if ldap.privilege.pattern is de
ldapUserContextMapper.profilMapping[Admin]=Administrator
ldapUserContextMapper.profilMapping[Editeur]=Reviewer
ldapUserContextMapper.profilMapping[Public]=RegisteredUser
```

For example, in the previous configuration, the attribute value Admin will be mapped to Administrator (which is a valid profile for the catalog).

Advanced privileges configuration An attribute could define both the profile and the group for a user. To extract this information, a custom pattern could be defined to populate user privileges according to that attribute:

```
# In config-security-overrides.properties
ldapUserContextMapper.mapping[privilege]=cat_privileges,sample

# In config-security.properties
ldap.privilege.pattern=CAT_(.*)_(.*)
ldap.privilege.pattern.idx.group=1
ldap.privilege.pattern.idx.profil=2
```

The LDAP attribute can contains the following configuration to define the different type of users:

```
-- Define a catalog admin:
cat_privileges=CAT_ALL_Administrator

-- Define a reviewer for the group GRANULAT
cat_privileges=CAT_GRANULAT_Reviewer

-- Define a reviewer for the group GRANULAT and editor for MIMEL
cat_privileges=CAT_GRANULAT_Reviewer
cat_privileges=CAT_MIMEL_Editor

-- Define a reviewer for the group GRANULAT and editor for MIMEL and RegisteredUser for
cat_privileges=CAT_GRANULAT_Reviewer
cat_privileges=CAT_MIMEL_Reviewer
cat_privileges=CAT_NATURA2000_RegisterdUser
```

```
-- Only a registered user for GRANULAT
cat_privileges=CAT_GRANULAT_RegisteredUser
```

Synchronization

A synchronization task is taking care of removing LDAP user which may be deleted. For example:

- T0: a user A sign in the catalog. A local user A is created in the user database
- T1: A is deleted from the LDAP (A could not sign in in the catalog anymore)
- T2: the synchronization task will check that all local LDAP users exist in LDAP:
  - if user is not owner of any records, it will be deleted
  - if user is owner of metadata records, warning message is avaialable on the catalog logging system. record's owner should be changed to another user before the task could remove the user.

By default the task is runned once every day. Configuration could be changed in config-security.properties:

```
# Run LDAP sync every day at 23:30
ldap.sync.cron=0 30 23 * * ?
```

Debugging

If connection fails, try to increase logging for LDAP in log4j.cfg:

```
log4j.logger.geonetwork.ldap          = DEBUG
log4j.logger.org.springframework = DEBUG, console, jeeves
log4j.logger.org.springframework.* = DEBUG
log4j.logger.org.springframework.security.ldap = DEBUG
```

3.2.2 CAS

To enable CAS support:

1. add the CAS base URL property in config-security.properties:

   ```
   cas.baseURL=https://localhost:8443/cas
   cas.ticket.validator.url=${cas.baseURL}
   cas.login.url=${cas.baseURL}/login
   cas.logout.url=${cas.baseURL}/logout?url=${geonetwork.https.url}/
   ```

2. add the following import to config-security.xml:

   ```
   <import resource="config-security-cas.xml"/>
   <import resource="config-security-cas-ldap.xml"/>
   ```

3.2.3 Shibboleth authentication

When using either the GeoNetwork database or LDAP for authentication, you can also configure shibboleth to allow authentication against access federations.

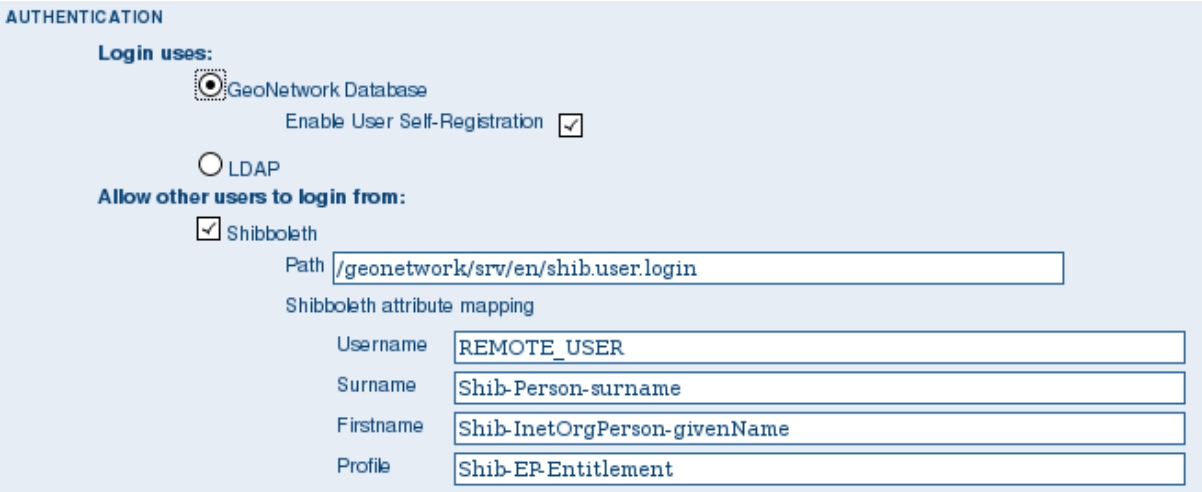


Figure 3.6: *The Shibboleth configuration options*

Shibboleth authentication requires interaction with Apache web server. In particular, the apache web server must be configured to require Shibboleth authentication to access the path entered in the configuration. The apache web server configuration will contain the details of the shibboleth server that works out where a user is located (sometimes called a 'where are you from' server).

The remainder of the shibboleth login configuration describes how shibboleth authentication attributes are mapped to GeoNetwork user database fields as once a user is authenticated against shibboleth, their details are copied to the local GeoNetwork database.

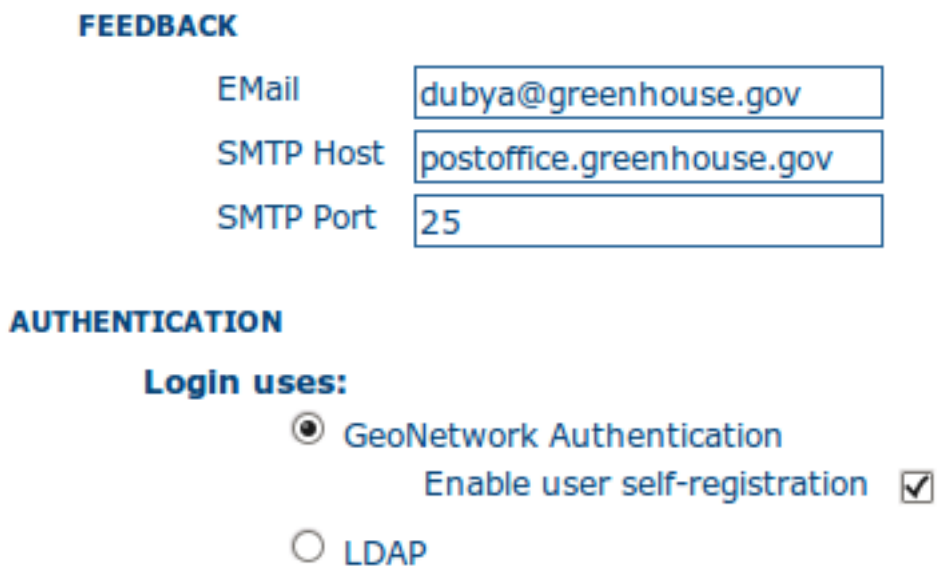
3.2.4 User Self-Registration

From Administration, system configuration, GeoNetwork has a self-registration function which allows a user to request a login which provides access to 'registered-user' functions. By default this capability is switched off. To configure this capability you must complete the following sections in the 'System configuration' menu:

- configure the site name and organization name as these will be used in emails from this GeoNetwork site to newly registered users. An example of how to config these fields at the top of the system configuration form is:

- configure feedback email address, SMTP host and SMTP port. The feedback email address will be sent an email when a new user registers and requests a profile other than 'Registered User'. An example of how to config these fields in the system configuration form is:
- check the box, enable user self-registration in the Authentication section of the system configuration form as follows:

When you save the system configuration form, return to the home page and log out as admin, your banner menu should now include two new options, 'Forgot your password?' and 'Register' (or their



translations into your selected language) as follows:

You should also configure the xml file that includes contact details to be displayed when an error occurs in the registration process. This file is localized - the english version is located in INSTALL_DIR/web/geonetwork/loc/en/xml/registration-sent.xml.

Finally, if you want to change the content of the email that contains registration details for new users, you should modify INSTALL_DIR/web/geonetwork/xsl/registration-pwd-email.xsl.

3.3 OGC CSW server configuration

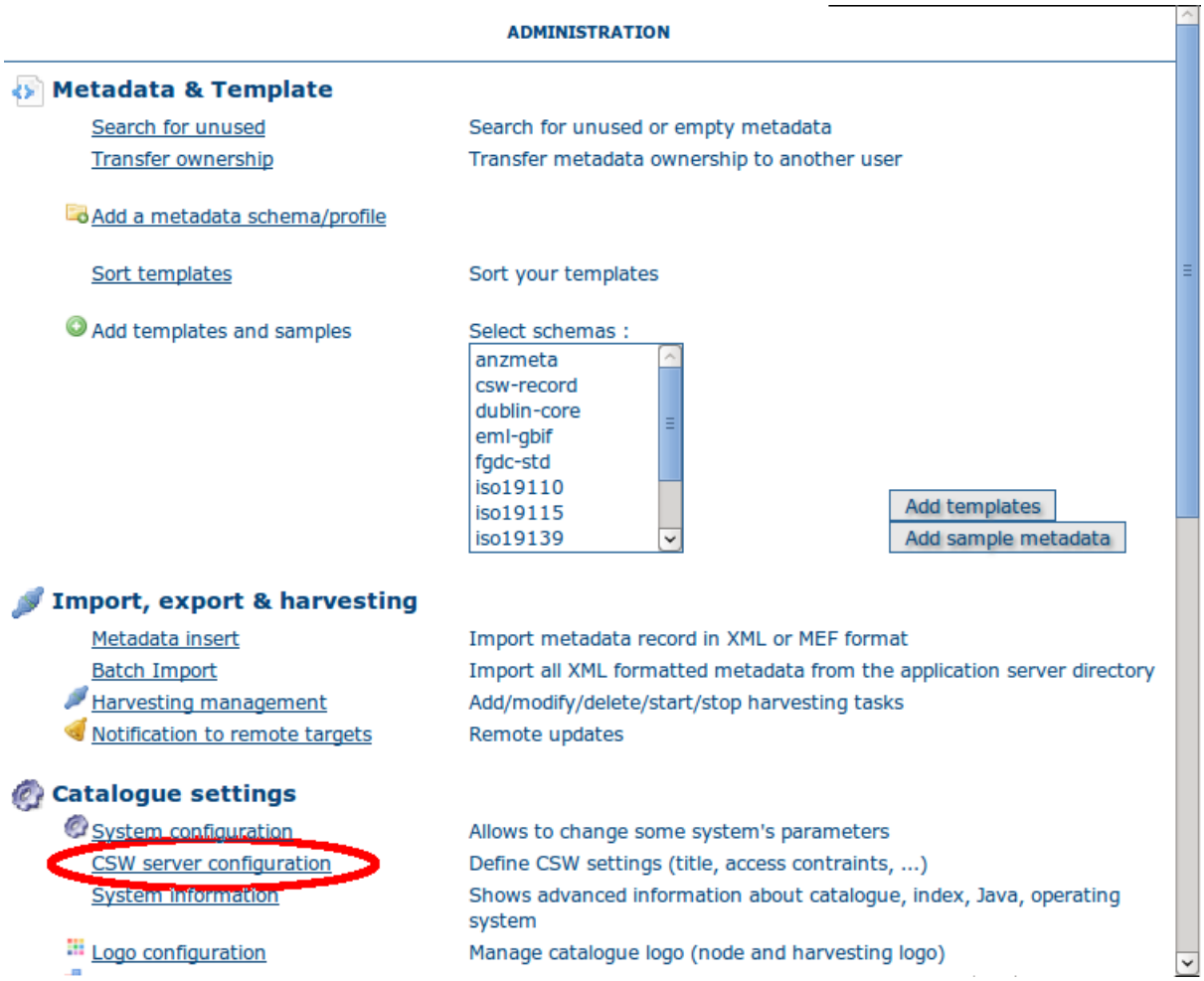
To get to the CSW server configuration, you must be logged on as administrator first. Open the Administration page and select CSW Server configuration (The link is surrounded with a red ellipse in the image below).

Clicking on this link will open a configuration menu that looks like the following.

The Open Geospatial Catalogue Service for the Web (OGC-CSW) service, is a self-describing service that allows query, update and insertion of metadata records. The service can be asked to provide a description of itself, the human who administers it and other information through a GetCapabilities request (eg. http://localhost:8080/geonetwork/srv/en/csw?request=GetCapabilities&service=CSW&version=2.0.2). This form allows you to configure the CSW server and fill out some of the properties returned in response to a GetCapabilities request. A description of each of the fields in this form now follows:

Enable: This option allows you to start or stop the CSW services. If this option is disabled, other catalogues cannot connect to the node using CSW protocol.

Inserted metadata is public: By default, metadata inserted with the CSW Harvest and CSW Transaction operations is not publicly viewable. A user with the appropriate access rights could do this after the CSW Harvest and CSW Transaction operations, but this is not always convenient. If this option is checked all metatada inserted using the CSW Harvest and CSW Transaction operations will be publicly viewable.


ADMINISTRATION
Metadata & Template
Search for unused
Search for unused or empty metadata
Transfer ownership
Transfer metadata ownership to another user
Add a metadata schema/profile
Sort templates
Sort your templates
Add templates and samples
Select schemas :
anzmeta
csw-record
dublin-core
eml-gbif
fgdc-std
iso19110
iso19115
iso19139
Add templates
Add sample metadata
Import, export & harvesting
Metadata insert
Import metadata record in XML or MEF format
Batch Import
Import all XML formatted metadata from the application server directory
Harvesting management
Add/modify/delete/start/stop harvesting tasks
Notification to remote targets
Remote updates
Catalogue settings
System configuration
Allows to change some system's parameters
CSW server configuration
Define CSW settings (title, access contraints, ...)
System information
Shows advanced information about catalogue, index, Java, operating system
Logo configuration
Manage catalogue logo (node and harvesting logo)

CSW SERVER CONFIGURATION

Enable	☑
Inserted metadata is public (Transaction)	☐
Contact	csw-server-admin (Admin CSW Server)
	English
Title	GeoNetwork Catalog CSW Service for Green House
Abstract	This CSW service serves up metadata records about datasets held in the Green House.
Fees	None
Access constraints	None

Save

CSW ISO Profile test Test interface for the CSW ISO Profile catalog interface

Contact: The drop down select list shows the current users in the local GeoNetwork catalog. The contact details of the user chosen from this list will be provided in the GetCapabilities document of the CSW service.

Language: The language that is used in the service description fields.

Title: The title of the CSW service.

Abstract: The abstract of the CSW service. The abstract can contain a brief description of what the service provides and who runs it.

Fees: If there are any fees for usage of the service then they should be detailed here.

Access constraints: If there are any constraints on access to the service then they should be detailed here.

The last function on this page is the CSW ISO Profile test. Clicking on this link brings up a javascript based interface that allows you to submit requests to the CSW server. The requests used by this interface are XML files in INSTALL_DIR/web/geonetwork/xml/csw/test.

3.3.1 Virtual CSW server entry points

This feature of CSW server adds the capability to create custom CSW entry points that apply extra criteria to the CSW requests, allowing to implement several useful cases like, for example:

- Define an INSPIRE CSW entry point to deliver only the INSPIRE related metadata stored in the catalog.
- Define CSW entry points to deliver only metadata related to specific theme/s: climate, boundaries, etc.

The CSW service entry points are defined in the configuration file `WEB-INF/config-csw-servers.xml` using the following syntax:

```
<service name="csw-with-my-filter-environment">
   <class name=".services.main.CswDispatcher" >
           <param name="filter" value="+inspirerelated:on +themekey:environment"/>
   </class>
</service>

<service name="csw-with-my-filter-climate">
   <class name=".services.main.CswDispatcher" >
           <param name="filter" value="+inspirerelated:on +themekey:climate"/>
   </class>
</service>
```

The `filter` parameter value should use the Lucene query parser syntax (see http://lucene.apache.org/java/2_9_1/queryparsersyntax.html) and is use in these CSW operations:

- *GetRecords*: the `filter` is applied with the CSW query as an extra query criteria.
- *GetRecordById*: the `filter` is applied with the metadata `id` requested as an extra query criteria.
- *GetDomain*: the `filter` is applied as a query criteria to retrieve the metadata properties requested.
- *GetCapabilities*: the `filter` is applied as a query criteria to fill the metadata keywords list in the GetCapabilities document.

The list of available Lucene index fields to use in the `filter` parameter can be obtained from the files `index-fields.xsl` in the schema folders located in `WEB-INF/xml/schemas`.

As Harvest and Transaction operations are not affected by `filter` parameter, to avoid confusion is better to use this feature as readonly CSW endpoints.

Configuration

Adding a new CSW entry point to GeoNetwork opensource requires these steps (suppose the new CSW entry point is call `csw-with-my-filter-environment`):

- Create the service definition in the configuration file `WEB-INF/config-csw-servers.xml` with the custom filter criteria as describe before:

```
<service name="csw-with-my-filter-environment">
   <class name=".services.main.CswDispatcher" >
           <param name="filter" value="+inspirerelated:on +themekey:environment"/>
   </class>
</service>
```

- Define permissions for the service in the file `WEB-INF/user-profiles.xml` file:

```
<profile name="Guest">
   <allow service="csw-with-my-filter-environment"/>
```

- Restart the application. The new CSW entry point is accessible in http://localhost:8080/srv/en/csw-with-my-filter-environment

Configuration using GeoNetwork overrides

In this section is described how to use GeoNetwork overrides feature to configure a new CSW entry point. This feature allows to use different configurations to handle multiple deployment platforms. See additional documentation of this feature in *Configuration override*.

- Add the next override to a configuration override file, for example `WEB-INF/config-overrides-csw.xml`:

```
<overrides xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
 <!-- Add custom CSW service -->
 <file name=".*/WEB-INF/config-csw-servers.xml">
   <addXML xpath="services">
     <service name="csw-with-my-filter-environment">
       <class name=".services.main.CswDispatcher">
         <param name="filter" value="+inspirerelated:on +themekey:environment"/>
       </class>
     </service>
   </addXML>
 </file>
 <file name=".*/WEB-INF/user-profiles.xml">
   <addXML xpath="profile[@name='Guest']">
     <allow service="csw-with-my-filter-environment"/>
   </addXML>
 </file>
</overrides>
```

For more information about configuration overrides see *Configuration override*

- Restart the application. The new CSW entry point is accessible in http://localhost:8080/srv/en/csw-with-my-filter-environment

3.4 Advanced configuration

3.4.1 Database configuration

GeoNetwork has two options for pooled connections to the relational database:

1. Manage its own database configuration and pool directly using Apache Commons Database Connection Pool (DBCP)
2. Use database configuration and pool managed by the web application server (also known as the container) such as tomcat or jetty via the Java Naming and Directory Interface (JNDI).

Managing the database connection pool through Apache DBCP

This option is the one that most users would use as it is the default option for managing the database in GeoNetwork. A typical configuration in the resources element of *INSTALL_DIR/web/geonetwork/WEB-INF/config.xml* uses the *jeeves.resources.dbms.ApacheDBCPool* class and looks something like:

```
<resource enabled="true">
        <name>main-db</name>
        <provider>jeeves.resources.dbms.ApacheDBCPool</provider>
        <config>
                <user>www-data</user>
                <password>www-data</password>
                <driver>org.postgis.DriverWrapper</driver>
                <url>jdbc:postgresql_postGIS://localhost:5432/geonetwork</url>
                <poolSize>10</poolSize>
                <validationQuery>SELECT 1</validationQuery>
        </config>
</resource>
```

The parameters that can be specified to control the Apache Database Connection Pool are described at http://commons.apache.org/dbcp/configuration.html. You can configure a subset of these parameters in your resource element. The parameters that can be specified are:

Parameter	Description	Default
maxActive	pool size/maximum number of active connections	10
maxIdle	maximum number of idle connections	maxActive
minIdle	minimum number of idle connections	0
maxWait	number of milliseconds to wait for a connection to become available	200
validationQuery	sql statement for verifying a connection, must return a least one row	no default
timeBetweenEviction-RunsMillis	time between eviction runs (-1 means next three params are ignored)	-1
testWhileIdle	test connections when idle	false
minEvictableIdle-TimeMillis	idle time before connection can be evicted	30 x 60 x 1000 msecs
numTestsPerEviction-Run	number of connections tested per eviction run	3
maxOpenPrepared-Statements	number of sql statements that can be cached for reuse (-1 none, 0 unlimited)	-1
defaultTransactionIso-lation	see http://en.wikipedia.org/wiki/Isolation_%28database_systems%29	READ_COMMITTED

For performance reasons you should set the following parameter *after* GeoNetwork has created and filled the database tables it has been configured to use:

- maxOpenPreparedStatements="300" (at least)

The following parameters are set by GeoNetwork and cannot be configured by the user:

- removeAbandoned - true
- removeAbandonedTimeout - 60 x 60 seconds = 1 hour
- logAbandoned - true
- testOnBorrow - true
- defaultReadOnly - false
- defaultAutoCommit - false
- initialSize - maxActive
- poolPreparedStatements - true, if maxOpenPreparedStatements >= 0, otherwise false

Note: Some firewalls kill idle connections to databases after say 1 hour (= 3600 secs). To keep idle connections alive by testing them with the validationQuery, set minEvictableIdleTimeMillis to something less than timeout, interval (eg. 2 mins = 120 secs = 120000 millisecs), set testWhileIdle to true and set timeBetweenEvictionRunsMillis and numTestsPerEvictionRun high enough to visit connections frequently eg 15 mins = 900 secs = 900000 millisecs and 4 connections per test. For example:

```
<testWhileIdle>true</testWhileIdle>
<minEvictableIdleTimeMillis>120000</minEvictableIdleTimeMillis>
<timeBetweenEvictionRunsMillis>900000</timeBetweenEvictionRunsMillis>
<numTestsPerEvictionRun>4</numTestsPerEvictionRun>
```

Note:

- When GeoNetwork manages the database connection pool, PostGIS database is the only database that can hold the spatial index in the database. All other database choices hold the spatial index as a shapefile. If using PostGIS, two pools of database connections are created. The first is managed

and configured using parameters in this section, the second is created by GeoTools and cannot be configured. This approach is now *deprecated*: if you want to use the database to hold the spatial index you *should* use the JNDI configuration described in the next section because it uses a single, configurable database pool through GeoTools as well as the more modern NG (Next Generation) GeoTools datastore factories.

- For more on transaction isolation see http://en.wikipedia.org/wiki/Isolation_%28database_systems%29.

Database connection pool managed by the container

A typical configuration in the resources element of *INSTALL_DIR/web/geonetwork/WEB-INF/config.xml* uses the *jeeves.resources.dbms.JNDIPool* class and looks something like:

```
<resource enabled="true">
        <name>main-db</name>
        <provider>jeeves.resources.dbms.JNDIPool</provider>
        <config>
                <context>java:/comp/env</context>
                <resourceName>jdbc/geonetwork</resourceName>
                <url>jdbc:oracle:thin:@localhost:1521:XE</url>
                <provideDataStore>true</provideDataStore>
        </config>
</resource>
```

The configuration parameters and their meanings are as follows:

Config Parameter	Description
context	The name of the context from which to obtain the resource - almost always this is java:/comp/env
resourceName	The name of the resource in the context to use
url	The URL of the database - this is needed to let GeoTools know the database type
provideDataStore	If set to true then the database will be used for the spatial index, otherwise a shapefile will be used

The remainder of the configuration is done in the container context. eg. for tomcat this configuration is in conf/context.xml in the resource called jdbc/geonetwork. Here is an example for the Oracle database:

```
<Resource name="jdbc/geonetwork"
        auth="Container"
        type="javax.sql.DataSource"
        username="system"
        password="oracle"
        factory="org.apache.commons.dbcp.BasicDataSourceFactory"
        driverClassName="oracle.jdbc.OracleDriver"
        url="jdbc:oracle:thin:@localhost:1521:XE"
        maxActive="10"
        maxIdle="10"
        removeAbandoned="true"
        removeAbandonedTimeout="3600"
        logAbandoned="true"
        testOnBorrow="true"
        defaultAutoCommit="false"
        validationQuery="SELECT 1 FROM DUAL"
        accessToUnderlyingConnectionAllowed="true"
/>
```

eg. for jetty, this configuration is in *INSTALL_DIR/web/geonetwork/WEB-INF/jetty-env.xml*. Here is an example for the Postgis database:

```
<Configure class="org.eclipse.jetty.webapp.WebAppContext">
  <New id="gnresources" class="org.eclipse.jetty.plus.jndi.Resource">
    <Arg></Arg>
    <Arg>jdbc/geonetwork</Arg>
    <Arg>
      <New class="org.apache.commons.dbcp.BasicDataSource">
        <Set name="driverClassName">org.postgis.DriverWrapper</Set>
        <Set name="url">jdbc:postgresql_postGIS://localhost:5432/gndb</Set>
        <Set name="username">geonetwork</Set>
        <Set name="password">geonetworkgn</Set>
        <Set name="validationQuery">SELECT 1</Set>
        <Set name="maxActive">10</Set>
        <Set name="maxIdle">10</Set>
        <Set name="removeAbandoned">true</Set>
        <Set name="removeAbandonedTimeout">3600</Set>
        <Set name="logAbandoned">true</Set>
        <Set name="testOnBorrow">true</Set>
        <Set name="defaultAutoCommit">false</Set>
        <!-- 2=READ_COMMITTED, 8=SERIALIZABLE -->
        <Set name="defaultTransactionIsolation">2</Set>
        <Set name="accessToUnderlyingConnectionAllowed">true</Set>
      </New>
      </Arg>
    <Call name="bindToENC">
      <Arg>jdbc/geonetwork</Arg>
    </Call>
      </New>
</Configure>
```

The parameters that can be specified to control the Apache Database Connection Pool used by the container are described at http://commons.apache.org/dbcp/configuration.html.

The following parameters *must* be set to ensure GeoNetwork operates correctly:

Tomcat Syntax	Jetty Syntax
defaultAutoCommit="false"	<Set name="defaultAutoCommit">false</Set>
accessToUnderlyingConnectionAllowed="true"	<Set name="accessToUnderlyingConnectionAllowed">true</Set>

For performance reasons you should set the following parameters *after* GeoNetwork has created and filled the database it has been configured to use:

Tomcat Syntax	Jetty Syntax
poolPreparedStatements="true"	<Set name="poolPreparedStatements">true</Set>
maxOpenPreparedStatements="300" (at least)	<Set name="maxOpenPreparedStatements">300</Set>

Notes:

- both PostGIS and Oracle will build and use a table in the database for the spatialindex if provideDataStore is set to true. Other databases could be made to do the same if a spatialindex table is created - see the definition for the spatialIndex table in *INSTALL_DIR/web/geonetwork/WEB-INF/classes/setup/sql/create/create-db-postgis.sql* for example.
- you should install commons-dbcp-1.3.jar and commons-pool-1.5.5.jar in the container class path (eg. *common/lib* for tomcat5 or *jetty/lib/ext* for Jetty) as the only supported DataSourceFactory in geotools is apache commons dbcp.

- the default tomcat-dbcp.jar version of apache commons dbcp for tomcat appears to work correctly for geotools and PostGIS but does not work for those databases that need to unwrap the connection in order to do spatial operations (eg. Oracle).
- Oracle ojdbc-14.jar or ojdbc5.jar or ojdbc6.jar (depending on the version of Java being used) and sdoapi.jar should also be installed in the container class path (for tomcat: *common/lib* or *lib* and for jetty: *jetty/lib/ext*).
- advanced: you should check the default transaction isolation level for your database driver. READ_COMMITTED appears to be a safe level of isolation to use with GeoNetwork for commonly used databases. Also note that McKoi can only support SERIALIZABLE (does anyone still use McKoi?). For more on transaction isolation see http://en.wikipedia.org/wiki/Isolation_%28database_systems%29.

Specific Database Issues

Oracle

ORACLE on Linux (x86_64): if your connection with the database takes a long time to establish or frequently times out then adding *-Djava.security.egd=file:/dev/../dev/urandom* to your JAVA_OPTS environment variable (for tomcat) or the start-geonetwork.sh script may help. For more information on this see https://kr.forums.oracle.com/forums/thread.jspa?messageID=3699989.

ORACLE returns *ORA-01000: maximum open cursors exceeded* whilst filling the tables in a newly created GeoNetwork database. This occurs because you have enabled the prepared statement pool in either the container database configuration or the GeoNetwork database configuration in *WEB-INF/config.xml*. Until the database fill statements used by GeoNetwork are refactored, you will not be able to use a prepared statement cache with ORACLE if you are creating and filling a new GeoNetwork database so you should set the DBCP maxOpenPreparedStatements parameter to -1. *However*, after the database has been created and filled, you *can* use a prepared statement cache so, you should stop GeoNetwork and configure the prepared statement cache as described above before restarting.

DB2

DB2 may produce an exception when GeoNetwork is started for the first time:

```
DB2 SQL error: SQLCODE: -805, SQLSTATE: 51002, SQLERRMC: NULLID.SYSLH203
```

There are two possible solutions to this problem:

- Setup the database manually using a procedure like the following:

  ```
  db2 create db geonet
  db2 connect to geonet user db2inst1 using mypassword
  db2 -tf INSTALL_DIR/WEB-INF/classes/setup/sql/create/create-db-db2.sql > res1.txt
  db2 -tf INSTALL_DIR/WEB-INF/classes/setup/sql/data/data-db-default.sql > res2.txt
  db2 connect reset
  ```

After execution, check **res1.txt** and **res2.txt** if errors have occurred.

- Drop the database, re-create it, locate the file db2cli.lst in the db2 installation folder and execute the following command:

```
db2 bind @db2cli.lst CLIPKG 30**
```

3.4.2 Advanced configuration for larger catalogs

There are a number of steps you must consider if you are going to use GeoNetwork for catalogs with 20,000 or more metadata records:

1. **Consider the hardware you have available** GeoNetwork uses a database as a transactional store and does all metadata searches using Lucene. Lucene is very fast and scale for large catalogs if you supply fast disk (solid state disk is best by far), lots of memory/RAM (16Gb+) and multiple processors as part of a 64bit environment. Linux is probably the best operating system to take advantage of such an environment.

2. **Build the spatial index into your database ie. Use PostGIS (Postgres+PostGIS) or Oracle as your database** GeoNetwork has to build a spatial index containing all metadata bounding boxes and polygons, in order to support spatial queries for the Catalog Services Web (CSW) interface eg. select all metadata records that intersect a search polygon. By default GeoNetwork uses a shapefile but the shapefile quickly becomes costly to maintain during reindexing usually after the number of records in the catalog exceeds 20,000. If you select PostGIS or Oracle as your database via JNDI (see previous section), GeoNetwork will build the spatial index in a table (called spatialindex). The spatialindex table in the database is much faster to reindex. But more importantly, if appropriate database hardware and configuration steps are taken, it should also be faster to query than the shapefile when the number of records in the catalog becomes very large.

3. **Consider the Java heap space** Typically as much memory as you can give GeoNetwork is the answer here. If you have a 32bit machine then you are stuck below 2Gb (or maybe a little higher with some hacks). A 64bit machine is best for large catalogs. Jetty users can set the Java heap space in *INSTALL_DIR/bin/start-geonetwork.sh* (see the -Xmx option: eg. -Xmx4g will set the heap space to 4Gb on a 64bit machine). Tomcat users can set an environment variable JAVA_OPTS eg. export JAVA_OPTS="-Xmx4g"

4. **Consider the number of processors you wish to allocate to GeoNetwork** GeoNetwork 2.8 allows you to use more than one system processor (or core) to speed up reindexing and batch operations on large numbers of metadata records. The records to be processed are split into groups with each group assigned to an execution thread. You can specify how many threads can be used in the system configuration menu. A reasonable value for the number of threads is the number of processors or cores you have allocated to the GeoNetwork Java Virtual Machine (JVM) or just the number of processors on the machine that you have dedicated to GeoNetwork.

5. **Consider the number of database connections to be allocated to GeoNetwork** GeoNetwork uses and reuses a pool of database connections. This is configured in *INSTALL_DIR/web/geonetwork/WEB-INF/config.xml* or in the container via JNDI. To arrive at a reasonable number for the pool size is not straight forward. You need to consider the number of concurrent harvesters you will run, the number of concurrent batch import and batch operations you expect to run and the number of concurrent users you are expecting to arrive. The default value of 10 is really only for small sites. The more connections you can allocate, the less time your users and other tasks will spend waiting for a free connection.

6. **Consider the maximum number of files your system will allow any process to have open** Most operating systems will only allow a process to open a limited number of files. If you are expecting a large number of records to be in your catalog then you should change the default value to something larger (eg. 4096) as the lucene index in GeoNetwork will occasionally require large numbers of open files during reindexing. In Linux this value can be changed using the ulimit

command (ulimit -a typically shows you the current setting). Find a value that suits your needs and add the appropriate ulimit command (eg. ulimit -n 4096) to the GeoNetwork startup script to make sure that the new limit is used when GeoNetwork is started.

7. **Raise the stack size limit for the postgres database** Each process has some memory allocated as a stack. The stack is used to store process arguments and variables as well as state when functions are called. Most operating systems limit the size that the stack can grow to. With large catalogs and spatial searches, very large SQL queries can be generated on the PostGIS spatial index table. This can cause postgres to exceed the process stack size limit (typically 8192k on smaller machines). You will know when this happens because a very long SQL query will be output to the GeoNetwork log file prefixed with a cryptic message something along the lines of:

   ```
   java.util.NoSuchElementException: Could not acquire
   feature:org.geotools.data.DataSourceException: Error Performing SQL query: SELECT .
   ```

   In Linux the stack size can be changed using the ulimit command (ulimit -a typically shows you the current setting). You will need to choose a value and set it (eg. ulimit -s 262140) in the shell startup script of the postgres user (eg. .bashrc if using the bash shell). The setting may also need to be added to the postgres config - see "max_stack_depth" in the postgresql.conf file for your system. You may also have to enable to postgres user to change the stack size in */etc/security/limits.conf*. After this has been done, restart postgres.

8. **If you need to support a catalog with more than 1 million records** GeoNetwork creates a directory for each record that in turn contains a *public* and a *private* directory for holding attached data and thumbnails. These directories are in the GeoNetwork *data* directory - typically: *INSTALL_DIR/web/geonetwork/WEB-INF/data* - see *GeoNetwork data directory*. This can exhaust the number of inodes available in a Linux file system (you will often see misleading error reports saying that the filesystem is 'out of space' - even though the filesystem may have lots of freespace). Check this using *df -i*. Since inodes are allocated statically when the filesystem is created for most common filesystems (including extfs4), it is rather inconvenient to have to backup all your data and recreate the filesystem! So if you are planning a large catalog with over 1 million records, make sure that you create a filesystem on your machine with the number of inodes set to at least 5x (and to be safe 10x) the number of records you are expecting to hold and let GeoNetwork create its *data* directory on that filesystem.

3.4.3 GeoNetwork data directory

When customizing Geonetwork for a specific deployment server you need to be able to modify the configuration for that specific server. One way is to modify the configuration files within Geonetwork web application, however this is a problematic method because you essentially need either a different web application for each deployment target or need to patch each after deployment. Geonetwork provides two methods for addressing this issue

1. GeoNetwork data directory
2. Configuration override files (See *Configuration override*)

The GeoNetwork data directory is the location on the file system where GeoNetwork stores much of its custom configuration. This configuration defines such things as: What thesaurus is used by GeoNetwork? What schema is plugged in GeoNetwork? The data directory also contains a number of support files used by GeoNetwork for various purposes (eg. Lucene index, spatial index, logos).

It is a good idea to define an external data directory when going to production in order to make upgrade easier.

Creating a new data directory

The data directory needs to be created before starting the catalogue. It must be readable and writable by the user starting the catalogue. If the data directory is an empty folder, the catalogue will initialize the directory default structure. The easiest way to create a new data directory is to copy one that comes with a standard installation - you can find this in `INSTALL_DIR/web/geonetwork/WEB-INF/data`.

Setting the data directory

The data directory variable can be set using:

- Java environment variable
- Servlet context parameter
- System environment variable

For java environment variable and servlet context parameter use:

- <webappName>.dir and if not set using geonetwork.dir

For system environment variable use:

- <webappName>_dir and if not set using geonetwork_dir

Resolution order is:

1. <webappname>.dir
1. Java environment variable (ie. -D<webappname>.dir=/a/data/dir)
2. Servlet context parameter (ie. web.xml)
3. Config.xml appHandler parameter (ie. config.xml)
4. System environment variable (ie. <webappname>_dir=/a/data/dir). "." is not supported in env variables
1. geonetwork.dir
1. Java environment variable (ie. -Dgeonetwork.dir=/a/data/dir)
2. Servlet context parameter (ie. web.xml)
3. Config.xml appHandler parameter (ie. config.xml)
4. System environment variable (ie. geonetwork_dir=/a/data/dir). "." is not supported in env variables

Java System Property

Depending on the servlet container used it is also possible to specify the data directory location with a Java System Property.

For Tomcat, configuration is:

```
CATALINA_OPTS="-Dgeonetwork.dir=/var/lib/geonetwork_data"
```

Run the web application in read-only mode

In order to run GeoNetwork with the webapp folder in read-only mode, the user needs to set two variables:

- <webappName>.dir or geonetwork.dir for the data folder.
- (optional) config overrides if configuration files need to be changed (See *Configuration override*).

For Tomcat, configuration could be:

```
CATALINA_OPTS="-Dgeonetwork.dir=/var/lib/geonetwork_data -Dgeonetwork.jeeves.configurati
```

Structure of the data directory

The structure of the data directory is:

```
data_directory/
 |--data
 |    |--metadata_data: The data related to metadata records
 |    |--resources:
 |    |      |--htmlcache
 |    |      |--images
 |    |      |    |--harvesting
 |    |      |    |--logos
 |    |      |    |--statTmp
 |    |
 |    |--metadata_subversion: The subversion repository
 |
 |--config: Extra configuration (eg. could contain overrides)
 |    |--schemaplugin-uri-catalog.xml
 |    |--codelist: The thesauri in SKOS format
 |    |--schemaPlugins: The directory used to store new metadata standards
 |
 |--index: All indexes used for search
 |    |--nonspatial: Lucene index
 |    |--spatialindex.*: ESRI Shapefile for the index (if not using spatial db)
 |
 |--removed: Folder with removed metadata.
```

Advanced data directory configuration

All sub-directories could be configured separately using java system property. For example, to put index directory in a custom location use:

- <webappName>.lucene.dir and if not set using:
- geonetwork.lucene.dir

Example:

- Add the following java properties to start-geonetwork.sh script:

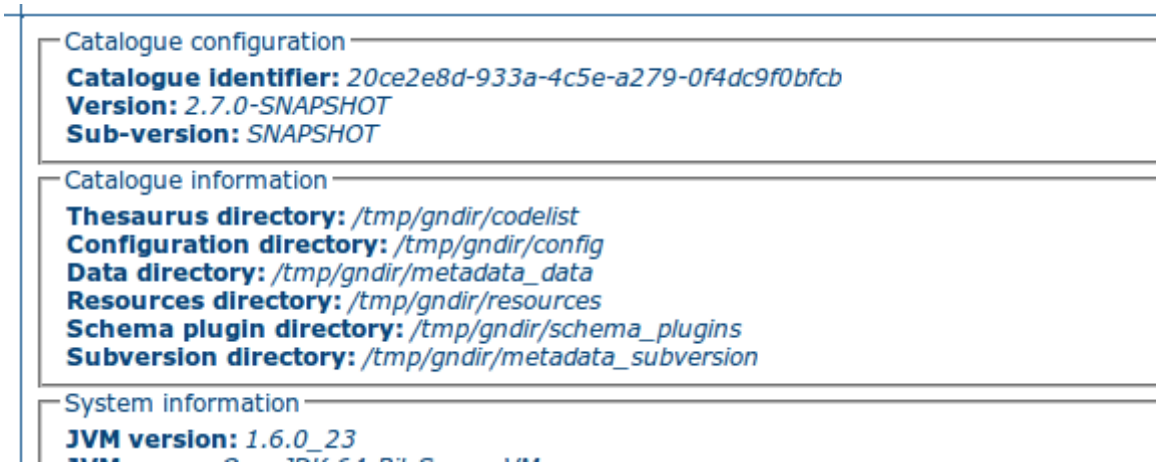
  ```
  java -Xms48m -Xmx512m -Xss2M -XX:MaxPermSize=128m -Dgeonetwork.dir=/app/geonetwork_
  ```

- Add the following system properties to start-geonetwork.sh script:

```
# Set custom data directory location using system property
export geonetwork_dir=/app/geonetwork_data_dir
export geonetwork_lucene_dir=/ssd/geonetwork_lucene_dir
```

System information

All catalogue configuration directory can be found using the `System Information` in the `Administration` page.



Other system properties

In Geonetwork there are several system properties that can be used to configure different aspects of Geonetwork. When a webcontainer is started the properties can be set. For example in Tomcat one can set either JAVA_OPTS or CATALINA_OPTS with -D<propertyname>=<value>.

- <webappname>.jeeves.configuration.overrides.file - See *Configuration override*
- jeeves.configuration.overrides.file - See *Configuration override*
- mime-mappings - mime mappings used by jeeves for generating the response content type
- http.proxyHost - The internal geonetwork Http proxy uses this for configuring how it can access the external network (Note for harvesters there is also a setting in the Settings page of the administration page)
- http.proxyPort - The internal geonetwork Http proxy uses this for configuring how it can access the external network (Note for harvesters there is also a setting in the Settings page of the administration page)
- geonetwork.sequential.execution - (true,false) Force indexing to occur in current thread rather than being queued in the ThreadPool. Good for debugging issues.

There is a usecase where multiple geonetwork instances might be ran in the same webcontainer, because of this many of the system properties listed above have <webappname>. When declaring the property this should be replaced with the webapp name the setting applies to. Typically this will be geonetwork.

3.4.4 Configuration override

Configuration override files allow nearly complete access to all the configuration allowing nearly any configuration parameter to be overridden for a particular deployment target. The concept behind configuration overrides is to have the basic configuration set in the geonetwork webapplication, the application is deployed and a particular set of override files are used for the deployment target. The override files only have the settings that need to be different for the deployment target, alleviating the need to deploy and edit the configuration files or have a different web application per deployment target.

Configuration override files are also useful for forked Geonetwork applications that regularly merge the changes from the true Geonetwork code base.

A common scenario is to have test and production instances with different configurations. In both configurations 90% of the configuration is the same but certain parts need to be updated.

An override file to be specified as a system property or as a servlet init parameter: jeeves.configuration.overrides.file.

The order of resolution is:

- System property with key: {servlet.getServletContext().getServletContextName()}.jeeves.configuration.over
- Servlet init parameter with key: jeeves.configuration.overrides.file
- System property with key: jeeves.configuration.overrides.file
- Servlet context init parameters with key: jeeves.configuration.overrides.file

The property should be a path or a URL. The method used to find a overrides file is as follows:

1. It is attempted to be used as a URL. if an exception occurs the next option is tried
2. It is assumed to be a path and uses the servlet context to look up the resources. If it can not be found the next option is tried
3. It is assumed to be a file. If the file is not found then an exception is thrown

An example of a overrides file is as follows:

```
<overrides>
    <!-- import values.  The imported values are put at top of sections -->
    <import file="./imported-config-overrides.xml" />
     <!-- properties allow some properties to be defined that will be substituted -->
     <!-- into text or attributes where ${property} is the substitution pattern -->
     <!-- The properties can reference other properties -->
     <properties>
         <enabled>true</enabled>
         <dir>xml</dir>
         <aparam>overridden</aparam>
     </properties>
     <!-- A regular expression for matching the file affected. -->
     <file name=".*WEB-INF/config\.xml">
         <!-- This example will update the file attribute of the xml element with the na
         <replaceAtt xpath="default/gui/xml[@name = 'countries']" attName="file" value="
         <!-- if there is no value then the attribute is removed -->
         <replaceAtt xpath="default/gui" attName="removeAtt"/>
         <!-- If the attribute does not exist it is added -->
         <replaceAtt xpath="default/gui" attName="newAtt" value="newValue"/>

         <!-- This example will replace all the xml in resources with the contained xml
```

```
        <replaceXML xpath="resources">
          <resource enabled="${enabled}">
            <name>main-db</name>
            <provider>jeeves.resources.dbms.DbmsPool</provider>
             <config>
                 <user>admin</user>
                 <password>admin</password>
                 <driver>oracle.jdbc.driver.OracleDriver</driver>
                 <!-- ${host} will be updated to be local host -->
                 <url>jdbc:oracle:thin:@${host}:1521:fs</url>
                 <poolSize>10</poolSize>
             </config>
          </resource>
        </replaceXML>
        <!-- This example simple replaces the text of an element -->
        <replaceText xpath="default/language">${lang}</replaceText>
        <!-- This examples shows how only the text is replaced not the nodes -->
        <replaceText xpath="default/gui">ExtraText</replaceText>
        <!-- append xml as a child to a section (If xpath == "" then that indicates the
             this case adds nodes to the root document -->
        <addXML xpath=""><newNode/></addXML>
        <!-- append xml as a child to a section, this case adds nodes to the root docum
        <addXML xpath="default/gui"><newNode2/></addXML>
        <!-- remove a single node -->
        <removeXML xpath="default/gui/xml[@name = countries2]"/>
        <!-- The logging files can also be overridden, although not as easily as other
             The files are assumed to be property files and all the properties are load
             The later properties overriding the previously defined parameters. Since t
             log file is not automatically located, the base must be also defined.  It
             shipped with geonetwork or another. -->
        <logging>
            <logFile>/WEB-INF/log4j.cfg</logFile>
            <logFile>/WEB-INF/log4j-jeichar.cfg</logFile>
        </logging>
    </file>
    <file name=".*WEB-INF/config2\.xml">
        <replaceText xpath="default/language">de</replaceText>
    </file>
    <!-- a normal file tag is for updating XML configuration files -->
    <!-- textFile tags are for updating normal text files like sql files -->
    <textFile name="test-sql.sql">
        <!-- each line in the text file is matched against the linePattern attribute an
        <update linePattern="(.*) Relations">$1 NewRelations</update>
        <update linePattern="(.*)relatedId(.*)">$1${aparam}$2</update>
    </textFile>
   <!-- configure the spring aspects of geonetwork -->
   <spring>
     <!-- import a complete spring xml file -->
     <import file="./config-spring-overrides.xml"/>
     <!-- declare a file as a spring properties override file: See http://static.spring
     <propertyOverrides file="./config-property-overrides.properties" />
     <!-- set a property on one bean to reference another bean -->
     <set>beanName.propertyName=beanName</set>
     <!-- add a references to a bean to a property on another bean.  This assumes the p
     <add>beanName.propertyName=beanName</add>
   </spring>
</overrides>
```

3.4.5 Lucene configuration

Lucene is the search engine used by GeoNetwork. All Lucene configuration is defined in WEB-INF/config-lucene.xml.

Add a search field

Indexed fields are defined on a per schema basis on the schema folder (eg. xml/schemas/iso19139) in index-fields.xsl file. This file define for each search criteria the corresponding element in a metadata record. For example, indexing the title of an ISO19139 record:

```
<xsl:for-each select="gmd:identificationInfo/gmd:MD_DataIdentification/
                      gmd:citation/gmd:CI_Citation/
                      gmd:title/gco:CharacterString">
    <Field name="mytitle" string="{string(.)}" store="true" index="true"/>
</xsl:for-each>
```

Usually, if the field is only for searching and should not be displayed in search results the store attribute could be set to false.

Once the field added to the index, user could query using it as a search criteria in the different kind of search services. For example using:

```
http://localhost:8080/geonetwork/srv/en/q?mytitle=africa
```

If user wants this field to be tokenized, it should be added to the tokenized section of config-lucene.xml:

```
<tokenized>
  <Field name="mytitle"/>
```

If user wants this field to be returned in search results for the search service, then the field should be added to the Lucene configuration in the dumpFields section:

```
<dumpFields>
  <field name="mytitle" tagName="mytitle"/>
```

Boosting documents and fields

Document and field boosting allows catalogue administrator to be able to customize default Lucene scoring in order to promote certain types of records.

A common use case is when the catalogue contains lot of series for aggregating datasets. Not promoting the series could make the series "useless" even if those records contains important content. Boosting this type of document allows to promote series and guide the end-user from series to related records (through the relation navigation).

In that case, the following configuration allows boosting series and minor importance of records part of a series:

```
<boostDocument name="org.fao.geonet.kernel.search.function.ImportantDocument">
  <Param name="fields" type="java.lang.String" value="type,parentUuid"/>
  <Param name="values" type="java.lang.String" value="series,NOTNULL"/>
  <Param name="boosts" type="java.lang.String" value=".2F,-.3F"/>
</boostDocument>
```

The boost is a positive or negative float value.

This feature has to be used by expert users to alter default search behavior scoring according to catalogue content. It needs tuning and experimentation to not promote too much some records. During testing, if search results looks different while being logged or not, it could be relevant to ignore some internal fields in boost computation which may alter scoring according to current user. Example configuration:

```
<fieldBoosting>
  <Field name="_op0" boost="0.0F"/>
  <Field name="_op1" boost="0.0F"/>
  <Field name="_op2" boost="0.0F"/>
  <Field name="_dummy" boost="0.0F"/>
  <Field name="_isTemplate" boost="0.0F"/>
  <Field name="_owner" boost="0.0F"/>
</fieldBoosting>
```

Boosting search results

By default Lucene compute score according to search criteria and the corresponding result set and the index content. In case of search with no criteria, Lucene will return top docs in index order (because none are more relevant than others).

In order to change the score computation, a boost function could be define. Boosting query needs to be loaded in classpath. A sample boosting class is available. RecencyBoostingQuery will promote recently modified documents:

```
<boostQuery name="org.fao.geonet.kernel.search.function.RecencyBoostingQuery">
  <Param name="multiplier" type="double" value="2.0"/>
  <Param name="maxDaysAgo" type="int" value="365"/>
  <Param name="dayField" type="java.lang.String" value="_changeDate"/>
</boostQuery>
```

3.4.6 Faceted search configuration

Faceted search provides a way to easily filter search:

In WEB-INF/config-summary.xml, catalogue administrator can configure the faceted search displayed in the search page.

In the hits section, new facet could be added:

