

## Lab 4 (2 hours): Librarian Interface: Advanced features

### Part I – GEMS: Modifying a metadata set

---

1. Start the Greenstone Editor for Metadata Sets (GEMS):

Start→All Programs→Greenstone Digital Library Software→Greenstone Editor for Metadata Sets

2. A list of all the available metadata sets is shown on the left hand side. Explore these metadata sets, and see what elements belong to each set. Double click on a folder icon to open the set. A list of elements will be displayed.
3. In this exercise, we will create a new metadata set. In order to save time, we will base it on an existing one: Development Library Subset. From the File menu, select New (**File→New**). A popup window appears: **Add Set**. Fill in the fields. Use “USP Workshop Metadata Set” for the **Name**, “usp” for the **Namespace**, and select “Development Library Subset Example Metadata” from the **Inherit from Metadataset** drop down list. Click <OK>.
4. A folder for the USP Metadata will appear at the bottom of the metadata set list. Double click the folder icon to see what elements it has. Since it was based on the dls metadata set, it contains all the elements from that set.

#### *Adding a new element to a metadata set*

5. Right click on the **USP Workshop Metadata (usp)** item in the list of metadata sets, and choose **Add element** from the menu that appears. In the popup window, type “Category” for the **Name**, and click <OK>. The new element will appear in the list.
6. Right click on the **usp.Category** element and select **Add Attribute** from the menu. Select **definition** from the **Name** drop down list, and enter “The category this resource belongs to” in the **Values** box. The GLI uses the element definitions when displaying information about a metadata set.
7. Save the new metadata set by **File→Save**, then close the GEMS by **File→Exit**. We will use the new metadata set in the GLI in the following exercise.

### Part II – Hierarchical metadata; Phind phrase index; GLI modes

---

#### *Revision:*

1. Start a new collection called **TudorX** (the ‘X’ is for extra). Fill out the requested fields with appropriate information. Choose **USP Workshop Metadata Set** as the metadata set to use with this collection (deselect **Dublin Core**).

Note that when you select a metadata set in this popup, the list of elements is displayed in the **Elements within selected set** box. Scroll down the list for **USP**

**Workshop Metadata Set** and you can see the new **Category** element and its definition.

2. In the **Gather** panel, open the *html\_large* folder in *workshop\_files* on the CD-ROM and copy *englishhistory.net* into your new collection.
3. Build the collection, preview it and check the extracted metadata.
4. You've probably noticed that the collection contains a few stray image files, as well as the HTML documents. This is a mistake. The issue is that many of the HTML documents include images, and although Greenstone attempts to determine which images belong to HTML pages and only considers other images for inclusion in the collection, in this case it hasn't been completely successful. (This is because the web site from which these files were downloaded occasionally departs from the usual convention of hierarchical structuring.)
5. Switch to the **Design** panel and select the **Document Plugins** section. Beside **plugin HTMLPlug** you will see *smart\_block*. This is the option that attempts to identify images in the HTML pages and block them from inclusion—in this case, it's not smart enough!
6. Select the **plugin HTMLPlug** line and click **<Configure Plugin>**. A popup window appears. Scroll down the page to locate the **smart\_block** option and switch it off. Click **<OK>**.
7. Switch to the **Create** panel and **build** and **preview** the collection. The collection is exactly as before except that these stray images are suppressed.

What is happening is that plugins operate as a pipeline: files are passed to each one in turn until one is found that can process it. By default (i.e. without *smart\_block*) the HTML plugin blocks *all* images, which is appropriate for this collection. With *smart\_block* on, it tries to block only images that are linked to from the web pages. This blocking doesn't work if the images are not contained in the same folder as (or in a subfolder of) the HTML page that links to them.

*Adding hierarchically-structured metadata and a Hierarchy classifier:*

8. Switch to the **Enrich** panel and open the *tudor* folder in the left-hand panel.
9. Click on the *citizens* folder and then set its **usp.Category** metadata to **Tudor Period|Citizens**. The vertical bar ("|") is a hierarchy marker. Adding metadata to a *folder* has the effect of setting this metadata value for all files contained in this folder, its subfolders, and so on. A popup alerts you to this fact. Click **<OK>** to close the popup.
10. Repeat the process for the *monarchs* and *relatives* folders, assigning the terms **Tudor Period|Monarchs** and **Tudor Period|Relatives** respectively. Note that the hierarchy appears in the **Existing values for usp.Category** area.

If you don't want to see the popup each time you add folder level metadata, tick the **Do not show this warning again** checkbox: it won't be displayed again.

11. Finally, select all remaining files—the ones that are not in the *monarchs*, *relative*, and *citizens* folders—by selecting the first and shift-clicking the last. Set their **usp.Category** metadata to **Tudor Period|Others**: this is done in a single operation (there is a short delay before it completes).

When multiple files are selected in the left hand collection tree, all metadata values for all files are shown on the right hand side. Items that are common to all files are displayed in black—e.g. **usp.Category**—while others that pertain to only one or some of the files are displayed in grey—e.g. any extracted metadata.

Metadata inherited from a parent folder is indicated by a folder icon to the left of the metadata name. Select one of the files in the *relative* folder to see this.

12. Now switch to the **Design** panel and choose the **Browsing Classifiers** item from the left-hand list. Choose **Hierarchy** from the **Select classifier to add** menu. Click **Add Classifier**. A window pops up to control the classifier's options. Change the **metadata** to **usp.Category**. Click **<OK>**.
13. For tidiness' sake, **remove** the **classifier** for **ex.Source** metadata (included by default) from the list of currently assigned classifiers, because this adds little to the collection.
14. Finally rebuild your collection and preview it. Choose the new **Category** link that appears in the navigation bar, and click the bookshelves to navigate around the four-entry hierarchy that you have created.
15. You'll notice that the Category link is displayed as text, rather than a nice button. This is because there are no predefined images for Category metadata. In **Lab 6** you will learn how to create the buttons for this metadata element.

#### *Adding a hierarchical phrase index (PHIND)*

16. To add a PHIND classifier (an interactive hierarchical phrase index), switch to the **Design** panel and choose the **Browsing Classifiers** item from the left-hand list.
17. Choose **Phind** from the **Select classifier to add** menu. Click **Add Classifier**. A window pops up asking for configuration options: leave the values at their preset defaults (this will base the phrase index on the full text) and click **<OK>**.
18. **Build** the collection again, **preview** it and try out the new **phrases** option in the navigation bar. An interesting search term for this collection is **king**.

#### *Changing modes in the Librarian Interface*

GLI can run in different modes to reflect different user's levels of experience. There are four levels and the current level is changed via the GLI preferences. For the next short section you will need to change the current mode.

19. Choose **File→Preferences...** Click on the **Mode** tab. Choose **Expert** mode. Click **Ok** to close the **Preferences** dialog.

You should be able to see that the mode has changed in the main GLI title bar at the top of the GLI window.

20. Sometimes the building procedure does not process files in the way you expect, and it is helpful to study the text output generated during building. This shows which plugin processed which files, what indexes are being built, and so on. Changing to Expert mode will result in more output being displayed. The volume of information displayed in Expert mode is controlled by the **verbosity** option on the **Create** panel.
21. Click on the **Create** panel. You will notice that its appearance has changed. The **Import Options** and **Build Options** tabs contain options to control the import and build processes. The **Message Log** tab shows the output from these processes. Select **