```
<hits>
  <item name="keyword" plural="keywords" indexKey="keyword" max="15"/>
```

An item element defines a facet with the following parameters:

- name: the name of the facet (ie. the tag name in the XML response)
- plural: the plural for the name (ie. the parent tag of each facet values)
- indexKey: the name of the field in the index
- (optional) sortBy: the ordering for the facet. Defaults is by count.
- (optional) sortOrder: asc or desc. Defaults is descendant.
- (optional) max: the number of values to be returned for the facet. Defaults is 10.

Organisations
Région wallonne (26)
Cellule SIG de la DGARNE (SPW - DGO3 - DEMNA - DCD) (11)
Cellule Sous-sol/Géologie (SPW - DGO3) (7)
Direction du Développement rural (SPW - DGO3 - DRCE) (6)
Cellule SIG de la DGARNE (SPW-DGO3-DEMNA-DCD) (5)
Direction de la coordination des données (SPW - DGO3 - DEMNA - DCD) (3)
Direction des Eaux souterraines (SPW - DGO3 - DESO) (3)
+ Voir tous les filtres

Type
Jeu de données (61)
Service (17)

Type de service
Application (11)
OGC:CSW (2)
Inconnu (2)
discovery (1)
OGC:WMS (1)

Niveaux d'échelle
20000 (2)
10000 (31)
2500 (2)

Mots-clés
(10) Nature et environnement (30)
geoscientificInformation (16)
(1030) Sol et sous-sol (16)
(1020) Eau (14)
(1010) Faune et flore (14)
Habitats et biotopes (14)
(20) Aménagement du territoire (13)
+ Voir tous les filtres

Années
2013 (11)
2012 (23)
2011 (3)
2009 (4)
2008 (12)
2007 (2)
2006 (1)
+ Voir tous les filtres

When an item is modified or added, reload the lucene configuration and rebuild the index from the administration panel.

For easier update, config overrides could be used to modify the config-summary file (See *Configuration override*).

3.4.7 Character Set

By default the character set of geonetwork is UTF-8. This works well for many locales in the world and is compatible with ASCII that is typically used in US and Canada. However, if UTF-8 is not a compatible characterset in your environment you can change the default.

To change it within GeoNetwork simply start the application with the system property geonetwork.file.encoding set to the desired character set name.

For example if you are running tomcat you can set

JAVA_OPTS="-Dgeonetwork.file.encoding=UTF-16"

to the startup script and the default codec in Geonetwork will be UTF-16.

It is also recommended to set the file.encoding parameter to the same codec as this dictates to the default encoding used in Java and the Web Server may reference at times use the default codec.

Finally, by default the URL parameters are typically interpretted as ASCII characters which can be a problem when searching for metadata that are not in the english language. Each Web Server will have a method for configuring the encoding used when reading the parameters. For example, in Tomcat the encoding/charset configuration is in the server.xml Connector element.

3.5 User and Group Administration

GeoNetwork uses the concept of Users, Groups and User Profiles.

- A User can be part of one or more Groups.
- A User has a User Profile.
- A User can only have one User Profile associated.

The combination of User Profile and Group defines what tasks the User can perform on the system or on specific metadata records.

3.5.1 Creating new user Groups

The administrator can create new groups of users. User groups can correspond to logical units within an organisation. For example groups for Fisheries, Agriculture, Land and Water, Health etcetera.

To create new groups you should be logged on with an account that has Administrative privileges.

1. Select the *Administration* button in the menu. On the Administration page, select *Group management*.

2. Select *Add a new group*. You may want to remove the *Sample* group;

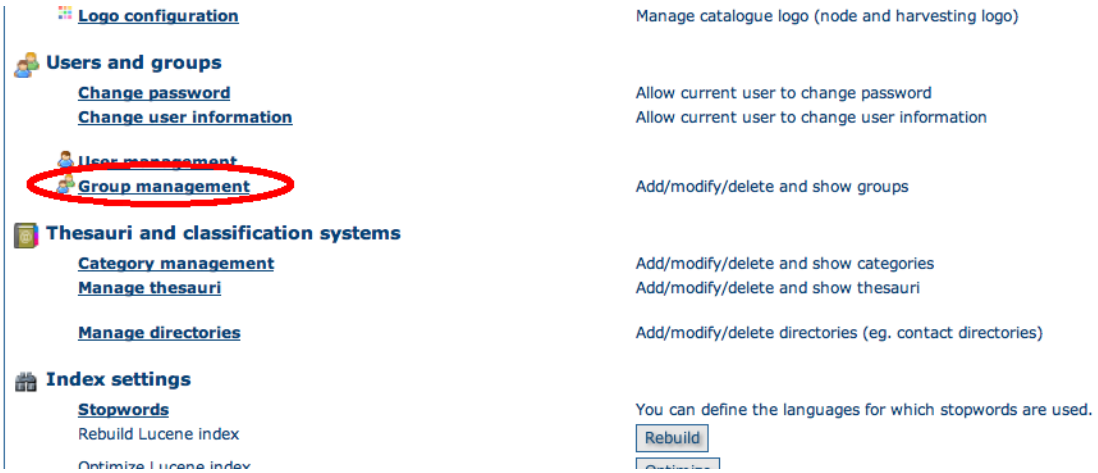


Figure 3.7: *Administration page - Group management*

Figure 3.8: *Group management*

3. Fill out the details. The email address will be used to send feedback on data downloads when they occur for resources that are part of the Group.

> **Warning:** The Name should *NOT* contain spaces! You can use the Localization panel to provide localized names for groups.

Figure 3.9: *Group edit form*

4. Click on *Save*

Access privileges can be set per metadata record. You can define privileges on a per Group basis.

Privileges that can be set relate to visibility of the Metadata (*Publish*), data *Download*, *Interactive Map* access and display of the record in the *Featured* section of the home page.

Editing defines the groups for which editors can edit the metadata record.

Notify defines what Groups are notified when a file managed by GeoNetwork is downloaded.

Below is an example of the privileges management table related to a dataset.

3.5.2 Creating new Users

To add a new user to the GeoNetwork system you do the following:

1. Select the *Administration* button in the menu. On the Administration page, select *User management*.

2. Click the button *Add a new user*;

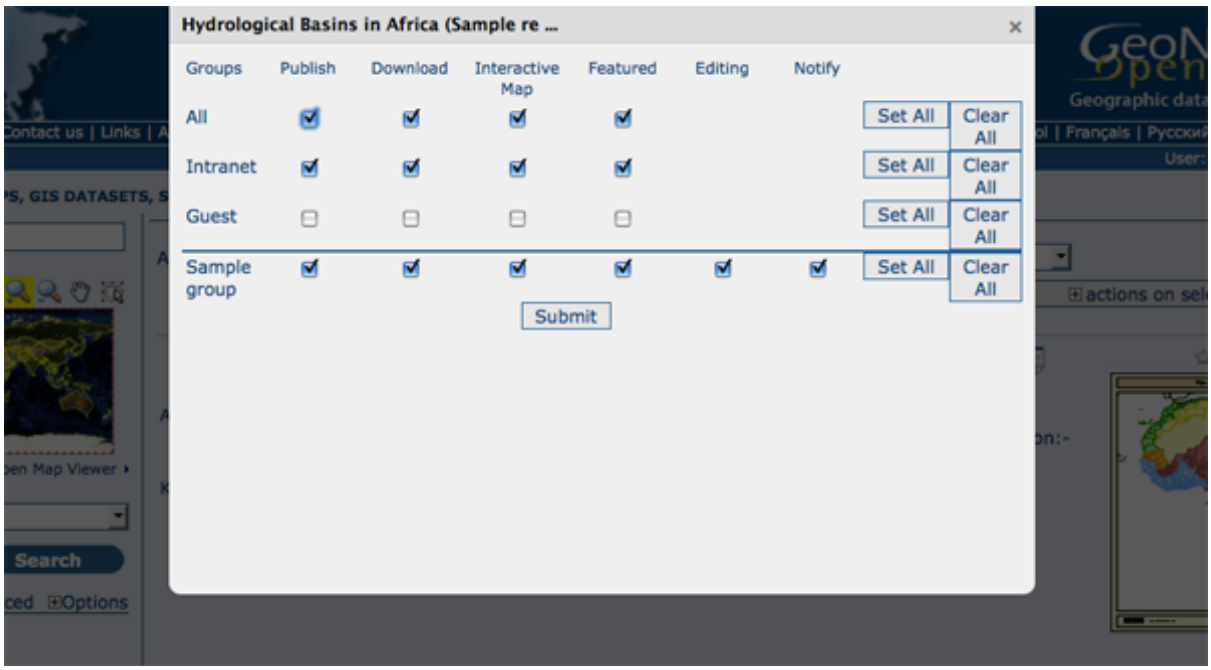


Figure 3.10: *Privilege settings*

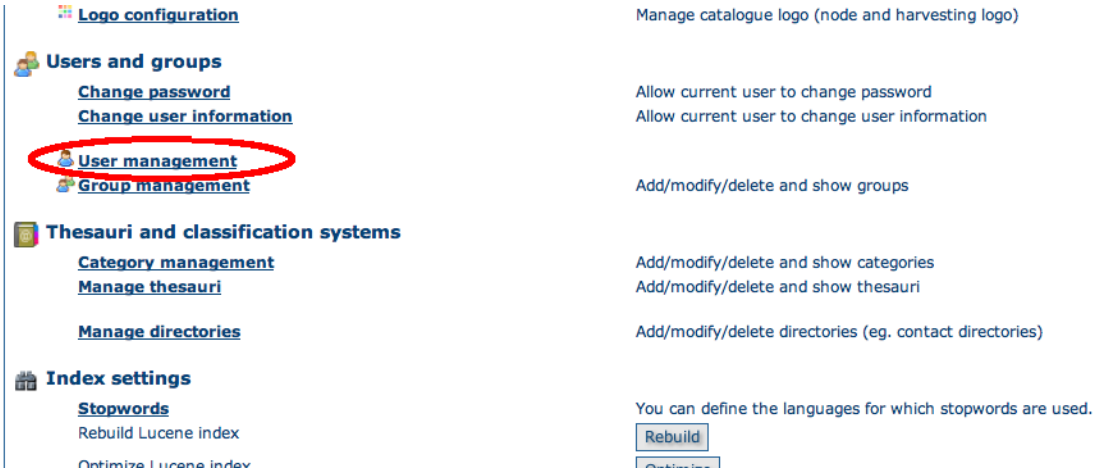


Figure 3.11: *Administration page - User management*

Figure 3.12: *User administration form*

1. Provide the *information* required for the new user;

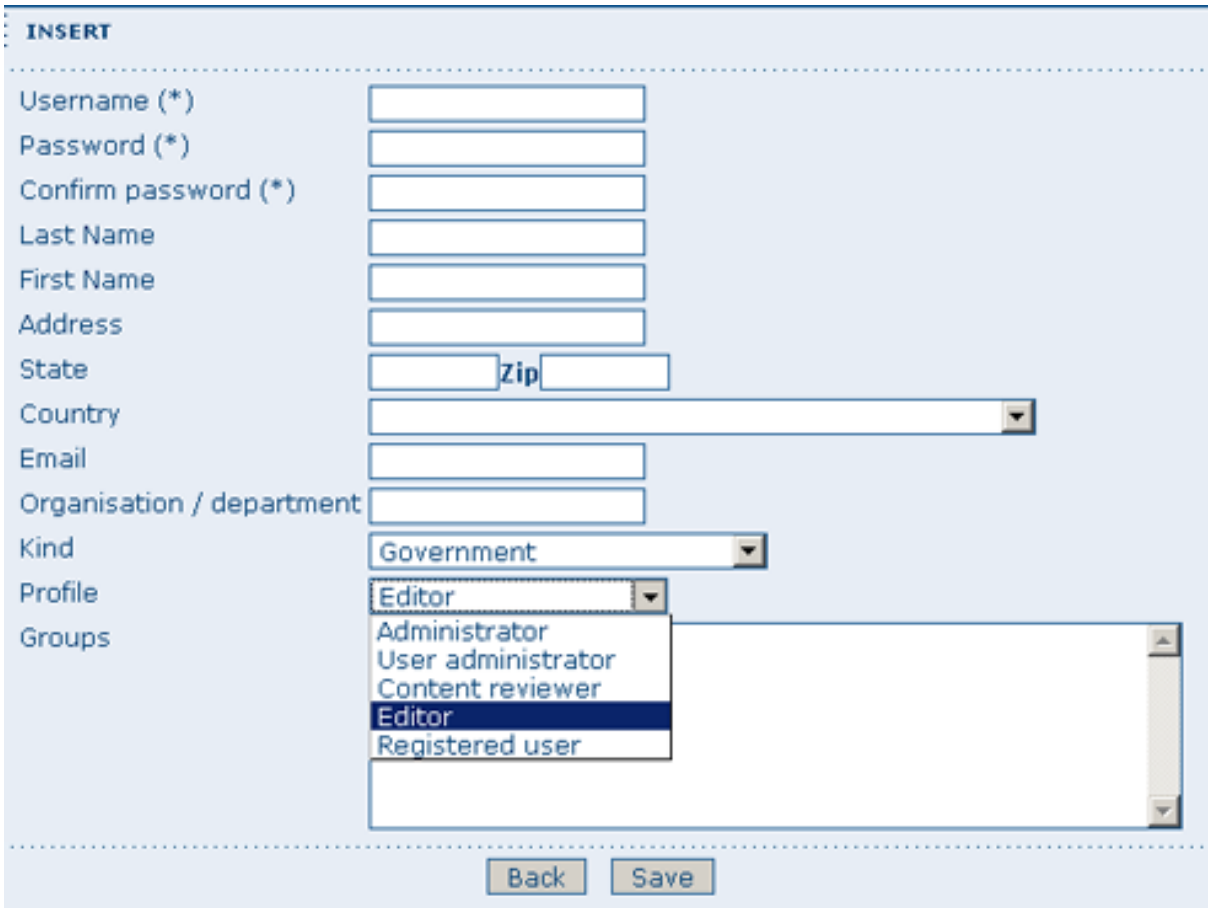


Figure 3.13: *User information form*

1. Assign the correct *profile*;
2. Assign the user to a *group*;
3. Click on *Save*.

3.5.3 User Profiles

Users can have different profiles depending on their role in the GeoNetwork system. A profile defines what tasks the user can perform.

User profiles are hierarchical and based on inheritance. This means that a user with an Editor profile can create and modify new metadata records, but can also use all functions a Registered user can use.

Rights associated with the profiles are illustrated in detail in the list below:

1. **Administrator Profile**

   The Administrator has special privileges that give access to all available functions.

   These include:

   - Full rights for creating new groups and new users

- Rights to change users/groups' profiles
- Full rights for creating/editing/deleting new/old metadata
- Perform system administration and configuration tasks.

2. **User Administrator Profile**

   The User Administrator is the administrator of his/her own group with the following privileges:

   - Full rights on creating new users within the own group
   - Rights to change users profiles within the own group
   - Full rights on creating/editing/ deleting new/old data within the own group

3. **Content Reviewer Profile**

   The content reviewer is the only person allowed to give final clearance on the metadata publication on the Intranet and/or on the Internet:

   - Rights on reviewing metadata content within the own group and authorising its publication

4. **Editor Profile**

   The editor works on metadata with following privileges:

   - Full rights on creating/editing/ deleting new/old data within the own group

5. **Registered User Profile**

   The Registered User has more access privileges than non-authenticated Guest users:

   - Right to download protected data

3.5.4 User Self-Registration

See *User Self-Registration Functions*.

3.6 Localization

3.6.1 Localization of dynamic user interface elements

The user interface of GeoNetwork can be localized into several languages through XML language files. Beside static text, there is also more dynamic text that can be added and changed interactively. This text is stored in the database and is translated using the Localization form that is part of the administrative functions.

The form allows you to localize the following entities: *Groups*, *Categories*, *Operations* and *Regions*. The localization form is subdivided in a left and a right panel.

The left panel allows you to choose which elements you want to edit. On the top, a dropdown let you choose which entity to edit. All elements of the selected type are shown in a list.

When you select an element from the list, the right panel will show the text as it will be displayed in the user interface. The text in the source language is read only while you can update the text in the target language field.

Figure 3.14: *How to open the Localization form*

Note: You can change the source and target languages to best suit your needs. Some users may for instance prefer to translate from French to Spanish, others prefer to work with English as the source language.

Use the Save button to store the updated label and move to the next element.

Warning: If the user changes a label and chooses another target language without saving, the label change is lost.

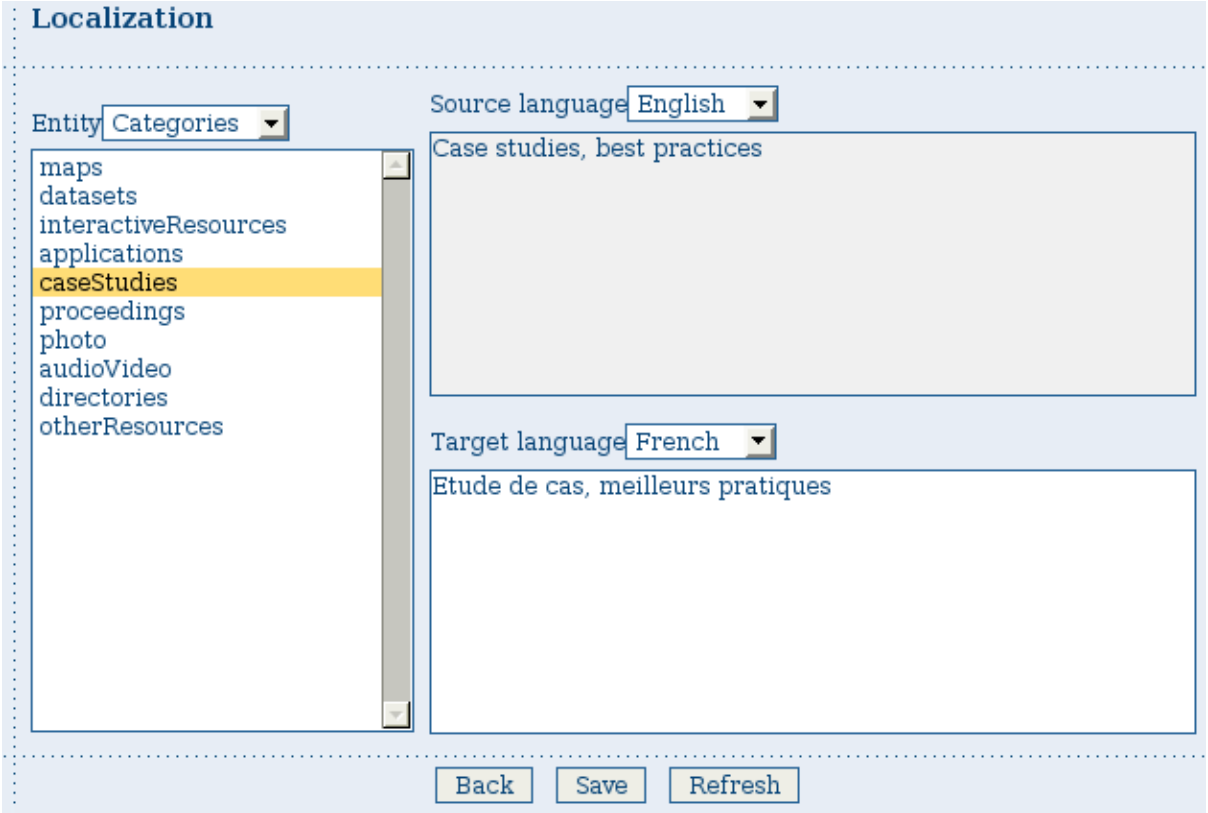


Figure 3.15: *The Localization form*

3.7 System Monitoring

The monitoring system provides automated monitoring of a Geonetwork web application to be able track the health of the system over time. The monitoring is based on the ‘‘Metrics’‘ library (http://metrics.codahale.com/) by Yammer and detailed explanation for developers desiring specific monitors can be found there.

The metrics are available via JMX or as JSON with http GET requests. The same information is available through both APIs. The web requests provided are:

- /monitor/metrics?[pretty=(true|false)][class=metric.name] - returns a json response with all of the registered metrics
- /monitor/threads - returns a text representation of the stack dump at the moment of the call

- /monitor/healthcheck - runs ALL health checks and returns 200 if all checks pass or 500 Internal Service Error if one fails (and human readable response of the failures)
- /criticalhealthcheck - runs only the critical (fast) health checks and returns 200 if all checks pass or 500 Internal Service Error if one fails
- /warninghealthcheck - runs only the non-critical health checks and returns 200 if all checks pass or 500 Internal Service Error if one fails
- /expensivehealthcheck - runs only the expensive critical health checks and returns 200 if all checks pass or 500 Internal Service Error if one fails
- /monitor - provide links to pages listed above.

Links to this data is also available in the geonetwork/srv/eng/config.info administration user interface as well.

By default the /monitor/* urls are protected and may only be accessed by an ''administrator" or ''monitor'', however it is possible in the web.xml to provide a whitelist of URLs or IP addresses of monitoring servers that are permitted to access the monitoring data without needing an administration account.

The monitors available are:

- Database Health Monitor - checks that the database is accessible
- Index Health Monitor - checks that the Lucene index is searchable
- Index Error Health Monitor - checks that there are no index errors in index (documents with _indexError field == 1)
- CSW GetRecords Health Monitor - Checks that GetRecords? does not return an error for a basic hits search
- CSW GetCapabilities Health Monitor - Checks that the GetCapabilities is returned and is not an error document
- Database Access timer - Time taken to access a DBMS instance. This gives and idea of the level of contention over the database connections
- Database Open Timer - Tracks the length of time a Database access is kept open
- Database Connection Counter - Counts the number of open Database connections
- Harvester Error Counter - Tracks errors that are raised during harvesting
- Service timer - Track the time of service execution
- Gui Services timer - Track the time of spend executing Gui services
- XSL output timer - Track the time of output xsl transform
- Log4j integration - monitors the frequency that logs are made for each log level so (for example) the rate that error are logged can be monitored. See http://metrics.codahale.com/manual/log4j

The monitors that are enabled are in the config-monitoring.xml file and if desired certain monitors can be disabled.

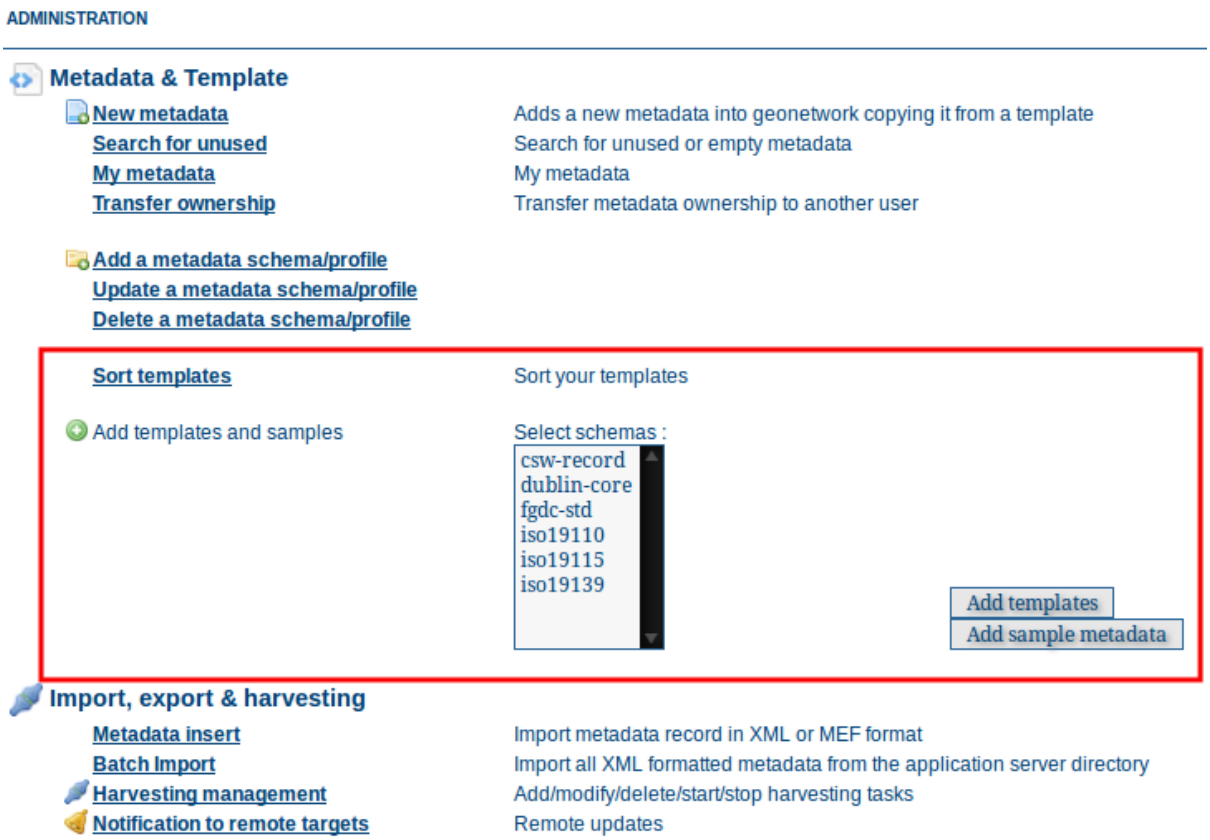
In the source code repository there are configuration files for collectd (and perhaps other monitoring software in the future).

CHAPTER 4

Managing Metadata

4.1 Templates

The *Metadata and Templates* options in the Administration page allows you to manage the metadata templates in the catalog. You have to be logged in as an administrator to access this page and function.



4.1.1 Sort templates

You can define the order in which Templates are listed when an Editor creates a new metadata record.

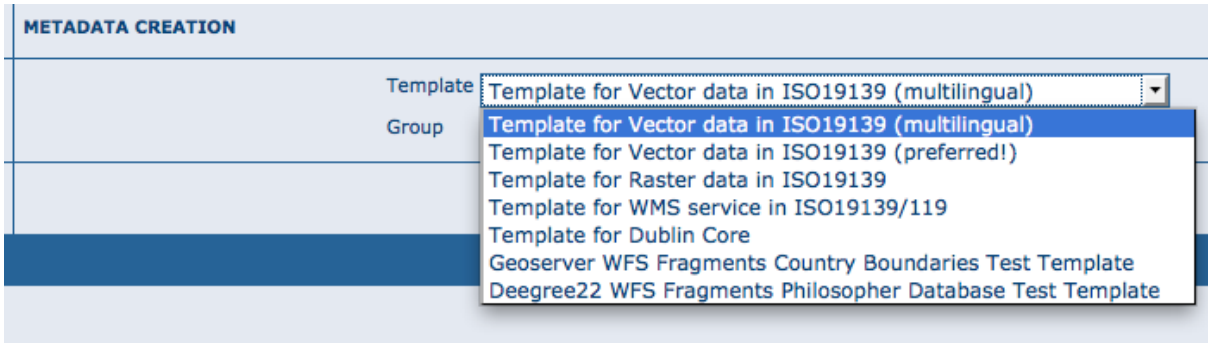


Figure 4.1: *The listing as shown to Editors*

Use drag and drop to re-order the templates.

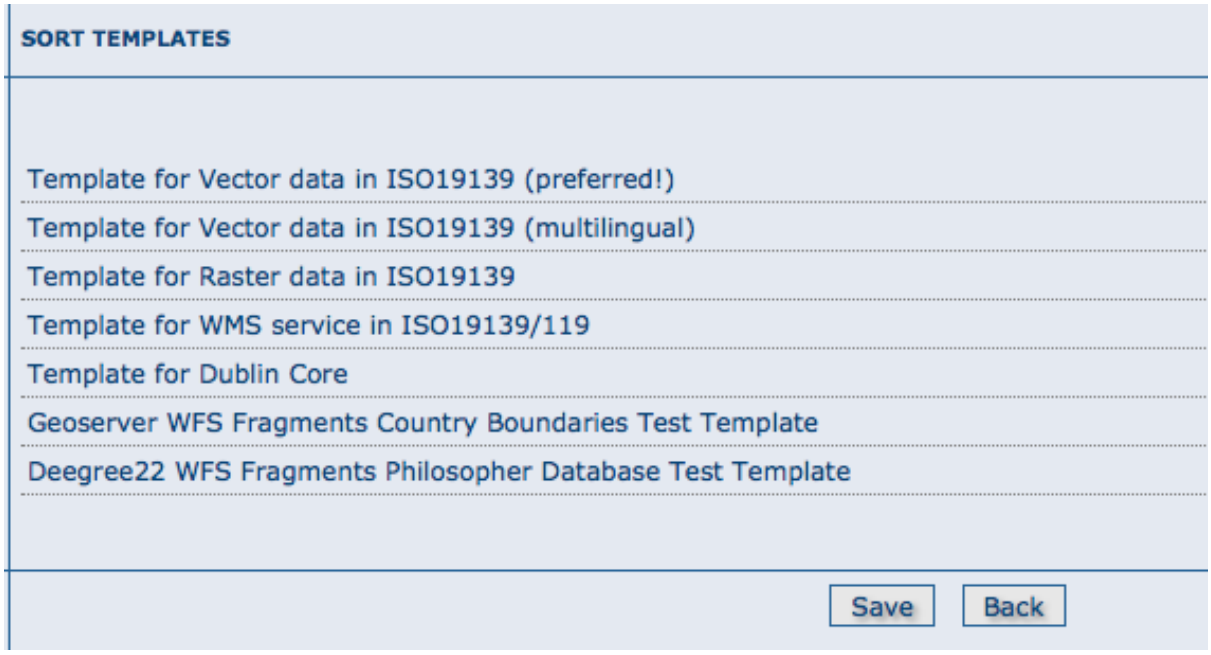


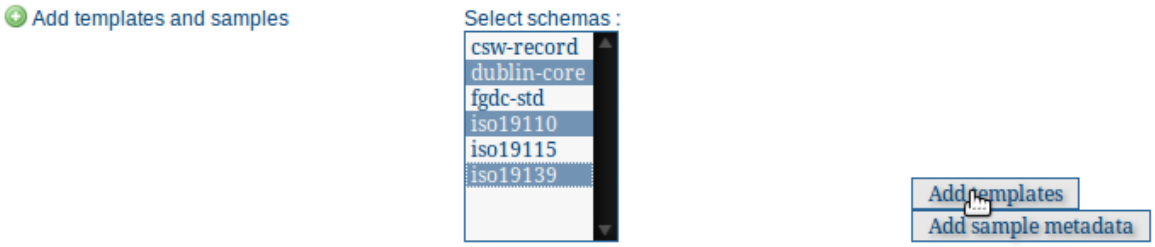
Figure 4.2: *Sort Templates panel*

4.1.2 Add templates

This option allows the user to select the metadata templates from any schema and add them to the catalogue.

> **Warning:** This will add the default templates available for each schema in `GEONETWORK_DATA_DIR/config/schema_plugins` - it should be used with care by an Administrator.

Select the metadata schemas to add templates from (multiple selections can be made) and click on the *Add templates* button to import them into the catalogue. They will then be available for creating new metadata records.



4.2 Ownership and Privileges

Please review and make sure that you understand *User Profiles* in the User and Group Administration section of this manual.

Note: A public metadata record is a metadata record that has the view privilege for the group named "All".

The following rules apply to Viewing and Editing permissions on a metadata record:

4.2.1 Viewing

An *administrator* can view any metadata.

A *content reviewer* can view a metadata if:

1. The metadata owner is member of one of the groups assigned to the reviewer.
2. She/he is the metadata owner.

A *user administrator* or an *editor* can view:

1. All metadata that has the view privilege selected for one of the groups she/he is member of.
2. All metadata created by her/him.

A *registered user* can view:

1. All metadata that has the view privilege selected for one of the groups she/he is member of.

Public metadata can be viewed by any user (logged in or not).

4.2.2 Editing

An *administrator* can edit any metadata.

A *reviewer* can edit a metadata if:

1. The metadata owner is member of one of the groups assigned to the reviewer.
2. She/he is the metadata owner.

A *User Administrator* or an *Editor* can only edit metadata she/he created.

4.2.3 Setting Privileges on a metadata record

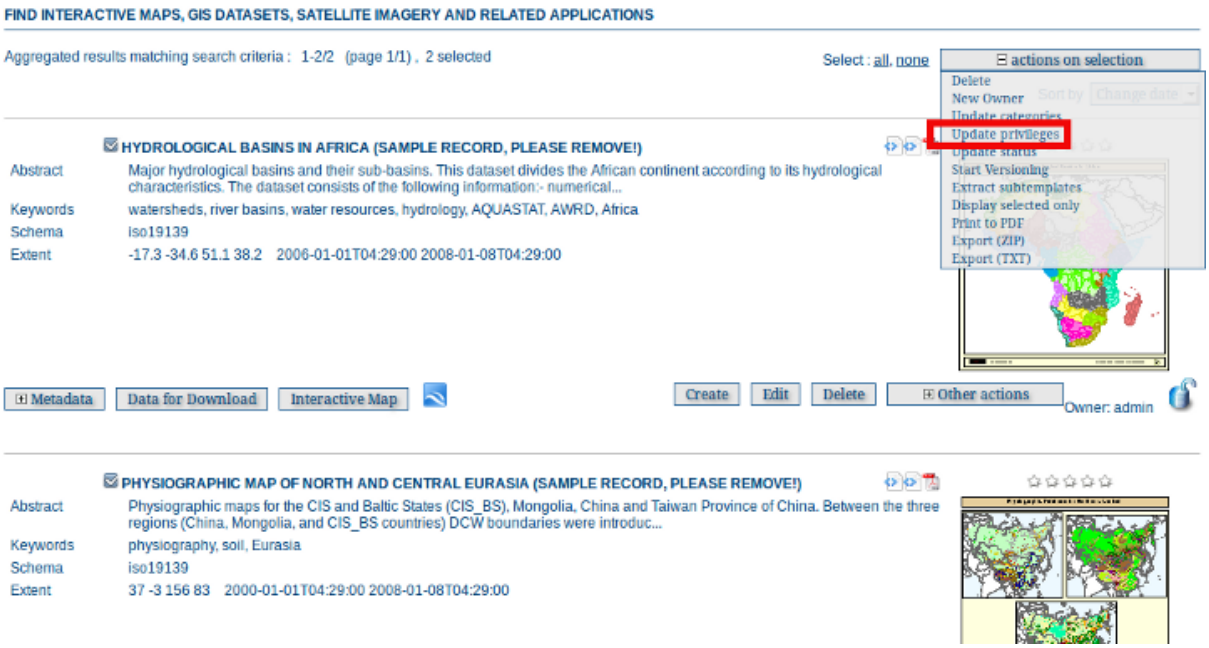
A button to access the Privileges page for a metadata record will appear in the search results or when the record is being viewed for:

1. All Administrators
2. All Reviewers that are member of one of the groups assigned to the metadata owner.
3. The Owner of the metadata

Privileges for the All and Intranet groups can only be edited by Administrators and Reviewers.

4.2.4 Setting Privileges on a selected set of metadata records

Privileges can be set on a selected set of records in the search results using the "actions on selected set" menu. The following screenshot shows how to access this function:



The following rules apply:

- the groups that will appear in the Privileges page will be those that the user belongs to
- the Privileges specified will only be applied to records that the user has ownership or administration rights on - any other records will be skipped.

4.2.5 Transfer Ownership

When metadata ownership needs to be transferred from one user to another for all or specific metadata records, the *Transfer Ownership* option is available. It is located in the Administration page and once selected, leads to the following page.

Initially, the page shows only a dropdown for a Source editor (the current metadata owner). The dropdown is filled with all GeoNetwork Users that have the Editor role and own some metadata. Selecting

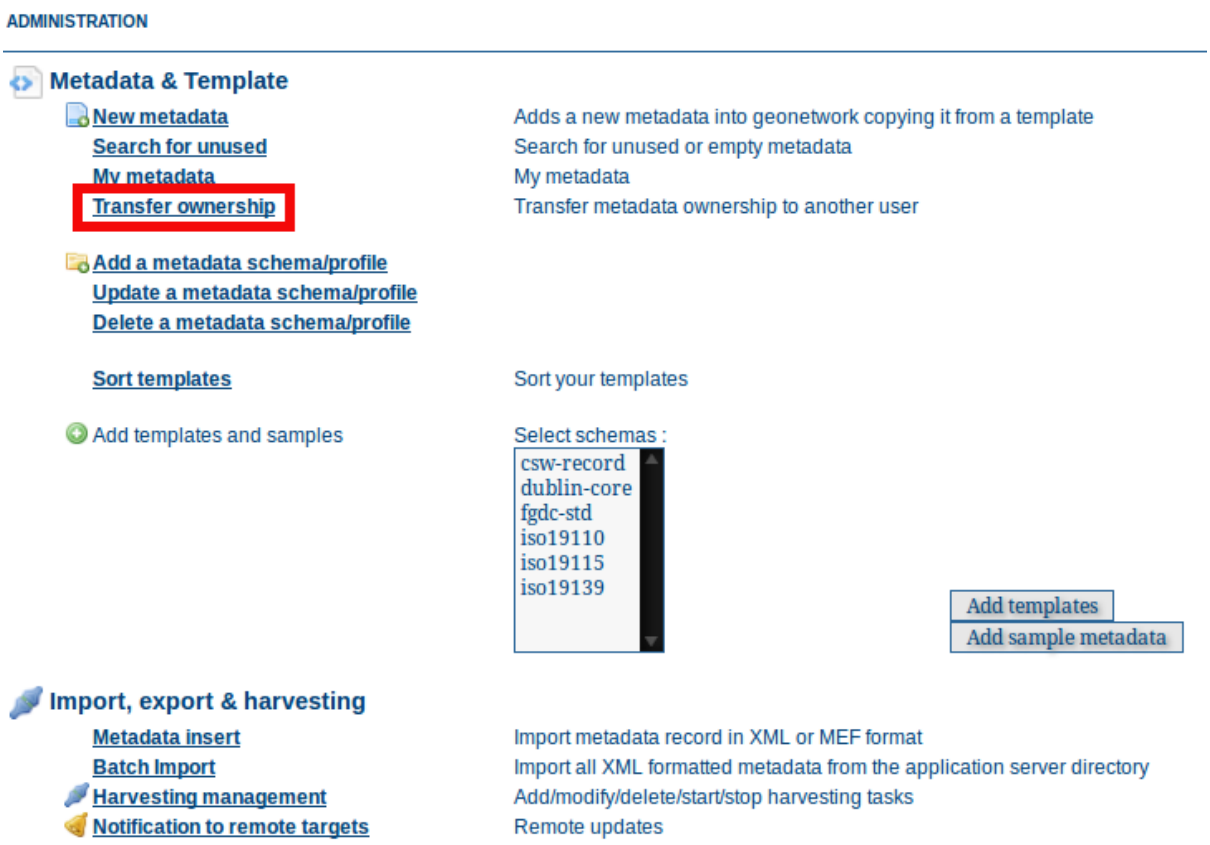


Figure 4.3: *How to open the Transfer Ownership page*

an Editor will select all metadata that is managed by that Editor. An empty dropdown means that there are no Editors with metadata associated and hence no transfer is possible.

Note: The drop down will be filled with all Editors visible to you. If you are not an Administrator, you will view only a subset of all Editors.

Figure 4.4: *The Transfer Ownership page*

Once a Source Editor has been selected, a set of rows is displayed. Each row refers to the group of the Editor for which there are privileges. The meaning of each column is the following:

1. *Source group*: This is a group that has privileges in the metadata that belong to the source editor. Put in another way, if one of the editor's metadata has privileges for one group, that group is listed here.

2. *Target group*: This is the destination group of the transferring process. All privileges relative to the source group are transferred to the target group. The target group drop down is filled with all

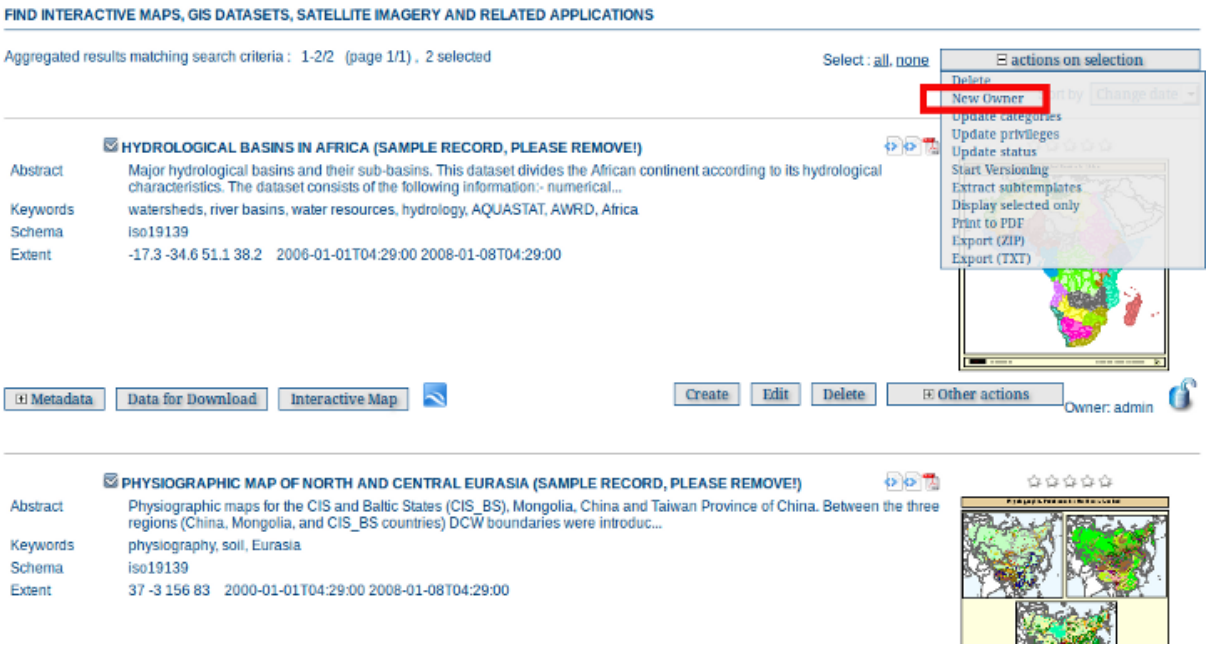
groups visible to the logged user (typically an administrator or a user administrator). By default, the Source group is selected in the target dropdown. Privileges to groups All and Intranet are not transferable.

3. *Target editor*: Once a Target group is selected, this drop down is filled with all editors that belong to that Target group.
4. *Operation*: Currently only the Transfer operation is possible.

By selecting the *Transfer* operation, if the Source group is different than the Target group, the system performs the Transfer of Ownership, shows a brief summary and removes the current row because now there are no privileges to transfer anymore.

4.2.6 Setting Ownership on a selected set of metadata records

Ownership can be set on a selected set of records in the search results using the "actions on selected set" menu. The following screenshot shows how to access this function:



The following rules apply:

- Only *administrators* or *user administrators* can set ownership on a selected set of records
- *administrators* can set ownership to any user
- *user administrators* can set ownership to any user in the same group(s) as them
- Ownership will only be transferred on those records that the ownership or administration rights on - any others will be skipped.

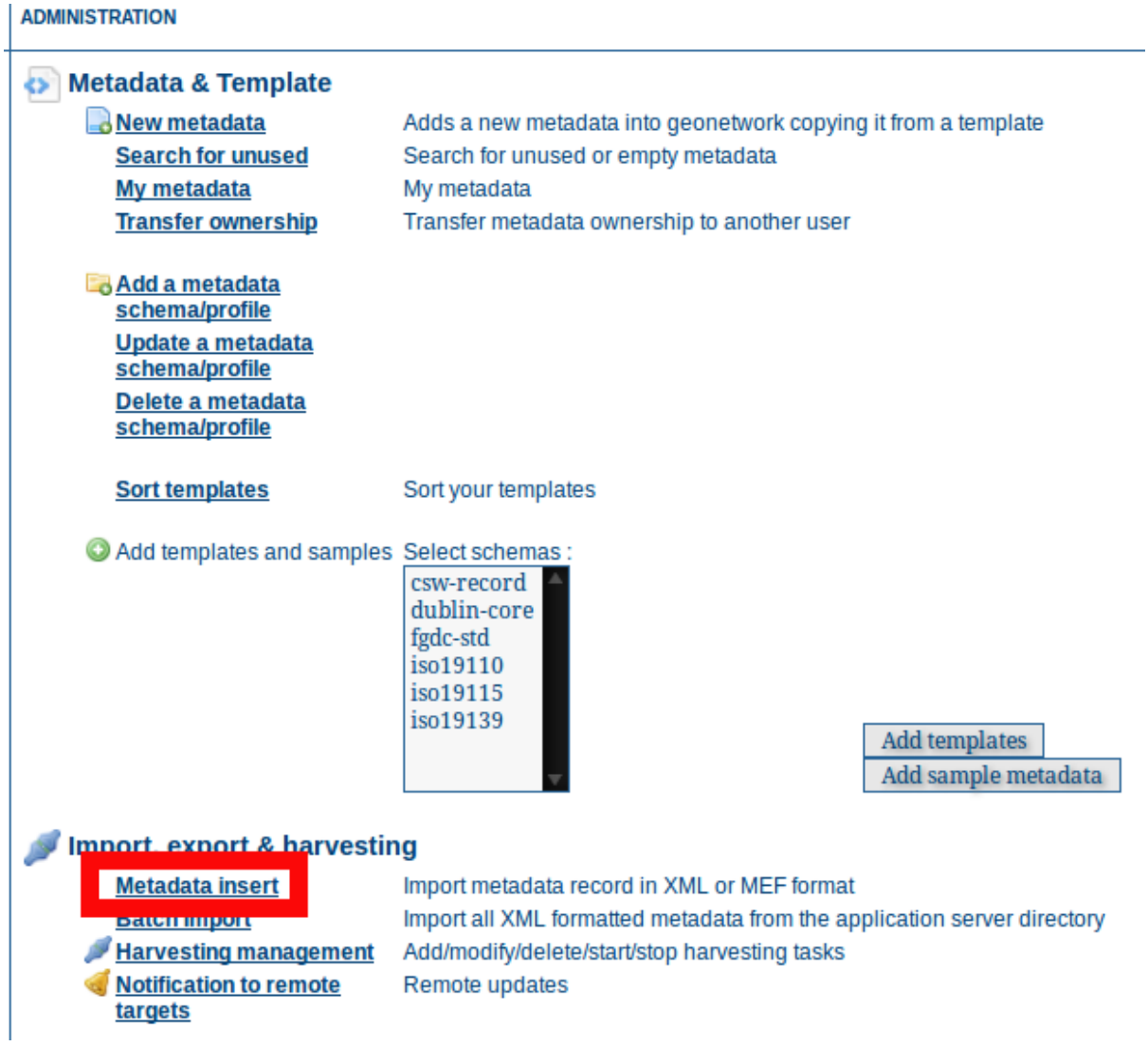
4.3 Import facilities

4.3.1 Importing a metadata record from XML or a MEF file

The file import facility allows you to import metadata records in three ways:

1. XML file from the filesystem on your machine.
2. MEF file from the filesystem on your machine
3. Copy/Paste XML

In order to use this facility, you have to be logged in as an editor. After the login step, go to the administration page and select the Metadata insert link.



Clicking the link will open the metadata import page. You will then have to specify a set of parameters. The following screenshot shows the parameters for importing an XML file.

We'll describe the options you see on this page because they are common ways you can import metadata records in this interface.

- *File Type* - First option is to choose the type of metadata record you are loading. The two choices are:
- *Metadata* - use when loading a normal metadata record
- *Template* - use when loading a metadata record that will be used as a template to build new records in the editor.

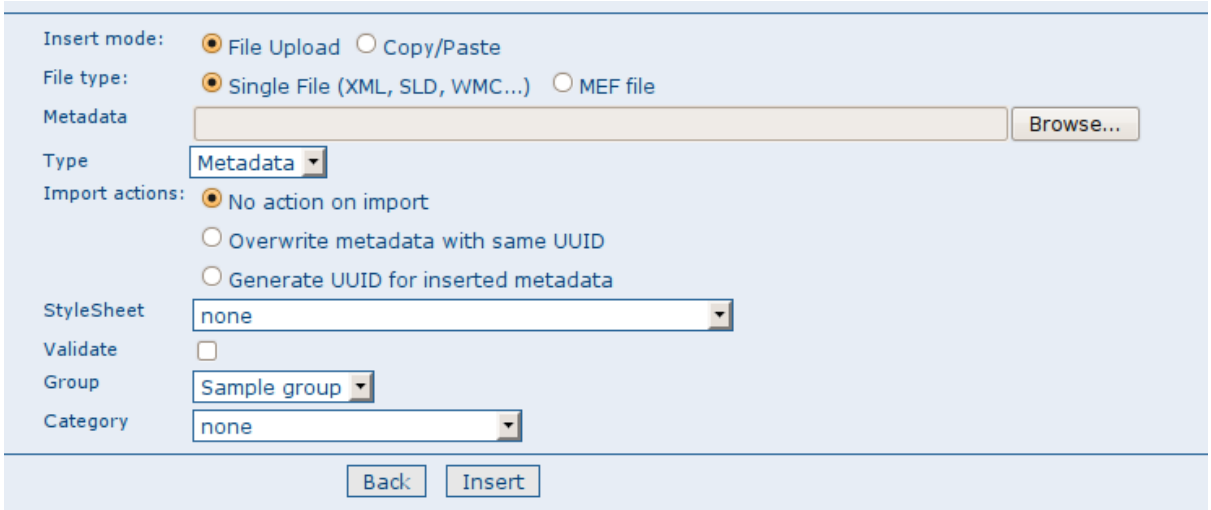


Figure 4.5: *The XML file import options*

- *Import Action* - This option group determines how to handle potential clashes between the UUID of the metadata record you are loading and the UUIDs of metadata records already present in the catalog. There are three actions and you can select one:
- *No action on import* - the UUID of the metadata record you are loading is left unchanged. If a metadata record with the same UUID is already present in the catalog, you will receive an error message.
- *Overwrite metadata with same UUID* - any existing metadata record in the catalog with the same UUID as the record you are loading will be replaced with the metadata record you are loading.
- *Generate UUID for inserted metadata* - create new a UUID for the metadata records you are loading.
- *Stylesheet* - Allows you to transform the metadata record using an XSLT stylesheet before loading the record. The drop down control is filled with the names of files taken from the *INSTALL_DIR/web/geonetwork/xsl/conversion/import folder*. (Files can be added to this folder without restarting GeoNetwork). As an example, you could use this option to convert a metadata into schema that is supported by GeoNetwork.
- *Validate* - The metadata is validated against its schema before loading. If it is not valid it will not be loaded.
- *Group* - Use this option to select a user group to assign to the imported metadata.
- *Category* - Use this option to select a local category to assign to the imported metadata. Categories are local to the catalogue you are using and are intended to provide a simple way of searching groups of metadata records.

MEF file import

If you select *MEF file* in the *File type* option, only the Import actions option group is show. See above for more details. Note: a MEF file can contain more than one metadata record.

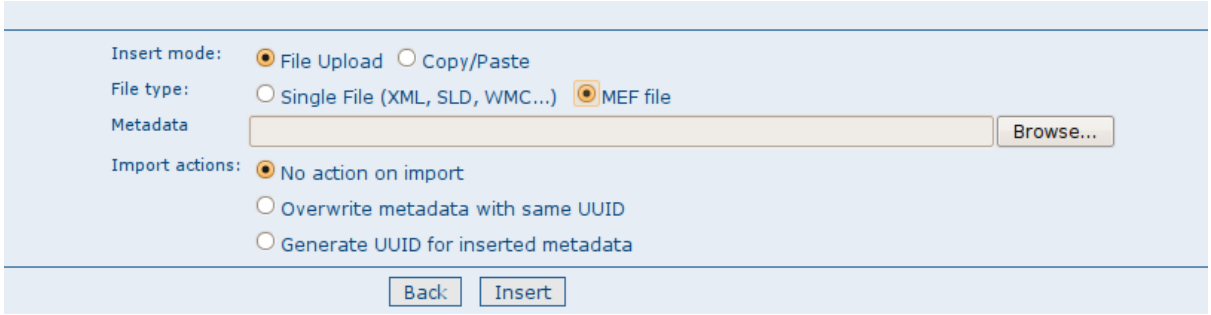


Figure 4.6: *The MEF file import options*

Copy/Paste XML

If you select *Copy/Paste* in the Insert mode option, then a text box appears. You can copy the XML from another window and paste it into that text box. The options for loading that XML are the same as those for loading an XML file - see above.

4.3.2 Batch import

The batch import facility allows you to import a set of metadata records in the form of XML or MEF files. In order to use this facility, you have to be logged in as an administrator. After the login step, go to the administration page and select the *Batch Import* link.

Clicking the link will open the batch import page. You will then have to specify a set of parameters. The following screenshot shows the parameters for batch import of a set of XML or MEF files.

- *Directory* This is the full path on the server's file system of the directory to scan. GeoNetwork will look for and try to import all XML or MEF files present into this directory. It is important to notice that this is the directory on the *server* machine and *not* on the client of the user that is doing the import.
- *File Type* - First option is to choose the type of metadata record you are loading. The two choices are:
- *Metadata* - use when loading a normal metadata record
- *Template* - use when loading a metadata record that will be used as a template to build new records in the editor.
- *Import Action* - This option group determines how to handle potential clashes between the UUID of the metadata record you are loading and the UUIDs of metadata records already present in the catalog. There are three actions and you can select one:
- *No action on import* - the UUID of the metadata record you are loading is left unchanged. If a metadata record with the same UUID is already present in the catalog, you will receive an error message.
- *Overwrite metadata with same UUID* - any existing metadata record in the catalog with the same UUID as the record you are loading will be replaced with the metadata record you are loading.
- *Generate UUID for inserted metadata* - create new a UUID for the metadata records you are loading.

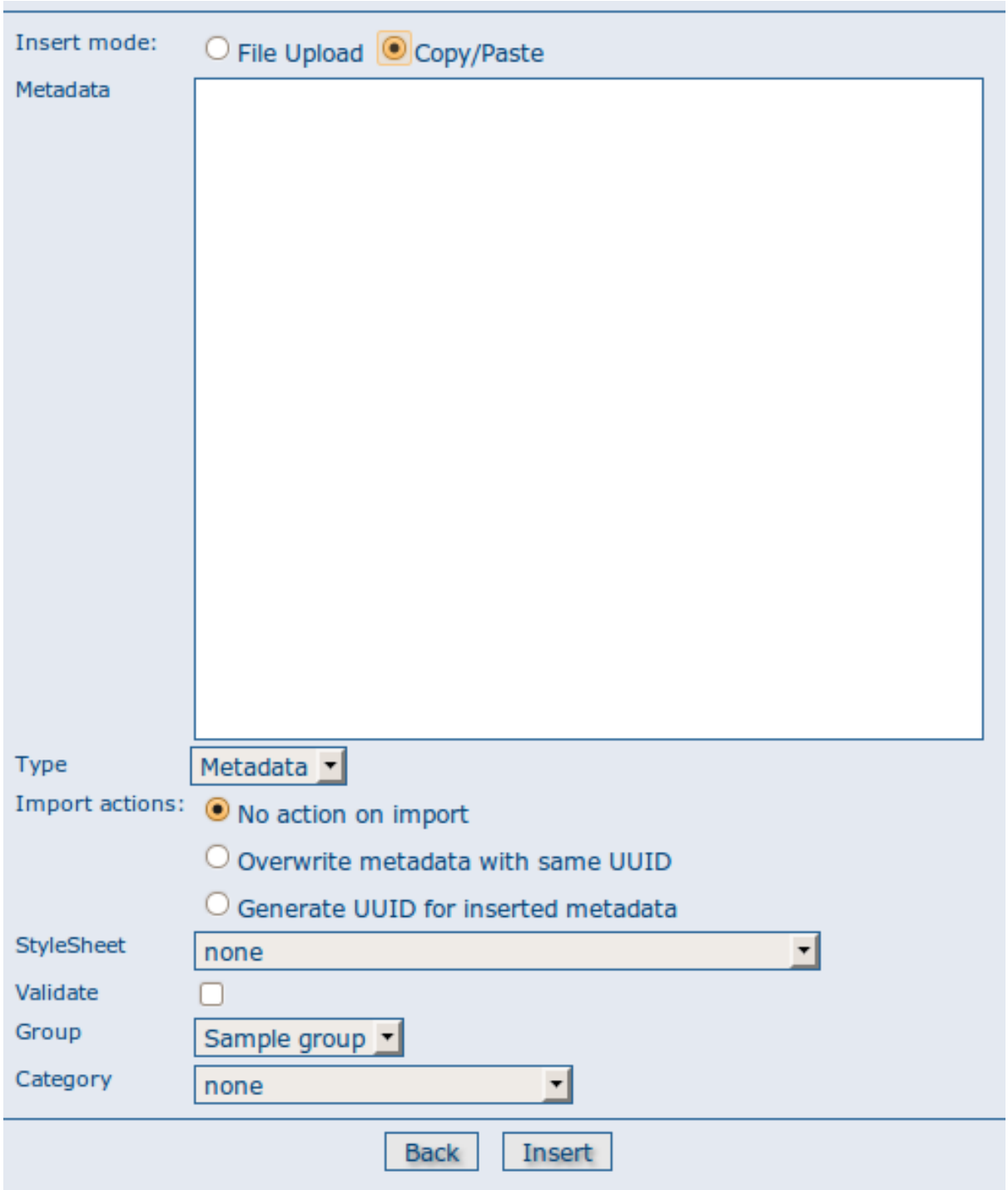


Figure 4.7: *The XML insert options*

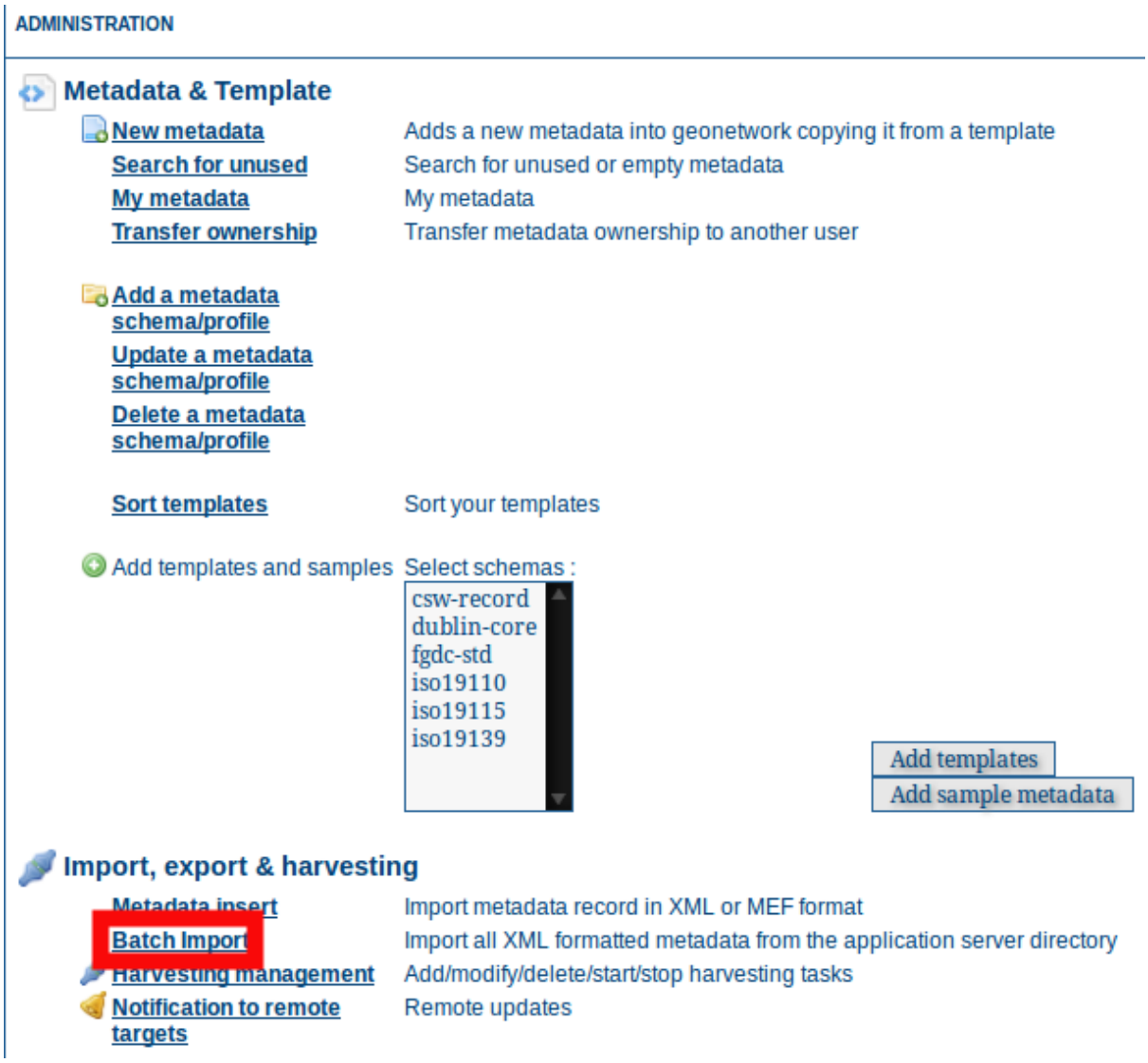


Figure 4.8: *How to reach the batch import page*

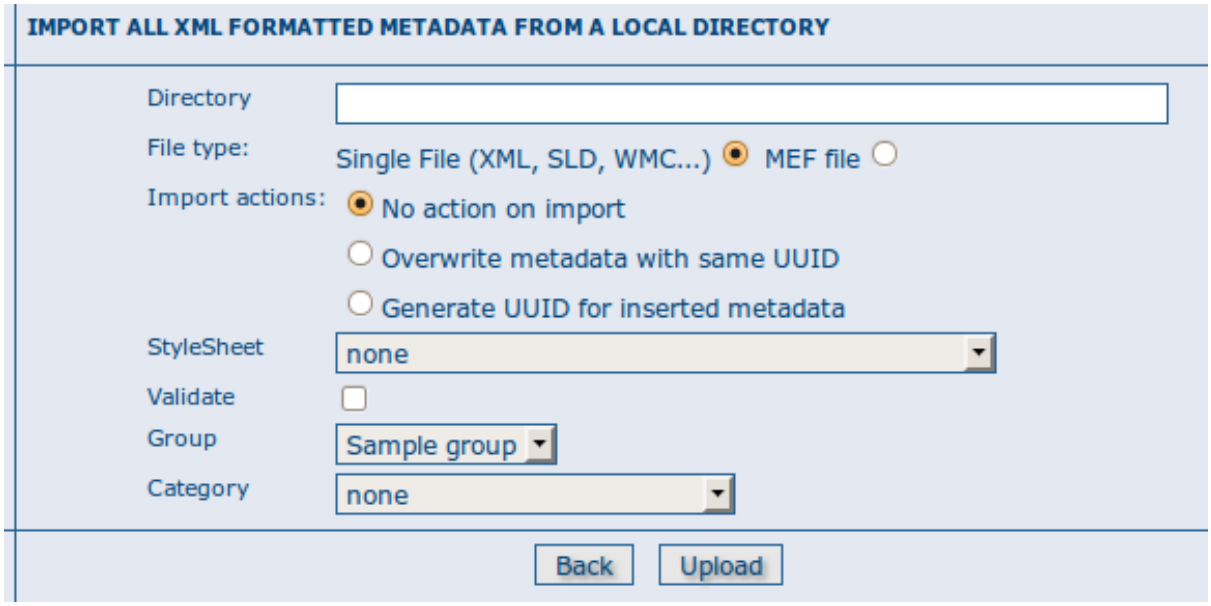


Figure 4.9: *The batch import options*

- *Stylesheet* - Allows you to transform the metadata record using an XSLT stylesheet before loading the record. The drop down control is filled with the names of files taken from the *INSTALL_DIR/web/geonetwork/xsl/conversion/import folder*. (Files can be added to this folder without restarting GeoNetwork). As an example, you could use this option to convert a metadata into schema that is supported by GeoNetwork.
- *Validate* - The metadata is validated against its schema before loading. If it is not valid it will not be loaded.
- *Group* - Use this option to select a user group to assign to the imported metadata.
- *Category* - Use this option to select a local category to assign to the imported metadata. Categories are local to the catalogue you are using and are intended to provide a simple way of searching groups of metadata records.

At the bottom of the page there are two buttons:

- *Back* Goes back to the administration form.
- *Upload* Starts the import process.

Notes on the batch import process

- When the import process ends, the total count of imported metadata will be shown
- The import is transactional: the metadata set will be fully imported or fully discarded (there are no partial imports)
- Files that start with '.' or that do not end with '.xml' or '.mef' are ignored

Structured batch import using import-config.xml

Finer control of the batch import process can be obtained by structuring the metadata files into directories mapped to categories and metadata schemas and describing the mapping in a file called import-

config.xml.

The import-config.xml should be placed in the directory from which you will batch import (see *Directory* parameter above). It has a config root element with the following children:

1. categoryMapping [1]: this element specifies the mapping of directories to categories.
   (a) mapping [0..n]: This element can appear 0 or more times and maps one directory name to a category name. It must have a "dir" attribute that indicates the directory and a "to" attribute that indicates the category name.
   (b) default [1]: This element specifies a default mapping of categories for all directories that do not match the other mapping elements. The default element can only have one attribute called "to".
2. schemaMapping [1]: this element specifies the mapping of directories to metadata schemas.
   (a) mapping [0..n]: This element can appear 0 or more times and maps one directory to the schema name that must be used when importing. The provided schema must match the one used by the metadata contained into the specified directory, which must all have the same schema. It must have a "dir" attribute that indicates the directory and a "to" attribute that indicates the schema name.
   (b) default [1]: default behaviour to use when all other mapping elements do not match. The default element can only have one attribute called "to".

Here is an example of the import-config.xml file:

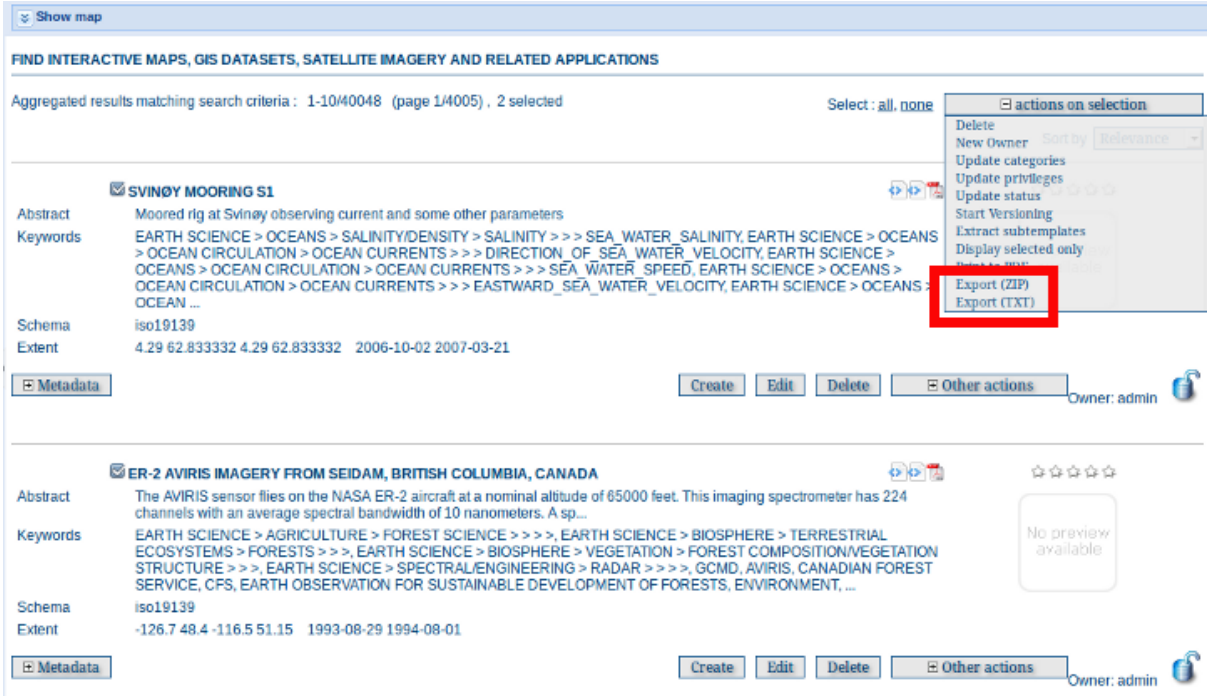
```
<config>
        <categoryMapping>
                <mapping dir="1" to="maps" />
                <mapping dir="3" to="datasets" />
                <mapping dir="6" to="interactiveResources" />
                <mapping dir="30" to="photo" />
                <default to="maps" />
        </categoryMapping>
        <schemaMapping>
                <mapping dir="3" to="fgdc-std" />
                <default to="dublin-core" />
        </schemaMapping>
</config>
```

As described above, the import procedure starts by scanning the specified *Directory*. Apart from the import-config.xml file, this directory should only contain subdirectories - these are the category directories referred to in the categoryMapping section of the import-config.xml file described above. Each of the category directories should only contain subdirectories - these are the schema directories referred to in the schemaMapping section of the import-config.xml file described above.

4.4 Export facilities

GeoNetwork has two different types of export function - both of which operate on selected sets of metadata from the search results. As such they are accessible from the "actions on selection" menu as shown in the following example:

The export functions: ZIP and CSV - highlighted



4.4.1 Export as a ZIP archive

When a selected set of metadata records is exported as a ZIP archive, each metadata record is inserted in the ZIP archive as a directory containing the metadata, any data uploaded with the metadata record and the thumbnails. This type of ZIP archive is the MEF (Metadata Exchange Format) Version 2.0. You can find more details of MEF V2 in the GeoNetwork Developers Manual.

4.4.2 Export as a CSV file

When a selected set of metadata records is exported as a CSV/TXT file, the following process takes place for each metadata record:

- a brief summary of some of the elements from each selected metadata record is generated by applying the brief template from the metadata schema eg. for an iso19139 metadata record the brief template from `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/present/metadata-iso19139` would be applied to the metadata record.
- the elements common to the brief summary elements for all metadata records are extracted (as they may differ according to the metadata schema)
- a title record with comma separated element names is created
- the content of each element is laid out in comma separated form. Where there is more than one child element in the brief element (eg. for geoBox), the content from each child element is separated using '###'.

An example of an ISO metadata record in CSV format is shown as follows:

```
"schema","uuid","id","title","geoBox","metadatacreationdate"
"iso19139","6ad9e3b8-907e-477b-9d82-1cc4efd2581d","17729","ER-2 AVIRIS Imagery","-126.7#
```

It is possible to override the brief summary of metadata elements by creating a special template in the presentation XSLT of the metadata schema. As an example of how to do this, we will override the brief summary for the iso19139 schema and replace it with just one element: gmd:title. To do this we create an XSLT template as follows:

```
<xsl:template match="gmd:MD_Metadata" mode="csv">
  <xsl:param name="internalSep"/>
  <metadata>
    <!-- add in our field -->
    <xsl:copy-of select="gmd:identificationInfo/gmd:MD_DataIdentification/gmd:citation/g
    <!-- copy geonet:info element in - has special metadata eg schema name  -->
    <xsl:copy-of select="geonet:info"/>
  </metadata>
</xsl:template>
```

This template, when added to `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/present/me` will replace the brief summary (produced by the brief template) with just one element, gmd:title.

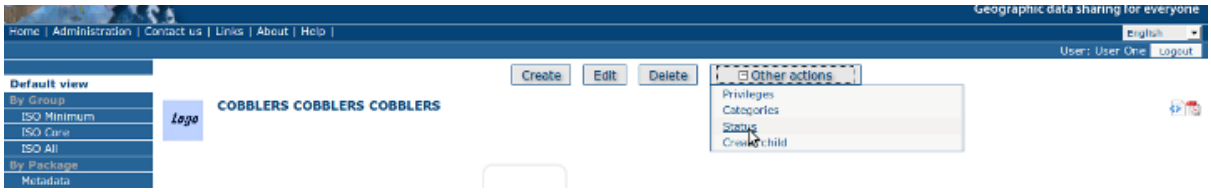
4.5 Status

Metadata records have a lifecycle that typically goes through one or more states. For example, when a record is created and edited by an 'Editor' user it is in the 'Draft' state. Whilst it is reviewed by a 'Content Reviewer' user it would typically be in a 'Submitted' state. If the record is found to be complete and correct by the 'Content Reviewer' it would be in the 'Approved' state and may be made available for casual search and harvest by assigning privileges to the GeoNetwork 'All' group. Eventually, the record may be superseded or replaced and the state would be 'Retired'.

GeoNetwork has (an extensible) set of states that a metadata record can have:

- **Unknown** - this is the default state - nothing is known about the status of the metadata record
- **Draft** - the record is under construction or being edited.
- **Submitted** - the record has been submitted for approval to a content review.
- **Approved** - the content reviewer has reviewed and approved the metadata record
- **Rejected** - the content reviewer has reviewed and rejected the metadata record
- **Retired** - the record has been retired

Status can be assigned to metadata records individually or as a selected set.



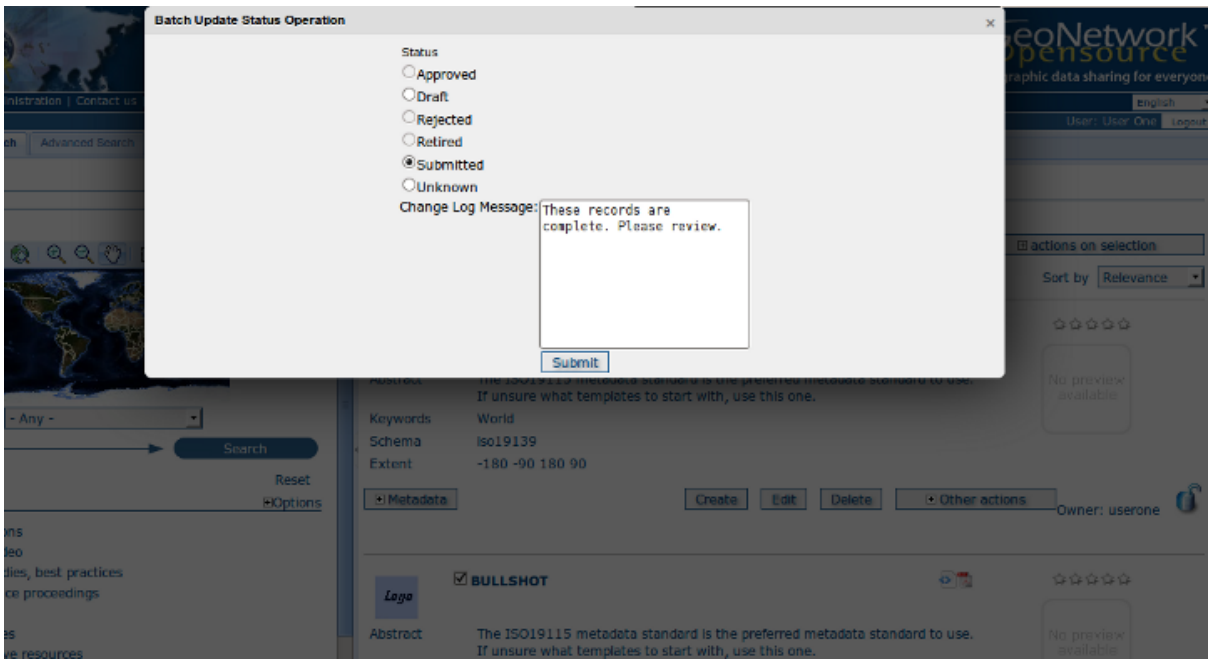
Initiating status change for a single metadata record

Initiating status change for a set of metadata records

The interface for setting the status looks like the following:

Changing the status of a set of metadata records

Geographic data sharing for everyone
Home | Administration | Contact us | Links | About | Help |
User: User One
Logout
Simple Search
Advanced Search
Show map
WHAT?
WHERE?
- Any -
Search
Reset
Options
Applications
Audio/Video
Case studies, best practices
Conference proceedings
Datasets
Directories
FIND INTERACTIVE MAPS, GIS DATASETS, SATELLITE IMAGERY AND RELATED APPLICATIONS
Aggregate Results matching search criteria : 1-2/2 (page 1/1) , 2 selected
Select : all, none
actions on selection
Delete
Update categories
Update privileges
Update status
Display selected only
Print to PDF
Export (ZIP)
Export (TXT)
COBBLERS COBBLERS COBBLERS
Abstract
The ISO19115 metadata standard is the preferred metadata standard to use.
If unsure what templates to start with, use this one.
Keywords
World
Schema
iso19139
Extent
-180 -90 180 90
Metadata
Create
Edit
Delete
Other actions
Owner: userone
BULLSHOT
Abstract
The ISO19115 metadata standard is the preferred metadata standard to use.


Batch Update Status Operation
Status
Approved
Draft
Rejected
Retired
Submitted
Unknown
Change Log Message:
These records are complete. Please review.
Submit
Sort by
Relevance
actions on selection
Keywords
World
Schema
iso19139
Extent
-180 -90 180 90
Metadata
Create
Edit
Delete
Other actions
Owner: userone
BULLSHOT
Abstract
The ISO19115 metadata standard is the preferred metadata standard to use.
If unsure what templates to start with, use this one.

It is also possible to search for metadata records with a particular status using a search restriction in the 'Advanced Search' menu.

4.5.1 Status actions

The status values shown above are held in a database table called MetadataStatus. Extra states can be added to this table if required.

There are two status change action hooks (in Java) that can be used by sites to provide specific behaviours:

1. *statusChange* - This action is called when status is changed by a user eg. when 'Draft' records are set to 'Submitted' and could be used for example to send notifications to other users affected by this change.
2. *onEdit* - This action is called when a record is edited and saved and could be used for example to reset records with an 'Approved' status to 'Draft' status. A default set of actions is provided. These can be customised or replaced by sites that wish to provide different or more extensive behaviour.

A default pair of metadata status change actions defined in Java is provided wit GeoNetwork - see the class `org.fao.geonet.services.metadata.DefaultStatusActions.java`.

statusChange: This action is called when status is changed by a user. What happens depends on the status change taking place:

- when an 'Editor' changes the state on a metadata record(s) from 'Draft' or 'Unknown' to 'Submitted', the Content Reviewers from the groupOwner of the record are informed of the status change via email which looks like the following. They can log in and click on the link supplied in the email to access the submitted records. Here is an example email sent by this action:

```
Date: Tue, 13 Dec 2011 12:58:58 +1100 (EST)
From: Metadata Workflow <feedback@localgeonetwork.org.au>
Subject: Metadata records SUBMITTED by userone@localgeonetwork.org.au (User One) on
To: "reviewer@localgeonetwork.org.au" <Reviewer@localgeonetwork.org.au>
Reply-to: User One <userone@localgeonetwork.org.au.au>
Message-id: <1968852534.01323741538713.JavaMail.geonetwork@localgeonetwork.org.au>

These records are complete. Please review.

Records are available from the following URL:
http://localgeonetwork.org.au/geonetwork/srv/en/main.search?_status=4&_statusChange
```

- when a 'Content Reviewer' changes the state on a metadata record(s) from 'Submitted' to 'Accepted' or 'Rejected', the owner of the metadata record is informed of the status change via email. The email received by the metadata record owner looks like the following. Again, the user can log in and use the link supplied in the email to access the approved/rejected records. Here is an example email sent by this action:

```
Date: Wed, 14 Dec 2011 12:28:01 +1100 (EST)
From: Metadata Workflow <feedback@localgeonetwork.org.au>
Subject: Metadata records APPROVED by reviewer@localgeonetwork.org.au (Reviewer) on
To: "User One" <userone@localgeonetwork.org.au>
Message-ID: <1064170697.31323826081004.JavaMail.geonetwork@localgeonetwork.org.au>
Reply-To: Reviewer <reviewer@localgeonetwork.org.au>
```

```
Records approved - please resubmit for approval when online resources attached

Records are available from the following URL:
http://localgeonetwork.org.au/geonetwork/srv/en/main.search?_status=2&_statusChange
```

onEdit: This action is called when a record is edited and saved by a user. If the user did not indicate that the edit changes were a 'Minor edit' and the current status of the record is 'Approved', then the default action is to set the status to 'Draft' and remove the privileges for the GeoNetwork group 'All'.

4.5.2 Changing the status actions

These actions can be replaced with different behaviours by:

1. writing Java code in the form of a new class that implements the interface defined in org.fao.geonet.services.metadata.!StatusActions.java and placing a compiled version of the class in the GeoNetwork class path
2. defining the name of the new class in the statusActionsClass configuration parameter in `INSTALL_DIR/web/geonetwork/WEB-INF/config.xml`

4.6 Versioning

There are many use cases where it is important to be able to track (over time):

- changes to the metadata record
- changes to properties of the metadata record eg. privileges, categories, status

GeoNetwork uses a subversion repository to capture these changes and allow the user to examine the changes through the various visual interfaces to subversion repositories that already exist eg. viewvc. Apart from the advantage of ready to use tools for examining the changes, the subversion approach is efficient for XML files and simple to maintain.

The database remains the point of truth for GeoNetwork. That is, changes will be tracked in subversion, but all services will continue to extract the latest version of the metadata record from the database.

4.6.1 Selecting records to version

Not all records in GeoNetwork are tracked as the compute and systems admin cost of this tracking for every record, particularly in large catalogs, is too high. Instead only those records selected by the user in the local GeoNetwork instance will be tracked in the subversion repository.

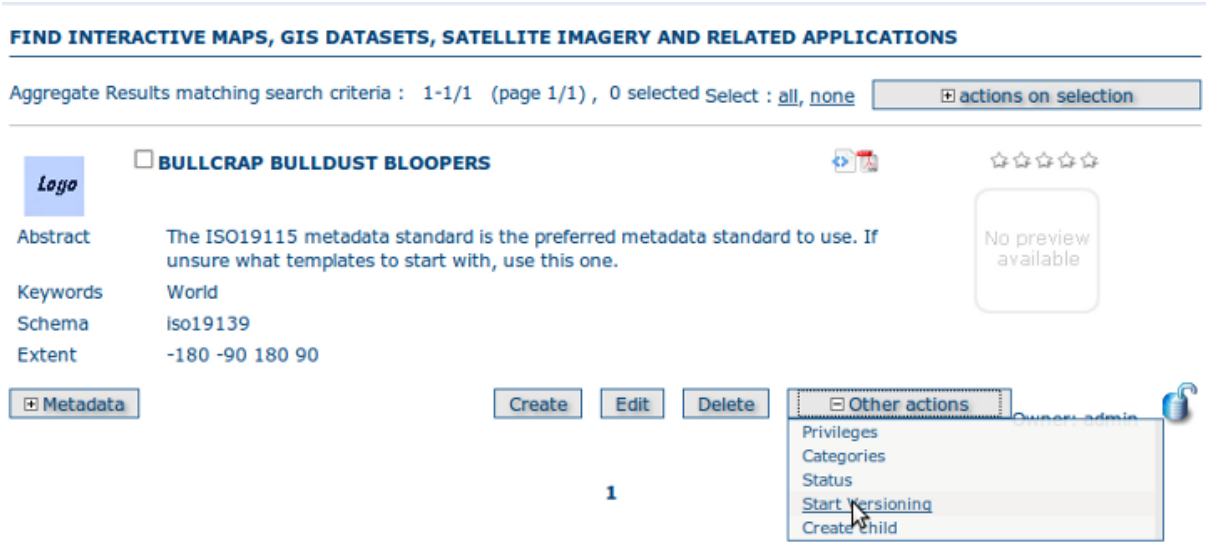
Records can be selected for versioning individually or by doing a search and selecting a set of records.

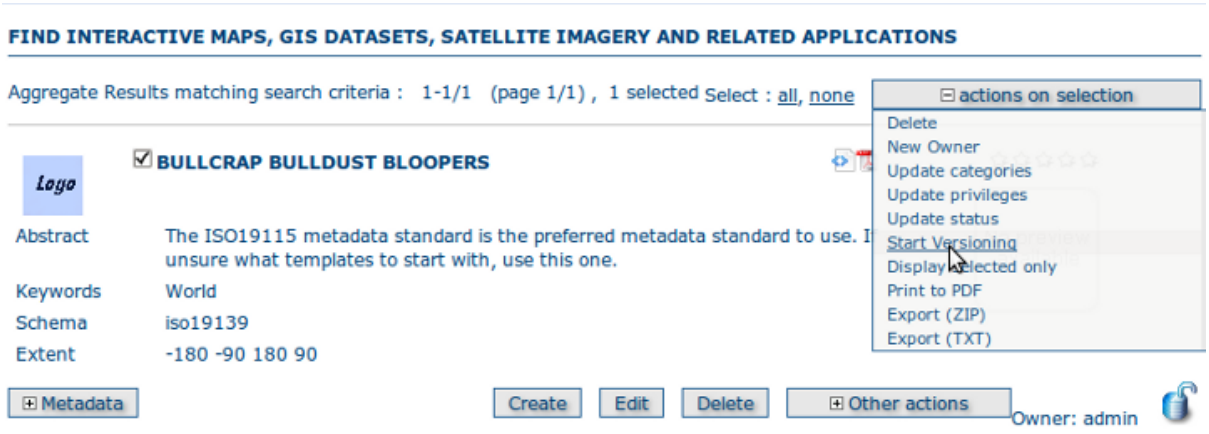
Starting versioning on a single record

Starting versioning on a selected set of records

4.6.2 When will a new version be created?

Metadata records that are processed by a GeoNetwork service are associated with a database session. When the database session is committed, the metadata XML and its properties (as XML) are selected


FIND INTERACTIVE MAPS, GIS DATASETS, SATELLITE IMAGERY AND RELATED APPLICATIONS
Aggregate Results matching search criteria : 1-1/1 (page 1/1) , 0 selected Select : all, none
actions on selection
Logo
BULLCRAP BULLDUST BLOOPERS
Abstract
The ISO19115 metadata standard is the preferred metadata standard to use. If unsure what templates to start with, use this one.
Keywords
World
Schema
iso19139
Extent
-180 -90 180 90
No preview available
Metadata
Create
Edit
Delete
Other actions
Privileges
Categories
Status
Start Versioning
Create child
1


FIND INTERACTIVE MAPS, GIS DATASETS, SATELLITE IMAGERY AND RELATED APPLICATIONS
Aggregate Results matching search criteria : 1-1/1 (page 1/1) , 1 selected Select : all, none
actions on selection
Delete
New Owner
Update categories
Update privileges
Update status
Start Versioning
Display selected only
Print to PDF
Export (ZIP)
Export (TXT)
Logo
BULLCRAP BULLDUST BLOOPERS
Abstract
The ISO19115 metadata standard is the preferred metadata standard to use.
unsure what templates to start with, use this one.
Keywords
World
Schema
iso19139
Extent
-180 -90 180 90
Metadata
Create
Edit
Delete
Other actions
Owner: admin

from the database and passed as a commit to the subversion repository, creating a new version in the repository. This process is automatic - at the moment the user cannot force a new version to be created, unless they change the metadata record or its properties.

Due to recent changes in the way in which GeoNetwork database sessions are committed (forced by the adoption of background threads for work tasks) and the implementation dependent way in which database transaction isolation is handled by different vendors, there is a small chance that database sessions may overlap. This may mean that the ordering of the changes committed to the subversion repository may not be correct in a small number of cases. After some discussion amongst the developers, the implementation may change to remove this possibility in the next version of GeoNetwork.

4.6.3 How the changes are held in the subversion repository

The metadata record and its properties are stored in the subversion repository as XML files. The structure of the XML files describing the properties of the metadata is that returned by SELECT statements on the relevant database tables. The typical structure of a directory for a metadata record in the repository consists of a directory (named after the id of the metadata record) which contains:

- metadata.xml - the XML metadata record
- owner.xml - an XML file describing the owner of the metadata record
- privileges.xml - an XML file describing the privileges of the metadata record
- categories.xml - an XML file describing the categories to which the metadata record has been assigned
- status.xml - an XML file describing the status of the metadata (eg. Approved, Rejected, etc)

A typical example of a privileges.xml file stored in the repository

```
<response>
 <record>
   <group_name>intranet</group_name>
   <operation_id>0</operation_id>
   <operation_name>view</operation_name>
 </record>
 <record>
   <group_name>sample</group_name>
   <operation_id>0</operation_id>
   <operation_name>view</operation_name>
 </record>
 <record>
   <group_name>all</group_name>
   <operation_id>0</operation_id>
   <operation_name>view</operation_name>
 </record>
 <record>
   <group_name>intranet</group_name>
   <operation_id>1</operation_id>
   <operation_name>download</operation_name>
 </record>
 <record>
   <group_name>all</group_name>
   <operation_id>1</operation_id>
   <operation_name>download</operation_name>
 </record>
```

```
 <record>
   <group_name>sample</group_name>
   <operation_id>3</operation_id>
   <operation_name>notify</operation_name>
 </record>
 <record>
   <group_name>intranet</group_name>
   <operation_id>5</operation_id>
   <operation_name>dynamic</operation_name>
 </record>
 <record>
   <group_name>all</group_name>
   <operation_id>5</operation_id>
   <operation_name>dynamic</operation_name>
 </record>
 <record>
   <group_name>intranet</group_name>
   <operation_id>6</operation_id>
   <operation_name>featured</operation_name>
 </record>
 <record>
   <group_name>all</group_name>
   <operation_id>6</operation_id>
   <operation_name>featured</operation_name>
 </record>
</response>
```

Difference between revisions 3 and 4 for the privileges.xml file for metadata record 10:

```
svn diff -r 3:4
Index: 10/privileges.xml
===================================================================
--- 10/privileges.xml   (revision 3)
+++ 10/privileges.xml   (revision 4)
@@ -1,12 +1,52 @@
 <response>
   <record>
+    <group_name>intranet</group_name>
+    <operation_id>0</operation_id>
+    <operation_name>view</operation_name>
+  </record>
+  <record>
     <group_name>sample</group_name>
     <operation_id>0</operation_id>
     <operation_name>view</operation_name>
   </record>
   <record>
+    <group_name>all</group_name>
+    <operation_id>0</operation_id>
+    <operation_name>view</operation_name>
+  </record>
+  <record>
+    <group_name>intranet</group_name>
+    <operation_id>1</operation_id>
+    <operation_name>download</operation_name>
+  </record>
+  <record>
+    <group_name>all</group_name>
```

```
+    <operation_id>1</operation_id>
+    <operation_name>download</operation_name>
+  </record>
+  <record>
     <group_name>sample</group_name>
     <operation_id>3</operation_id>
     <operation_name>notify</operation_name>
   </record>
+  <record>
+    <group_name>intranet</group_name>
+    <operation_id>5</operation_id>
+    <operation_name>dynamic</operation_name>
+  </record>
+  <record>
+    <group_name>all</group_name>
+    <operation_id>5</operation_id>
+    <operation_name>dynamic</operation_name>
+  </record>
+  <record>
+    <group_name>intranet</group_name>
+    <operation_id>6</operation_id>
+    <operation_name>featured</operation_name>
+  </record>
+  <record>
+    <group_name>all</group_name>
+    <operation_id>6</operation_id>
+    <operation_name>featured</operation_name>
+  </record>
 </response>
```

Examination of this diff file shows that privileges for the 'All' and 'Intranet' groups have been added between revision 3 and 4 - in short, the record has been published.

Here is an example of a change that has been made to a metadata record:

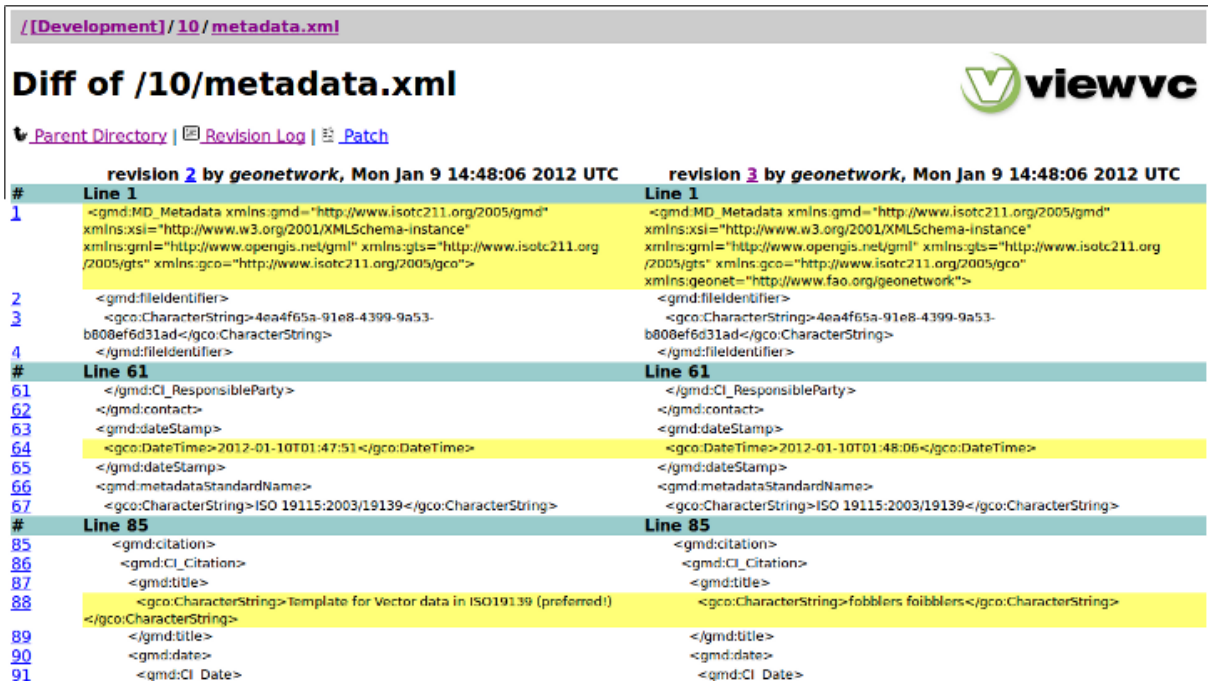
```
svn diff -r 2:3
Index: 10/metadata.xml
===================================================================
--- 10/metadata.xml     (revision 2)
+++ 10/metadata.xml     (revision 3)
@@ -61,7 +61,7 @@
     </gmd:CI_ResponsibleParty>
   </gmd:contact>
   <gmd:dateStamp>
-    <gco:DateTime>2012-01-10T01:47:51</gco:DateTime>
+    <gco:DateTime>2012-01-10T01:48:06</gco:DateTime>
   </gmd:dateStamp>
   <gmd:metadataStandardName>
     <gco:CharacterString>ISO 19115:2003/19139</gco:CharacterString>
@@ -85,7 +85,7 @@
       <gmd:citation>
         <gmd:CI_Citation>
           <gmd:title>
-            <gco:CharacterString>Template for Vector data in ISO19139 (preferr
ed!)</gco:CharacterString>
+            <gco:CharacterString>fobblers foibblers</gco:CharacterString>
           </gmd:title>
           <gmd:date>
```

```
<gmd:CI_Date>
```

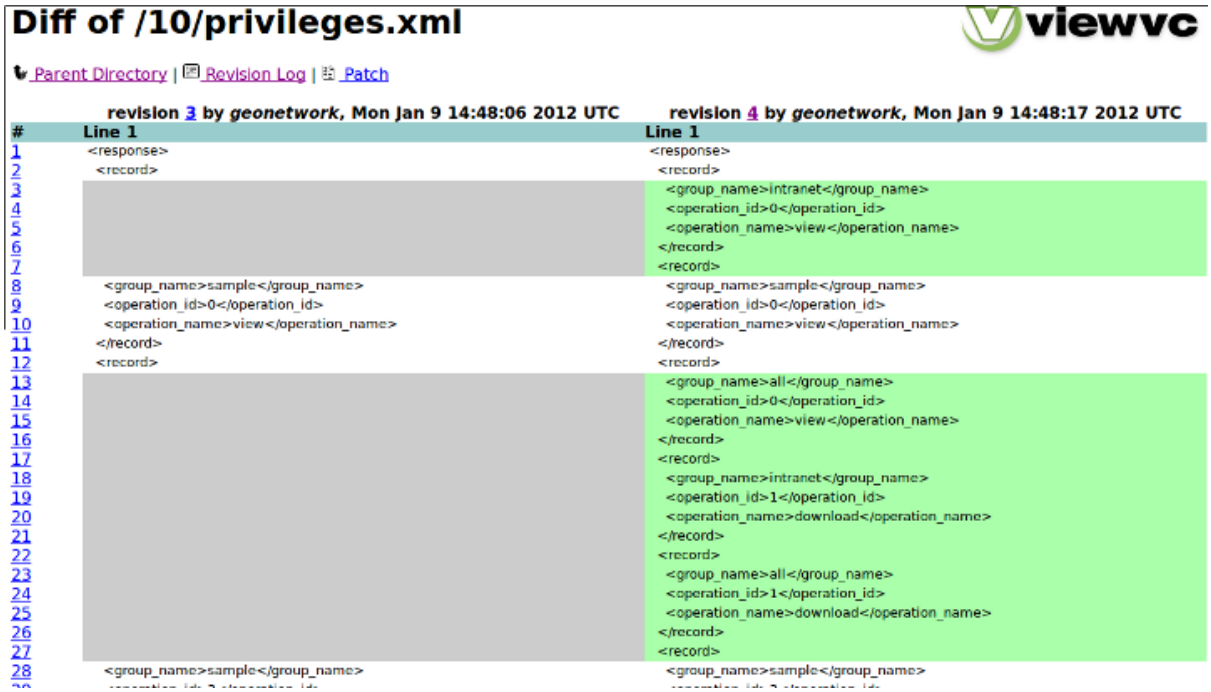
This example shows that the editor has made a change to the title and the dateStamp.

4.6.4 Looking at the revision history using viewvc - a graphical user interface

The viewvc subversion repository tool has a graphical interface that allows side-by-side comparison of changes/differences between files:



Looking at the changes in a metadata record using browser to query viewvc



Looking at the changes in the privileges set on a metadata record using browser to query viewvc

4.6.5 XLink support

Metadata fragments (from directories local to GeoNetwork or from external URLs on the internet) can be linked into metadata records to support reuse. A record is said to be resolved when all available fragments have been copied into the record. With regard to XLinks the current implementation:

- supports versioning of resolved records only
- cannot version fragments of metadata held by GeoNetwork
- will not create a new version of a metadata record when a change is made to one of its component fragments. Instead these changes will be picked up next time the record or its properties are changed.

Support for these corner cases may be added in future versions of GeoNetwork.

4.7 Harvesting

There has always been a need to share metadata between GeoNetwork nodes and bring metadata into GeoNetwork from other sources eg. self-describing web services that deliver data and metadata or databases with organisational metadata etc.

Harvesting is the process of collecting metadata from a remote source and storing it locally in GeoNetwork for fast searching via Lucene. This is a periodic process to do, for example, once a week. Harvesting is not a simple import: local and remote metadata are kept aligned.

GeoNetwork is able to harvest from the following sources (for more details see below):

1. Another GeoNetwork node (version 2.1 or above). See *GeoNetwork Harvesting*
2. An old GeoNetwork 2.0 node (deprecated). See *GeoNetwork 2.0 Harvester*
3. A WebDAV server. See *WEBDAV Harvesting*
4. A CSW 2.0.1 or 2.0.2 catalogue server. See *CSW Harvesting*
5. A GeoPortal 9.3.x or 10.x server. See *GeoPortal REST Harvesting*
6. A File system acessible by GeoNetwork. See *Local File System Harvesting*
7. An OAI-PMH server. See *OAIPMH Harvesting*
8. An OGC service using its GetCapabilities document. These include WMS, WFS, WPS and WCS services. See *Harvesting OGC Services*
9. An ArcSDE server. See *Harvesting an ARCSDE Node*
10. A THREDDS catalog. See *THREDDS Harvesting*
11. An OGC WFS using a GetFeature query. See *WFS GetFeature Harvesting*
12. One or more Z3950 servers. See *Z3950 Harvesting*

4.7.1 Mechanism overview

The harvesting mechanism relies on the concept of a *universally unique identifier (UUID)*. This is a special id because it is not only unique locally to the node that generated it but it is globally unique. It is a combination of the network interface MAC address, the current date/time and a random number. Every time you create a new metadata record in GeoNetwork, a new UUID is generated and assigned to it.

Another important concept behind the harvesting is the *last change date*. Every time you change a metadata record, the last change date is updated. Just storing this parameter and comparing it with a new one allows any system to find out if the metadata record has been modified since last update.

These two concepts allow GeoNetwork to fetch remote metadata, check if it has been updated and remove it locally if it has been removed remotely. Furthermore, thanks to UUIDs, a hierarchy of harvesting nodes can be built where B harvests from C and A harvests from B.

4.7.2 Harvesting life cycle

When a harvester is first set up, there is no harvested metadata. During the first run, all remote matching metadata are retrieved and stored locally. For some harvesters, after the first run, only metadata that has changed will be retrieved.

Harvested metadata are (by default) not editable for the following reasons:

1. The harvesting is periodic so any local change to harvested metadata will be lost during the next run.
2. The change date may be used to keep track of changes so if the metadata gets changed, the harvesting mechanism may be compromised.

Metadata properties (like categories, privileges etc...) on harvested metadata records cannot be changed.

Note: if you really want to edit harvested metadata records and aren't worried by the possible issues described above, there is now a configuration setting which will permit this. See *Harvesting* for more details.

The harvesting process goes on until one of the following situations arises:

1. An administrator stops (deactivates) the harvester.
2. An exception arises. In this case the harvester is automatically stopped.

When a harvester is removed, all metadata records associated with that harvester are removed.

4.7.3 Multiple harvesting and hierarchies

Catalogues that use UUIDs to identify metadata records (eg. GeoNetwork) can be harvested several times without having to take care about metadata overlap.

As an example, consider the GeoNetwork harvesting type which allows one GeoNetwork node to harvest metadata records from another GeoNetwork node and the following scenario:

1. Node (A) has created metadata (a)
2. Node (B) harvests (a) from (A)

3. Node (C) harvests (a) from (B)
4. Node (D) harvests from both (A), (B) and (C)

In this scenario, Node (D) will get the same metadata (a) from all 3 nodes (A), (B), (C). The metadata will flow to (D) following 3 different paths but thanks to its UUID only one copy will be stored. When (a) is changed in (A), a new version will flow to (D) but, thanks to the change date, the copy in (D) will be updated with the most recent version.

4.7.4 Harvesting Fragments of Metadata to support re-use

All the harvesters except for the THREDDS and OGC WFS GetFeature harvester create a complete metadata record that is inserted into or replaces an existing record in the catalog. However, it's often the case that:

- the metadata harvested from an external source is really only one or more fragments of the metadata required to describe a resource such as a dataset
- you might want to combine harvested fragments of metadata with manually entered or static metadata in a single record
- a fragment of metadata harvested from an external source may be required in more than one metadata record

For example, you may only be interested in harvesting the geographic extent and/or contact information from an external source and manually entering or maintaining the remainder of the content in the metadata record. You may also be interested in re-using the contact information for a person or organisation in more than one metadata record.

To support this capability, both the WFS GetFeature Harvester and the THREDDS harvester, allow fragments of metadata to be harvested and linked or copied into a template record to create metadata records. Fragments that are saved into the GeoNetwork database are called subtemplates and can be used in more than one metadata record. These concepts are shown in the diagram below.

As shown above, an example of a metadata fragment is the gmd:contactInfo element of an iso19139 document. This element contains contact details for an individual or an organisation. If a fragment is stored in the geonetwork database as a subtemplate for a given person or organisation, then this fragment can be referenced in metadata records where this organisation or individual is specified using an XML linking mechanism called XLink. An example of an XLink is shown in the following diagram.

4.7.5 HTTPS support

Harvesting between GeoNetwork nodes may require the HTTPS protocol. If harvesting from an https GeoNetwork URL, the server will need to have a trusted certificate available in a JVM keystore accessible to the GeoNetwork node running the harvest.

If you don't have a trusted certificate in the JVM keystore being used by GeoNetwork, the harvester may issue an exception like this when you try to harvest from the https GeoNetwork:

```
javax.net.ssl.SSLHandshakeException:
   sun.security.validator.ValidatorException: PKIX path building failed:
   sun.security.provider.certpath.SunCertPathBuilderException:
   unable to find valid certification path to requested target

Caused by: sun.security.validator.ValidatorException:
```

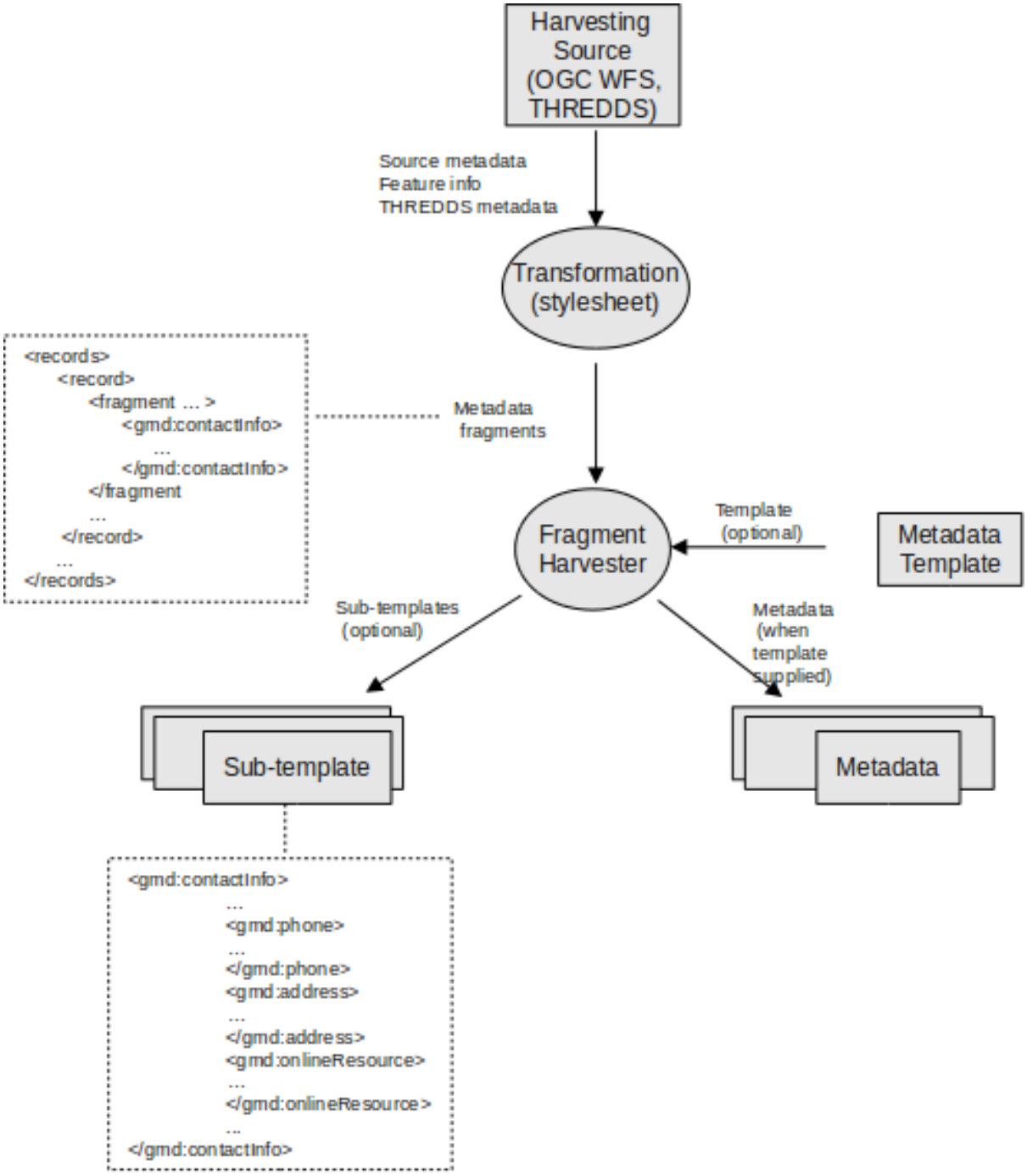
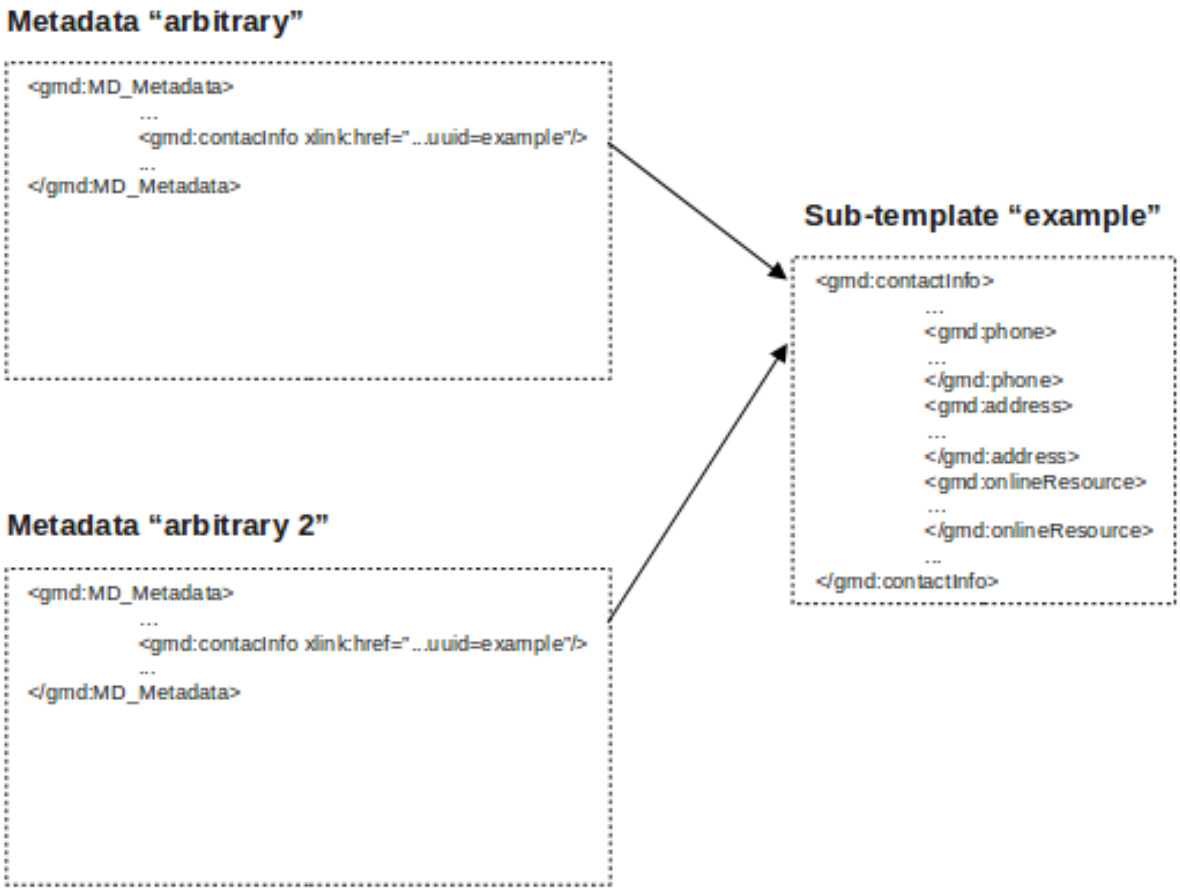


Figure 4.10: *Harvesting Metadata Fragments*




```
   PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderExceptior
   unable to find valid certification path to requested target

Caused by: sun.security.provider.certpath.SunCertPathBuilderException:
   unable to find valid certification path to requested target
```

The server certificate for the GeoNetwork server being harvested needs to be added to the JVM keystore with keytool in order to be trusted.

An alternative way to add the certificate is to use a script like:

```
## JAVA SSL Certificate import script
## Based on original MacOs script by LouiSe@louise.hu : http://louise.hu
##
## Usage: ./ssl_key_import.sh <sitename> <port>
##
## Example: ./ssl_key_import.sh mail.google.com 443 (to read certificate from https://ma

## Compile and start
javac InstallCert.java
java InstallCert $1:$2

## Copy new cert into local JAVA keystore
echo "Please, enter admnistrator password:"
sudo cp jssecacerts $JAVA_HOME/jre/lib/security/jssecacerts
# Comment previous line and uncomment next one for MacOs
#sudo cp jssecacerts /Library/Java/Home/lib/security/
```

To use the script, the Java compiler must be installed and the file InstallCert.java, must be downloaded and placed in the same directory as the script.

The script will add the certificate to the JVM keystore, if you run it as follows::

```
$ ./ssl_key_import.sh https_server_name 443
```

Note: Use this script at your own risk! Before installing a certificate in the JVM keystore as trusted, make sure you understand the security implications.

Note: After adding the certificate you will need to restart GeoNetwork.

4.7.6 The main page

To access the harvesting main page you have to be logged in as an administrator. From the administration page, click the link shown below with a red rectangle.

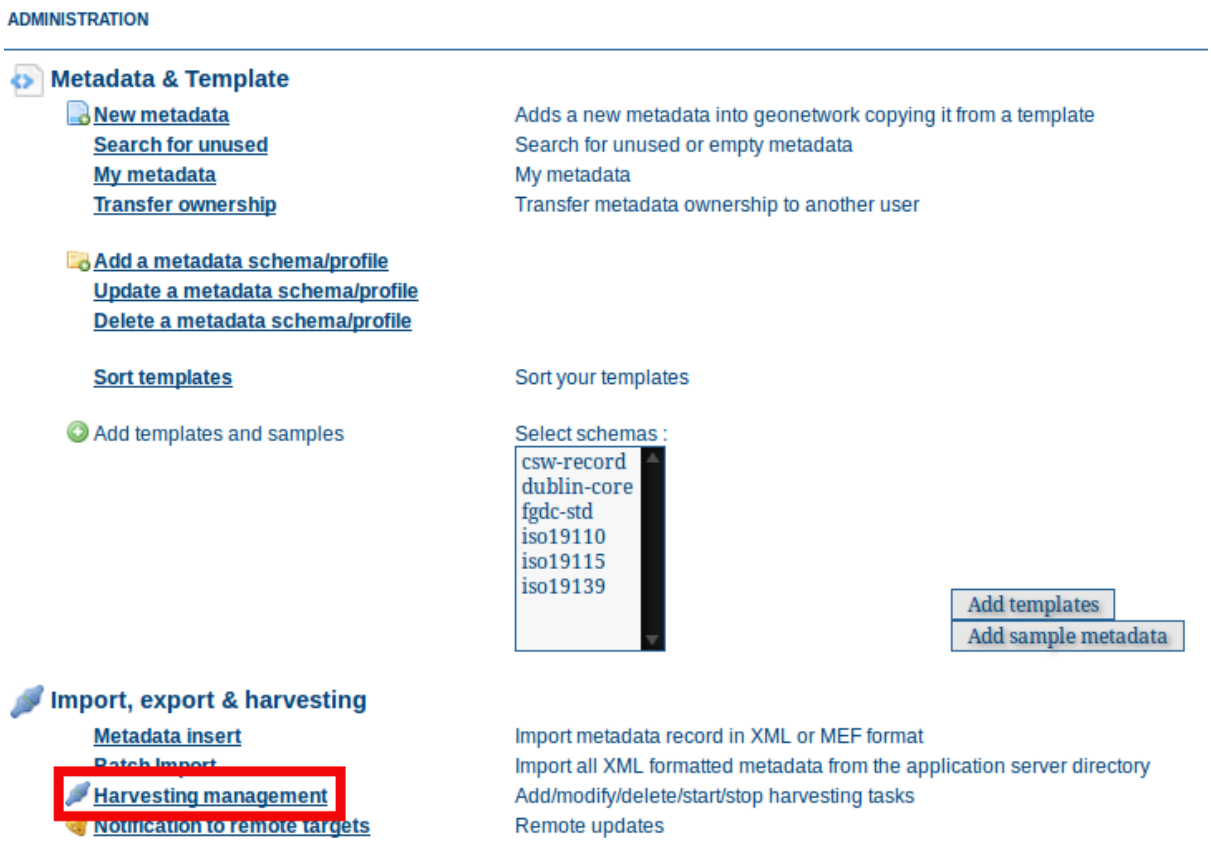


Figure 4.11: *How to access the harvesting main page*

The harvesting main page will then be displayed.

The page shows a list of the currently defined harvesters and a set of buttons for management functions. The meaning of each column in the list of harvesters is as follows:

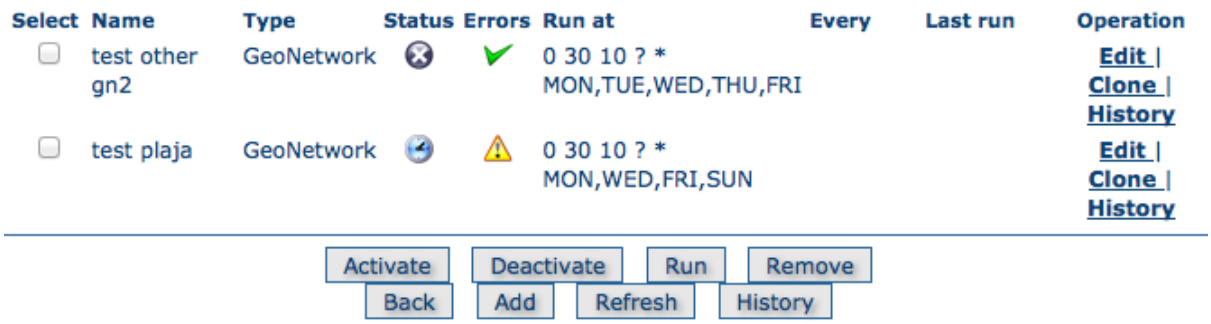


Figure 4.12: *The harvesting main page*

1. *Select* Check box to select one or more harvesters. The selected harvesters will be affected by the first row of buttons (activate, deactivate, run, remove). For example, if you select three harvesters and press the Remove button, they will all be removed.
2. *Name* This is the harvester name provided by the administrator.
3. *Type* The harvester type (eg. GeoNetwork, WebDAV etc...).
4. *Status* An icon showing current status. See *Harvesting Status and Error Icons* for the different icons and status descriptions.
5. *Errors* An icon showing the result of the last harvesting run, which could have succeeded or not. See *Harvesting Status and Error Icons* for the different icons and error descriptions. Hovering the cursor over the icon will show detailed information about the last harvesting run.
6. *Run at* and *Every*: Scheduling of harvester runs. Essentially the time of the day + how many hours between repeats and on which days the harvester will run.
7. *Last run* The date, in ISO 8601 format, of the most recent harvesting run.
8. *Operation* A list of buttons/links to operations on a harvester.

- Selecting *Edit* will allow you to change the parameters for a harvester.
- Selecting *Clone* will allow you to create a clone of this harvester and start editing the details of the clone.
- Selecting *History* will allow you to view/change the harvesting history for a harvester - see *Harvest History*.

At the bottom of the list of harvesters are two rows of buttons. The first row contains buttons that can operate on a selected set of harvesters. You can select the harvesters you want to operate on using the check box in the Select column and then press one of these buttons. When the button finishes its action, the check boxes are cleared. Here is the meaning of each button:

1. *Activate* When a new harvester is created, the status is *inactive*. Use this button to make it *active* and start the harvester(s) according to the schedule it has/they have been configured to use.
2. *Deactivate* Stops the harvester(s). Note: this does not mean that currently running harvest(s) will be stopped. Instead, it means that the harvester(s) will not be scheduled to run again.
3. *Run* Start the selected harvesters immediately. This is useful for testing harvester setups.
4. *Remove* Remove all currently selected harvesters. A dialogue will ask the user to confirm the action.

The second row contains general purpose buttons. Here is the meaning of each button:

1. *Back* Simply returns to the main administration page.
2. *Add* This button creates a new harvester.
3. *Refresh* Refreshes the current list of harvesters from the server. This can be useful to see if the harvesting list has been altered by someone else or to get the status of any running harvesters.
4. *History* Show the harvesting history of all harvesters. See *Harvest History* for more details.

4.7.7 Harvesting Status and Error Icons

Icon	Status	Description
	Inactive	The harvester is stopped.
	Active	The harvesting engine is waiting for the next scheduled run time of the harvester.
	Running	The harvesting engine is currently running, fetching metadata. When the process is finished, the result of the harvest will be available as an icon in the *Errors* column

Possible status icons

Icon	Description
	The harvesting was OK, no errors were found. In this case, a tool tip will show some harvesting results (like the number of harvested metadata etc...).
	The harvesting was aborted due to an unexpected condition. In this case, a tool tip will show some information about the error.

Possible error icons

4.7.8 Harvesting result tips

When a harvester runs and completes, a tool tip showing detailed information about the harvesting process is shown in the **Errors** column for the harvester. If the harvester succeeded then hovering the cursor over the tool tip will show a table, with some rows labelled as follows:

- **Total** - This is the total number of metadata found remotely. Metadata with the same id are considered as one.
- **Added** - Number of metadata added to the system because they were not present locally.
- **Removed** - Number of metadata that have been removed locally because they are not present in the remote server anymore.
- **Updated** - Number of metadata that are present locally but that needed to be updated because their last change date was different from the remote one.
- **Unchanged** - Local metadata left unchanged. Their remote last change date did not change.
- **Unknown schema** - Number of skipped metadata because their format was not recognised by GeoNetwork.

- **Unretrievable** - Number of metadata that were ready to be retrieved from the remote server but for some reason there was an exception during the data transfer process.
- **Bad Format** - Number of skipped metadata because they did not have a valid XML representation.
- **Does not validate** - Number of metadata which did not validate against their schema. These metadata were harvested with success but skipped due to the validation process. Usually, there is an option to force validation: if you want to harvest these metadata anyway, simply turn/leave it off.
- **Thumbnails/Thumbnails failed** - Number of metadata thumbnail images added/that could not be added due to some failure.
- **Metadata URL attribute used** - Number of layers/featuretypes/coverages that had a metadata URL that could be used to link to a metadata record (OGC Service Harvester only).
- **Services added** - Number of ISO19119 service records created and added to the catalogue (for THREDDS catalog harvesting only).
- **Collections added** - Number of collection dataset records added to the catalogue (for THREDDS catalog harvesting only).
- **Atomics added** - Number of atomic dataset records added to the catalogue (for THREDDS catalog harvesting only).
- **Subtemplates added** - Number of subtemplates (= fragment visible in the catalog) added to the metadata catalog.
- **Subtemplates removed** - Number of subtemplates (= fragment visible in the catalog) removed from the metadata catalog.
- **Fragments w/Unknown schema** - Number of fragments which have an unknown metadata schema.
- **Fragments returned** - Number of fragments returned by the harvester.
- **Fragments matched** - Number of fragments that had identifiers that in the template used by the harvester.
- **Existing datasets** - Number of metadata records for datasets that existed when the THREDDS harvester was run.
- **Records built** - Number of records built by the harvester from the template and fragments.
- **Could not insert** - Number of records that the harvester could not insert into the catalog (usually because the record was already present eg. in the Z3950 harvester this can occur if the same record is harvested from different servers).

Result vs harvesting type	GeoNetwork	WebDAV	CSW	OAI-PMH	OGC Service	OGC WFS Features	THREDDS	Z3950 Server(s)	GeoPortal REST
Total	✔	✔	✔	✔	✔	✔	✔	✔	✔
Added	✔	✔	✔	✔	✔			✔	✔
Removed	✔	✔	✔	✔	✔			✔	✔
Updated	✔	✔	✔	✔				✔	✔
Unchanged	✔	✔	✔	✔				✔	✔
Unknown schema	✔	✔	✔	✔	✔			✔	✔
Unretrievable	✔	✔	✔	✔	✔		✔	✔	✔
Bad Format		✔		✔				✔	✔
Does Not Validate		✔		✔		✔		✔	✔
Thumbnails / Thumbnails failed					✔		✔		
Metadata URL attribute used					✔				
Services Added							✔		
Collections Added							✔		
Atomics Added							✔		
Subtemplates Added						✔	✔		
Subtemplates removed						✔	✔		
Fragments w/Unknown Schema						✔	✔		
Fragments Returned						✔	✔		
Fragments Matched						✔	✔		
Existing datasets							✔		
Records Built						✔	✔		

Result information supported by harvesting types

4.7.9 Adding new harvesters

The Add button in the main page allows you to add new harvesters. A drop down list is then shown with all the available harvester protocols.

Figure 4.13: *Adding a new harvester*

You can choose the type of harvest you intend to perform and press *Add* to begin the process of adding the harvester. The supported harvesters and details of what to do next are in the following sections:

GeoNetwork Harvesting

This is the standard and most powerful harvesting protocol used in GeoNetwork. It is able to log in into the remote site, to perform a standard search using the common query fields and to import all matching metadata. Furthermore, the protocol will try to keep both remote privileges and categories of the harvested metadata if they exist locally.

Adding a GeoNetwork harvester

Adding a GeoNetwork Harvester

A description of the options follows:

- **Site** - Information about the GeoNetwork site you wish to harvest from. The **URL** should have the format: `http[s]://server[:port]/geonetwork`. If you want to search privileged metadata you have to specify account credentials that are valid on the remote GeoNetwork site. The **Name** parameter is a short description of the remote site that will be used as the name for this instance of the GeoNetwork harvester.
  - **Set categories if exist locally** - This option allows to maintain category assignments. If the metadata belongs to a category on the remote site and a category with the same name is present on the local site, then the harvested metadata will be assigned to that category if this option is checked on.
  - **Use full MEF format** - If checked, then the remote site will include any thumbnails and data files with the metadata record they are attached too. The option refers to the fact that the MEF file type used to in this case will be the **full** export type.
  - **XSL filter name** - This option will apply a custom XSL filter to the metadata record before it is inserted into the local database. A common use case is to anoymize metadata records using the anonymizer process which remove or rename contact personal information (See the *Processing* section for more information).

SITE

Name: test

URL: http://plaja-hf.hba.marine.csiro.

Set categories if exist locally ☑

Use full MEF format ☑

XSL filter name

Use account ☑

Username: admin

Password: •••••

SEARCH CRITERIA

Criteria

Free text

Title

Abstract

Keywords

Digital ☐

Hardcopy ☐

Source

Add | Retrieve sources

OPTIONS

Run at: 0 : 00 (hour of day : minute eg. 23:15)

Will run again every: 3 hours

☑Monday ☑Tuesday ☑Wednesday ☑Thursday ☑Friday ☑Saturday ☑Sunday

One run only ☐

HARVESTED CONTENT

Validate ☐

PRIVILEGES

Remote group	Copy policy
GUEST	Don't copy
all	Don't copy
sample	Don't copy

Retrieve groups

CATEGORIES

- Applications
- Audio/Video
- Case studies, best practices
- Datasets
- Directories
- Interactive resources
- Maps & graphics
- Other information resources

Back | Save

- **Search criteria** - In this section you can specify search parameters to select metadata records for harvesting. The parameters are the same or similar to those found on the GeoNetwork search form.
  - *source*: A GeoNetwork site can contain both its own metadata and metadata harvested from other sources. Use the **Retrieve sources** button to retrieve the sources from the remote site. You can then choose a source name to constrain the search to a particular source. eg. You could constrain the search to the source representing metadata that has not been harvested from other sites. Leaving **source** blank will retrieve *all* metadata from the remote site.

  You can add multiple search criteria through the **Add** button: multiple searches will be performed and results merged. Search criteria sets an be removed using the small cross button at the top left of the criteria set. If no search criteria are added, a global unconstrained search will be performed.
- **Options** - Scheduling Options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Harvested Content**
  - *Validate* - if checked then harvested metadata records will be validated against the relevant metadata schema. Invalid records will be rejected.
- **Privileges** - Use this section to handle remote group privileges. Press the **Retrieve groups** button and the list of groups on the remote site will be returned. You can then assign a copy policy to each group.
  - The **All** group has a different policy to the other groups:
    1. Copy: Privileges are copied.
    2. Copy to Intranet: Privileges are copied but to the Intranet group. This allows public metadata to be protected.
    3. **Don't copy: Privileges are not copied and harvested metadata will** not be publicly visible.
  - For all other groups the policies are these:
    1. Copy: Privileges are copied only if there is a local group with the same (not localised) name as the remote group.
    2. Create and copy: Privileges are copied. If there is no local group with the same name as the remote group then it is created.
    3. Don't copy: Privileges are not copied.

Note: The Intranet group is not considered because it does not make sense to copy its privileges.

- **Categories**
  - Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories (except where the **Set categories if exist locally** option described above causes the metadata to be assigned to a matching local category).

Notes

- This harvester will not work if the remote site has a version prior to GeoNetwork 2.1 eg. GeoNetwork 2.0.2.
- During harvesting, site icons are harvested and local copies are updated. Icons are propagated to new sites as soon as those sites harvest from this one.
- The metadata record uuid is taken from the info.xml file of the MEF bundle.
- in order to be successfully harvested, metadata records retrieved from the remote site must match a metadata schema in the local GeoNetwork instance

WEBDAV Harvesting

This harvesting type uses the WebDAV (Distributed Authoring and Versioning) protocol or the WAF (web accessible folder) protocol to harvest metadata from a web server. It can be useful to users that want to publish their metadata through a web server that offers a DAV interface. The protocol permits retrieval of the contents of a web page (a list of files) along with the change date.

Adding a WebDAV harvester

- **Site** - Options about the remote site.
  - *Subtype* - Select WebDAV or WAF according to the type of server being harvested.
  - *Name* - This is a short description of the remote site. It will be shown in the harvesting main page as the name for this instance of the WebDAV harvester.
  - *URL* - The remote URL from which metadata will be harvested. Each file found that ends with .xml is assumed to be a metadata record.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing search results.
  - *Use account* - Account credentials for basic HTTP authentication on the WebDAV/WAF server.
- **Options** - Scheduling options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Options** - Specific harvesting options for this harvester.
  - *Validate* - If checked, the metadata will be validated after retrieval. If the validation does not pass, the metadata will be skipped.
  - *Recurse* - When the harvesting engine will find folders, it will recursively descend into them.
- **Privileges** - Assign privileges to harvested metadata.

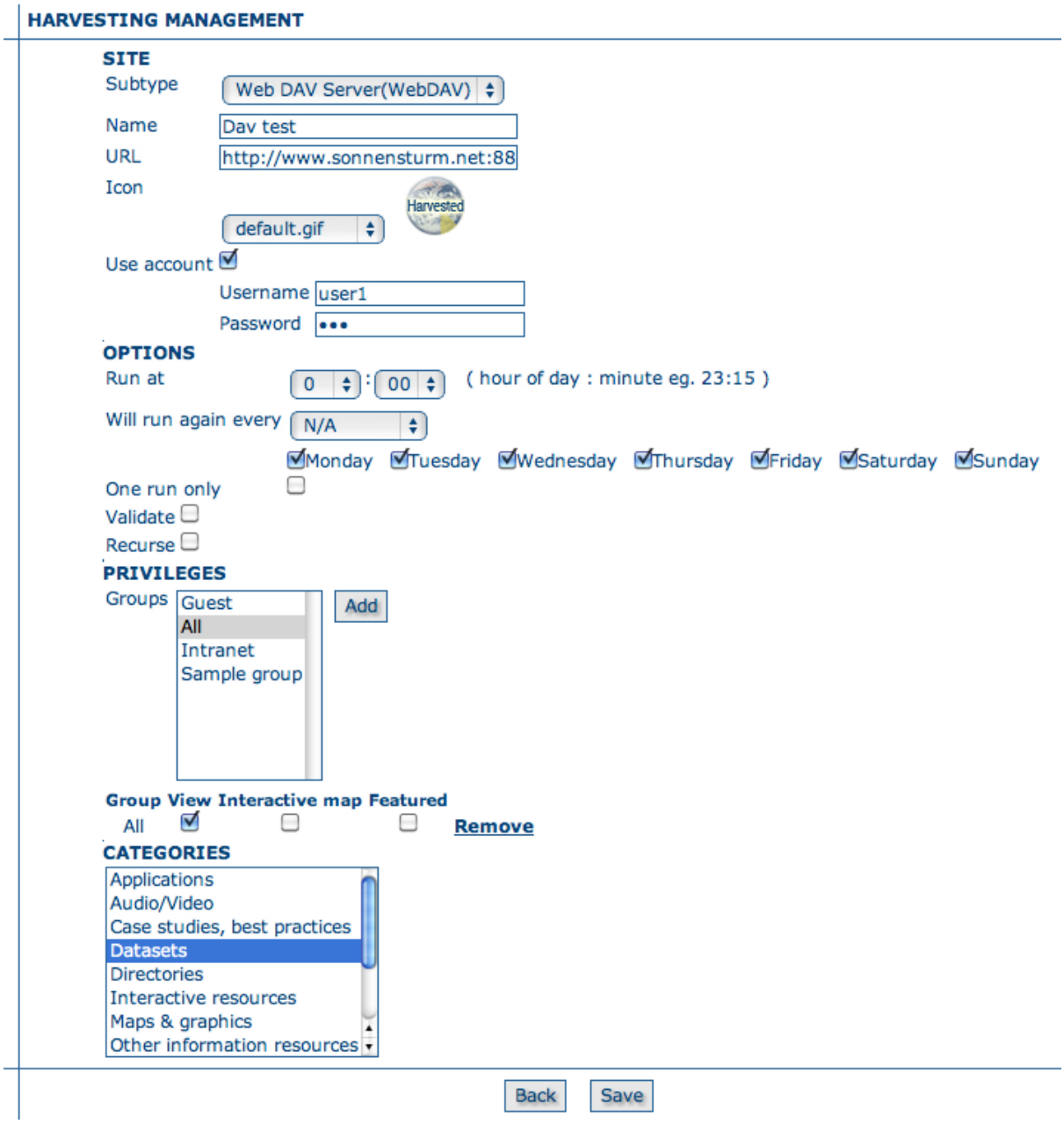


Figure 4.14: *Adding a WebDAV harvester*

- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

Notes

- The same metadata could be harvested several times by different instances of the WebDAV harvester. This is not good practise because copies of the same metadata record will have a different UUID.
- in order to be successfully harvested, metadata records retrieved from the remote site must match a metadata schema in the local GeoNetwork instance

CSW Harvesting

This harvester will connect to a remote CSW server and retrieve metadata records that match the query parameters specified.

Adding a CSW harvester

The figure above shows the options available:

- **Site** - Options about the remote site.
  - *Name* - This is a short description of the remote site. It will be shown in the harvesting main page as the name for this instance of the CSW harvester.
  - *Service URL* - The URL of the capabilities document of the CSW server to be harvested. eg. http://geonetwork-site.com/srv/eng/csw?service=CSW&request=GetCabilities&version=2.0.2. This document is used to discover the location of the services to call to query and retrieve metadata.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing harvested metadata records in the search results.
  - *Use account* - Account credentials for basic HTTP authentication on the CSW server.
- **Search criteria** - Using the Add button, you can add several search criteria. You can query only the fields recognised by the CSW protocol.
- **Options** - Scheduling options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.

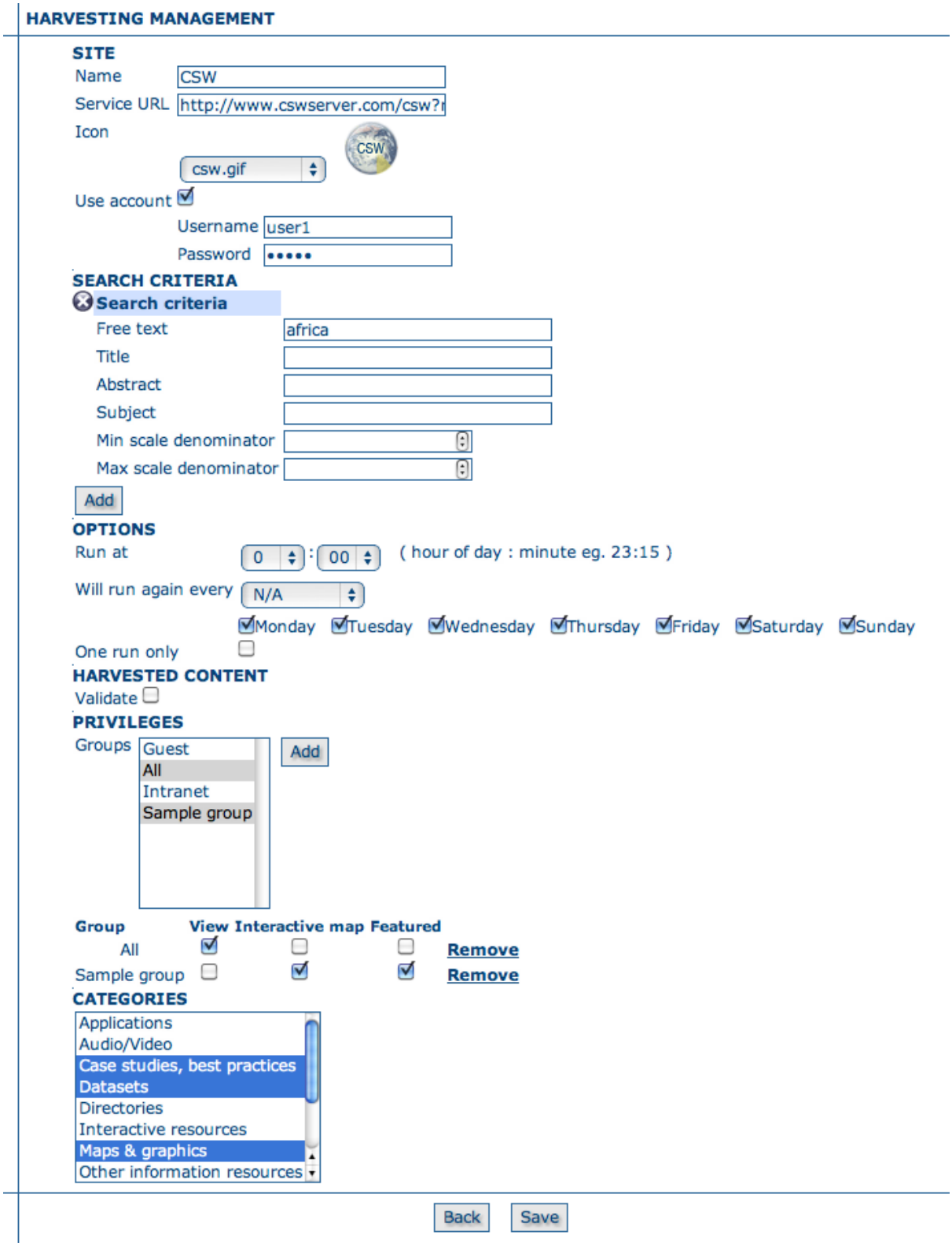


Figure 4.15: *Adding a Catalogue Services for the Web harvesting node*

- **Options** - Specific harvesting options for this harvester.
  - *Validate* - If checked, the metadata will be validated after retrieval. If the validation does not pass, the metadata will be skipped.
- **Privileges** - Assign privileges to harvested metadata.
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

Notes

- in order to be successfully harvested, metadata records retrieved from the remote site must match a metadata schema in the local GeoNetwork instance

GeoPortal REST Harvesting

This harvester will connect to a remote GeoPortal version 9.3.x or 10.x server and retrieve metadata records that match the query parameters specified using the GeoPortal REST API.

Adding a GeoPortal REST harvester

The figure above shows the options available:

- **Site** - Options about the remote site.
  - *Name* - This is a short description of the remote site. It will be shown in the harvesting main page as the name for this instance of the GeoPortal REST harvester.
  - *Base URL* - The base URL of the GeoPortal server to be harvested. eg. http://yourhost.com/geoportal. The harvester will add the additional path required to access the REST services on the GeoPortal server.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing harvested metadata records in the search results.
- **Search criteria** - Using the Add button, you can add several search criteria. You can query any field on the GeoPortal server using the Lucene query syntax described at http://webhelp.esri.com/geoportal_extension/9.3.1/index.htm#srch_lucene.htm.
- **Options** - Scheduling options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.

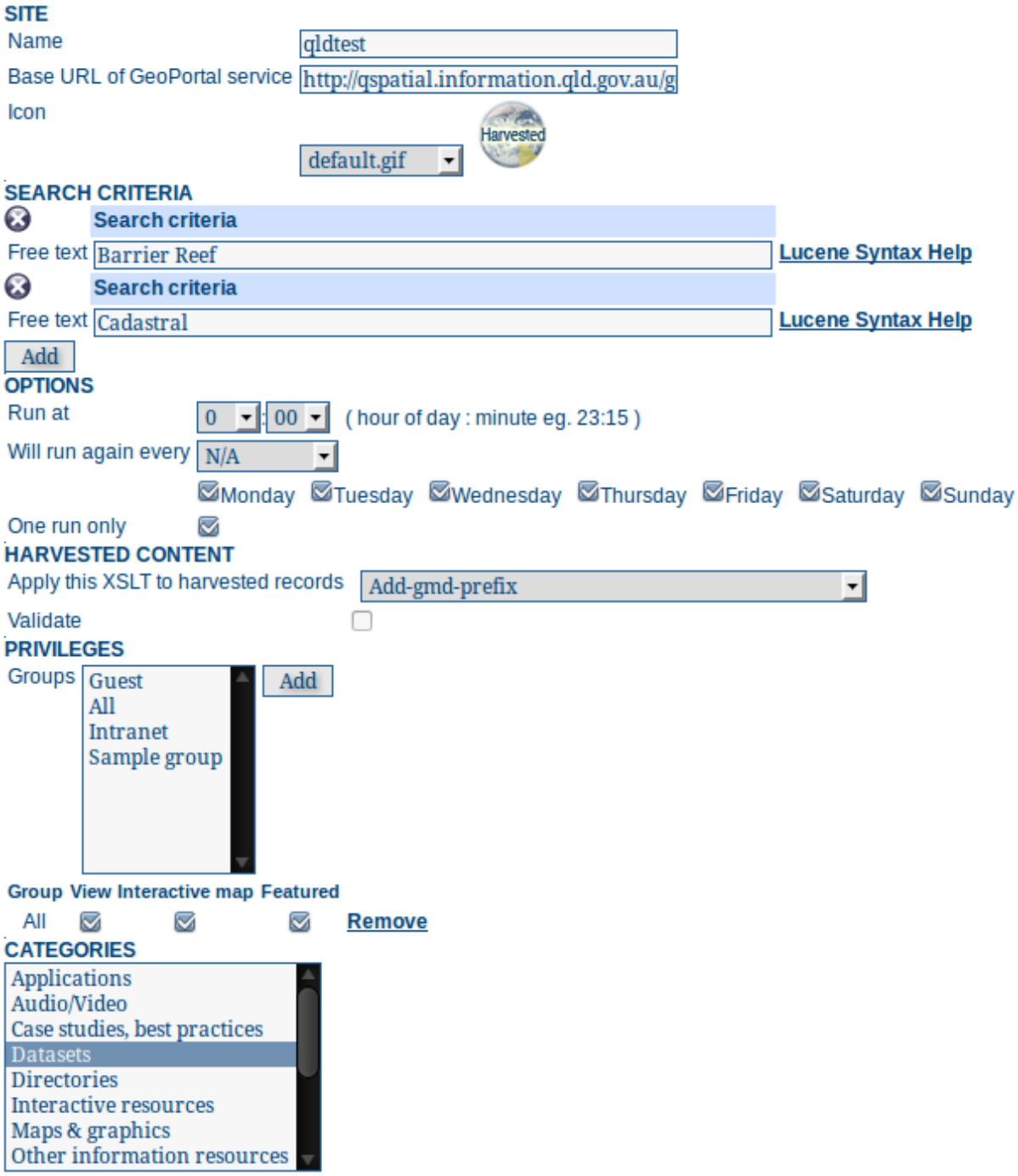


Figure 4.16: *Adding a GeoPortal REST harvester*

- **Harvested Content** - Options that are applied to harvested content.
  - *Apply this XSLT to harvested records* - Choose an XSLT here that will convert harvested records to a different format. See notes section below for typical usage.
  - *Validate* - If checked, the metadata will be validated after retrieval. If the validation does not pass, the metadata will be skipped.
- **Privileges** - Assign privileges to harvested metadata.
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

Notes

- in order to be successfully harvested, metadata records retrieved from the remote site must match a metadata schema in the local GeoNetwork instance
- this harvester uses two REST services from the GeoPortal API:
- `rest/find/document` with searchText parameter to return an RSS listing of metadata records that meet the search criteria (maximum 100000)
- `rest/document` with id parameter from each result returned in the RSS listing
- this harvester has been tested with GeoPortal 9.3.x and 10.x. It can be used in preference to the CSW harvester if there are issues with the handling of the OGC standards etc.
- typically ISO19115 metadata produced by the Geoportal software will not have a 'gmd' prefix for the namespace `http://www.isotc211.org/2005/gmd`. GeoNetwork XSLTs will not have any trouble understanding this metadata but will not be able to map titles and codelists in the viewer/editor. To fix this problem, please select the *Add-gmd-prefix* XSLT for the *Apply this XSLT to harvested records* in the **Harvested Content** set of options described earlier

Local File System Harvesting

This harvester will harvest metadata as XML files from a filesystem available on the machine running the GeoNetwork server.

Adding a Local File System harvester

The figure above shows the options available:

- **Site** - Options about the remote site.
  - *Name* - This is a short description of the filesystem harvester. It will be shown in the harvesting main page as the name for this instance of the Local Filesystem harvester.

SITE
Name test
Base URL of GeoPortal service http://www.ngdc.noaa.gov/geoportal
Icon default.gif Harvested
SEARCH CRITERIA
Search criteria
Free text Charleston Lucene Syntax Help
Add
OPTIONS
Run at 0 : 00 (hour of day : minute eg. 23:15)
Will run again every N/A
Monday Tuesday Wednesday Thursday Friday Saturday Sunday
One run only
HARVESTED CONTENT
Apply this XSLT to harvested records gmiTogmd
Validate
PRIVILEGES
Groups Guest All Intranet Sample group Add
Group View Interactive map Featured
All Remove
CATEGORIES
Applications
Audio/Video
Case studies, best practices
Datasets
Directories
Interactive resources
Maps & graphics
Other information resources
Back Save

Figure 4.17: *GeoPortal REST harvester for a version 10 site*

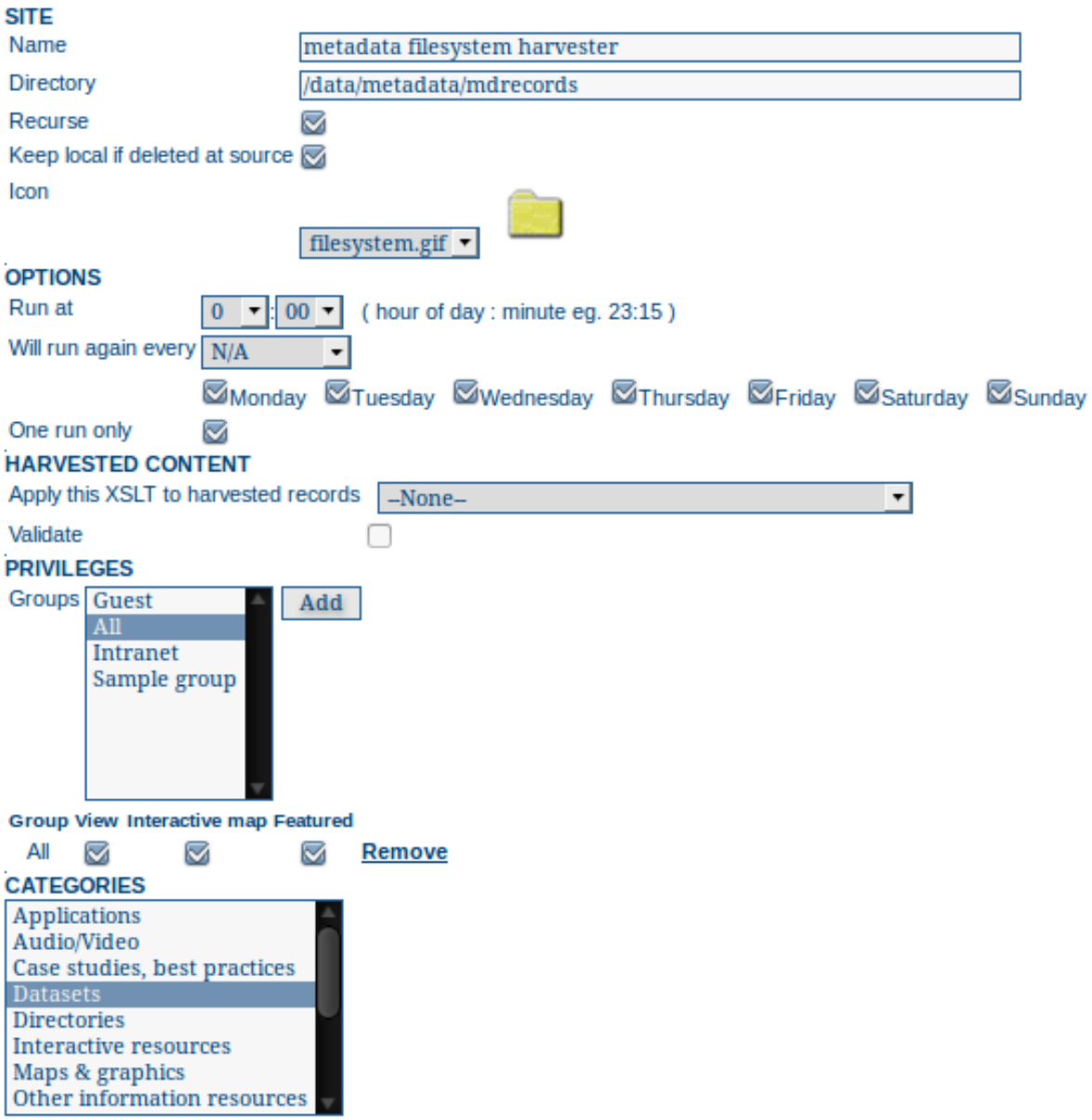


Figure 4.18: *Adding a Local Filesystem harvester*

  - *Directory* - The path name of the directory containing the metadata (as XML files) to be harvested.
  - *Recurse* - If checked and the *Directory* path contains other directories, then the harvester will traverse the entire file system tree in that directory and add all metadata files found.
  - *Keep local if deleted at source* - If checked then metadata records that have already been harvested will be kept even if they have been deleted from the *Directory* specified.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing harvested metadata records in the search results.
- **Options** - Scheduling options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Harvested Content** - Options that are applied to harvested content.
  - *Apply this XSLT to harvested records* - Choose an XSLT here that will convert harvested records to a different format.
  - *Validate* - If checked, the metadata will be validated after retrieval. If the validation does not pass, the metadata will be skipped.
- **Privileges** - Assign privileges to harvested metadata.
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

Notes

- in order to be successfully harvested, metadata records retrieved from the file system must match a metadata schema in the local GeoNetwork instance

GeoNetwork 2.0 Harvester

GeoNetwork 2.1 introduced a new powerful harvesting engine which is not compatible with GeoNetwork version 2.0 based catalogues. Old 2.0 servers can still harvest from 2.1 servers but harvesting metadata from a v2.0 server requires this harvesting type. Due to the fact that GeoNetwork 2.0 was released more than 5 years ago, this harvesting type is deprecated.

OAIPMH Harvesting

This is a harvesting protocol that is widely used among libraries. GeoNetwork implements version 2.0 of the protocol.

Adding an OAI-PMH harvester

An OAI-PMH server implements a harvesting protocol that GeoNetwork, acting as a client, can use to harvest metadata.

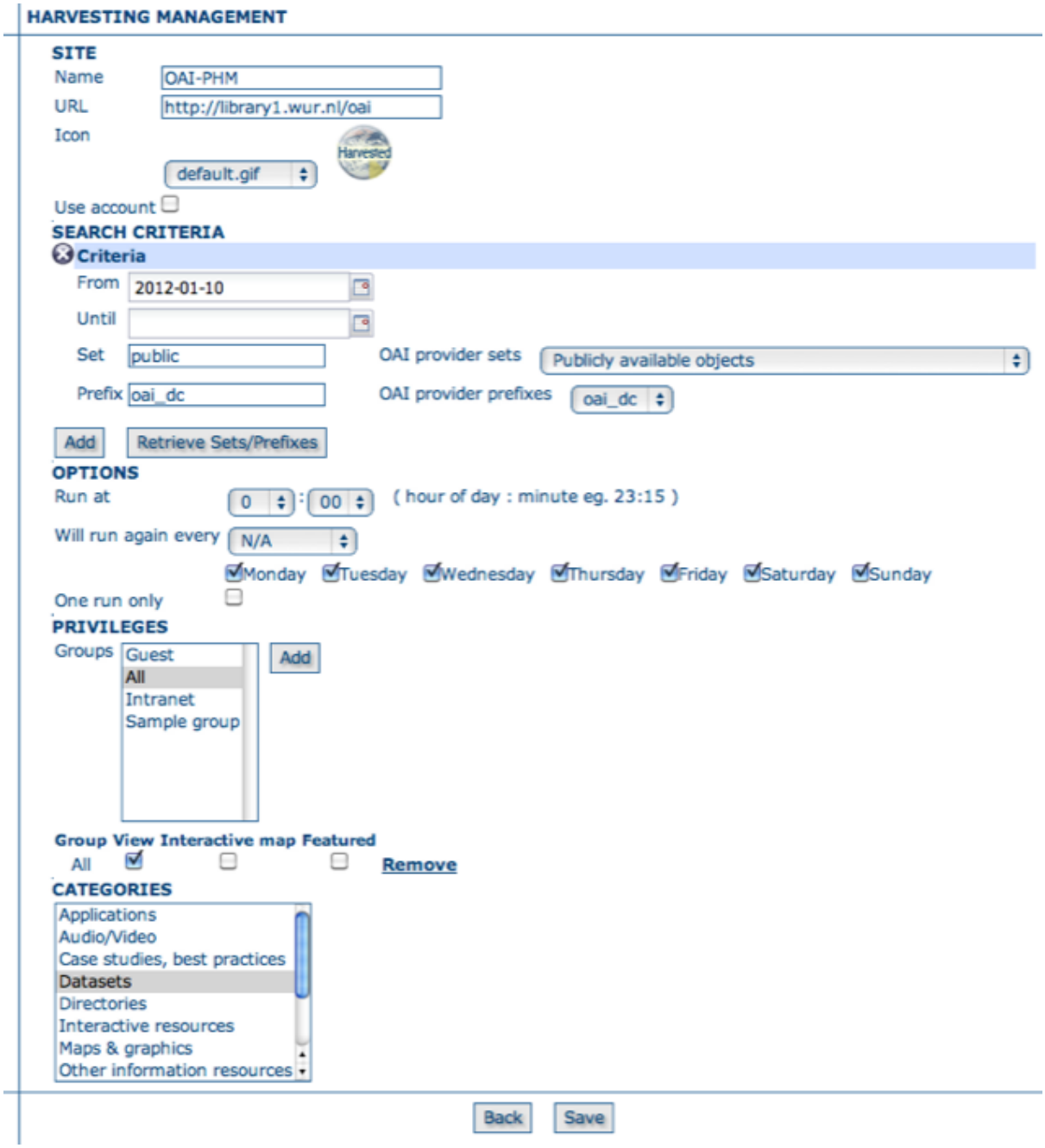


Figure 4.19: *Adding an OAI-PMH harvesting harvester*

Configuration options:

- **Site** - Options describing the remote site.
  - *Name* - This is a short description of the remote site. It will be shown in the harvesting main page as the name for this instance of the OAIPMH harvester.
  - *URL* - The URL of the OAI-PMH server from which metadata will be harvested.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing search results.
  - *Use account* - Account credentials for basic HTTP authentication on the OAIPMH server.
- **Search criteria** - This allows you to select metadata records for harvest based on certain criteria:
  - *From* - You can provide a start date here. Any metadata whose last change date is equal to or greater than this date will be harvested. To add or edit a value for this field you need to use the icon alongside the text box. This field is optional so if you don't provide a start date the constraint is dropped. Use the icon to clear the field.
  - *Until* - Functions in the same way as the *From* parameter but adds an end constraint to the last change date search. Any metadata whose last change data is less than or equal to this data will be harvested.
  - *Set* - An OAI-PMH server classifies metadata into sets (like categories in GeoNetwork). You can request all metadata records that belong to a set (and any of its subsets) by specifying the name of that set here.
  - *Prefix* - 'Prefix' means metadata format. The oai_dc prefix must be supported by all OAI-PMH compliant servers.
  - You can use the Add button to add more than one Search Criteria set. Search Criteria sets can be removed by clicking on the small cross at the top left of the set.

Note: the 'OAI provider sets' drop down next to the *Set* text box and the 'OAI provider prefixes' drop down next to the *Prefix* textbox are initially blank. After specifying the connection URL, you can press the **Retrieve Info** button, which will connect to the remote OAI-PMH server, retrieve all supported sets and prefixes and fill the drop downs with these values. Selecting a value from either of these drop downs will fill the appropriate text box with the selected value.

- **Options** - Scheduling Options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**

- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

Notes

- if you request the oai_dc output format, GeoNetwork will convert it to Dublin Core format.
- when you edit a previously created OAIPMH harvester instance, both the *set* and *prefix* drop down lists will be empty. You have to press the retrieve info button again to connect to the remote server and retrieve set and prefix information.
- the id of the remote server must be a UUID. If not, metadata can be harvested but during hierarchical propagation id clashes could corrupt harvested metadata.
- in order to be successfully harvested, metadata records retrieved from the remote site must match a metadata schema in the local GeoNetwork instance

Harvesting OGC Services

An OGC service implements a GetCapabilities operation that GeoNetwork, acting as a client, can use to produce metadata for the service (ISO19119) and resources delivered by the service (ISO19115/19139). This harvester supports the following OGC services and versions:

- Web Map Service (WMS) - versions 1.0.0, 1.1.1, 1.3.0
- Web Feature Service (WFS) - versions 1.0.0 and 1.1.0
- Web Coverage Service (WCS) - version 1.0.0
- Web Processing Service (WPS) - version 0.4.0 and 1.0.0
- Catalogue Services for the Web (CSW) - version 2.0.2
- Sensor Observation Service (SOS) - version 1.0.0

Adding an OGC Service Harvester

Configuration options:

- **Site**
  - *Name* - The name of the catalogue and will be one of the search criteria.
  - *Type* - The type of OGC service indicates if the harvester has to query for a specific kind of service. Supported type are WMS (1.0.0, 1.1.1, 1.3.0), WFS (1.0.0 and 1.1.0), WCS (1.0.0), WPS (0.4.0 and 1.0.0), CSW (2.0.2) and SOS (1.0.0).
  - *Service URL* - The service URL is the URL of the service to contact (without parameters like "REQUEST=GetCapabilities", "VERSION=", ...). It has to be a valid URL like http://your.preferred.ogcservice/type_wms.
  - *Metadata language* - Required field that will define the language of the metadata. It should be the language used by the OGC web service administrator.
  - *ISO topic category* - Used to populate the topic category element in the metadata. It is recommended to choose one as the topic category is mandatory for the ISO19115/19139 standard if the hierarchical level is "datasets".

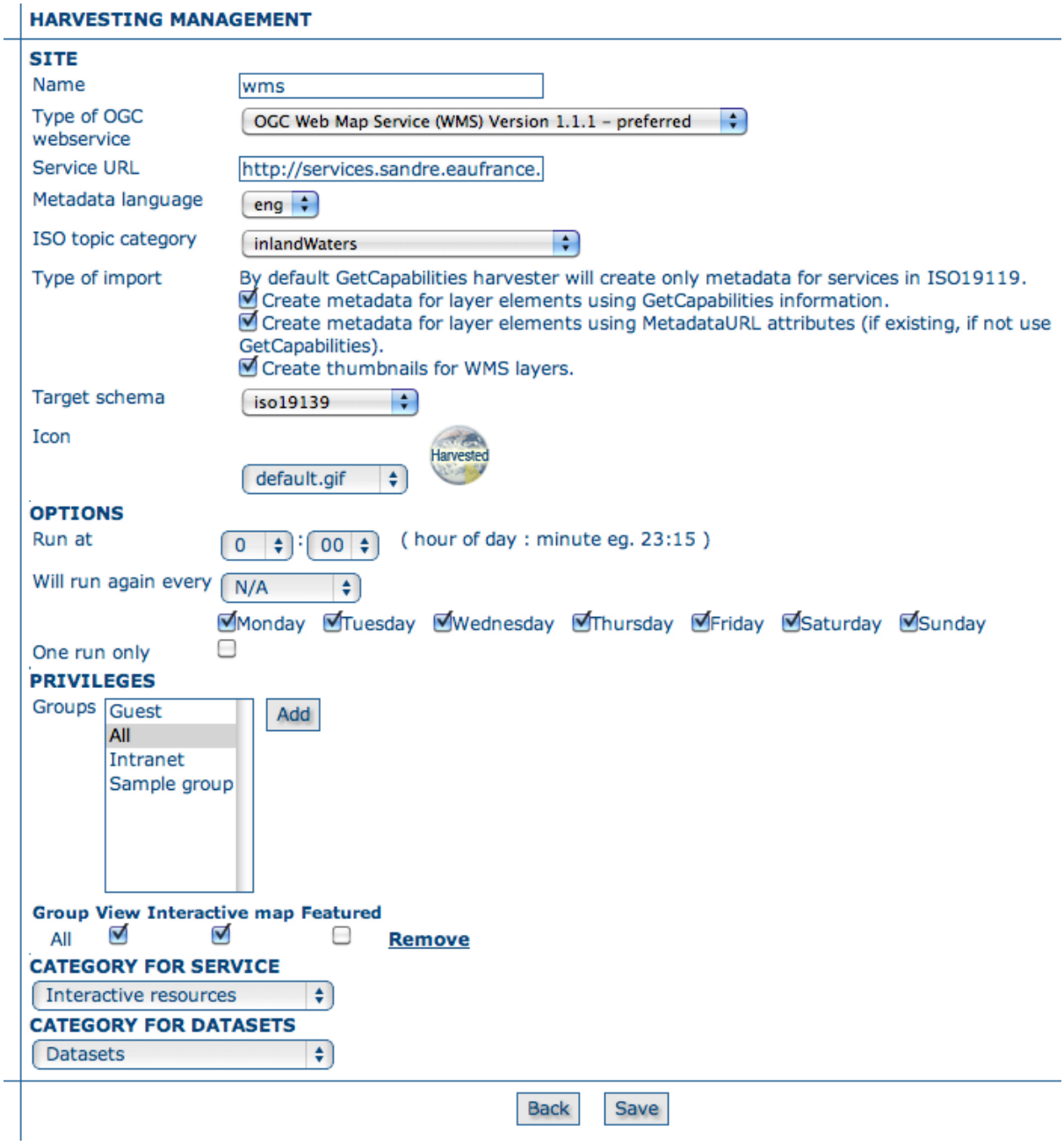


Figure 4.20: *Adding an OGC service harvesting node*

  - *Type of import* - By default, the harvester produces one service metadata record. Check boxes in this group determine the other metadata that will be produced.
    * *Create metadata for layer elements using GetCapabilities information*: Checking this option means that the harvester will loop over datasets served by the service as described in the GetCapabilities document.
    * *Create metadata for layer elements using MetadataURL attributes*: Checkthis option means that the harvester will generate metadata from an XML document referenced in the MetadataUrl attribute of the dataset in the GetCapabilities document. If the document referred to by this attribute is not valid (eg. unknown schema, bad XML format), the GetCapabilites document is used as per the previous option.
    * *Create thumbnails for WMS layers*: If harvesting from an OGC WMS, then checking this options means that thumbnails will be created during harvesting.
  - *Target schema* - The metadata schema of the dataset metadata records that will be created by this harvester.
  - *Icon* - The default icon displayed as attribution logo for metadata created by this harvester.
- **Options** - Scheduling Options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Category for service** - Metadata for the harvested service is assigned to the category selected in this option (eg. "interactive resources").
- **Category for datasets** - Metadata for the harvested datasets is assigned to the category selected in this option (eg. "datasets").

Notes

- every time the harvester runs, it will remove previously harvested records and create new records. GeoNetwork will generate the uuid for all metadata (both service and datasets). The exception to this rule is dataset metadata created using the MetadataUrl tag is in the GetCapabilities document, in that case, the uuid of the remote XML document is used instead
- thumbnails can only be generated when harvesting an OGC Web Map Service (WMS). The WMS should support the WGS84 projection
- the chosen *Target schema* must have the support XSLTs which are used by the harvester to convert the GetCapabilities statement to metadata records from that schema. If in doubt, use iso19139.

Harvesting an ARCSDE Node

This is a harvesting protocol for metadata stored in an ArcSDE installation.

Adding an ArcSDE harvester

Note: Additional installation steps are required to use the ArcSDE harvester because it needs proprietary ESRI java api jars to be installed. See the Developers Manual, Settings for Harvester type arcsde

---

The harvester identifies the ESRI metadata format: ESRI ISO, ESRI FGDC to apply the required xslts to transform metadata to ISO19139

Configuration options:

- **Site**
  - *Name* - This is a short description of the node. It will be shown in the harvesting main page.
  - *Server* - ArcSde server IP address or name
  - *Port* - ArcSde service port (typically 5151)
  - *Username* - Username to connect to ArcSDE server
  - *Password* - Password of the ArcSDE user
  - *Database name* - ArcSDE instance name (typically esri_sde)
- **Options**
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Harvested Content**
  - *Validate* - if checked then harvested metadata records will be validated against the relevant metadata schema. Invalid records will be rejected.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

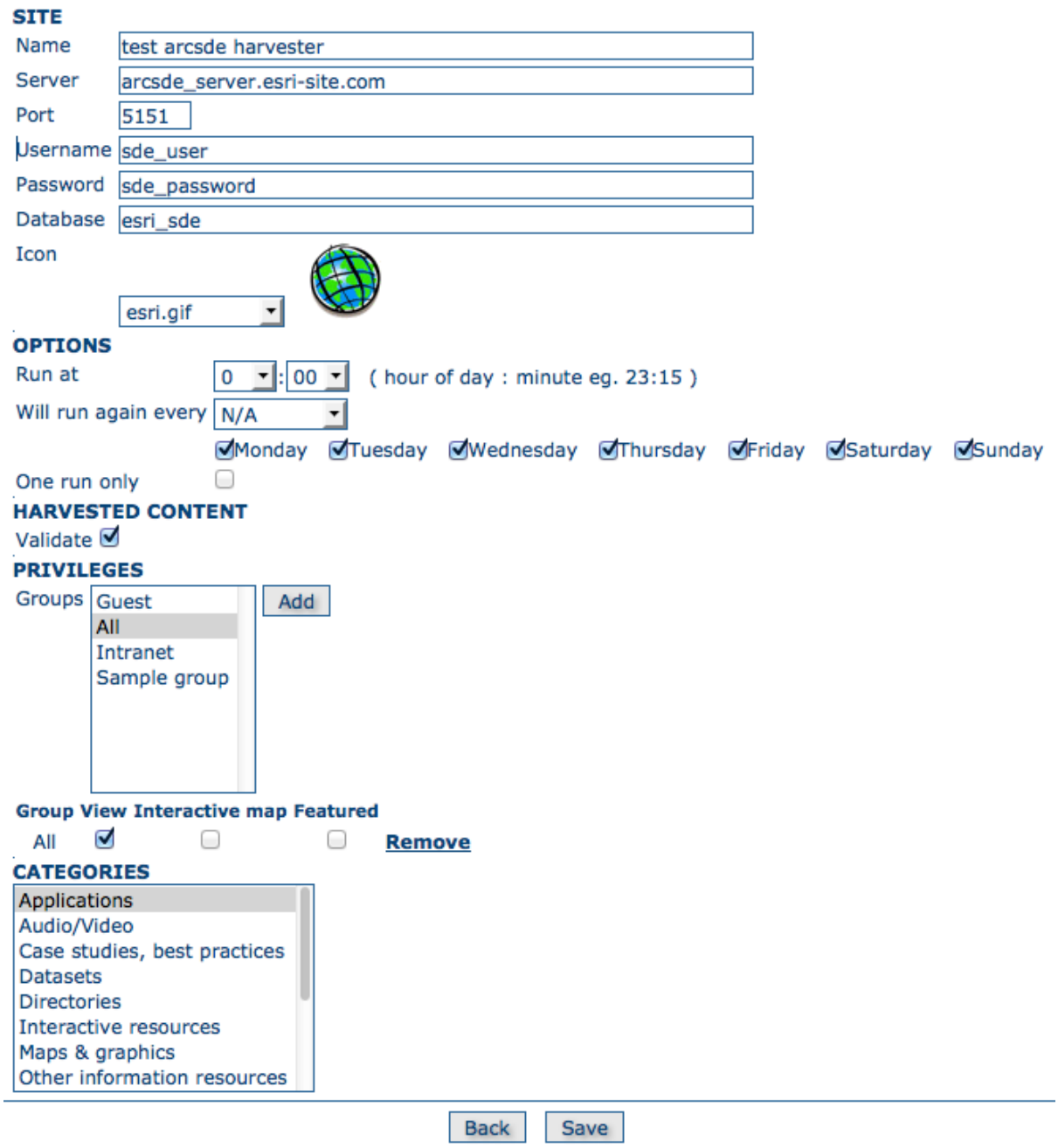


Figure 4.21: *Adding an ArcSDE harvesting node*

THREDDS Harvesting

THREDDS catalogs describe inventories of datasets. They are organised in a hierarchical manner, listing descriptive information and access methods for each dataset. They typically catalog netCDF datasets but are not restricted to these types of files. This harvesting type crawls through a THREDDS catalog harvesting metadata for datasets and services described in it or in referenced netCDF datasets. This harvesting type can extract fragments of metadata from the THREDDS catalog, allowing the user to link or copy these fragments into a template to create metadata records.

Adding a THREDDS Catalog Harvester

The available options are:

- **Site**
  - *Name* - This is a short description of the THREDDS catalog. It will be shown in the harvesting main page as the name of this THREDDS harvester instance.
  - *Catalog URL* - The remote URL of the THREDDS Catalog from which metadata will be harvested. This must be the xml version of the catalog (i.e. ending with .xml). The harvester will crawl through all datasets and services defined in this catalog creating metadata for them as specified by the options described further below.
  - *Metadata language* - Use this option to specify the language of the metadata to be harvested.
  - *ISO topic category* - Use this option to specify the ISO topic category of service metadata.
  - *Create ISO19119 metadata for all services in catalog* - Select this option to generate iso19119 metadata for services defined in the THREDDS catalog (eg. OpenDAP, OGC WCS, ftp) and for the THREDDS catalog itself.
  - *Create metadata for Collection datasets* - Select this option to generate metadata for each collection dataset (THREDDS dataset containing other datasets). Creation of metadata can be customised using options that are displayed when this option is selected as described further below.
  - *Create metadata for Atomic datasets* - Select this option to generate metadata for each atomic dataset (THREDDS dataset not containing other datasets – for example cataloguing a netCDF dataset). Creation of metadata can be customised using options that are displayed when this option is selected as described further below.
    - *Ignore harvesting attribute* - Select this option to harvest metadata for selected datasets regardless of the harvest attribute for the dataset in the THREDDS catalog. If this option is not selected, metadata will only be created for datasets that have a harvest attribute set to true.
    - *Extract DIF metadata elements and create ISO metadata* - Select this option to generate ISO metadata for datasets in the THREDDS catalog that have DIF metadata elements. When this option is selected a list of schemas is shown that have a DIFToISO.xsl stylesheet available (see for example `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/DIFToISO.x` Metadata is generated by reading the DIF metadata items in the THREDDS into a DIF format metadata record and then converting that DIF record to ISO using the DIFToISO stylesheet.

SITE

Name: test thredds

Catalog URL: http://motherlode.ucar.edu:808

Metadata language: eng

ISO topic category: environment

☑ Create ISO19119 metadata for all services in the catalog

☑ Create metadata for Collection Datasets

- ☑ Ignore harvesting attribute
- ○ Extract DIF metadata elements and create ISO metadata
- ◉ Extract Unidata dataset discovery metadata using fragments
  - Select schema for output metadata records: iso19139
    - Stylesheet to create metadata fragments: (iso19139) thredds-metadata
    - ☑ Create subtemplates for fragments and XLink them into template
    - Select template to combine with fragments: (iso19139) HARVESTING TEMPLATE - THREDDS - UNIDATA DISCOVERY

☑ Create metadata for Atomic Datasets

- ☑ Ignore harvesting attribute
- ○ Extract DIF metadata elements and create ISO metadata
- ◉ Extract Unidata dataset discovery metadata using fragments
  - Select schema for output metadata records: iso19139
    - Stylesheet to create metadata fragments: (iso19139) thredds-metadata
    - ☑ Create subtemplates for fragments and XLink them into template
    - Select template to combine with fragments: (iso19139) HARVESTING TEMPLATE - THREDDS - UNIDATA DISCOVERY
- ☑ Harvest new or modified datasets only

☑ Create thumbnails for any datasets delivered by WMS

Icon: thredds.gif

OPTIONS

Run at: 0 : 00 (hour of day : minute eg. 23:15)

Will run again every: N/A

☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday ☑ Saturday ☑ Sunday

One run only: ☐

PRIVILEGES

Groups: Guest, All, Intranet, Sample group — Add

Group	View	Interactive map	Featured	
All	☑	☐	☐	Remove

CATEGORY FOR SERVICE

Interactive resources

CATEGORY FOR DATASETS

Datasets

Back Save

Figure 4.22: *Adding a THREDDS catalog harvester*

    * *Extract Unidata dataset discovery metadata using fragments* - Select this option when the metadata in your THREDDS or netCDF/ncml datasets follows Unidata dataset discovery conventions (see http://www.unidata.ucar.edu/software/netcdf-java/formats/DataDiscoveryAttConvention.html). You will need to write your own stylesheets to extract this metadata as fragments and define a template to combine with the fragments. When this option is selected the following additional options will be shown:

      · *Select schema for output metadata records* - choose the ISO metadata schema or profile for the harvested metadata records. Note: only the schemas that have THREDDS fragment stylesheets will be displayed in the list (see the next option for the location of these stylesheets).

      · *Stylesheet to create metadata fragments* - Select a stylesheet to use to convert metadata for the dataset (THREDDS metadata and netCDF ncml where applicable) into metadata fragments. These stylesheets can be found in the directory convert/ThreddsToFragments in the schema directory eg. for iso19139 this would be `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/Thredds`

      · *Create subtemplates for fragments and XLink them into template* - Select this option to create a subtemplate (=metadata fragment stored in GeoNetwork catalog) for each metadata fragment generated.

      · *Template to combine with fragments* - Select a template that will be filled in with the metadata fragments generated for each dataset. The generated metadata fragments are used to replace referenced elements in the templates with an xlink to a subtemplate if the *Create subtemplates* option is checked. If *Create subtemplates* is not checked, then the fragments are simply copied into the template metadata record.

      · For Atomic Datasets , one additional option is provided *Harvest new or modified datasets only*. If this option is checked only datasets that have been modified or didn't exist when the harvester was last run will be harvested.

  – *Create Thumbnails* - Select this option to create thumbnails for WMS layers in referenced WMS services

  – *Icon* - An icon to assign to harvested metadata. The icon will be used when showing search results.

- **Options** - Scheduling Options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.

- **Category for Service** - Select the category to assign to the ISO19119 service records for the THREDDS services.
- **Category for Datasets** - Select the category to assign the generated metadata records (and any subtemplates) to.

At the bottom of the page there are the following buttons:

- **Back** - Go back to the main harvesting page. The harvesting definition is not added.
- **Save** - Saves this harvester definition creating a new harvesting instance. After the save operation has completed, the main harvesting page will be displayed.

More about harvesting THREDDS DIF metadata elements with the THREDDS Harvester

THREDDS catalogs can include elements from the DIF metadata standard. The Unidata netcdf-java library provides a DIFWriter process that can create a DIF metadata record from these elements. GeoNetwork has a DIFToISO stylesheet to transform these DIF records to ISO. An example of a THREDDS Catalog with DIF-compliant metadata elements is shown below.


```
-<catalog name="Argo floats - bodc" version="1.0.1">
  -<service name="WCS" serviceType="WCS" base="/thredds/wcs/">
     <property name="description" value="Thredds Web Coverage Service"/>
   </service>
  -<service name="thisDODS" serviceType="OPENDAP" base="/thredds/dodsC/">
     <property name="description" value="Thredds OPeNDAP data server"/>
   </service>
  -<dataset name="1900083_bodc" harvest="true" ID="82f5c5dd-f551-4c2b-a66f-4f59dd1b3ea1">
     <documentation type="sensormaker">SeaBird, SeaBird, SeaBird</documentation>
     <documentation type="sensormodel">SBE-41, SBE-41, SBE-41</documentation>
     <documentation type="project">Argo UK</documentation>
     <documentation type="principalinvestigator">Jon Turton</documentation>
    -<documentation type="profiles">
       http://localhost:6666/thredds/catalog/dac/bodc/1900083/profiles/catalog.html
     </documentation>
    -<documentation type="summary">
       Argo Float Dataset: 1900083_bodc Spatial extent: -33.272, 31.792 to -28.867, 35.873 Dates Active: 2001-08-16T10
       (dbar), TEMP (Celsius_degree), PSAL (psu) Principal Investigator: Jon Turton Project: Argo UK Sensor Model: SE
       SeaBird
     </documentation>
    -<geospatialCoverage>
     -<northsouth>
        <start>-33.272</start>
        <size>4.404999999999998</size>
        <units>degrees_north</units>
      </northsouth>
     -<eastwest>
        <start>31.792</start>
        <size>4.080999999999989</size>
        <units>degrees_east</units>
      </eastwest>
     </geospatialCoverage>
    -<timeCoverage>
       <start>2001-08-16 00:00:00Z</start>
       <end>2003-04-28 15:35:54Z</end>
     </timeCoverage>
    -<variables vocabulary="argo" xlink:href="http://www.ifremer.fr/coriolis/cdc/argo/argo-dm-user-manual.pdf">
       <variable name="PRES" vocabulary_name="argo" units="dbar" vocabulary_id="argo"/>
       <variable name="TEMP" vocabulary_name="argo" units="Celsius_degree" vocabulary_id="argo"/>
       <variable name="PSAL" vocabulary_name="argo" units="psu" vocabulary_id="argo"/>
     </variables>
   </dataset>
```


Figure 4.23: *A THREDDS catalog with DIF compliant metadata elements*

More about harvesting Unidata dataset discovery metadata with the THREDDS Harvester

The options described above for the *Extract Unidata dataset discovery metadata using fragments* (see http://www.unidata.ucar.edu/software/netcdf-java/formats/DataDiscoveryAttConvention.html for more details of these conventions) invoke the following process for each collection dataset or atomic dataset in the THREDDS catalog:

1. The harvester bundles up the catalog URI, a generated uuid, the THREDDS metadata for the dataset (generated using the catalog subset web service) and the ncml for netCDF datasets into a single xml document. An example is shown below.
2. This document is then transformed using the specified stylesheet (see *Stylesheet* option above) to obtain a metadata fragments document.
3. The metadata fragment harvester is then called to create subtemplates and/or metadata for the each dataset as requested

```
<root>
      <catalogUri>http://someserver.com/thredds/catalog.xml</catalog>
      <uuid>uuid-generated-for-dataset</uuid>
      <catalog xmlns="http://www.unidata.ucar.edu/namespaces/thredds/InvCatalog/v1.0"
               xmlns:xlink="http://www.w3.org/1999/xlink" version="1.0.1">
            ... subset of catalog containing dataset as the top dataset - refer Unidata catalog ...
            ... specification ...
      </catalog>
      <netcdf xmlns="http://www.unidata.ucar.edu/namespaces/netcdf/ncml-2.2"  location="example1.nc">
            ... ncml generated for netcdf dataset - refer Unidata ncml specification...
            ... atomic datasets only ...
      </netcdf>
</root>
```

Figure 4.24: *An example THREDDS dataset document created by the THREDDS fragment harvester*

Example

DIF Metadata elements on datasets in THREDDS catalogs are not as widely used as metadata elements that follow the Unidata dataset discovery metadata conventions. This example will show how to harvest metadata elements that follow the Unidata data discovery conventions (see http://www.unidata.ucar.edu/software/netcdf-java/formats/DataDiscoveryAttConvention.html).

Two reference stylesheets are provided as examples of how to harvest metadata fragments from a THREDDS catalog. One of these stylesheets, thredds-metadata.xsl, is for generating iso19139 metadata fragments from THREDDS metadata following Unidata dataset discovery conventions. The other stylesheet, netcdf-attributes.xsl, is for generating iso19139 fragments from netCDF datasets following Unidata dataset discovery conventions. These stylesheets are designed for use with the 'HARVESTING TEMPLATE – THREDDS – DATA DISCOVERY' template and can be found in the schema 'convert' directory eg. for ISO19139 this is `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/ThreddsToFragments`.

A sample template 'HARVESTING TEMPLATE – THREDDS – DATA DISCOVERY' has been provided for use with the stylesheets described above for the iso19139 metadata schema. This template is in the schema 'templates' directory eg. for ISO19139, this is `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/templates/thredds-harvester-u`

Before attempting to run this example, you should make sure that this template and others from the iso19139 schema have been loaded into GeoNetwork using the 'Add templates' function in the Administration menu.

We'll now give an example of how to set up a harvester and harvest THREDDS metadata from one of the public unidata motherlode catalogs at http://motherlode.ucar.edu:8080/thredds/catalog/satellite/3.9/WEST-CONUS_4km/catalog.xml. If you were to paste this URL into your browser, you would see the XML representation of this THREDDS catalog. This is the document that is read and converted into metadata by the THREDDS harvester. A snippet of this catalog is shown below.

In GeoNetwork, go into the Administration menu, choose Harvesting Management as described earlier. Add a THREDDS Catalog harvester. Fill out the harvesting management form as shown in the form below.

The first thing to notice is that the *Service URL* should be http://motherlode.ucar.edu:8080/thredds/catalog/satellite/3.9/WEST-CONUS_4km/catalog.xml. Make sure that you use the xml version of the catalog. If you use an html version, you will not be able to harvest any metadata.

Now because this unidata motherload THREDDS catalog has lots of file level datasets (many thousands in fact), we will only harvest collection metadata. To do this you should check *Create metadata for Collection Datasets* and ignore the atomic datasets.

Next, because the metadata in this catalog follows Unidata data discovery conventions, so we will choose *Extract Unidata dataset discovery metadata using fragments*.

Next, we will check *Ignore harvesting attribute*. We do this because datasets in the THREDDS catalog can have an attribute indicating whether the dataset should be harvested or not. Since none of the datasets in this catalog have the harvesting attribute, we will ignore it. If we didn't check this box, all the datasets would be skipped.

Next we will select the metadata schema that the harvested metadata will be written out in. We will choose *iso19139* here because this is the schema for which we have stylesheets that will convert THREDDS metadata to fragments of iso19139 metadata and a template into which these fragments of metadata can be copied or linked. After choosing *iso19139*, choices will appear that show these stylesheets and templates.

The first choice is the stylesheet that will create iso19139 metadata fragments. Because we are interested in the thredds metadata elements in the THREDDS catalog, we will choose the *(iso19139) thredds-metadata* (located in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/ThreddsToFragments`) to convert these elements to iso19139 metadata fragments.

For the purposes of this demonstration, we will *not* check *Create subtemplates for fragments (xlinks...)*. This means that the fragments of metadata created by the stylesheet will be copied directly into the metadata template. They will not be able to be reused (eg. shared between different metadata records). See the earlier section on metadata fragments if you are not sure what this means.

Finally, we will choose *HARVESTING TEMPLATE - THREDDS - UNIDATA DISCOVERY* as the template metadata record that will be combined with the metadata fragments to create the output records. This template will have been loaded into GeoNetwork from `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/templates/thredds-harvester-u` through the Add Templates function in the Administration interface. This template could be filled out with metadata common to all records before the harvester is run. The process by which the template is used to create metadata records is as follows:

```
-<catalog version="1.0.1">
  -<service name="all" serviceType="Compound" base="">
      <service name="ncdods" serviceType="OPENDAP" base="/thredds/dodsC/"/>
      <service name="wcs" serviceType="WCS" base="/thredds/wcs/"/>
      <service name="wms" serviceType="WMS" base="/thredds/wms/"/>
      <service name="ncss" serviceType="NetcdfSubset" base="/thredds/ncss/grid/"/>
      <service name="HTTPServer" serviceType="HTTPServer" base="/thredds
      /fileServer/"/>
      <service name="ncml" serviceType="NCML" base="/thredds/ncml/"/>
      <service name="uddc" serviceType="UDDC" base="/thredds/uddc/"/>
      <service name="iso" serviceType="ISO" base="/thredds/iso/"/>
    </service>
  -<dataset name="WEST-CONUS_4km" ID="SSEC/IDD-Satellite/3.9/WEST-CONUS_4km">
    -<metadata inherited="true">
        <serviceName>all</serviceName>
        <authority>edu.ucar.unidata</authority>
        <dataType>GRID</dataType>
        <dataFormat>GINI</dataFormat>
        <documentation xlink:href="http://www.unidata.ucar.edu
        /data/satellite_data_2.html" xlink:title="Satellite Data Available Through Unidata "/>
        <documentation xlink:href="http://www.nesdis.noaa.gov/satellites.html"
        xlink:title="NESDIS Satellite page"/>
      -<documentation type="summary">
          National Environmental Satellite, Data, And Information Service (NESDIS)
          Geostationary Operational Environmental Satellites (GOES) Data is remapped to
          standard projections.
        </documentation>
        <documentation type="rights">Freely available</documentation>
      -<creator>
          <name>NOAA/NESDIS</name>
          <contact url="http://www.nesdis.noaa.gov/satellites.html"
          email="http://www.nesdis.noaa.gov/contact.html"/>
        </creator>
      -<publisher>
          <name vocabulary="DIF">UCAR/UNIDATA</name>
          <contact url="http://www.unidata.ucar.edu/"
          email="support@unidata.ucar.edu"/>
        </publisher>
      -<timeCoverage>
          <end>present</end>
          <duration>7 days</duration>
        </timeCoverage>
      </metadata>
      <catalogRef xlink:href="current/catalog.xml" xlink:title="current" ID="SSEC/IDD-
      Satellite/3.9/WEST-CONUS_4km/current" name=""/>
      <catalogRef xlink:href="20110216/catalog.xml" xlink:title="20110216"
      ID="SSEC/IDD-Satellite/3.9/WEST-CONUS_4km/20110216" name=""/>
```


Figure 4.25: *Example XML THREDDS catalog*

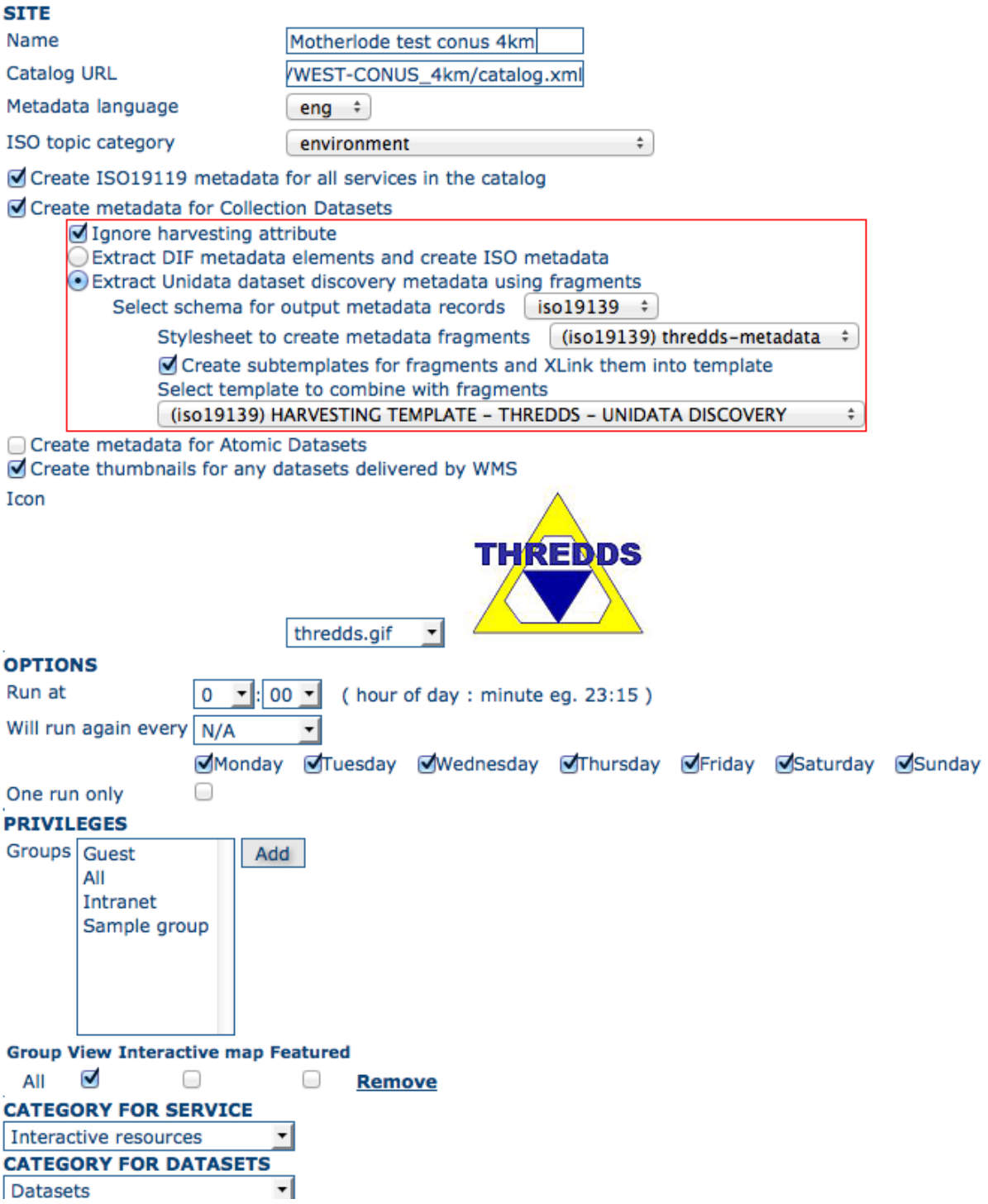


Figure 4.26: *THREDDS harvester form for motherlode THREDDS catalog example*

1. For each dataset in the THREDDS catalog, the template will be copied to create a new iso19139 metadata record
2. Each fragment of metadata harvested from a THREDDS dataset will be copied into the new iso19139 metadata record by matching an identifier in the template with an identifier in the fragment (this match is created by the developer of the template and the stylesheet).
3. The new record is then inserted into the GeoNetwork metadata catalog and the record content is indexed in Lucene for searching.

You can then fill out the remainder of the form according to how often you want the harvested metadata to be updated, what categories will be assigned to the created metadata record, which icon will be displayed with the metadata records in the search results and what the privileges on the created metadata records will be.

Save the harvester screen. Then from the harvesting management screen, check the box beside the newly created harvester, *Activate* it and then *Run* it. After a few moments (depending on your internet connection and machine) you should click on *Refresh*. If your harvest has been successful you should see a results panel appear something like the one shown in the following screenshot.

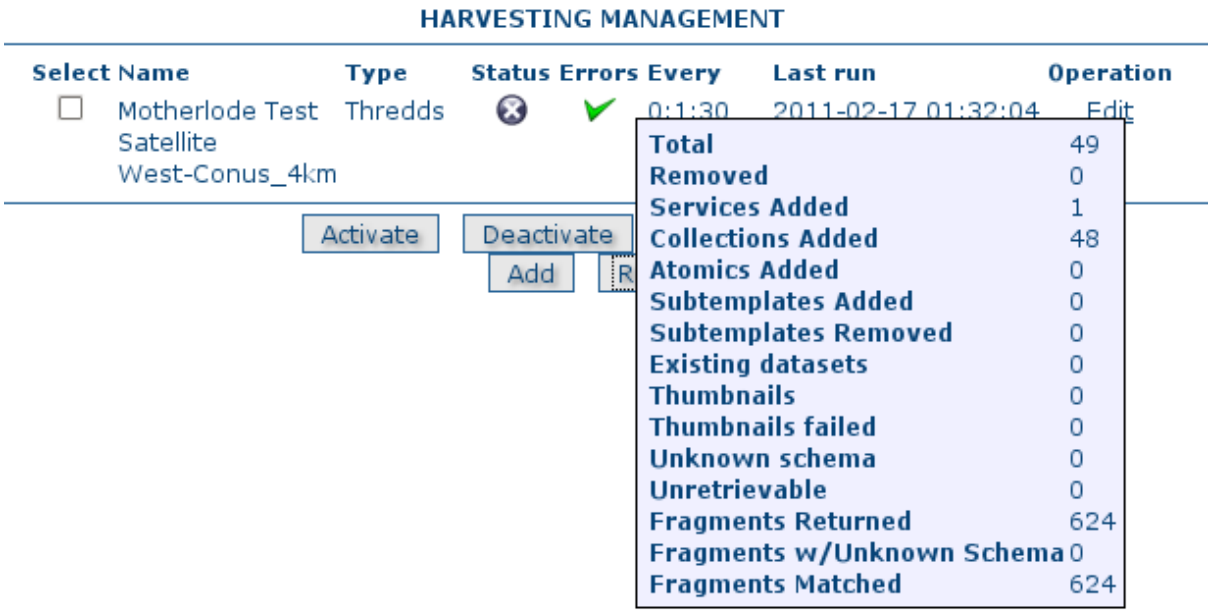


Figure 4.27: *Results of harvesting collection records from a motherlode THREDDS catalog*

Notice that there were 48 metadata records created for the 48 collection level datasets in this THREDDS catalog. Each metadata record was formed by duplicating the metadata template and then copying 13 fragments of metadata into it - hence the total of 624 fragments harvested.

An example of one of the collection level metadata records created by the harvester in this example and rendered by GeoNetwork is shown below.

IDENTIFICATION INFO

Title: SSEC/IDD-Satellite/3.9/WEST-CONUS_4km/current

Cited responsible party

Individual name: UCAR/UNIDATA
OnLine resource: http://www.unidata.ucar.edu/
Role: **Publisher**: Party who published the resource

Cited responsible party

Individual name: NOAA/NESDIS
OnLine resource: http://www.nesdis.noaa.gov/satellites.html
Role: **Originator**: Party who created the resource

Abstract: National Environmental Satellite, Data, And Information Service (NESDIS) Geostationary Operational Environmental Satellites (GOES) Data is remapped to standard projections.
Use constraints: **Other restrictions**: Limitation not listed
Other constraints: Freely available
Language: English
Character set: **UTF8**: 8-bit variable size UCS Transfer Format, based on ISO/IEC 10646
Topic category code: Oceans

Extent

Temporal Extent

Begin date: 2011-02-09T14:32:08Z
End date: 2011-02-16T14:32:08Z

DISTRIBUTION INFORMATION

Download Summary

Show File Download Chooser

Transfer options

Data URL: THREDDS Metadata
Data URL: THREDDS Catalog

METADATA INFO

File identifier: edu.ucar.unidata:SSEC/IDD-Satellite/3.9/WEST-CONUS_4km/current
Metadata language: English
Character set: **UTF8**: 8-bit variable size UCS Transfer Format, based on ISO/IEC 10646
Hierarchy level: **Dataset**: Information applies to the dataset
Date stamp: 2011-02-16T14:32:08.244Z

Figure 4.28: *ISO Metadata record harvested from a motherlode THREDDS catalog*

WFS GetFeature Harvesting

Metadata can be present in the tables of a relational databases, which are commonly used by many organisations. Putting an OGC Web Feature Service (WFS) over a relational database will allow metadata to be extracted via standard query mechanisms. This harvesting type allows the user to specify a GetFeature query and map information from the features to fragments of metadata that can be linked or copied into a template to create metadata records.

Adding an OGC WFS GetFeature Harvester

An OGC web feature service (WFS) implements a GetFeature query operation that returns data in the form of features (usually rows from related tables in a relational database). GeoNetwork, acting as a client, can read the GetFeature response and apply a user-supplied XSLT stylesheet to produce metadata fragments that can be linked or copied into a user-supplied template to build metadata records.

The available options are:

- **Site**
- *Name* - This is a short description of the harvester. It will be shown in the harvesting main page as the name for this WFS GetFeature harvester.
- *Service URL* - The bare URL of the WFS service (no OGC params required)
- *Metadata language* - The language that will be used in the metadata records created by the harvester
- *OGC WFS GetFeature Query* - The OGC WFS GetFeature query used to extract features from the WFS.
- *Schema for output metadata records* - choose the metadata schema or profile for the harvested metadata records. Note: only the schemas that have WFS fragment stylesheets will be displayed in the list (see the next option for the location of these stylesheets).
  - *Stylesheet to create fragments* - User-supplied stylesheet that transforms the GetFeature response to a metadata fragments document (see below for the format of that document). Stylesheets exist in the WFSToFragments directory which is in the convert directory of the selected output schema. eg. for the iso19139 schema, this directory is `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/WFSToFragment`
  - *Save large response to disk* - Check this box if you expect the WFS GetFeature response to be large (eg. greater than 10MB). If checked, the GetFeature response will be saved to disk in a temporary file. Each feature will then be extracted from the temporary file and used to create the fragments and metadata records. If not checked, the response will be held in RAM.
  - *Create subtemplates* - Check this box if you want the harvested metadata fragments to be saved as subtemplates in the metadata catalog and xlink'd into the metadata template (see next option). If not checked, the fragments will be copied into the metadata template.
  - *Template to use to build metadata using fragments* - Choose the metadata template that will be combined with the harvested metadata fragments to create metadata records. This is a standard GeoNetwork metadata template record.
- *Category for records built with linked fragments* - Choose the metadata template that will be combined with the harvested metadata fragments to create metadata records. This is a standard

SITE

Name: deegree22 philosopher example

Service URL: http://localhost:6060/deegree/s

Icon: wfs.gif

Metadata language: eng

OGC WFS GetFeature Query:

```
<wfs:GetFeature service="WFS" version="1.1.0"
      xmlns:wfs="http://www.opengis.net/wfs">
   <wfs:Query typeName="app:Philosopher"/>
</wfs:GetFeature>
```

Schema for output metadata records: iso19139

Stylesheet to create fragments: (iso19139) deegree22_philosopher_fragments

☐ Save large response to disk

☑ Create subtemplates

Select template to combine with fragments: (iso19139) Deegree22 WFS Fragments Philosopher Database T

Category for output metadata records: Datasets

OPTIONS

Run at: 0 : 00 (hour of day : minute eg. 23:15)

Will run again every: N/A

☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday ☑ Saturday ☑ Sunday

One run only ☑

PRIVILEGES

Groups: Guest, All, Intranet, Sample group — Add

Group	View	Interactive map	Featured	
All	☑	☑	☑	Remove

CATEGORY FOR SUBTEMPLATES

Datasets

Figure 4.29: *Adding an OGC WFS GetFeature harvester*

GeoNetwork metadata template record.

- **Options**
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Category for subtemplates** - When fragments are saved to GeoNetwork as subtemplates they will be assigned to the category selected here.

More about turning the GetFeature Response into metadata fragments

The structure of the metadata fragments document that your XSLT (see *Stylesheet used to create fragments* above) must produce from the GetFeature response is shown below.

Within the root <record> element there can be zero to many <record> elements. When generating metadata, each record element will result in the generation of one metadata document, otherwise, the <record> element is used to group metadata fragments as required only (e.g. fragments generated for a dataset or feature).

Within a <record> element there can be zero to many <fragment> elements and zero to many <replacementGroup> elements. A <replacementGroup> element can itself contain zero to many <fragment> elements. Ordering of <fragment> elements and <replacementGroup> elements within a <record> or <replacementGroup> element is not important.

<fragment> elements contain individual xml fragments. The content of the <fragment> can be any xml element from a supported geonetwork schema with the proviso that the element must contain enough relevant metadata to allow the target schema to be identified (i.e. distinguishing namespaces).

<replacementGroup> elements have significance during metadata generation only. They are used to group zero or more fragments for insertion into or creation of links in a copy of the metadata template used to generate the metadata. Where the <replacementGroup> element contains no <fragment> elements, the referenced element in the template copy will be removed, otherwise it will be replaced with the contents of the fragment.

Valid attributes on these elements and their function is as follows:

```
<records>
  <record uuid="metadata-uuid">
    <fragment id="template-id" uuid="sub-template-uuid"
                title="sub-template-title">
                ... metadata fragment - single element ...
                ... from target metadata schema ...
    </fragment>
    <fragment id="template-id" uuid="sub-template-uuid"
                title="sub-template-title">
                ... metadata fragment - single element ...
                ... from target metadata schema ...
    </fragment>
    ...
    <replacementGroup id="template-id">
          ... zero or more fragments as above ...
          ... id attribute not required ...
    </replacementGroup>
    <replacementGroup id="template-id">
          ... zero or more fragments as above ...
          ... id attribute not required ...
    </replacementGroup>
    ...
  </record>
</records>
```

Figure 4.30: *An example metadata fragments document produced by a user-supplied XSLT*

Element	Attribute	Description
Record	Uuid	Uuid of the generated metadata record (optional - one will be assigned by the harvester otherwise)
Fragment	Id	Id of element in metadata template to replace/link from. Ignored when fragment is within a replacementGroup.
	Uuid	Uuid to use for generated subtemplate (used to link to this subtemplate from metadata)
	Title	Title of fragment – used as title of xlink
ReplacementGroup	Id	Id of element in metadata template to delete, replace or link from to contained fragments

Finally, two examples of how to harvest metadata from the Features of an OGC WFS harvester can be given using stylesheets and templates supplied with GeoNetwork.

Bundled GeoServer Boundaries Harvest example

This example assumes that you have installed the bundled GeoServer that comes with GeoNetwork. The end result of this example will be 251 ISO19139 metadata records that link in 1506 fragments (6 per record) created from a GetFeature response on the boundaries shapefile in the GeoServer instance supplied with GeoNetwork. The records created contain metadata about the countries of the world.

The procedure to follow is:

- From the Administration->System Configuration menu, enable the XLink Resolver and *Save* the configuration to the database.
- Add an *OGC WFS GetFeature response* harvester from the Administration->Harvesting menu.
- Give it a *Name* (eg. gboundaries) and enter the URL of the wfs service from the bundled geoserver (eg. http://localhost:8080/geoserver/wfs) in the *Service URL* field.
- We'll use a simple GetFeature query to select all countries from the boundaries shapefile behind the WFS. The XML for such a query (which is to be entered in the *GetFeature Query* textarea) is:

```
<wfs:GetFeature service="WFS" version="1.1.0"
        xmlns:wfs="http://www.opengis.net/wfs">

 <wfs:Query typeName="gboundaries"/>

</wfs:GetFeature>
```

- Choose an output schema - we'll choose *iso19139* as this schema has the example stylesheets and templates we need for this example. Notice that after this option is chosen the following options become visible and we'll take the following actions:
  - Choose the supplied 'geoserver_boundary_fragments' stylesheet to extract fragments from the GetFeature response in the *Stylesheet to use to create fragments* pull-down list. This stylesheet can be found in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/WFSToFragment`
  - Select the supplied 'Geoserver WFS Fragments Country Boundaries Test Template' template from the *Template to use to build metadata using fragments* pull-down list. This template can be found in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/templates/geoserver_f`

- Choose a category for the records created by the harvester, check the *One run only* box, add some privileges (simplest is to let All users have View rights). At this stage your harvester entry form should look like the following screenshot.
- *Save* the harvester entry form.
- You will be returned to the harvester operations menu where you can *Activate* the harvester and then *Run* it.

After the harvester has been run you should see a results screen that looks something like the following screenshot.

WFS GetFeature Harvesting - Results for geoserver boundaries example

The results page shows that there were 1506 fragments of metadata harvested from the WFS GetFeature response. They were saved to the GeoNetwork database as subtemplates and linked into the metadata template to form 251 new metadata records.

Deegree Version 2.x Philosopher Database example

This example assumes that you have downloaded Deegree version 2.x and loaded the Philosopher example database. The end result of this example will be 7 ISO19139 metadata records that link in 42 fragments (6 per record) created from the GetFeature response from your deegree installation. The records contain metadata about 7 famous philosophers.

The procedure to follow is:

- From the Administration->System Configuration menu, enable the XLink Resolver and *Save* the configuration to the database.
- Add an *OGC WFS GetFeature response* harvester from the Administration->Harvesting menu.
- Give it a *Name* (eg. deegree22-philosopher-test) and enter the URL of your deegree 2.2 installation in the *Service URL* field.
- We'll use a simple GetFeature query to select all philosophers from the database under the WFS. The XML for such a query (which is to be entered in the *GetFeature Query* textarea) is:

```
<wfs:GetFeature version="1.1.0" xmlns:app="http://www.deegree.org/app"
           xmlns:wfs="http://www.opengis.net/wfs">

 <!-- request all Philosopher instances -->
 <wfs:Query typeName="app:Philosopher"/>

</wfs:GetFeature>
```

- Choose an output schema - we'll choose *iso19139* as this schema has the example stylesheets and templates we need for this example. Notice that after this option is chosen the following options become visible and we'll take the following actions:
  - Choose the supplied 'deegree2_philosopher_fragments' stylesheet to extract fragments from the GetFeature response in the *Stylesheet to use to create fragments* pull-down list. This stylesheet can be found in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/convert/WFSToFragment`
  - Select the supplied 'Deegree 22 WFS Fragments Philosopher Database Test Template' template from the *Template to use to build meta-*

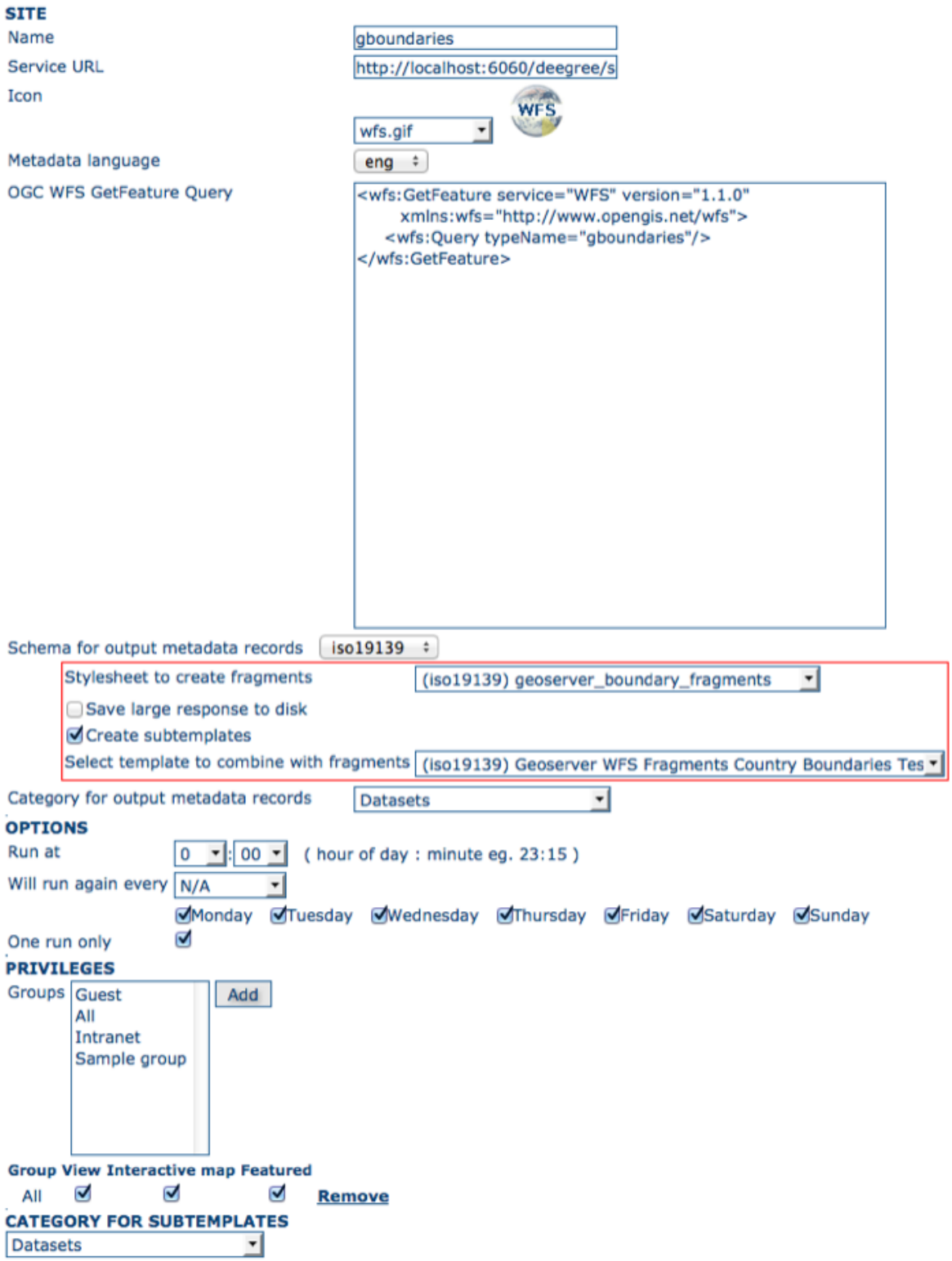
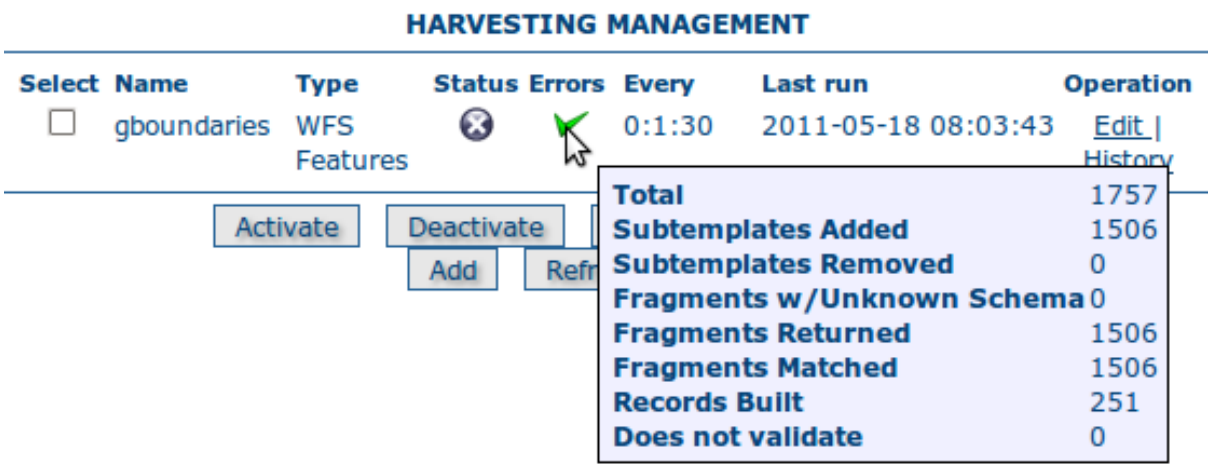


Figure 4.31: *Adding an OGC WFS GetFeature harvester - boundaries example*



data using fragments pull-down list. This template can be found in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/templates/deegree_fra`

- Choose a category for the records created by the harvester, check the *One run only* box, add some privileges (simplest is to let All users have View rights). At this stage your harvester entry form should look like the following screenshot.
- *Save* the harvester entry form.
- You will be returned to the harvester operations menu where you can *Activate* the harvester and then *Run* it.

After the harvester has been run you should see a results screen that looks something like the following screenshot.

WFS GetFeature Harvesting - Results for deegree philosopher database example

The results page shows that there were 42 fragments of metadata harvested from the WFS GetFeature response. They were saved to the GeoNetwork database as subtemplates and linked into the metadata template to form 7 new metadata records.

Z3950 Harvesting

Z3950 is a remote search and harvesting protocol that is commonly used to permit search and harvest of metadata. Although the protocol is often used for library catalogs, significant geospatial metadata catalogs can also be searched using Z3950 (eg. the metadata collections of the Australian Government agencies that participate in the Australian Spatial Data Directory - ASDD). This harvester allows the user to specify a Z3950 query and retrieve metadata records from one or more Z3950 servers.

Adding a Z3950 Harvester

The available options are:

- **Site**
  - *Name* - A short description of this Z3950 harvester. It will be shown in the harvesting main page using this name.

SITE

Name: deegree22 philosopher example

Service URL: http://localhost:6060/deegree/s

Icon: wfs.gif

Metadata language: eng

OGC WFS GetFeature Query:

```
<wfs:GetFeature service="WFS" version="1.1.0"
      xmlns:wfs="http://www.opengis.net/wfs">
   <wfs:Query typeName="app:Philosopher"/>
</wfs:GetFeature>
```

Schema for output metadata records: iso19139

Stylesheet to create fragments: (iso19139) deegree22_philosopher_fragments

☐ Save large response to disk

☑ Create subtemplates

Select template to combine with fragments: (iso19139) Deegree22 WFS Fragments Philosopher Database T

Category for output metadata records: Datasets

OPTIONS

Run at: 0 : 00 (hour of day : minute eg. 23:15)

Will run again every: N/A

☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday ☑ Saturday ☑ Sunday

One run only ☑

PRIVILEGES

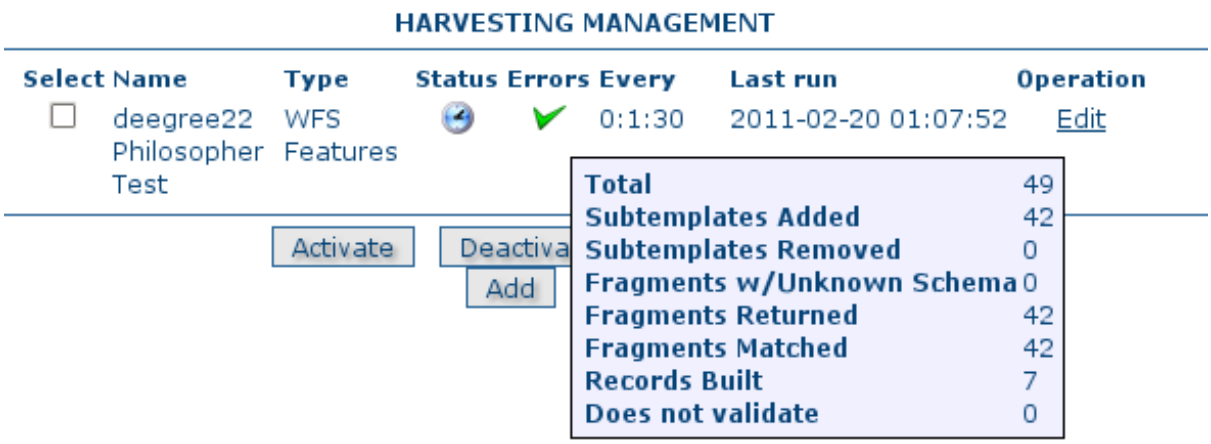
Groups: Guest, All, Intranet, Sample group — Add

Group	View	Interactive map	Featured	
All	☑	☑	☑	Remove

CATEGORY FOR SUBTEMPLATES

Datasets

Figure 4.32: *Adding an OGC WFS GetFeature harvester - philosopher example*



  - *Z3950 Server(s)* - These are the Z3950 servers that will be searched. You can select one or more of these servers.
  - *Z3950 Query* - Specify the Z3950 query to use when searching the selected Z3950 servers. At present this field is known to support the Prefix Query Format (also known as Prefix Query Notation) which is described at this URL: http://www.indexdata.com/yaz/doc/tools.html#PQF. See below for more information and some simple examples.
  - *Icon* - An icon to assign to harvested metadata. The icon will be used when showing search results.
- **Options** - Scheduling options.
- *Run at* - The time when the harvester will run.
- *Will run again every* - Choose an interval from the drop down list and then select the days for which this scheduling will take place.
- *One run only* - Checking this box will cause the harvester to run only when manually started using the `Run` button on the Harvesting Management page.
- **Harvested Content**
  - *Apply this XSLT to harvested records* - Choose an XSLT here that will convert harvested records to a different format.
  - *Validate* - If checked, records that do not/cannot be validated will be rejected.
- **Privileges**
- *Groups* - Groups can be selected from the scrolling list. When the `Add` button is pushed, a row of privileges will be created below the scrolling list for each group. Privileges can then be checked/unchecked for each group as required.
- *Remove* - To remove a row click on the `Remove` button on the right of the row.
- **Categories**
- Select one or more categories from the scrolling list. The harvested metadata will be assigned to the selected categories.

SITE

Name: Z3950 Test Harvest

Z3950 Server(s):
- BRS and Australian Natural Resources Data Library (ANRDL)
- Bureau of Meteorology
- CSIRO Marine and Atmospheric Research
- DEWHA Discover Information Geographically (DIG)
- Geoscience Australia
- IndexGeo Pty Ltd - Eco Companion catalogue
- Local GeoNetwork
- Local GeoNetwork - applications category
- Murray-Darling Basin Authority

Z3950 Query: @attrset geo @attr 1=1016 @attr 4=6 "Franklin"

Icon: Z3950.gif

OPTIONS

Run at: 0 : 00 (hour of day : minute eg. 23:15)

Will run again every: N/A

☑ Monday ☑ Tuesday ☑ Wednesday ☑ Thursday ☑ Friday ☑ Saturday ☑ Sunday

One run only ☑

HARVESTED CONTENT

Apply this XSLT to harvested records: --None--

Validate ☐

PRIVILEGES

Groups: Guest, All, Intranet, Sample group — Add

Group	View	Interactive map	Featured	
All	☑	☐	☐	Remove

CATEGORIES

- Applications
- Audio/Video
- Case studies, best practices
- Datasets
- Directories
- Interactive resources
- Maps & graphics
- Other information resources

Figure 4.33: *Adding a Z3950 harvester*

Note: this harvester automatically creates a new Category named after each of the Z3950 servers that return records. Records that are returned by a server are assigned to the category named after that server.

More about PQF Z3950 Queries

PQF is a rather arcane query language. It is based around the idea of attributes and attribute sets. The most common attribute set used for geospatial metadata in Z3950 servers is the GEO attribute set (which is an extension of the BIB-1 and GILS attribute sets - see http://www.fgdc.gov/standards/projects/GeoProfile). So all PQF queries to geospatial metadata Z3950 servers should start off with @attrset geo.

The most useful attribute types in the GEO attribute set are as follows:

@attr number	Meaning	Description
1	Use	What field to search
2	Relation	How to compare the term specified
4	Structure	What type is the term? eg. date, numeric, phrase
5	Truncation	How to truncate eg. right

In GeoNetwork the numeric values that can be specified for `@attr 1` map to the lucene index field names as follows:

@attr 1=	Lucene index field	ISO19139 element
1016	any	All text from all metadata elements
4	title, altTitle	gmd:identificationInfo//gmd:citation//gmd:title/gco:CharacterString
62	abstract	gmd:identificationInfo//gmd:abstract/gco:CharacterString
1012	_changeDate	Not a metadata element (maintained by GeoNetwork)
30	createDate	gmd:MD_Metadata/gmd:dateStamp/gco:Date
31	publicationDate	gmd:identificationInfo//gmd:citation//gmd:date/gmd:CI_DateCode/@codeListValu
2072	tempExtentBegin	gmd:identificationInfo//gmd:extent//gmd:temporalElement//gml:begin(Position)
2073	tempExtentEnd	gmd:identificationInfo//gmd:extent//gmd:temporalElement//gml:end(Position)
2012	fileId	gmd:MD_Metadata/gmd:fileIdentifier/*
12	identifier	gmd:identificationInfo//gmd:citation//gmd:identifier//gmd:code/*
21,29,2002,3121,3122	keyword	gmd:identificationInfo//gmd:keyword/*
2060	northBL,eastBL,southBL,westBL	gmd:identificationInfo//gmd:extent//gmd:EX_GeographicBoundingBox/gmd:west (etc)

Note that this is not a complete set of the mappings between Z3950 GEO attribute set and the GeoNetwork lucene index field names for ISO19139. Check out INSTALL_DIR/web/geonetwork/xml/search/z3950Server.xsl and INSTALL_DIR/web/geonetwork/xml/schemas/iso19139/index-fields.xsl for more details and annexe A of the GEO attribute set for Z3950 at http://www.fgdc.gov/standards/projects/GeoProfile/annex_a.html for more details.

Common values for the relation attribute (`@attr=2`):

@attr 2=	Description
1	Less than
2	Less than or equal to
3	Equals
4	Greater than or equal to
5	Greater than
6	Not equal to
7	Overlaps
8	Fully enclosed within
9	Encloses
10	Fully outside of

So a simple query to get all metadata records that have the word 'the' in any field would be:

```
@attrset geo @attr 1=1016 the
```

- `@attr 1=1016` means that we are doing a search on any field in the metadata record

A more sophisticated search on a bounding box might be formulated as:

```
@attrset geo @attr 1=2060 @attr 4=201 @attr 2=7 "-36.8262 142.6465
-44.3848 151.2598
```

- `@attr 1=2060` means that we are doing a bounding box search
- `@attr 4=201` means that the query contains coordinate strings
- `@attr 2=7` means that we are searching for records whose bounding box overlaps the query box specified at the end of the query

Notes

- Z3950 servers must be configured for GeoNetwork in `INSTALL_DIR/web/geonetwork/WEB-INF/classes/JZKitConfig.xml.tem`
- every time the harvester runs, it will remove previously harvested records and create new ones.

4.7.10 Harvest History

Each time a harvester is run, it generates a status report of what was harvested and/or what went wrong (eg. exception report). These reports are stored in a table in the database used by GeoNetwork. The entire harvesting history for all harvesters can be recalled using the History button on the Harvesting Management page. The harvest history for an individual harvester can also be recalled using the History link in the Operations for that harvester.

Once the harvest history has been displayed it is possible to:

- expand the detail of any exceptions
- sort the history by harvest date (or in the case of the history of all harvesters, by harvester name)
- delete any history entry or the entire history

Figure 4.34: *An example of the Harvesting Management Page with History functions*

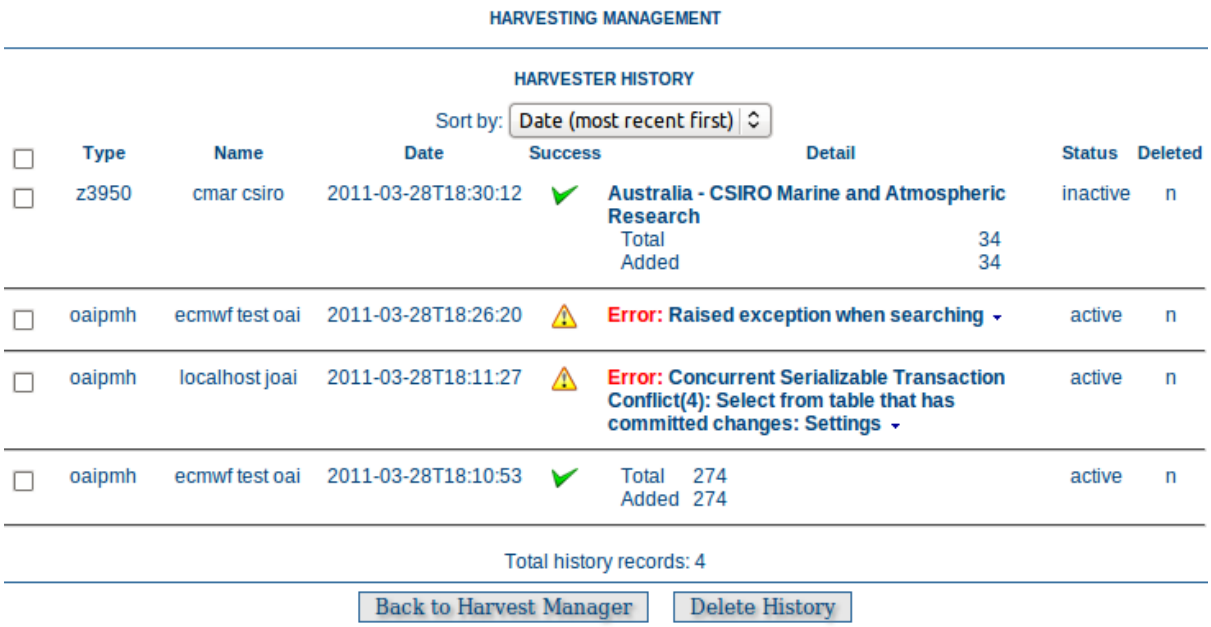


Figure 4.35: *An example of the Harvesting History for all harvesters*

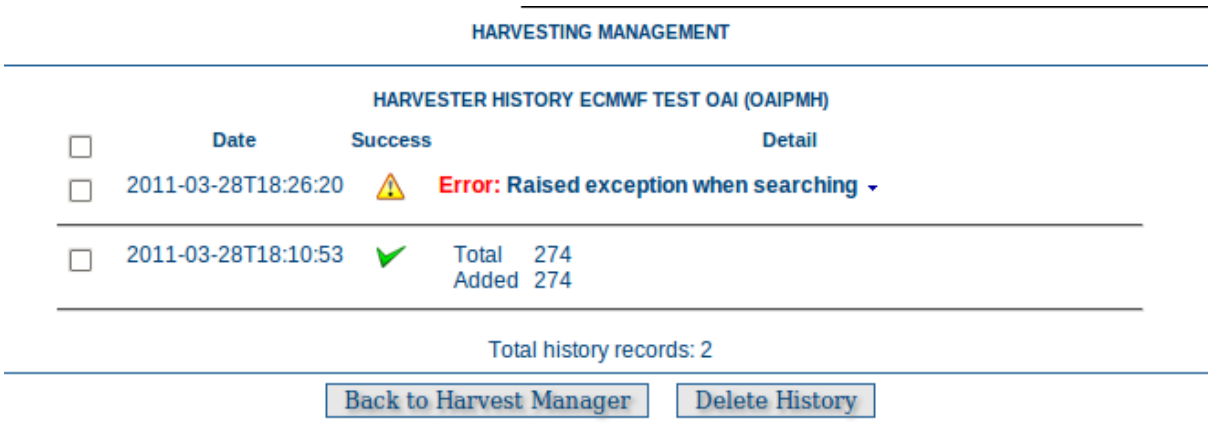


Figure 4.36: *An example of the Harvesting History for a harvester*

4.8 Formatter

4.8.1 Introduction

The metadata.show service (the metadata viewer) displays a metadata document using the default metadata display stylesheets. However it can be useful to provide alternate stylesheets for displaying the metadata. Consider a central catalog that is used by several partners. Each partner might have special branding and wish to emphasize particular components of the metadata document.

To this end the metadata.formatter.html and metadata.formatter.xml services allow an alternate stylesheet to be used for displaying the metadata. The urls of interest to an end-user are:

- /geonetwork/srv/<langCode>/metadata.formatter.html?xsl=<formatterId>&id=<metadataId>
- Applies the stylesheet identified by xsl parameter to the metadata identified by id param and returns the document with the *html* contentType
- /geonetwork/srv/<langCode>/metadata.formatter.xml?xsl=<formatterId>&id=<metadataId>
- Applies the stylesheet identified by xsl parameter to the metadata identified by id param and returns the document with the *xml* contentType
- /geonetwork/srv/<langCode>/metadata.formatter.list
- Lists all of the metadata formatter ids

Another use-case for metadata formatters is to embed the metadata in other websites. Often a metadata document contains a very large amount of data and perhaps only a subset is interesting for a particular website or perhaps the branding/stylesheets needs to be customized to match the website.

4.8.2 Administration

A metadata formatter is a bundle of files that can be uploaded to Geonetwork as a zip file (or in the simplest case just upload the xsl).

An administration user interface exists for managing these bundles. The starting page of the ui contains a list of the available bundles and has a field for uploading new bundles. There are three upload options:

- *Single xsl file* - A new bundle will be created for the xsl file and the name of the bundle will be based on the xsl file name
- *Zip file (flat)* - A zip file which contains a view.xsl file and other required resources at the root of the zip file so that when unzipped the files will be unzipped into the current directory
- *Zip file (folder)* - A zip file with a single folder that contains a view.xsl file and the other required resources so that when unzipped a single directory will be created that contains the formatter resources.

If a bundle is uploaded any existing bundles with the same name will be replaced with the new version.

See Bundle format section below for more details about what files can be contained in the format bundle.

When a format in the formatter list is selected the following options become enabled:

- Delete - Delete the format bundler from Geonetwork
- Download - Download the bundle. This allows the administrator to download the bundle and edit the contents then upload at a later date

- Edit - This provides some online edit capabilities of the bundle. At the moment it allows editing of *existing* text files. Adding new files etc... maybe added in the future but is not possible at the moment. When edit is clicked a dialog with a list of all editable files are displayed in a tree and double clicking on a file will open a new window/tab with a text area containing the contents of the file. The webpage has buttons for saving the file or viewing a metadata with the style. The view options do *NOT* save the document before execution, that must be done before pressing the view buttons.

4.8.3 Bundle format

A format bundle is at minimum a single xsl file. If the xsl file is uploaded it can have any name. On the server a folder will be created that contains the xsl file but renamed to view.xsl.

If a zip file is uploaded the zip file must contain a file view.xsl. The view.xsl file is the entry point of the transformation. It can reference other xsl stylesheets if necessary as well as link to css stylesheets or images that are contained within the bundle or elsewhere.

The view.xsl stylesheet is executed on an xml file with essentially the following format:

- root
  - url - text of the url tag is the base url to make requests to geonetwork. An example is /geonetwork/
  - locUrl - text of the url tag is the localised url to make requests to geonetwork. An example is /geonetwork/srv/eng/
  - resourceUrl - a base url for accessing a resource from the bundle. An example of image tag might be:

    ```
    <img src="{/root/resourceURL}/img.png"/>
    ```

  - <metadata> - the root of the metadata document
  - loc
  - lang - the text of this tag is the lang code of the localization files that are loaded in this section
  - <bundle loc file> - the contents of the bundles loc/<locale>/*.xml files
  - strings - the contents of geonetwork/loc/<locale>/xml/strings.xml
  - schemas
    - <schema> - the name of the schema of the labels and codelists strings to come
    - labels - the localised labels for the schema as defined in the schema_plugins/<schema>/loc/<locale>/labels.xml
    - codelists - the localised codelists labels for the schema as defined in the schema_plugins/<schema>/loc/<locale>/codelists.xml
    - strings - the localised strings for the schema as defined in the schema_plugins/<schema>/loc/<locale>/strings.xml

If the view.xsl output needs to access resources in the formatter bundle (like css files or javascript files) the xml document contains a tag: resourceUrl that contains the url for obtaining that resource. An example of an image tag is:

```
<img src="{/root/resourceURL}/img.png"/>
```

By default the strings, labels, etc... will be localized based on the language provided in the URL. For example if the url is /geonetwork/srv/eng/metadata.formatter.html?xsl=default&id=32 then the language code that is used to look up the localization will be eng. However if the language code does not exist it will fall back to the Geonetwork platform default and then finally just load the first local it finds.

Schemas and geonetwork strings all have several different translations but extra strings, etc... can be added to the formatter bundle under the loc directory. The structure would be:

```
loc/<langCode>/strings.xml
```

The name of the file does not have to be strings.xml. All xml files in the loc/<langCode>/ directory will be loaded and added to the xml document.

The format of the formatter bundle is as follows:

```
config.properties
view.xsl
loc/<langCode>/
```

Only the view.xsl is required. If a single xsl file is uploaded then the rest of the directory structure will be created and some files will be added with default values. So a quick way to get started on a bundle is to upload an empty xsl file and then download it again. The downloaded zip file will have the correct layout and contain any other optional files.

4.8.4 Config.properties

The config.properties file contains some configuration options used when creating the xml document. Some of the properties include:

- *fixedLang* - sets the language of the strings to the fixed language, this ensures that the formatter will always use the same language for its labels, strings, etc... no matter what language code is in the url.
- *loadGeonetworkStrings* - if true or non-existent then geonetwork strings will be added to the xml document before view.xsl is applied. The default is true so if this parameter is not present then the strings will be loaded
- *schemasToLoad* - defines which schema localization files should be loaded and added to the xml document before view.xsl is applied
- if a comma separated list then only those schemas will be loaded
- if all then all will be loaded
- if none then no schemas will be loaded
- *applicableSchemas* - declares which schemas the bundle can format
- A comma separated list indicates specifically which schemas the bundle applies to
- If the value is all (or value is empty) then all schemas are considered supported

4.9 Processing

GeoNetwork can batch process metadata records by applying an XSLT. The processing XSLTs are schema dependent and must be stored in the process folder of each metadata schema. For example, the process folder for the iso19139 metadata schema can be found in `GEONETWORK_DATA_DIR/config/schema_plugins/iso19139/process`.

Some examples of batch processing are:

- Filtering harvested records from another GeoNetwork node (See *GeoNetwork Harvesting* in the Harvesting section of this manual)
- Suggesting content for metadata elements (editor suggestion mechanism)
- Applying an XSLT to a selected set of metadata records by using the xml.batch.processing service (this service does not have a user interface, it is intended to be used with an http submitter such as curl).

4.9.1 Process available

Anonymizer

- schema: ISO19139
- usage: Harvester

Anonymiser is an XSL transformation provided for ISO19139 records which removes all resource contacts except point of contact. In addition, it has three custom options to replace email addresses, remove keywords and remove internal online resources. These options are controlled by the following parameters:

- protocol: Protocol of the online resources that must be removed
- email: Generic email to use for all email addresses in a particular domain (ie. after @domain.org).
- thesaurus: Portion of thesaurus name for which keywords should be removed

It could be used in the GeoNetwork harvesting XSL filter configuration using:

```
anonymizer?protocol=DB:&email=gis-service@myorganisation.org&thesaurus=MYINTERNALTHESAUR
```

Scale denominator formatter

- schema: ISO19139
- usage: Suggestion

Format scale which contains " ", "/" or ":" characters.

Add extent form geographic keywords

- schema: ISO19139
- usage: Suggestion

Compute extent based on keyword of type place using installed thesaurus.

WMS synchronizer

- schema: ISO19139
- usage: Suggestion

If an OGC WMS server is defined in distribution section, suggest that the user add extent, CRS and graphic overview based on that WMS.

Add INSPIRE conformity

- schema: ISO19139
- usage: Suggestion

If INSPIRE themes are found, suggest that the user add an INSPIRE conformity section.

Add INSPIRE data quality report

- schema: ISO19139
- usage: Suggestion

Suggest the creation of a default topological consistency report when INSPIRE theme is set to Hydrography, Transport Networks or Utility and governmental services

Keywords comma exploder

- schema: ISO19139
- usage: Suggestion

Suggest that comma separated keywords be expanded to remove the commas (which is better for indexing and searching).

Keywords mapper

- schema: ISO19139
- usage: Batch process

Process records and map keyword define in a mapping table (to be defined manually in the process).

Linked data checker

- schema: ISO19139
- usage: Suggestion

Check URL status and suggest to remove the link on error.

Thumbnail linker

- schema: ISO19139
- usage: Batch process

This batch process creates a browse graphic or thumbnail for all metadata records.

Process parameters:

- prefix: thumbnail URL prefix (mandatory)
- thumbnail_name: Name of the element to use in the metadata for the thumbnail file name (without extension). This element should be unique in a record. Default is gmd:fileIdentifier (optional).
- thumbnail_desc: Thumbnail description (optional).
- thumbnail_type: Thumbnail type (optional).
- suffix: Thumbnail file extension. Default is .png (optional).

Inserted fragment is:

```
<gmd:graphicOverview>
    <gmd:MD_BrowseGraphic>
      <gmd:fileName>
        <gco:CharacterString>$prefix + $thumbnail_name + $suffix</gco:CharacterString>
      </gmd:fileName>
      <gmd:fileDescription>
        <gco:CharacterString>$thumbnail_desc</gco:CharacterString>
      </gmd:fileDescription>
      <gmd:fileType>
        <gco:CharacterString>$thumbnail_type</gco:CharacterString>
      </gmd:fileType>
    </gmd:MD_BrowseGraphic>
</gmd:graphicOverview>
```

4.10 Fragments

GeoNetwork supports metadata records that are composed from fragments of metadata. The idea is that the fragments of metadata can be used in more than one metadata record.

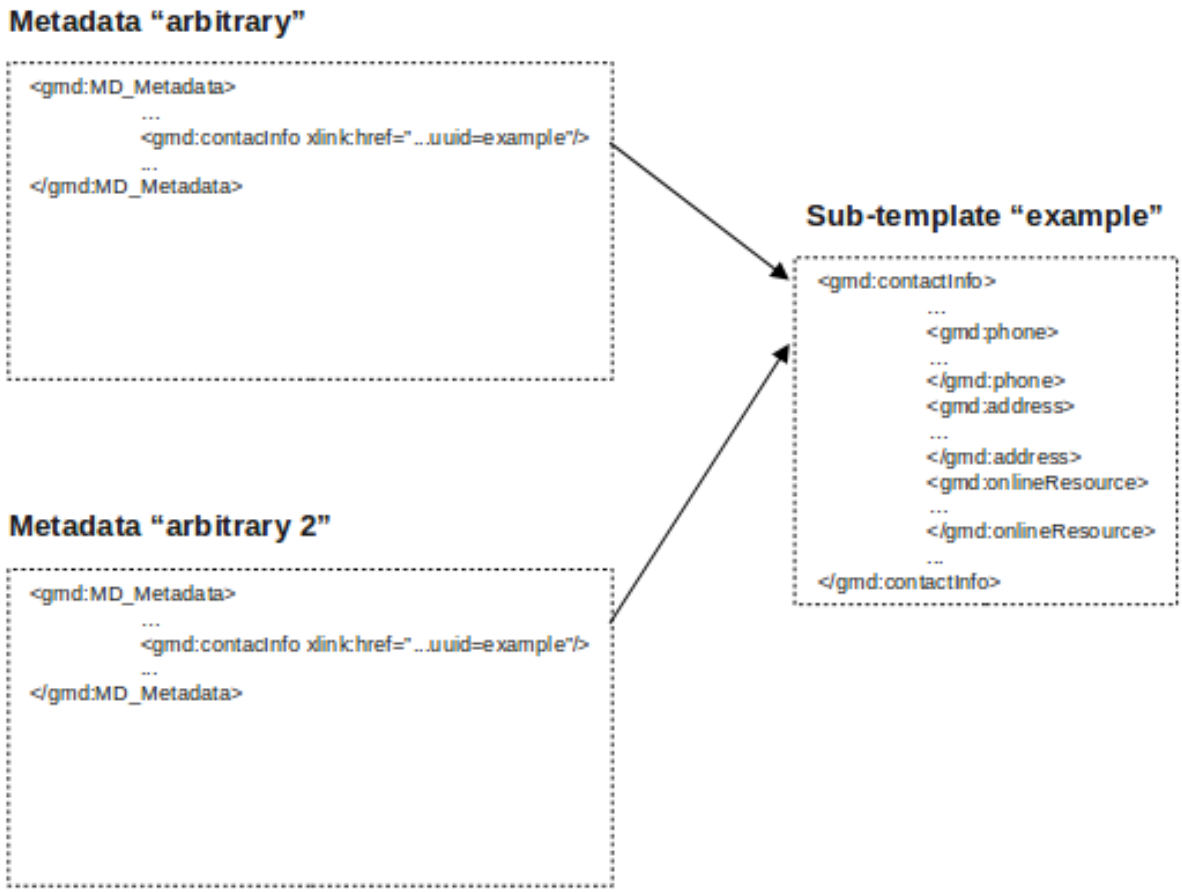
Here is a typical example of a fragment. This is a responsible party and it could be used in the same metadata record more than once or in more than one metadata record if applicable.

```
<gmd:CI_ResponsibleParty xmlns:gmd="http://www.isotc211.org/2005/gmd" xmlns:gco="http://
  <gmd:individualName>
    <gco:CharacterString>John D'ath</gco:CharacterString>
  </gmd:individualName>
  <gmd:organisationName>
    <gco:CharacterString>Mulligan &amp; Sons, Funeral Directors</gco:CharacterString>
  </gmd:organisationName>
  <gmd:positionName>
    <gco:CharacterString>Undertaker</gco:CharacterString>
  </gmd:positionName>
  <gmd:role>
    <gmd:CI_RoleCode codeList="./resources/codeList.xml#CI_RoleCode" codeListValue="poin
```

```
  </gmd:role>
</gmd:CI_ResponsibleParty>
```

Metadata fragments that are saved in the GeoNetwork database are called subtemplates. This is mainly for historical reasons as a subtemplate is like a template metadata record in that it can be used as a 'template' for constructing a new metadata record.

Fragments are not handled by GeoNetwork unless xlink support is enabled. See *XLink resolver* in the 'System Configuration' section of this manual. The reason for this is that XLinks are the main mechanism by which fragments of metadata can be included in metadata records.



Fragments may be created by harvesting (see *Harvesting Fragments of Metadata to support re-use*), used in register thesauri (see *Preparing to edit an ISO19135 register record*) and linked into metadata records using the GeoNetwork editor in the javascript widget interface.

This section of the manual will describe:

- how to manage directories of subtemplates
- how to extract fragments from an existing set of metadata records and store them as subtemplates
- how to manage the fragment cache that GeoNetwork uses to speed up access to fragments that are not in the local catalogue

4.10.1 Managing Directories of subtemplates

There are some differences between the handling of subtemplates and metadata records in GeoNetwork. Unlike metadata records, subtemplates do not have a consistent root element, the metadata schema they use may not be recognizable, they do not appear in search results (unless they are part of a metadata record) and they cannot be assigned privileges. But like metadata records, they are allocated an integer id and are stored in the GeoNetwork metadata table (with template field set to "s").

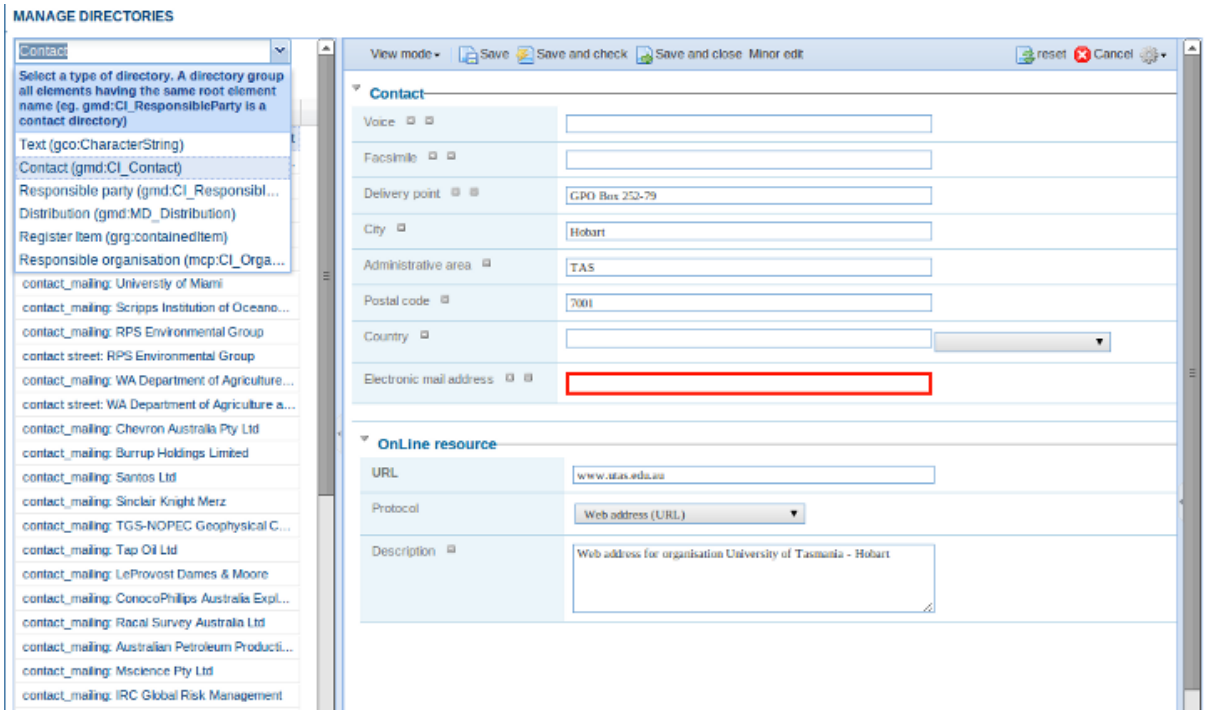
Because of these differences, a separate interface has been built to search, display and edit subtemplates in directories based upon their root element. The interface is accessed from the GeoNetwork "Administration" page. To access this page you need to be logged in as a GeoNetwork Administrator. The relevant part of the GeoNetwork Administration page is shown in the following screenshot with the directory interface highlighted.

The subtemplate directory function on the Administration page

Clicking on this link will bring up the directory interface. The directory interface allows you to browse the available subtemplates according to their root element or search for any subtemplate with content content containing the search term. A typical directory for a site is shown in the following screenshot.

Notice in the screenshot that we have selected the directory of subtemplates with the root element gmd:CI_Contact? The other directories for this particular site are also shown.

The edit interface shown in the right hand panel is self-explanatory.



4.10.2 Extracting subtemplates from a metadata record

Many sites have existing metadata records with common information eg. contact information in an ISO CI_Contact element. Fragments such as these can be extracted from a selected set of metadata records using the "Extract subtemplates" function provided in the "actions on selected set menu".

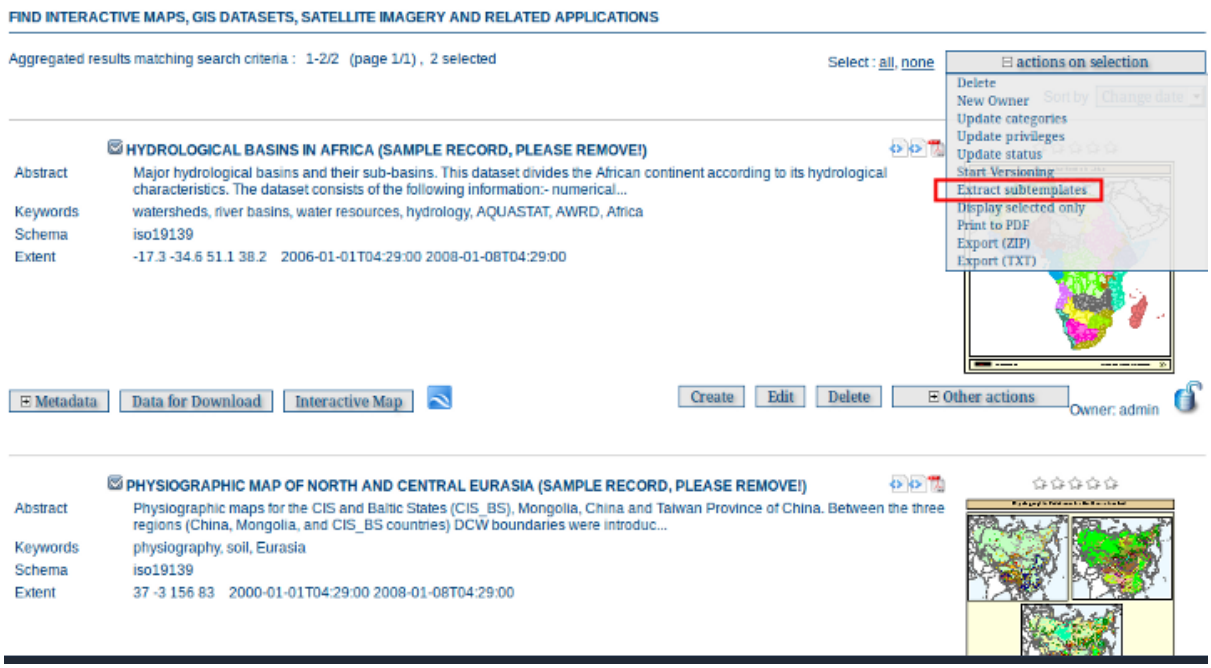
To use this function the following set of steps should be followed:

- Make sure you understand what an XPath is - see http://www.w3schools.com/xpath/default.asp for example.
- Identify fragments of metadata that they would like to manage as reusable subtemplates in the metadata record. This can be done using an XPath. eg. the XPath `/gmd:MD_Metadata/gmd:contact/gmd:CI_ResponsibleParty/gmd:contactInfo/gmd:CI_` identifies metadata contact information in iso19139 metadata records. An example of such a fragment (taken from one of the GeoNetwork sample records) is shown in the following example:

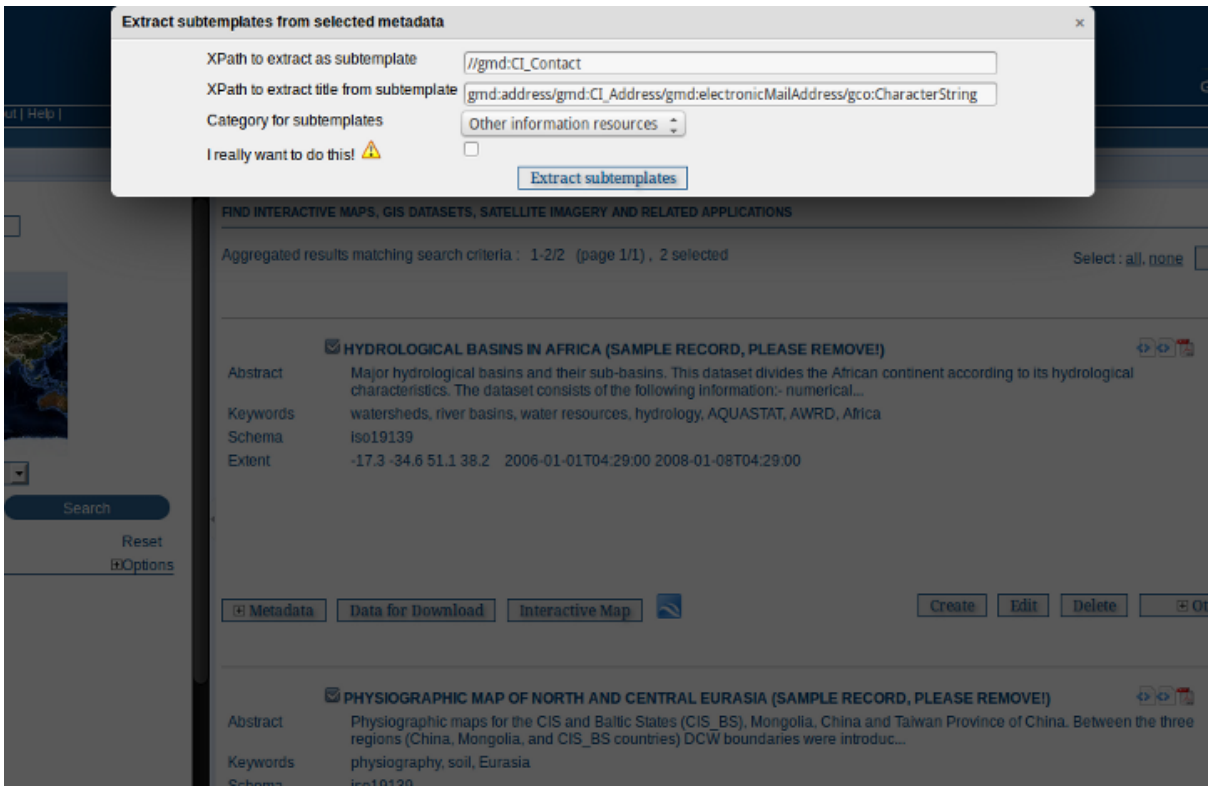
```
<gmd:CI_Contact>
  <gmd:phone>
    <gmd:CI_Telephone>
      <gmd:voice>
        <gco:CharacterString/>
      </gmd:voice>
      <gmd:facsimile>
        <gco:CharacterString/>
      </gmd:facsimile>
    </gmd:CI_Telephone>
  </gmd:phone>
  <gmd:address>
    <gmd:CI_Address>
      <gmd:deliveryPoint>
        <gco:CharacterString>Viale delle Terme di Caracalla</gco:CharacterString>
      </gmd:deliveryPoint>
      <gmd:city>
```

```
        <gco:CharacterString>Rome</gco:CharacterString>
      </gmd:city>
      <gmd:administrativeArea>
        <gco:CharacterString/>
      </gmd:administrativeArea>
      <gmd:postalCode>
        <gco:CharacterString>00153</gco:CharacterString>
      </gmd:postalCode>
      <gmd:country>
        <gco:CharacterString>Italy</gco:CharacterString>
      </gmd:country>
      <gmd:electronicMailAddress>
        <gco:CharacterString>jippe.hoogeveen@fao.org</gco:CharacterString>
      </gmd:electronicMailAddress>
    </gmd:CI_Address>
  </gmd:address>
</gmd:CI_Contact>
```

- Identify and record the XPath of a field or fields within the fragment which text content will be used as the title of the subtemplate. It is important to choose a set of fields that will allow a human to identify the subtemplate when they choose to either reuse the subtemplate in a new record or edit in the subtemplate directories interface. This XPath should be relative to the root element of the fragment identified in the previous step. So for example, in the fragment above we could choose `gmd:address/gmd:CI_Address/gmd:electronicMailAddress/gco:CharacterString` as the title for the fragments to be created.
- On the GeoNetwork home page, search for and then select the records from which the subtemplates will be extracted. Choose "Extract subtemplates" from the "actions on selection" menu as shown in the following screenshot:

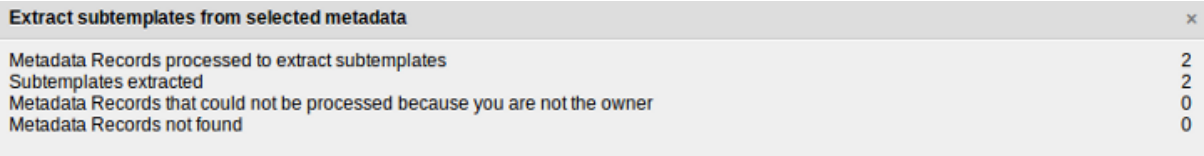


- Fill in the form with the information collected in the previous steps. It should look something like the following:
- Run the extract subtemplate function in test mode (ie. without checking the "I really want to do this" box). This will test whether your XPaths are correct by extracting one subtemplate from the

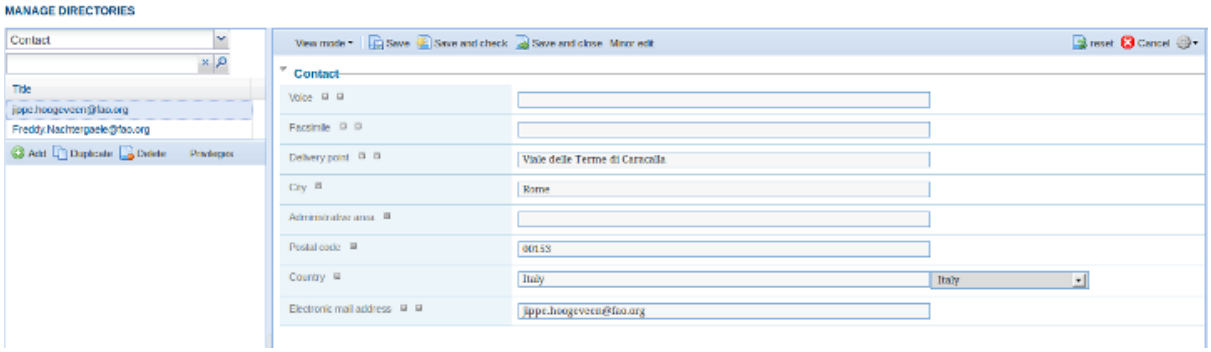


selected set of records and displaying the results.

- If you are happy with the test results, go ahead with the actual extraction by checking the "I really want to do this" checkbox. After the extraction completes you should see some results.



- Finally, go to the subtemplate directory management interface and you should be able to select the root element of your subtemplates to examine the extracted subtemplates.



- The metadata records from which the subtemplates were extracted now have xlinks to the subtemplates.

4.10.3 Managing the fragment cache

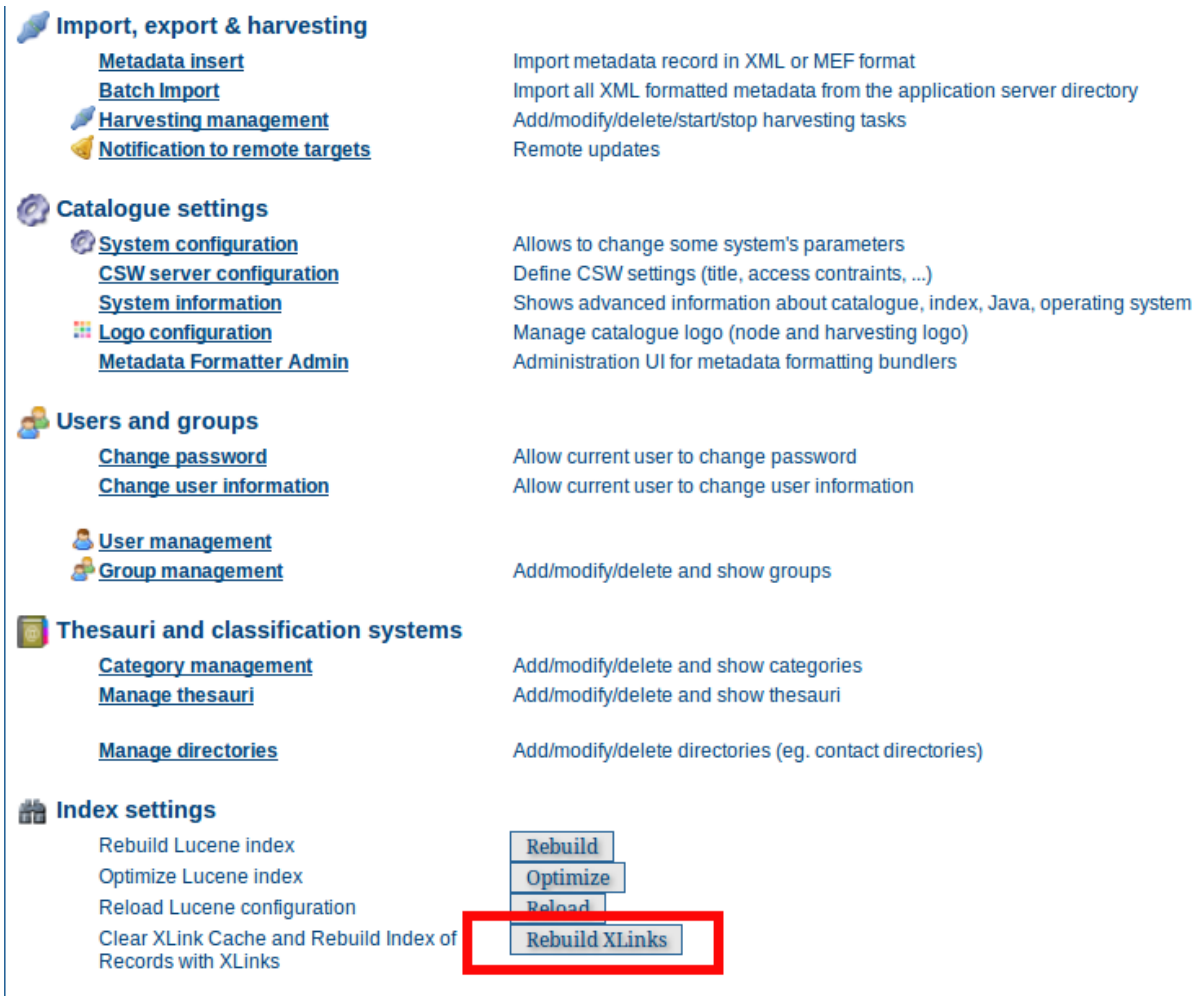
If metadata records in your catalog link in fragments from external sites, GeoNetwork caches these fragments after the first look up so as to reduce the amount of network traffic and speed up the display of metadata records in search results.

The cache is handled automatically using the Java Cache System (JCS). JCS handles large caches intelligently by:

- defining a maximum number of cached objects
- using as much main memory as possible before moving to secondary storage (disk)
- providing cache persistence: the cache is saved to disk when the GeoNetwork web application is shutdown and restores the cache from disk when GeoNetwork restarts

You can configure JCS parameters in GeoNetwork using the JCS configuration file in **INSTALL_DIR/web/geonetwork/WEB-INF/classes/cache.ccf**.

Some operations in GeoNetwork (such as harvesting) that generate metadata fragments, will automatically refresh the XLink cache when a new fragment is generated. However, if you are linking fragments from an external site, then depending on how often the change, you will need to manually refresh the XLink cache. To do this you should navigate to the Administration page and select the "Clear XLink Cache and Rebuild Index of Records with XLinks" function as highlighted in the following screenshot of the "Administration" page.



Function to clearing the XLink cache on the Administration page

Note: finer control of the XLink cache will be implemented in a future version of GeoNetwork.

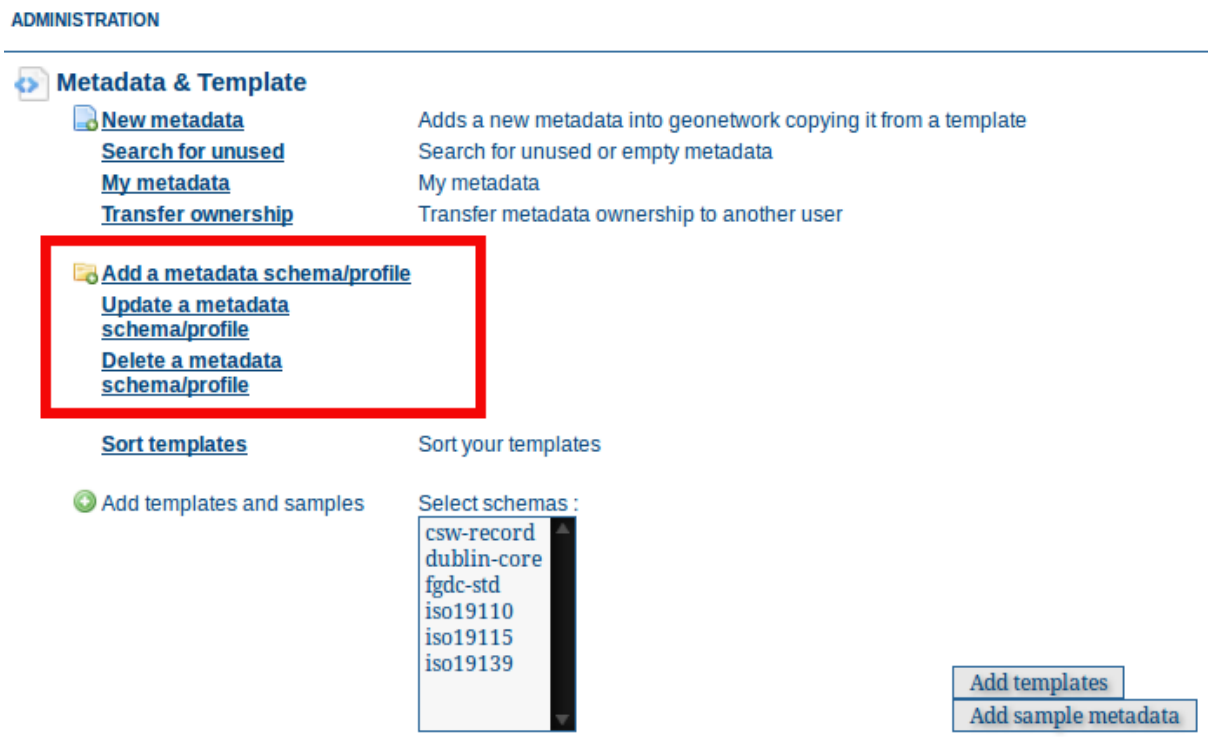
4.11 Schemas

Metadata records in GeoNetwork are described by a schema. The schema sets out the structuring of the metadata record and provides all the ancillary data and functions to use the schema in GeoNetwork.

A metadata schema plugin capability has been introduced in GeoNetwork 2.8.0. This allows the administrator to add, update and delete metadata schemas in GeoNetwork without the need to stop and restart GeoNetwork.

Note: Adding a metadata schema to GeoNetwork that is incorrect or invalid can thoroughly break your GeoNetwork instance. This section is for catalogue administrators who are confident about metadata schemas and understand the different technologies involved with a GeoNetwork metadata schema.

A detailed description of what constitutes a metadata schema for GeoNetwork can be found in the GeoNetwork Developers Manual. This section will describe how to access the schema add, update and delete functions and how those functions should be used. To access these functiuons you need to be logged in to GeoNetwork as an Administrator. The schema functions are on the Administration page as shown below.



The Administration page with the metadata schema functions highlighted

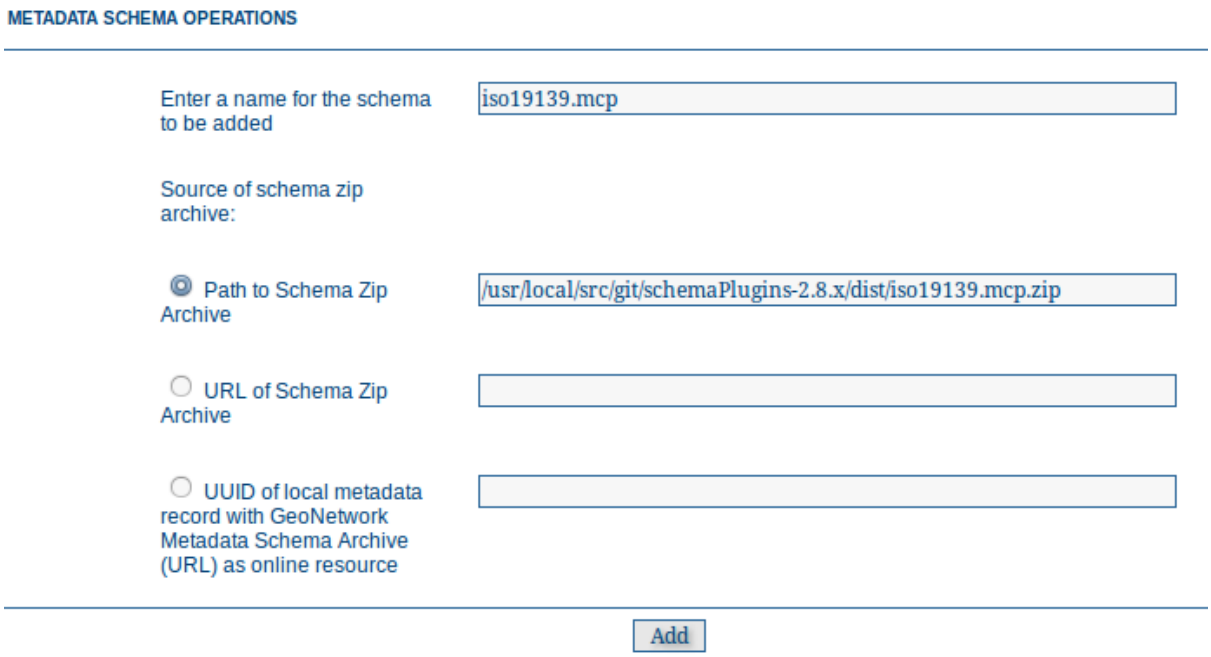
Note: Metadata schemas should be thoroughly tested in a development instance of GeoNetwork **before**

they are added to a production instance. Errors in a schema plugin (particularly in the presentation XSLTs) may make your GeoNetwork instance unusable.

4.11.1 Adding a schema

To a metadata schema to GeoNetwork, click on the **Add a metadata schema/profile** link in the Administration page as shown above.

This will bring up a menu from which you can specify the location of a metadata schema in a ZIP archive to add to GeoNetwork.



There are three possible locations for the ZIP archive:

1. on the server filesystem - you specify the path of the ZIP archive on the server filesystem
2. on a web server accessible via a http link - you specify the URL of the ZIP archive on the web server.
3. Attached to a metadata record describing the schema which is present in the local GeoNetwork catalog - you specify the UUID of that metadata record which must be an iso19139 metadata record.

4.11.2 Updating a schema

To update a metadata schema in GeoNetwork, click on the **Update a metadata schema/profile** link in the Administration page as shown above. You will be presented with a menu that is the same as the one for adding a metadata schema, except that instead of a text box for typing in the name of the new metadata schema, you select a metadata schema to update from a drop down menu of those already present in GeoNetwork.

4.11.3 Deleting a schema

To delete a metadata schema from GeoNetwork, click on the **Delete a metadata schema/profile** link in the Administration page as shown above. You will be presented with a drop down list of the current metadata schemas in GeoNetwork from which you can select one to delete.

Note: You cannot delete a metadata schema if there are records that belong to that schema in the catalog. You must delete all the records that belong to that schema first before you can delete the schema itself.

Note: You cannot delete a metadata schema if another schema depends upon that schema eg. you cannot delete the iso19139 schema if the iso19139.mcp schema is present because the iso19139.mcp schema is a profile that depends on iso19139. Schema dependencies can be found/specified in the schema-ident.xml file.

CHAPTER 5

Features

5.1 Multilingual search

5.1.1 Introduction

GeoNetwork supports *multilingual* search. Depending on the configuration, this influences which search results are returned and how they are presented:

Enable auto-detecting search request language: If this option is selected, Geonetwork will analyse the search query and attempt to detect the language that is used before defaulting to the GUI language.

Search only in requested language The options in this section determines how documents are sorted/prioritised relative to the language in the document compared to the search language.

- *All documents in all languages (No preferences)* - The search language is ignored and will have no effect on the ordering of the results
- *Prefer documents with translations requested language* - Documents with a translation in the search language (anywhere in the document) will be prioritized over documents without any elements in the search language
- *Prefer documents whose language is the requested language* - Documents that are the same language as the search language (IE the documents that are specified as being in the same language as the search language) or prioritized over documents that are not.
- *Translations in requested language* - The search results will only contain documents that have some translations in the search language.
- *Document language is the requested language* - The search results will contain documents whose metadata language is specified as being the in search language

Administrator users can change these settings in the System Configuration page:

5.1.2 Requested language

The requested language is determined as follows (in this order):

- request parameter: in the GUI, the user may select a language:

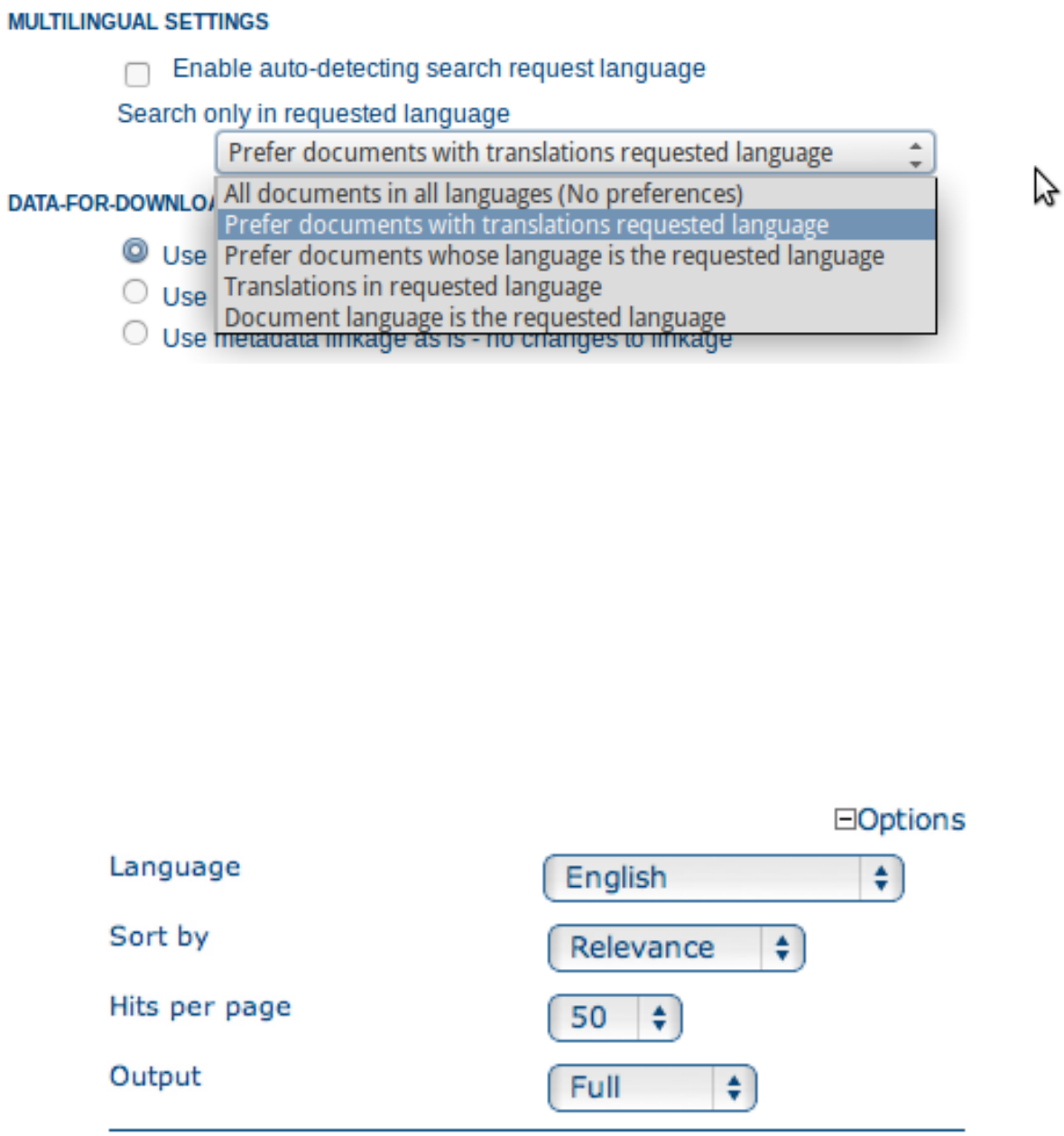
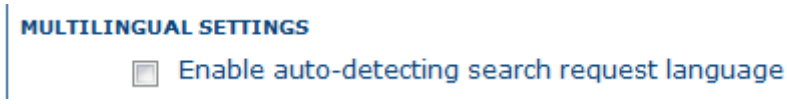


Figure 5.1: For XML searches, the client may add:

```
<requestedLanguage>language-code</requestedLanguage>
```

where language-code is one of the ISO 639-2 (three-character) language codes, see http://www.loc.gov/standards/iso639-2/php/code_list.php.

- if the request parameter is not sent (the user selected "any" language, or it's not in the XML request), the requested language may be automatically detected, if an Administrator user has enabled this in the System Configuration:



The auto-detection feature uses Language Detection Library for Java, see https://code.google.com/p/language-detection/. This library tries to detect the language of search terms in parameter 'any'. This may not work very well, depending on the language, if there is only one or very few search terms. This is why this feature is disabled by default. At the time of writing the auto-detection supports these languages:

- Afrikaans
- Arabic
- Bulgarian
- Bengali
- Czech
- Danish
- German
- Greek (modern)
- English
- Spanish
- Estonian
- Persian
- Finnish
- French
- Gujarati
- Hebrew
- Hindi
- Croatian
- Hungarian
- Indonesian
- Italian
- Japanese
- Kannada
- Korean

- Lithuanian
- Latvian
- Macedonian
- Malayalam
- Marathi
- Nepali
- Dutch
- Norwegian
- Punjabi
- Polish
- Portuguese
- Romanian
- Russian
- Slovak
- Slovenian
- Somali
- Albanian
- Swedish
- Swahili
- Tamil
- Telugu
- Thai
- Tagalog
- Turkish
- Ukrainian
- Urdu
- Vietnamese
- Chinese (traditional)
- Chinese (simplified)
- if autodetecting the language is disabled (the default), the current language of the user's GUI is used as the requested language
- if there is no GUI, the requested language is hardcoded to be English

5.1.3 Stopwords

Stopwords are words that are considered to carry little or no meaning relevant to search. To improve relevance ranking of search results, stopwords are often removed from search terms. In GeoNetwork stopwords are automatically used if a stopwords list for the requested language is available; if not, no stopwords are used. At the time of writing there are stopword lists for:

- Arabic
- Bulgarian
- Bengali
- Catalan
- Czech
- Danish
- German
- Greek (modern)
- English
- Spanish
- Persian
- Finnish
- French
- Hindi
- Hungarian
- Italian
- Japanese
- Korean
- Marathi
- Malay
- Dutch
- Norwegian
- Polish
- Portuguese
- Romanian
- Russian
- Swedish
- Turkish
- Chinese

System administrators may add additional languages' stopword lists by placing them in the directory <geonetwork>/web/resources/stopwords. The filenames should be <ISO 639-2 code>.txt. If you do add a stopwords list for another language, please consider contributing it for inclusion in GeoNetwork.

Likewise, to disable stopwords usage for one or more languages, the stopword list files should be removed or renamed.

5.2 Search Statistics

Searches made through the user interface on the local catalog are logged to the GeoNetwork database. The database holds metadata about the search request (eg. date on which it was made, simple or advanced, query, ip address of the client making the request) and the details of the search terms/parameters and values used in the search query.

5.2.1 Querying search statistics

The search statistics page is part of the Administration menu. You need to be logged in as an Administrator to access it **and** search statistics needs to be **enabled** in the System Configuration. See *Search Statistics* in the 'System Configuration' section of the manual.

Finding search statistics in the Administration page

As delivered the search statistics page delivers a number of indicators and some reports.

5.2.2 Adding your own search statistics

The indicators and reports that are needed/desired at your site might be different to those provided with GeoNetwork on the search statistics page. For that reason the search statistics implementation has been designed with extensibility in mind.

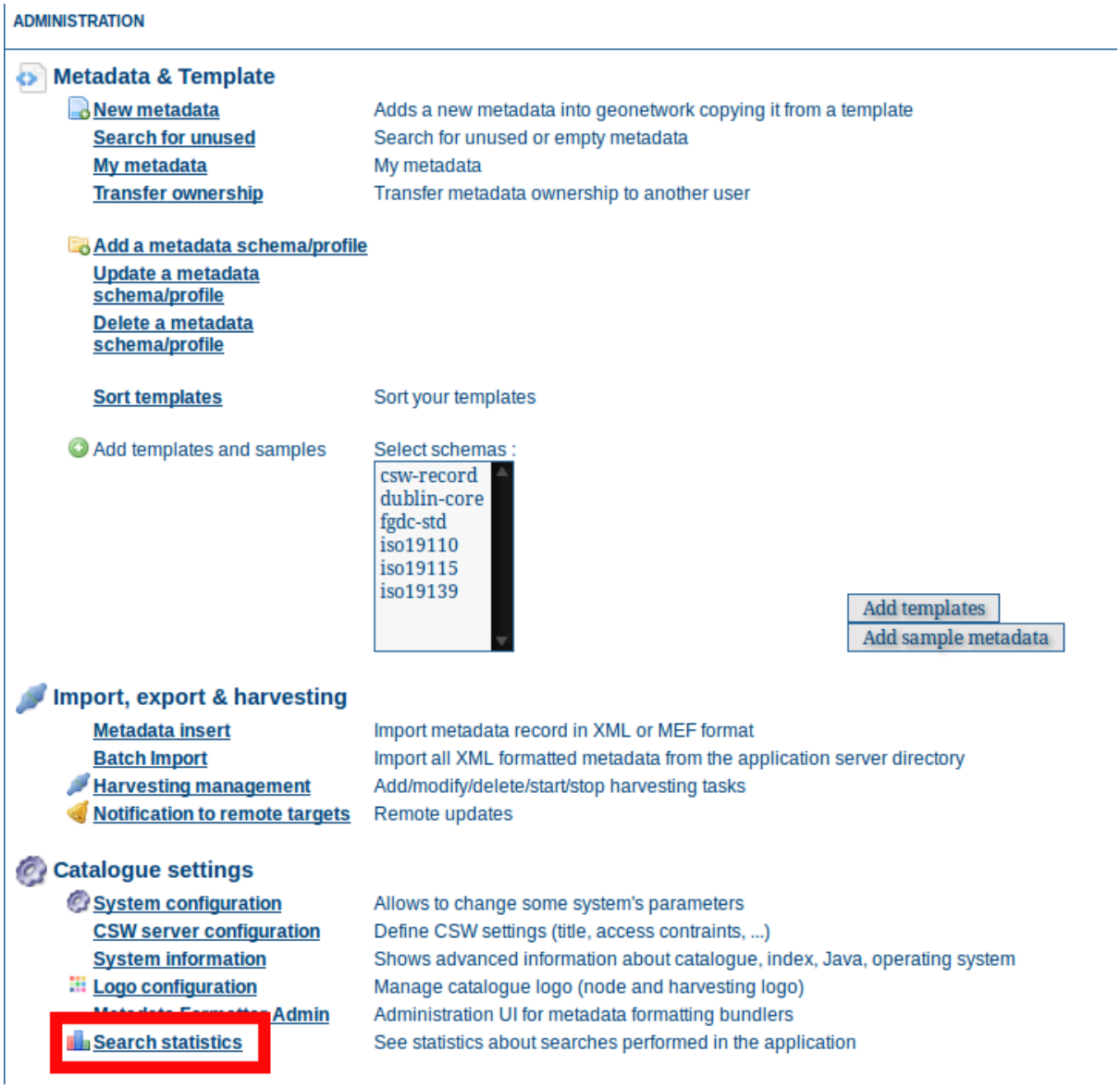
There are two types of search statistic services:

- pure XML: you specify a query on the database and a stylesheet to process the XML output from the query and display a web page in HTML - this could be used to add your own indicators to the search statistics page (or your own custom stats page).
- Java+XSLT: this is the more traditional GeoNetwork service, where you need to code a Jeeves service in Java which produces some XML and provide an XSLT to style the output from that service - this could be used in conjunction with the JFreeChart Java API to produce a chart or report from the search statistics

The service definitions for the default search statistics provide examples of both these types of service. You can find the service definitions in the GeoNetwork release at **INSTALL_DIR/web/geonetwork/WEB-INF/config-statistics.xml**. The XSLTs that style the output from these services are in **INSTALL_DIR/web/geonetwork/xsl/statistics**.

5.2.3 Exporting the search statistics as a CSV file

If you feel so inclined, you may want to export the search statistics from the GeoNetwork database as a CSV file and process them in a spreadsheet. The search statistics page provided with GeoNetwork has this capability.


ADMINISTRATION
Metadata & Template
New metadata
Adds a new metadata into geonetwork copying it from a template
Search for unused
Search for unused or empty metadata
My metadata
My metadata
Transfer ownership
Transfer metadata ownership to another user
Add a metadata schema/profile
Update a metadata schema/profile
Delete a metadata schema/profile
Sort templates
Sort your templates
Add templates and samples
Select schemas :
csw-record
dublin-core
fgdc-std
iso19110
iso19115
iso19139
Add templates
Add sample metadata
Import, export & harvesting
Metadata insert
Import metadata record in XML or MEF format
Batch Import
Import all XML formatted metadata from the application server directory
Harvesting management
Add/modify/delete/start/stop harvesting tasks
Notification to remote targets
Remote updates
Catalogue settings
System configuration
Allows to change some system's parameters
CSW server configuration
Define CSW settings (title, access contraints, ...)
System information
Shows advanced information about catalogue, index, Java, operating system
Logo configuration
Manage catalogue logo (node and harvesting logo)
Admin
Administration UI for metadata formatting bundlers
Search statistics
See statistics about searches performed in the application

5.3 Thesaurus

5.3.1 Introduction

A thesaurus is a list of words (or concepts) from a specialized field of knowledge. In a metadata catalog, words from a thesaurus can be assigned to a metadata record (as keywords) as a way of associating it with one or more concepts from a field of knowledge. For example, a record may be assigned a keyword 'AGRICULTURE - Crops' meaning that the record describes a resource or activity relating to crops in the field of Agriculture.

In GeoNetwork, the process of assigning keywords to a metadata record takes place in the metadata editor. The user can choose words from one or more thesauri to associate the record with the concepts described by those words. This process is supported for both ISO19115/19139 and dublin core metadata records using an extjs based thesaurus picker.

Concepts within a field of knowledge or in different fields of knowledge may be related or even be equivalent. For example, in a thesaurus describing geographic regions, the Australian state of 'Tasmania' is a specialization of the country of Australia. As an example of overlapping concepts in different fields, a thesaurus describing science activities in the field of global change may have concepts relating to agricultural activities that will be equivalent to terms from a thesaurus that describes the themes used in a map series.

In GeoNetwork, thesauri are represented as SKOS (Simple Knowledge Organisation System). SKOS (more on this below) captures concepts and relationships between concepts. SKOS thesauri can be imported from standalone files or they can be generated from ISO19135 register records in the local GeoNetwork catalog. ISO19135 (more on this below) not only captures the concepts and relationships between the concepts, but (amongst other things) how the concepts have evolved and most importantly, who has contributed to and managed the evolution of the concepts and the thesauri itself.

5.3.2 External, Local and Register Thesauri

There are three types of thesaurus in GeoNetwork. The different types are based on where the thesaurus comes from:

- *External*: A thesaurus managed by an external organisation and imported as a SKOS file. It is flagged to "external" which means that users are not allowed to edit the thesaurus.
- *Local*: A thesaurus built in the GeoNetwork thesaurus editor and stored as a SKOS file. It is flagged as "local" which means that users are allowed to edit the thesaurus.
- *Register*: A SKOS thesaurus created from an ISO19135 register record. Users can edit the thesaurus by changing the content of the ISO19135 register record in the GeoNetwork metadata editor and then regenerating the thesaurus. Users cannot edit the thesaurus in thesaurus manager.

5.3.3 ISO19115/19139 Thesaurus Categories

All thesauri in GeoNetwork are categorized using the codelist values for the gmd:MD_KeywordTypeCode element from ISO19115/19139. The categories and their meanings are given below but can also be found in http://www.isotc211.org/2005/resources/gmxCodelist.xml:

ISO Thesaurus Category	Description
place	Thesaurus has concepts identifying a location
stratum	Thesaurus has concepts identifying layers of any deposited substance
temporal	Thesaurus has concepts identifying a time period
theme	Thesaurus has concepts identifying a particular subject or topic
discipline	Thesaurus has concepts identifying a branch of instruction or specialized learning

5.3.4 SKOS format

The Simple Knowledge Organisation Systems (SKOS) http://www.w3.org/2004/02/skos/ is an area of work developing specifications and standards to support the use of knowledge organisation systems (KOS) such as thesauri, classification schemes. This format is used by GeoNetwork to store thesaurus information.

A concept is defined by an identifier, a preferred label, a definition and links with other concepts. Labels and definitions could be stored in multiple languages (using the xml:lang attributes). Three type of links between concepts have been defined in the SKOS format:

- related terms
- broader terms
- narrower terms

For example, a concept "ABLETTE" could be defined as follow with a label in French and English, linked to broader concept:

```
<skos:Concept rdf:about="http://www.oieau.org/concept#c4fc54576dc00227b82a709287ac3681">
    <skos:prefLabel xml:lang="fr">ABLETTE</skos:prefLabel>
    <skos:prefLabel xml:lang="en">BLEAK</skos:prefLabel>
    <skos:broader rdf:resource="http://www.oieau.org/concept#9f25ece36d04776e09492c66627
</skos:Concept>
```

GeoNetwork supports multilingual thesauri (e.g. Agrovoc). Search and editing takes place in the current user interface language (i.e. if the interface is in English, when editing metadata, GeoNetwork will only search for concept in English).

We use SKOS to represent thesauri in GeoNetwork because:

- it provides a simple and compact method of describing concepts and relationships between concepts from a field of knowledge
- SKOS concepts can be queried and managed by the sesame/openRDF software used by GeoNetwork

5.3.5 ISO19135 register format

ISO19135 is an ISO standard that describes procedures for registering an item and the schema for describing a list (or register) of items and the processes by which the items can be created and evolve. This schema is available as a plugin for use in GeoNetwork. To use it, you must download and load the iso19135 plugin schema into GeoNetwork. FIXME: We need a standard way of referring to plugin schemas and a standard place from which they can be downloaded.

A typical ISO19135 register record describes:

- the name and a description of the register
- version and language information
- contact information of those that have a role in the register (eg. manager, contributor, custodian, publisher etc)
- the elements used to describe an item in the register
- the items

The standard information used to describe a register item includes:

- identifier
- name and description of item
- field of application
- lineage and references to related register items

An example of a register item from register of the NASA GCMD (Global Change Master Directory) science keywords is shown below.

```
<grg:RE_RegisterItem uuid="d1e7">
   <grg:itemIdentifier>
      <gco:Integer>7</gco:Integer>
   </grg:itemIdentifier>
   <grg:name>
      <gco:CharacterString>Aquaculture</gco:CharacterString>
   </grg:name>
   <grg:status>
      <grg:RE_ItemStatus>valid</grg:RE_ItemStatus>
   </grg:status>
   <grg:dateAccepted>
      <gco:Date>2006</gco:Date>
   </grg:dateAccepted>
   <grg:definition gco:nilReason="missing"/>
   <grg:itemClass xlink:href="#Item_Class"/>
   <grg:specificationLineage>
      <grg:RE_Reference>
         <grg:itemIdentifierAtSource>
            <gco:CharacterString>5</gco:CharacterString>
         </grg:itemIdentifierAtSource>
         <grg:similarity>
            <grg:RE_SimilarityToSource codeListValue="generalization"
               codeList="http://ww.../lists.xml#RE_SimilarityToSource"/>
         </grg:similarity>
      </grg:RE_Reference>
   </grg:specificationLineage>
</grg:RE_RegisterItem>
```

As mentioned earlier, to use a thesaurus described by an ISO19135 register record, GeoNetwork uses an XSLT called xml_iso19135ToSKOS.xsl (from the convert subdirectory in the iso19135 plugin schema) to extract the following from the ISO19135 register record:

- valid concepts (grg:itemIdentifier, grg:name, grg:status)
- relationships to other concepts (grg:specificationLineage)

- title, version and other management info

This information is used build a SKOS file. The SKOS file is then available for query and management by the sesame/openRDF software used in GeoNetwork.

5.3.6 Creating or Importing a Thesaurus

External and local thesauri are created or imported using the thesaurus manager. You can use the thesaurus manager by:

- logging in as an administrator
- navigating to the 'Administration' page and clicking on the link "Manage thesauri"

The thesaurus manager page will show a list of thesauri that have been created or imported. The upper part of the page provides the user with functions to edit, add, modify or search a thesaurus. The lower part provides a function to upload an external thesaurus in SKOS format.

Creating a local thesaurus

To create a local thesaurus, click the "+" sign on the category you want your thesaurus to be in. Once created, the thesaurus can be updated through the edit interface. The meaning of each column is as follows:

- **Type** - This is an identifier assigned to the thesaurus in GeoNetwork. It is composed of the ISO category to which the thesaurus has been assigned (see the codelist for the gmd:MD_KeywordTypeCode element in http://www.isotc211.org/2005/resources/gmxCodelist.xml), whether the thesaurus is a local, external or register thesaurus and the filename of the SKOS file that holds the thesaurus. (Note: the name of the file used to hold a register thesaurus is the uuid of the ISO19135 register record that describes the thesaurus).
- **Name** - This is the name of the thesaurus which is the administrator on creation or the filename if the thesaurus is ting a thesaurus, the name of the thesaurus will be the filename of the thesaurus.

Figure 5.2: *Administration interface for thesaurus*

For each thesaurus the following buttons are available:

- **Download** - Link to the SKOS RDF file.
- **Delete** - Remove thesaurus from the current node.
- **View** - If type is external, the view button allows to search and view concepts.
- **Edit** - If type is local, the edit button allows to search, add, remove and view concepts.

Import an external thesaurus

GeoNetwork allows thesaurus import in SKOS format. Once uploaded, an external thesaurus cannot be updated. Select the category, browse for the thesaurus file and click upload. The SKOS file will be in `GEONETWORK_DATA_DIR/config/codelist/external/thesauri/<category>`.

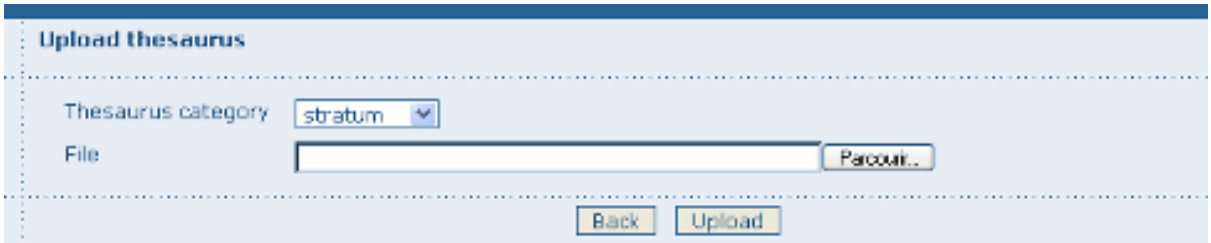


Figure 5.3: *Upload interface for thesaurus*

At the bottom of the page there are the following buttons:

1. *Back*: Go back to the main administration page.
2. *Upload*: Upload the selected RDF file to the node. Then it will list all thesaurus available on the node.

Creating a register thesaurus

An ISO19135 record in the local GeoNetwork catalog can be turned into a SKOS file and used as a thesaurus in GeoNetwork. ISO19135 records not in the local catalog can be harvested from other catalogs (eg. the catalog of the organisation that manages the register). Once the ISO19135 register record is in the local catalog, the process of turning it into a thesaurus for use in the keywords selector begins a search for the record. Having located the record in the search results, one of the actions on the record is to 'Create/Update Thesaurus'.

After selecting this action, you can choose the ISO thesaurus category appropriate for this thesaurus:

After selecting the ISO thesaurus category, the ISO19135 register record is converted to a SKOS file and installed as a thesaurus ready for use in the metadata editor. As described above in the section on ISO19135, only the valid register items are included in the thesaurus. This behaviour and any of the mappings between ISO19135 register items and the SKOS thesaurus file can be changed or inspected by looking at the XSLT xml_iso19135TOSKOS.xsl in the convert subdirectory of the iso19135 schema plugin.

Figure 5.4: *Search results showing ISO19135 record with thesaurus creation action*

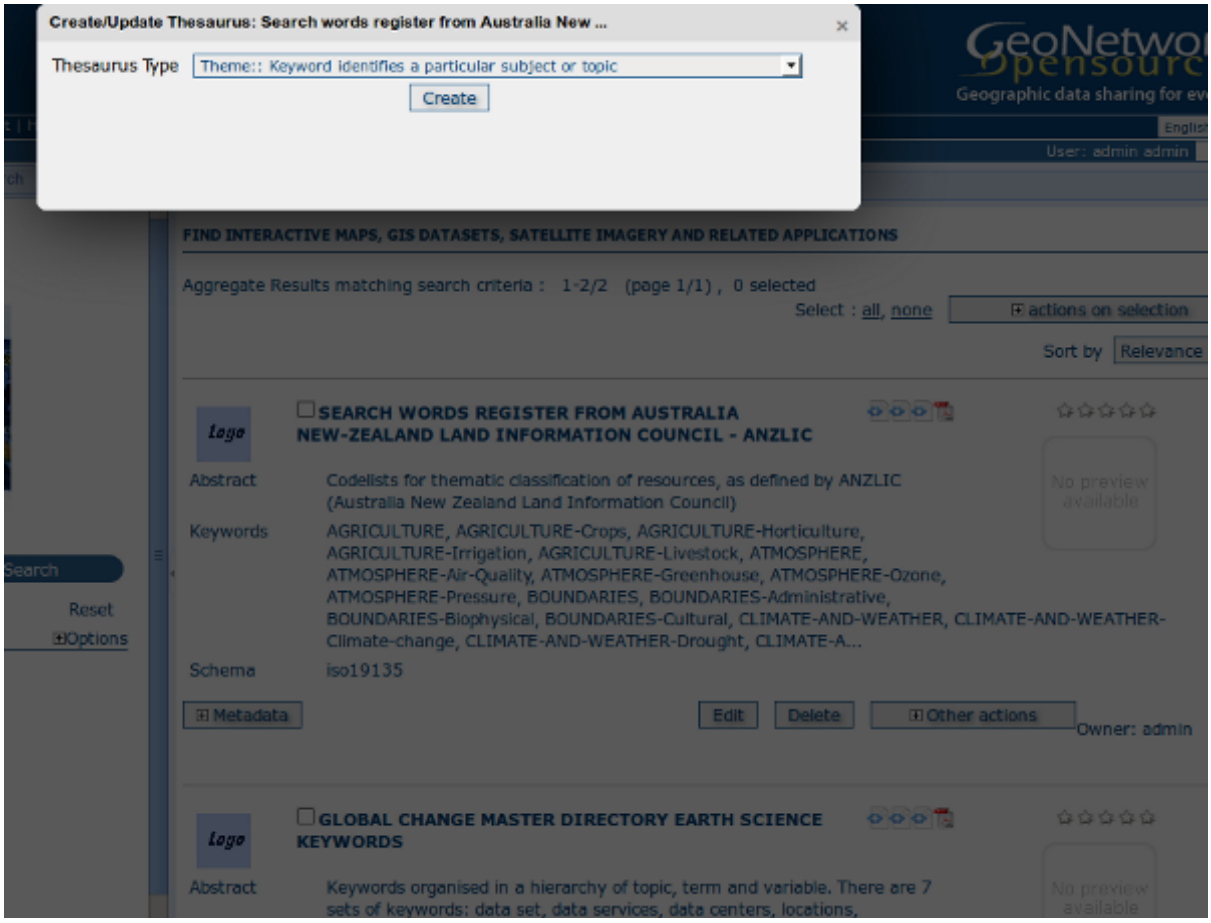


Figure 5.5: *Selecting the ISO thesaurus category when creating a thesaurus*

5.3.7 Editing/browsing a local or external thesaurus: add/remove/browse keywords

From the thesaurus administration interface, click on the edit button for a local thesaurus or the view button for an external thesaurus. This interface allows:

- keywords search
- add/remove keywords for local thesaurus.

Use the textbox and the type of search in order to search for keywords.

Figure 5.6: *Browse interface for thesaurus*

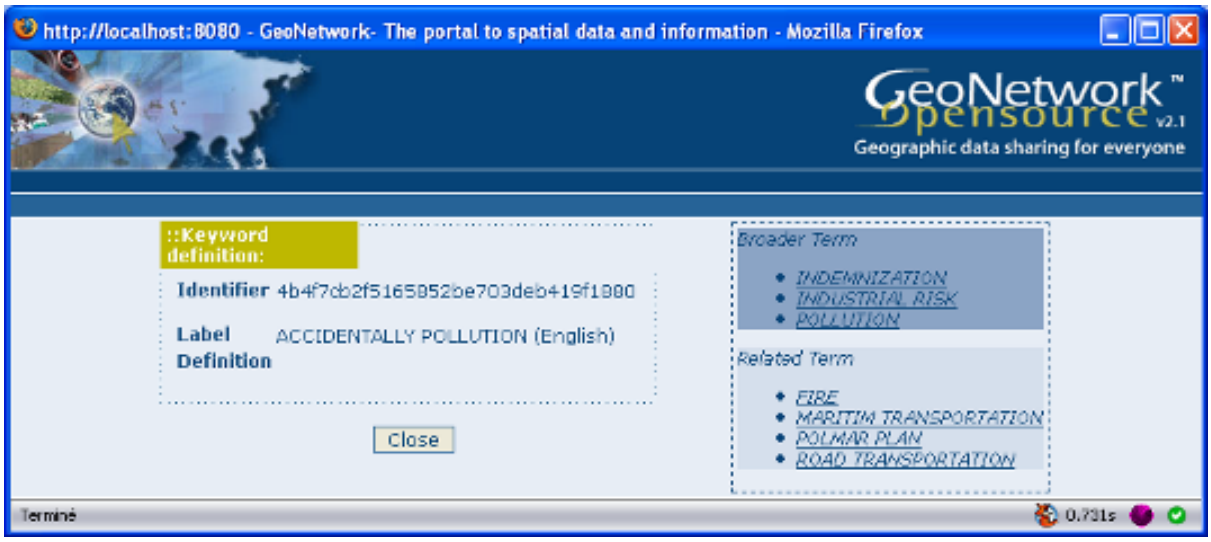


Figure 5.7: *Keyword description*

5.3.8 Editing a register thesaurus

A register thesaurus is created from an ISO19135 metadata record as described above, so a register thesaurus is updated by editing the ISO19135 metadata record and then regenerating the register thesaurus. The ISO19135 metadata record can be created and edited in the GeoNetwork editor.

Preparing to edit an ISO19135 register record

Register records can be very large. For example, a register record describing the ANZLIC Geographic Extent Names register has approx 1800 register items. Each register item holds not only the name of the geographic extent, but also its geographic extent and details of the lineage, relationships to other terms and potentially, the evolution of the extent (changes to name, geographic extent) including the details of changes and why those changes occurred. Editing such a large record in the GeoNetwork editor can cause performance problems for both the browser and the server because the editor constructs an HTML form describing the entire record. Fortunately a much more scaleable approach exists which is based on extracting the register items from the ISO19135 register record and storing them as subtemplates (essentially small metadata records with just the content of the register item). The process for extracting register items from an ISO19135 register record is as follows:

- search for and select the register record
- choose 'Extract register items' from the 'Actions on selected set' menu

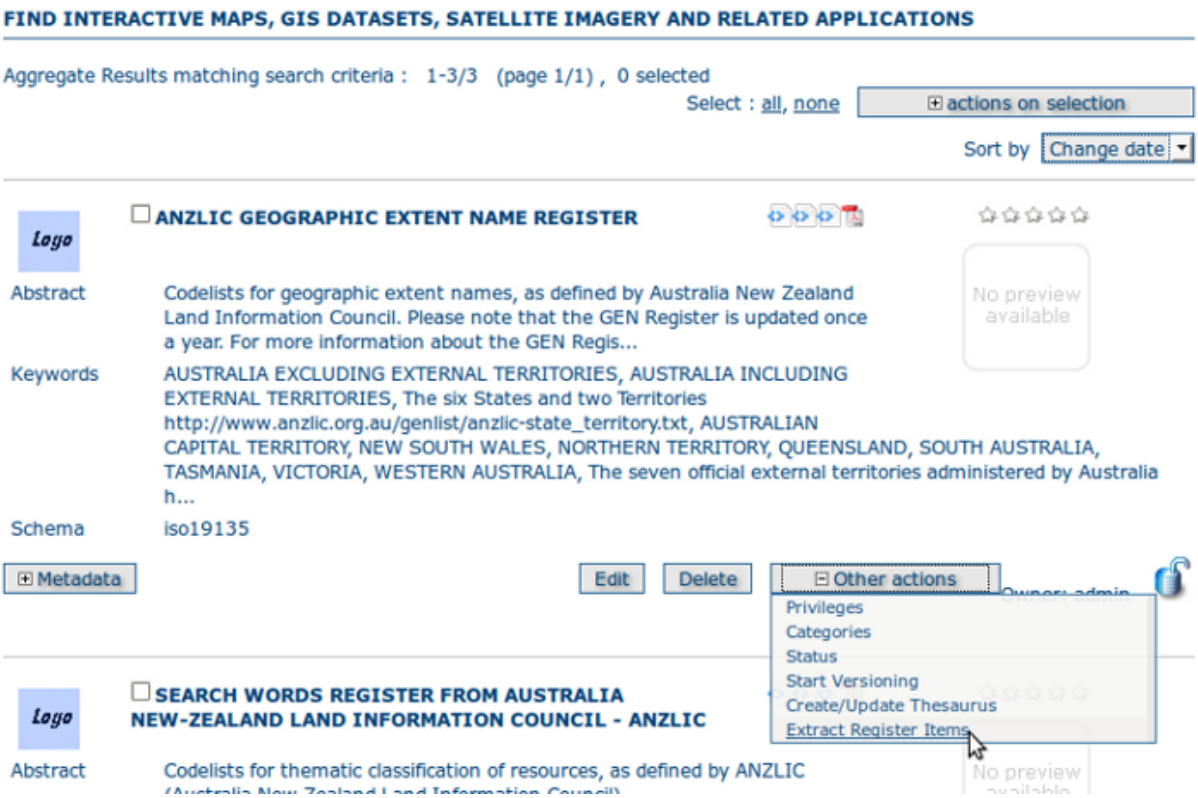
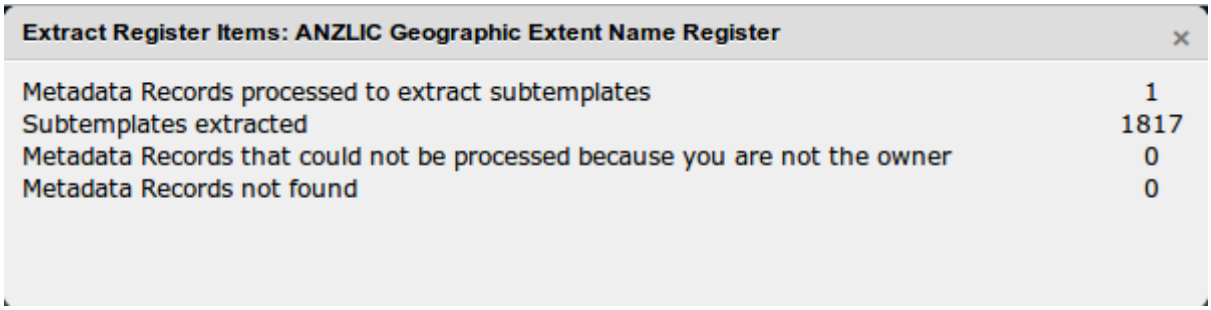


Figure 5.8: *Extracting subtemplates from a register record*

- After the register items have been extracted, you should see a results summary like the following.
- The figure for 'Subtemplates extracted' is the number of register items extracted from the ISO19135 register record.



Editing a register item

To edit/change any of the register items that have been extracted as subtemplates, you can use the Directory management interface. This interface is accessed from the 'Administration' menu, under 'Manage Directories'. In this interface:

- select 'Register Item (GeoNetwork)' as the type of subtemplate to edit as follows.
- enter a search term or just select the search option to return the first 50 register items.
- register items will appear in the left hand side bar, selecting on one will open an editing interface in the right hand panel.

Editing global register information

To edit/change any of the global register information (eg. register owner, manager, version, languages), edit the register record in the normal GeoNetwork metadata editing interface.

5.3.9 Metadata editing: adding keywords

When editing an ISO metadata record, a keyword (or concept) picker can be used which allows the editor to:

- do searches for keywords in one or more thesauri in the catalog (search results are displayed on the left).
- select one or more keywords and add them to the selected items list (using arrows or drag & drop) on the right.
- add the selected keywords directly into metadata, grouping keywords by thesaurus.

The editor can also control how many keywords from searches are displayed in the keyword picker (default is 50).

Notice that a URL pointing to the source thesaurus is included in the Thesaurus Name citation (the actual element used for this is gmd:otherCitationDetails/gmx:FileName). The thesaurus can be downloaded as a SKOS file if it is a local or external thesaurus. For register thesauri the URL refers to the ISO19135 register record from which the thesaurus was created.

5.3.10 Search criteria: keywords

You can search on keywords in the advanced search interface. To help select a keyword you can click in the keyword search field to bring up a list of all the keywords that have been used in the metadata

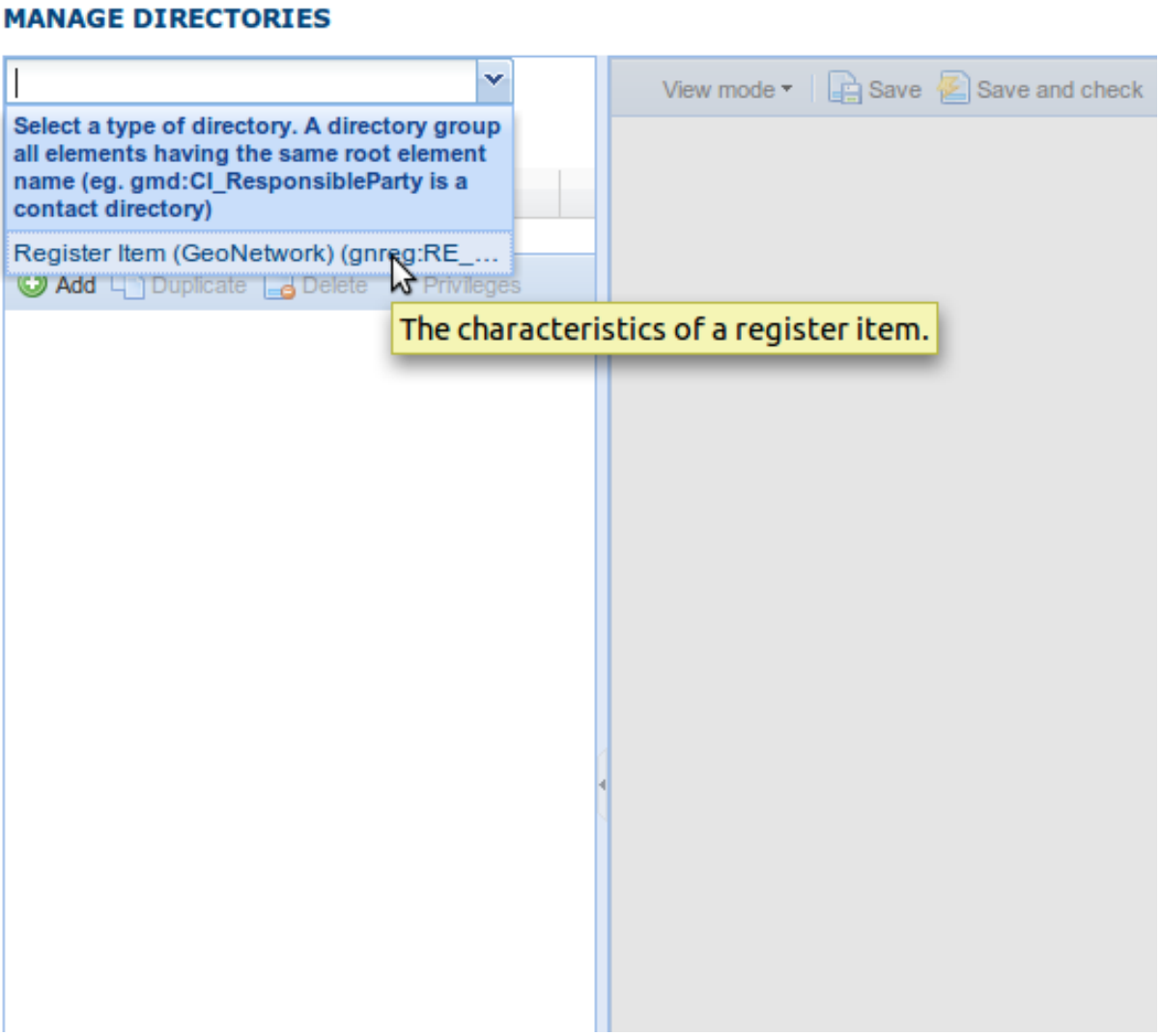


Figure 5.9: *Managing a Directory of subtemplates, selecting 'Register Item' subtemplates*

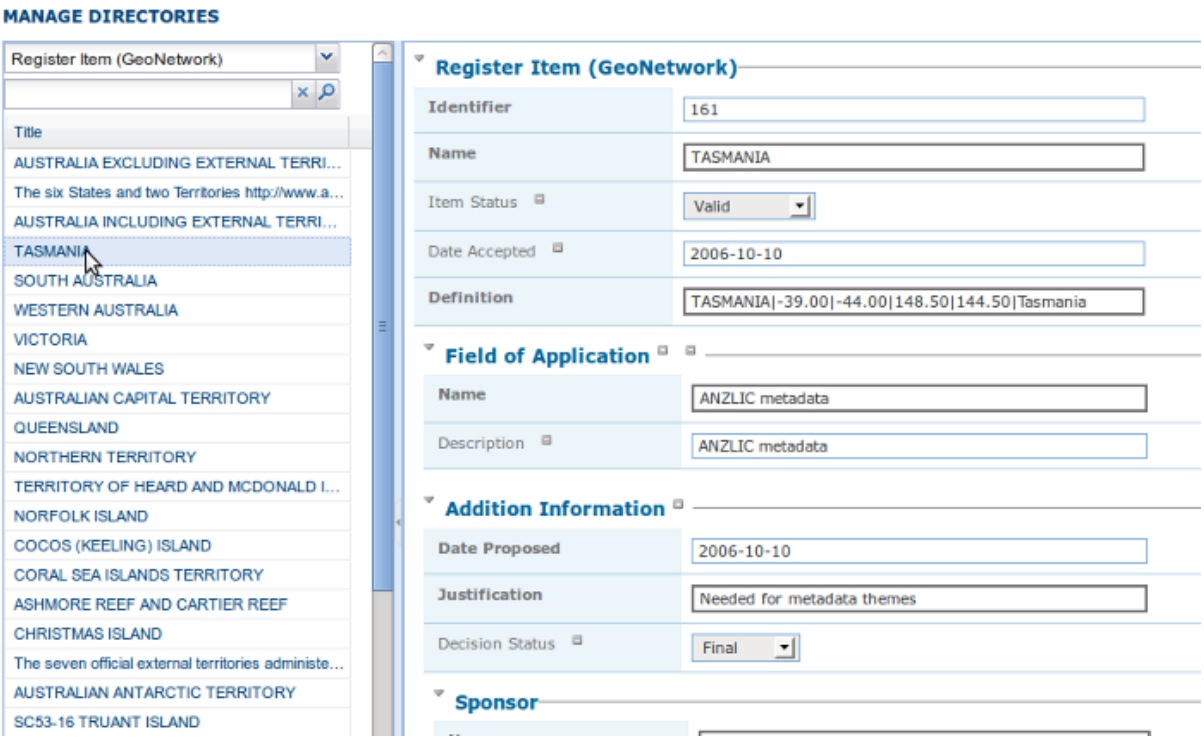


Figure 5.10: *Managing a Directory of subtemplates, opening a Register Item for editing*

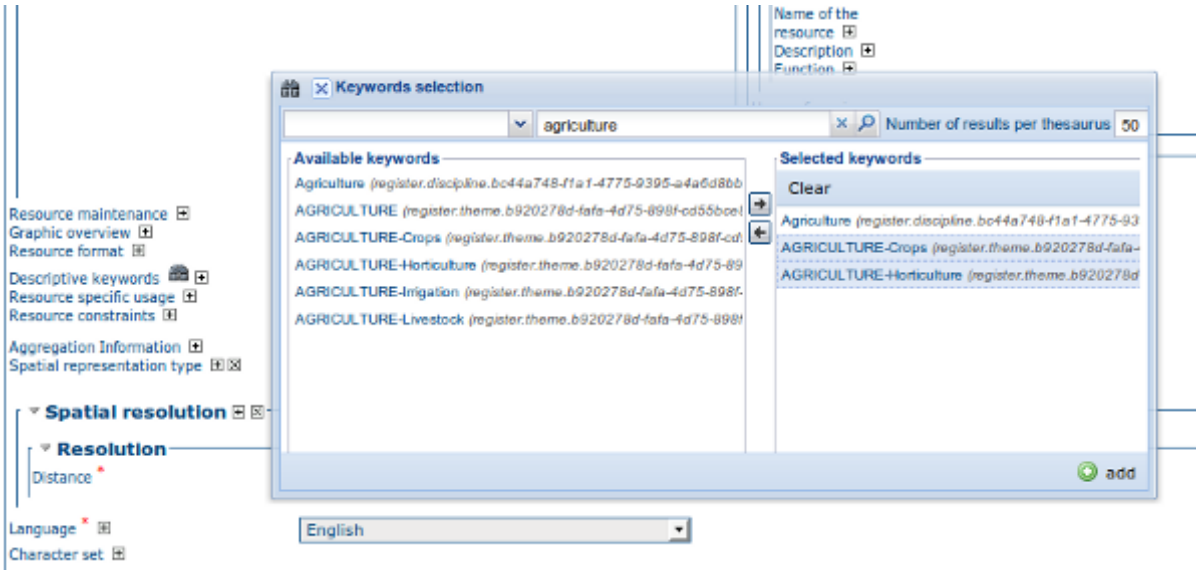


Figure 5.11: *Keyword selection interface (editing mode)*

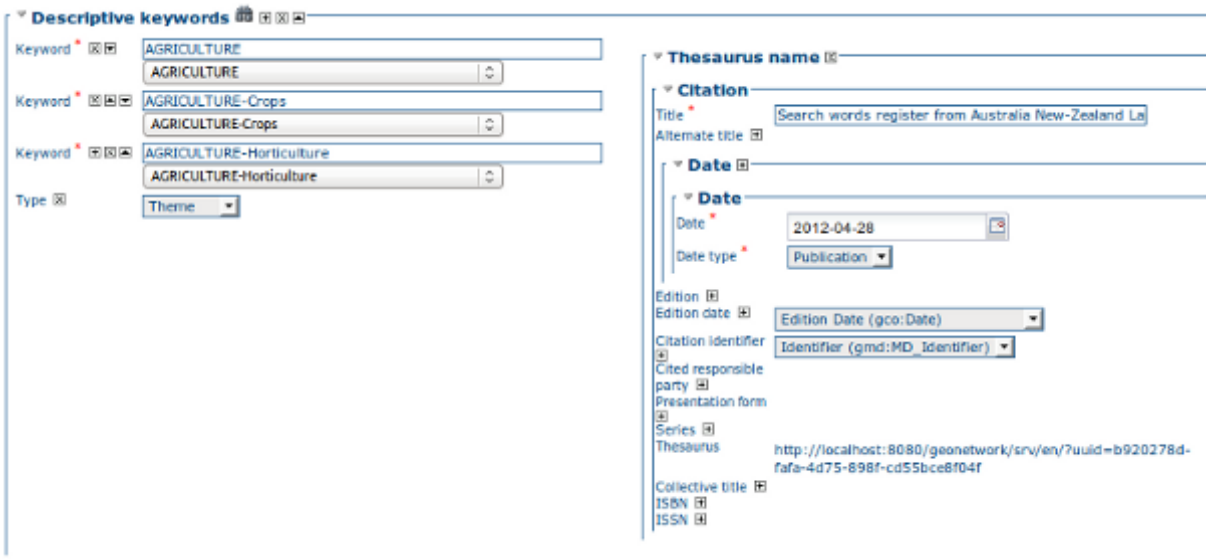


Figure 5.12: *Keywords in Metadata Record (editing mode)*

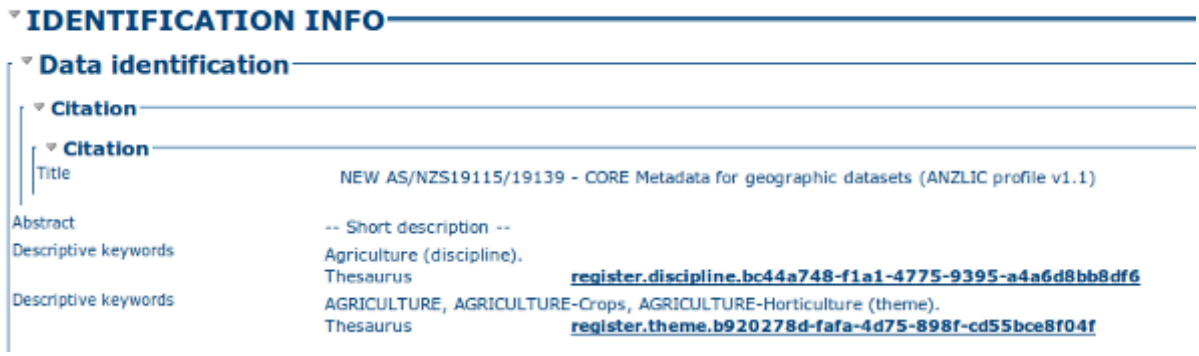


Figure 5.13: *Keywords in Metadata Record (view mode)*

records in this catalog. These keywords are indexed by Lucene on creation/update of metadata. Each keyword in the list has the number of records that use the keyword displayed next to it.

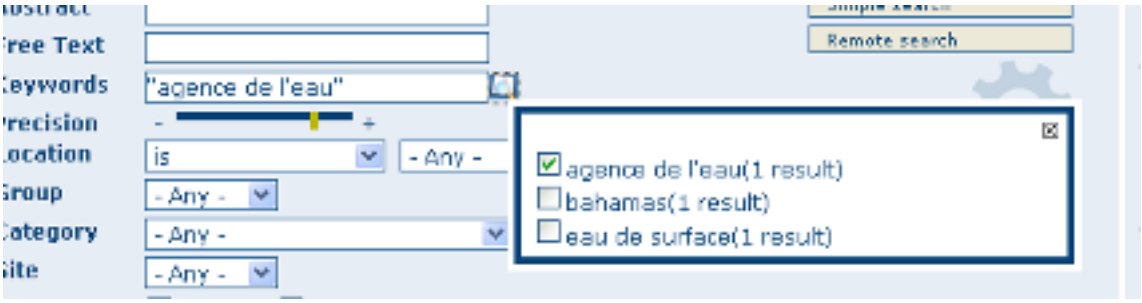


Figure 5.14: *Thesaurus search interface*

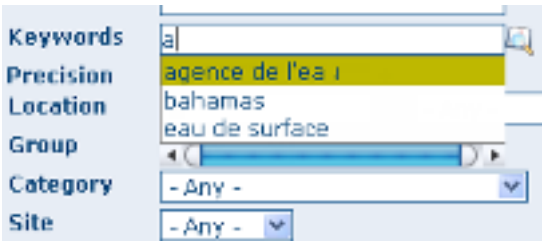


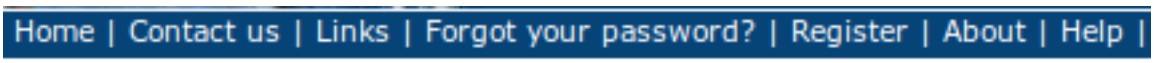
Figure 5.15: *Auto-complete function in thesaurus search interface*

If an XML element named keyword-select-panel is present as a child of the search element in the *config-gui.xml* file (in the WEB-INF directory), then search for keyword using the keyword selection panel is available as in the metadata editor:

```
<search>
 <!-- Display or not keyword selection panel in advanced search panel
  <keyword-selection-panel/>
 -->
</search>
```

5.4 User Self-Registration Functions

To enable the self-registration functions, see *config_user_self_registration* section of this manual. When self-registration is enabled, the banner menu functions shown to a user who has not logged in should contain two additional choices: 'Forgot your password?' and 'Register' as follows:



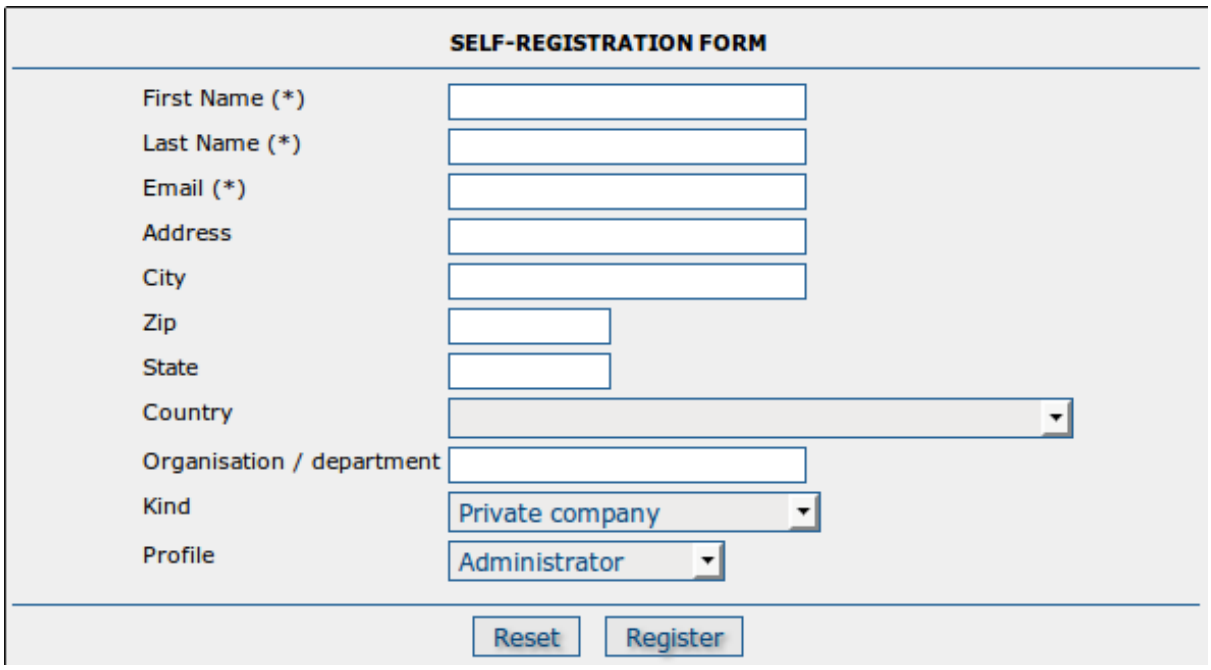
If 'Register' is chosen the user will be asked to fill out a form as follows:

The fields in this form are self-explanatory except for the following:

Email: The user's email address. This is mandatory and will be used as the username.

Profile: By default, self-registered users are given the 'Registered User' profile (see previous section). If any other profile is selected:

- the user will still be given the 'Registered User' profile



- an email will be sent to the Email address nominated in the Feedback section of the 'System Administration' menu, informing them of the request for a more privileged profile

5.4.1 What happens when a user self-registers?

When a user self-registration occurs, the user receives an email with the new account details that looks something like the following:

```
Dear User,

  Your registration at The Greenhouse GeoNetwork Site was successful.

  Your account is:
  username :    dubya.shrub@greenhouse.gov
  password :    011003
  usergroup:    GUEST
  usertype :    REGISTEREDUSER

  You've told us that you want to be "Editor", you will be contacted by our office soon.

  To log in and access your account, please click on the link below.
  http://greenhouse.gov/geonetwork

  Thanks for your registration.

        Yours sincerely,
        The team at The Greenhouse GeoNetwork Site
```

Notice that the user has requested an 'Editor' profile. As a result an email will be sent to the Email address nominated in the Feedback section of the 'System Administration' menu which looks something like the following:

Notice also that the user has been added to the built-in user group 'GUEST'. This is a security restriction. An administrator/user-administrator can add the user to other groups if that is required later.

If you want to change the content of this email, you should modify INSTALL_DIR/web/geonetwork/xsl/registration-pwd-email.xsl.

```
Dear Admin,

        Newly registered user dubya.shrub@greenhouse.gov has requested "Editor" access f

        Instance:     The Greenhouse GeoNetwork Site
        Url:          http://greenhouse.gov/geonetwork

        User registration details:

        Name:         Dubya
        Surname:      Shrub
        Email:        dubya.shrub@greenhouse.gov
        Organisation: The Greenhouse
        Type:         gov
        Address:      146 Main Avenue, Creationville
        State:        Clerical
        Post Code:    92373
        Country:      Mythical

Please action.

The Greenhouse GeoNetwork Site
```

If you want to change the content of this email, you should modify INSTALL_DIR/web/geonetwork/xsl/registration-prof-email.xsl.

5.4.2 The 'Forgot your password?' function

This function allows users who have forgotten their password to request a new one. For security reasons, only users that have the 'Registered User' profile can request a new password.

If a user takes this option they will receive an email inviting them to change their password as follows:

```
You have requested to change your Greenhouse GeoNetwork Site password.

You can change your password using the following link:

http://localhost:8080/geonetwork/srv/en/password.change.form?username=dubya.shrub@greenh

This link is valid for today only.

Greenhouse GeoNetwork Site
```

GeoNetwork has generated a changeKey from the forgotten password and the current date and emailed that to the user as part of a link to a change password form.

If you want to change the content of this email, you should modify INSTALL_DIR/web/geonetwork/xsl/password-forgotten-email.xsl.

When the user clicks on the link, a change password form is displayed in their browser and a new password can be entered. When that form is submitted to GeoNetwork, the changeKey is regenerated and checked with the changeKey supplied in the link, if they match then the password is changed to the new password supplied by the user.

The final step in this process is a verification email sent to the email address of the user confirming that a change of password has taken place:

```
Your Greenhouse GeoNetwork Site password has been changed.

If you did not change this password contact the Greenhouse GeoNetwork Site helpdesk

The Greenhouse GeoNetwork Site team
```

If you want to change the content of this email, you should modify INSTALL_DIR/web/geonetwork/xsl/password-changed-email.xsl.

CHAPTER 6

Glossary of Metadata Fields Description

This glossary provides you with brief descriptions of the minimum set of metadata fields required to properly describe a geographical data as well as some optional elements highly suggested for a more extensive standard description.

Access constraints Access constraints applied to assure the protection of privacy or intellectual property, and any special restrictions or limitations on obtaining the resource

Abstract Brief narrative summary of the content of the resource(s)

Administrative area State, province of the location

Temporal extent - Begin date Formatted as 2007-09-12T15:00:00 (YYYY-MM-DDTHH:mm:ss)

Character set Full name of the character coding standard used for the metadata set

Grid spatial representation - Cell geometry Identification of grid data as point or cell

City City of the location

Reference System Info - Code Alphanumeric value identifying an instance in the namespace

Country Country of the physical address

Data quality info Provides overall assessment of quality of a resource(s)

Date Reference date and event used to describe it (YYYY-MM-DD)

Date stamp Date that the metadata was created (YYYY-MM-DDThh:mm:ss)

Date type Event used for reference date

Delivery point Address line for the location (as described in ISO 11180, annex A)

Equivalent scale - Denominator The number below the line in a vulgar fraction

Data Quality - Description Description of the event, including related parameters or tolerances

OnLine resource - Description Detailed text description of what the online resource is/does

Descriptive keywords Provides category keywords, their type, and reference source

Grid spatial representation - Dimension name Name of the axis i.e. row, column

Grid spatial representation - Dimension size Number of elements along the axis

Dimension size Resolution Number of elements along the axis

Distribution info Provides information about the distributor of and options for obtaining the resource(s)

Geographic bounding box - East bound longitude Eastern-most coordinate of the limit of the dataset extent, expressed in longitude in decimal degrees (positive east)

Edition Version of the cited resource

Electronic mail address Address of the electronic mailbox of the responsible organisation or individual

Temporal extent - End date Formatted as 2007-09-12T15:00:00 (YYYY-MM-DDTHH:mm:ss)

Equivalent scale Level of detail expressed as the scale of a comparable hardcopy map or chart

Extent Information about spatial, vertical, and temporal extent

Facsimile Telephone number of a facsimile machine for the responsible organisation or individual

File identifier Unique identifier for this metadata file

Vector spatial representation - Geometric object type Name of point and vector spatial objects used to locate zero-, one-and two-dimensional spatial locations in the dataset

Vector spatial representation - Geometric object count Total number of the point or vector object type occurring in the dataset

Geographic bounding box Geographic position of the dataset

Grid spatial representation Information about grid spatial objects in the dataset

Grid spatial representation - Resolution value Degree of detail in the grid dataset

Grid spatial representation - Transformation parameter availability Indication of whether or not parameters for transformation exists

Data Quality - Hierarchy level Hierarchical level of the data specified by the scope

Identification info Basic information about the resource(s) to which the metadata applies

Point of Contact - Individual name Name of the responsible person- surname, given name, title separated by a delimiter

Keyword Commonly used word(s) or formalised word(s) or phrase(s) used to describe the subject

Data Language Language used for documenting data

Metadata - Language Language used for documenting metadata

Data Quality - Lineage Non-quantitative quality information about the lineage of the data specified by the scope. Mandatory if report not provided

OnLine resource - Linkage Location (address) for on-line access using a Uniform Resource Locator address or similar addressing scheme such as http://www.statkart.no/isotc211

Maintenance and update frequency Frequency with which changes and additions are made to the resource after the initial resource is completed

Metadata author Party responsible for the metadata information

Metadata standard name Name of the metadata standard (including profile name) used

Metadata standard version Version (profile) of the metadata standard used

OnLine resource - Name Name of the online resource

Geographic bounding box - North bound latitude Northern-most, coordinate of the limit of the dataset extent expressed in latitude in decimal degrees (positive north)

Grid spatial representation - Number of dimensions Number of independent spatial-temporal axes

Distribution Info - OnLine resource Information about online sources from which the resource can be obtained

Point of Contact - Organisation name Name of the responsible organisation

Other constraints Other restrictions and legal prerequisites for accessing and using the resource

Point of contact Identification of, and means of communication with, person(s) and organisations(s) associated with the resource(s)

Point of contact - Position name Role or position of the responsible person

Postal code ZIP or other postal code

Presentation form Mode in which the resource is represented

OnLine resource - Protocol Connection protocol to be used

Purpose Summary of the intentions with which the resource(s) was developed

Reference system info Description of the spatial and temporal reference systems used in the dataset-Data

Data Quality - Report Quantitative quality information for the data specified by the scope. Mandatory if lineage not provided

Grid spatial representation - Resolution value Degree of detail in the grid dataset

Point of contact - Role Function performed by the responsible party

Geographic bounding box - South bound latitude Southern-most coordinate of the limit of the dataset extent, expressed in latitude in decimal degrees (positive north)

Spatial representation info Digital representation of spatial information in the dataset

Spatial representation type Method used to spatially represent geographic information

Data Quality - Statement General explanation of the data producer's knowledge about the lineage of a dataset

Status Status of the resource(s)

Supplemental Information Any other descriptive information about the dataset

Temporal Extent Time period covered by the content of the dataset

Title Name by which the cited resource is known

Topic category code High-level geographic data thematic classification to assist in the grouping and search of available geographic data sets. Can be used to group keywords as well. Listed examples are not exhaustive. NOTE It is understood there are overlaps between general categories and the user is encouraged to select the one most appropriate.

Grid spatial representation - Transformation parameter availability Indication of whether or not parameters for transformation exists

Vector spatial representation - Topology level Code which identifies the degree of complexity of the spatial relationships

Type Subject matter used to group similar keywords

URL Unified Resource Locator

Use constraints Constraints applied to assure the protection of privacy or intellectual property, and any special restrictions or limitations or warnings on using the resource

Vector spatial representation Information about the vector spatial objects in the dataset

Voice Telephone number by which individuals can speak to the responsible organisation or individual

Geographic bounding box - West bound longitude Western-most coordinate of the limit of the dataset extent, expressed in longitude in decimal degrees (positive east)

CHAPTER 7

ISO Topic Categories

Iso Topic Categories and Keywords

Isotopic Category	Main Topic	Examples	Keywords
Base Maps	Base Maps, Scanned Maps, and Charts		Base Map
Biota	Biologic and Ecologic Information Flora and/or fauna in natural environment	wildlife, vegetation, biological sciences, ecology, wilderness, sea life, wetlands, habitat	
Boundaries	Administrative Legal land descriptions and Political Boundaries	political and administrative boundaries	Administrative boundaries,
Climatology Meteorology Atmosphere	cloud cover, weather, climate, atmospheric conditions, climate change, precipitation		NDVI, Drought, Floods
Earth Cover	Earth Surface Characteristics and Land Cover		Land Cover
Economy	Business and Economic Information, Economic activities, conditions and employment	production, labour, revenue, commerce, industry, tourism and ecotourism, forestry, fisheries, commercial or subsistence hunting, exploration and exploitation of resources such as minerals, oil and gas	
Elevation	Elevation and Derived Products, Height above or below sea level	altitude, bathymetry, digital elevation models, slope, derived products	Digital Elevation Model
Environment	Environmental Monitoring and Modelling, Environmental resources, protection and conservation	environmental pollution, waste storage and treatment, environmental impact assessment, monitoring environmental risk, nature reserves, landscape	
Farming	Agriculture and Farming Rearing of animals and/or cultivation of plants	agriculture, irrigation, aquaculture, plantations, herding, pests and diseases affecting crops and livestock	Agriculture, Crop Production, Livestock
Geoscientific Information	Geologic and Geophysical Information, Information pertaining to earth sciences	geophysical features and processes, geology, minerals, sciences dealing with the composition, structure and origin of the earth's rocks, risks of earthquakes, volcanic activity, landslides, gravity	

		information, soils, permafrost, hydrogeology, erosion	
Health	Human Health and Health, health services, human ecology, and Disease safety	disease and illness, factors affecting health, hygiene, substance abuse, mental and physical health, health services	Malnutrition, Wasting, Stunting, Underweight, Food Deficit, Crop Disease, Livestock Disease,
Imagery	Images and Photographs		
Imagery Base Maps Earth Cover	Base maps	land cover, topographic maps, imagery, unclassified images, annotations	
Inland Waters	Inland Water Resources and Characteristics, Inland water features, drainage systems and their characteristics	rivers and glaciers, salt lakes, water utilisation plans, dams, currents, floods, water quality, hydrographic charts	Rivers,
Intelligence Military	Military bases, structures, activities	barracks, training grounds, military transportation, information collection	
Location	Geodetic Networks Positional information and services and Control Points	addresses, geodetic networks, control points, postal zones and services, place names	Cartography
Oceans	Ocean and Estuarine Resources and Characteristics(excluding inland waters), Features and characteristics of salt water bodies	tides, tidal waves, coastal information, reefs	
Planning Cadastre	Cadastral and Legal Land Descriptions. Information used for appropriate actions for future use of the land	land use maps, zoning maps, cadastral surveys, land ownership	
Recreation	Tourism and Recreation		
Society	Society and Cultural and Demographic Information, Characteristics of society and cultures	Settlements, anthropology, archaeology, education, traditional beliefs, manners and customs, demographic data, recreational areas and activities, social impact assessments, crime and justice, census information	Vulnerability, Early Warning, Emergency, IDPS, Refugees, Population, Poverty, Food Security, Regional Bureaux, WFP Facilities, School feeding
Structure	Facilities, Man-made construction, Buildings and Structures	buildings, museums, churches, factories, housing, monuments, shops, towers	
Transportation	Transportation Means and aids for conveying persons and/or goods, Networks and Models	roads, airports/airstrips, shipping routes, tunnels, nautical charts, vehicle or vessel location, aeronautical	Infrastructure, COMPAS, Food Aid, Food Beneficiaries, Railways, Roads, Shipments

		charts, railways	
Utilities Communication	Utility Distribution Networks, Energy, water and waste systems and communications infrastructure and services	hydroelectricity, geothermal, solar and nuclear sources of energy, water purification and distribution, sewage collection and disposal, electricity and gas distribution, data communication, telecommunication, radio, communication networks	

CHAPTER 8

Free and Open Source Software for Geospatial Information Systems

A range of related software packages can be used in addition to GeoNetwork opensource to deploy a full Spatial Data Infrastructure. These include Web Map Server software, GIS desktop applications and Web Map Viewers.

Below you will find some examples of open source software available for each categories.

8.1 Web Map Server software

- GeoServer (All)
- MapServer (All)
- MapGuide Open Source (Windows & Linux)
- Deegree (All)

8.2 GIS Desktop software

- GRASS (All)
- gvSIG (All)
- uDig (All)
- Quantum GIS (All)
- OSSIM (Windows & OSX)

8.3 Web Map Viewer and Map Server Management

- OpenLayers (All)
- MapBender (All)

Note: All = The Windows, Linux and Mac OS X operating systems.

CHAPTER 9

Frequently Asked Questions

This is a list of frequently encountered problems, suggestions that help to find the cause of the problem and possible solutions. The list is by no means exhaustive. Feel free to contribute by submitting new problems and their solutions to the developer mailing list.

Note: <install directory> is a placeholder for the GeoNetwork web application directory (eg. <your_tomcat>/webapps/geonetwork or <your_jetty>/web/geonetwork). <some file> should be read as some random file name.

Warning: Be very careful when issuing commands on the terminal! You can easily damage your operating system with no way back. If you are not familiar with using the terminal: **don't do it, contact an expert instead!** Make a backup of your data before you make any of the suggested changes below!

9.1 HTTP Status 400 Bad request

Check the availability and write permissions of the data and tmp directories.

See *The data/tmp directory* and *What/Where is the GeoNetwork data directory?*.

9.2 Metadata insert fails

Inserting an XML or MEF file through the Metadata insert form fails silently. Verify if the data directory is available and writable.

See *The data/tmp directory* and *What/Where is the GeoNetwork data directory?*.

9.3 Thumbnail insert fails

Nothing happens when inserting a thumbnail through the wizard in the metadata editor.

Error in your log file looks like:

```
HTTP Status 400 - Cannot build ServiceRequest Cause : <install directory>/data/tmp/<some
```

Then check *The data/tmp directory* and *What/Where is the GeoNetwork data directory?*.

9.4 The data/tmp directory

This directory is used as a staging area for file uploads and image/thumbnail operations. On Linux or OS X systems verify from a terminal if the `<install directory>/data/tmp` directory exists and is writable.

ls -la <install directory>/data

This should show the permissions on the data directory. For example, if you are running tomcat:

```
total 0
drwxr-xr-x   6 tomcat  tomcat  204 19 jan 15:34 .
drwxr-xr-x   8 tomcat  tomcat  272 23 dec 19:30 ..
drwxr-xr-x   3 tomcat  tomcat  102 19 jan 15:47 tmp
```

The above example shows that only the user tomcat has write access on the directories listed. All other users have read (and execute) rights only. See http://en.wikipedia.org/wiki/Filesystem_permissions for more details on file permissions.

Make sure your web server is running as user tomcat. Check this with the command:

ps aux | grep tomcat

You should see the processes that have tomcat in their description. Something like this:

```
bash-3.2# ps aux | grep tomcat
tomcat     22253   0,7  0,0  2435120    532 s000  S+    5:03pm   0:00.00 grep tomcat
tomcat     22251   0,0  1,9  2861960  80596 s000  S     5:03pm   0:03.85 /System/Library
```

If all is well, the user referred to at the start of this string (in this case tomcat) is the same user that has write permissions on the data and tmp directories.

You now have two possible solutions:

- Make the data and temporary directories writable to all users. You can change this using the command:

chmod -R a+w <install directory>/data

Your permissions should now look like this:

```
drwxrwxrwx   6 tomcat  tomcat  204 19 jan 15:34 .
etc..
```

Note: the 'w' refers to 'write' access

- The second solution is to ensure the user running the webserver is the same user that holds write access to the data directory (in this case tomcat). For this, you can (a) change the user running the process, or (b) change ownership of the directory using the chown command:

chown -R tomcat:tomcat <install directory>/data

9.5 What/Where is the GeoNetwork data directory?

At GeoNetwork 2.8:

- metadata data (files uploaded with the metadata and thumbnails)
- the Lucene index
- plugin configurations (schema plugins, thesauri etc)

have been moved into a single directory. By default, this directory is `<install directory>/WEB-INF/data`, but it can be located on any filesystem accessible to the GeoNetwork server and the different subdirectories can even be placed in different directories. See *GeoNetwork data directory* for more details. For the purposes of this FAQ, we'll assume that the GeoNetwork data directory is `<install directory>/WEB-INF/data` because the same principles apply no matter where the data directory is located.

Check that the user running your webserver (eg. tomcat) has permissions over this directory.

ls -la <install directory>/WEB-INF/data

Your should see something like the following:

```
total 0
drwxr-xr-x   5 tomcat tomcat  170 Jan  8 01:17 .
drwxr-xr-x  48 tomcat tomcat 1632 Jan  8 01:17 ..
drwxr-xr-x   5 tomcat tomcat  170 Jan  8 01:17 config
drwxr-xr-x   5 tomcat tomcat  170 Jan  8 01:17 data
drwxr-xr-x   9 tomcat tomcat  306 Jan  8 10:04 index
```

If all is well, then the tomcat user will have write permissions on all sub directories.

If not then you should ensure that the user running the webserver is the same user that holds write access to the GeoNetwork data directory (in this case tomcat). For this, you can (a) change the user running the process, or (b) change ownership of the directory using the chown command:

chown -R tomcat:tomcat <install directory>/WEB-INF/data

9.6 The base maps are not visible

GeoServer may not have started properly. Confirm this by trying to connect to http://<yourdomain>:8080/geoserver (on your local machine this is http://localhost:8080/geoserver)

9.6.1 Native JAI error on Jetty

Error in output.log:

```
sun.misc.ServiceConfigurationError: javax.imageio.spi.ImageOutputStreamSpi: Provider com
```

Jetty by default ships with a classloader that does not conform to the Java classloading model: you'll notice because Geoserver will fail all (*JAI*) usage attempt with a "sealing violation" exception. It can be restored to standard behaviour locating the etc/jetty-webapps.xml configuration file and changing the web app context configuration to look like the following:

```
      <Configure id="Server" class="org.eclipse.jetty.server.Server">
  <Ref id="DeploymentManager">
        <Call id="webappprovider" name="addAppProvider">
          <Arg>
            <New class="org.eclipse.jetty.deploy.providers.WebAppProvider">
              <Set name="monitoredDir"><Property name="jetty.home" default="." />/../web
              <Set name="defaultsDescriptor"><Property name="jetty.home" default="."/>/e
              <Set name="scanInterval">1</Set>
              <Set name="contextXmlDir"><Property name="jetty.home" default="." />/conte
              <Set name="extractWars">true</Set>

              <!-- uncomment in case of a JAI usage attempt with a "sealing violation" e
              <Set name="parentLoaderPriority">true</Set>

            </New>
          </Arg>
        </Call>
  </Ref>
</Configure>
```

Note: The important line is the one where the **parentLoaderPriority** property is set to **true**

CHAPTER 10

Glossary

ebRIM Enterprise Business Registry Information Model..

CSW Catalog Service for the Web. The OGC Catalog Service defines common interfaces to discover, browse, and query metadata about data, services, and other potential resources.

ISO International Standards Organisation is an international-standard-setting body composed of representatives from various national standards organizations. http://www.iso.org

ISO TC211 ISO/TC 211 is a standard technical committee formed within ISO, tasked with covering the areas of digital geographic information (such as used by geographic information systems) and geomatics. It is responsible for preparation of a series of International Standards and Technical Specifications numbered in the range starting at 19101.

GeoNetwork GeoNetwork opensource is a standards based, Free and Open Source catalog application to manage spatially referenced resources through the web. http://geonetwork-opensource.org

GeoServer GeoServer is an open source software server written in Java that allows users to share and edit geospatial data. Designed for interoperability, it publishes data from any major spatial data source using open standards.

GPL The GNU General Public License is a free, copyleft license for software and other kinds of works. GeoNetwork opensource is released under the GPL 2 license. http://www.gnu.org/licenses/old-licenses/gpl-2.0.html

Creative Commons GeoNetwork documentation is released under the Creative Commons Attribution-ShareAlike 3.0 Unported License. Find more information at http://creativecommons.org/licenses/by-sa/3.0/

XML Extensible Markup Language is a general-purpose specification for creating custom markup languages.

XSD XML Schema, published as a W3C recommendation in May 2001, is one of several XML schema languages. http://en.wikipedia.org/wiki/XSD

ebXML Enterprise Business XML.

DAO Data Access Object.

CRUD Create Read Update and Delete.

DB (or DBMS) A database management system (DBMS) is computer software that manages databases. DBMSes may use any of a variety of database models, such as the network model

or relational model. In large systems, a DBMS allows users and other software to store and retrieve data in a structured way.

SOA Service Oriented Architecture provides methods for systems development and integration where systems package functionality as interoperable services. A SOA infrastructure allows different applications to exchange data with one another.

FGDC The Federal Geographic Data Committee (FGDC) is an interagency committee that promotes the coordinated development, use, sharing, and dissemination of geospatial data on a national basis in the USA. See http://www.fgdc.gov

JMS Java Messaging Service.

TDD Test Driven Development.

JIBX Binding XML to Java Code.

HQL Hibernate Query Language.

OO Object Oriented.

EJB Enterprise Java Beans.

SOAP Simple Object Access Protocol is a protocol specification for exchanging structured information in the implementation of Web Services in computer networks.

OGC Open Geospatial Consortium. A standards organization for geospatial information systems http://www.opengeospatial.org

OSGeo The Open Source Geospatial Foundation (OSGeo), is a non-profit non-governmental organization whose mission is to support and promote the collaborative development of open geospatial technologies and data. http://www.osgeo.org

FAO Food and Agriculture Organisation of the United Nations is a specialised agency of the United Nations that leads international efforts to defeat hunger. http://www.fao.org

WFP World Food Programme of the United Nations is the food aid branch of the United Nations, and the world's largest humanitarian organization. http://www.wfp.org

UNEP The UN Environment Programme (UNEP) coordinates United Nations environmental activities, assisting developing countries in implementing environmentally sound policies and encourages sustainable development through sound environmental practices. http://www.unep.org

OCHA United Nations Office for the Coordination of Humanitarian Affairs is designed to strengthen the UN's response to complex emergencies and natural disasters. http://ochaonline.un.org/

URL A Uniform Resource Locator specifies where an identified resource is available and the mechanism for retrieving it.

GAST GeoNetwork Administrator Survival Tool. A desktop application that allows administrators of a GeoNetwork catalog to perform simple database configuration using a GUI.

WebDAV Web-based Distributed Authoring and Versioning. WebDAV is a set of extensions to the Hypertext Transfer Protocol (HTTP) that allows users to edit and manage files collaboratively on remote World Wide Web servers.

OAI-PMH Open Archive Initiative Protocol for Metadata Harvesting. It is a protocol developed by the Open Archives Initiative. It is used to harvest (or collect) the metadata descriptions of the records in an archive so that services can be built using metadata from many archives.

WMS Web Map Service is a standard protocol for serving georeferenced map images over the Internet that are generated by a map server using data from a GIS database. The specification was developed and first published by the Open Geospatial Consortium in 1999.

WFS Web Feature Service provides an interface allowing requests for geographical features across the web using platform-independent calls. One can think of geographical features as the "source code" behind a map.

WCS Web Coverage Service provides an interface allowing requests for geographical coverages across the web using platform-independent calls. The coverages are objects (or images) in a geographical area

WPS Web Processing Service is designed to standardize the way that GIS calculations are made available to the Internet. WPS can describe any calculation (i.e. process) including all of its inputs and outputs, and trigger its execution as a Web Service.

UUID A Universally Unique Identifier (UUID) is an identifier standard used in software construction, standardized by the Open Software Foundation (OSF) as part of the Distributed Computing Environment (DCE).

MAC address Media Access Control address (MAC address) is a unique identifier assigned to most network adapters or network interface cards (NICs) by the manufacturer for identification, and used in the Media Access Control protocol sublayer. See also http://en.wikipedia.org/wiki/MAC_address on Wikipedia

MEF Metadata Exchange Format. An export format developed by the GeoNetwork community. More details can be found in this manual in Chapter Metadata Exchange Format.

SKOS The Simple Knowledge Organisation Systems (SKOS) is an area of work developing specifications and standards to support the use of knowledge organisation systems (KOS) such as thesauri, classification schemes. http://www.w3.org/2004/02/skos/

Z39.50 protocol Z39.50 is a client-server protocol for searching and retrieving information from remote computer databases. It is covered by ANSI/NISO standard Z39.50, and ISO standard 23950. The standard's maintenance agency is the Library of Congress.

SMTP Simple Mail Transfer Protocol is an Internet standard for electronic mail (e-mail) transmission across Internet Protocol (IP) networks.

LDAP Lightweight Directory Access Protocol is an application protocol for querying and modifying directory services running over TCP/IP.

Shibboleth The Shibboleth System is a standards based, open source software package for web single sign-on across or within organisational boundaries. It allows sites to make informed authorisation decisions for individual access of protected online resources in a privacy-preserving manner.

DC The Dublin Core metadata element set is a standard for cross-domain information resource description. It provides a simple and standardised set of conventions for describing things online in ways that make them easier to find.

ESA European Space Agency is an intergovernmental organisation dedicated to the exploration of space. http://www.esa.int

FOSS Free and Open Source Software, also F/OSS, FOSS, or FLOSS (free/libre/open source software) is software which is liberally licensed to grant the right of users to study, change, and improve its design through the availability of its source code. http://en.wikipedia.org/wiki/FOSS

JDBC The Java Database Connectivity (JDBC) API is the industry standard for database-independent connectivity between the Java programming language and a wide range of databases – SQL

databases and other tabular data sources, such as spreadsheets or flat files. The JDBC API provides a call-level API for SQL-based database access. JDBC technology allows you to use the Java programming language to exploit "Write Once, Run Anywhere" capabilities for applications that require access to enterprise data. With a JDBC technology-enabled driver, you can connect all corporate data even in a heterogeneous environment.

JAI Java Advanced Imaging (JAI) is a Java platform extension API that provides a set of object-oriented interfaces that support a simple, high-level programming model which allows developers to create their own image manipulation routines without the additional cost or licensing restrictions, associated with commercial image processing software.

Index

C

Creative Commons, **241**
CRUD, **241**
CSW, **241**

D

DAO, **241**
DB (or DBMS), **241**
DC, **243**

E

ebRIM, **241**
ebXML, **241**
EJB, **242**
ESA, **243**

F

FAO, **242**
FGDC, **242**
FOSS, **243**

G

GAST, **242**
GeoNetwork, **241**
GEONETWORK_DATA_DIR, 92
GeoServer, **241**
GPL, **241**

H

HQL, **242**

I

import
    MEF, 116
    XML, 116
ISO, **241**
ISO TC211, **241**

J

JAI, **244**
JDBC, **243**
JIBX, **242**
JMS, **242**

L

LDAP, **243**

M

MAC address, **243**
MEF, **243**
    import, 116

O

OAI-PMH, **242**
OCHA, **242**
OGC, **242**
OO, **242**
OSGeo, **242**

S

Shibboleth, **243**
SKOS, **243**
SMTP, **243**
SOA, **242**
SOAP, **242**

T

TDD, **242**

U

UNEP, **242**
URL, **242**
UUID, **243**

W

WCS, **243**

WebDAV, **242**
WFP, **242**
WFS, **243**
WMS, **243**
WPS, **243**

X

XML, **241**
    import, 116
XSD, **241**

Z

Z39.50 protocol, **243**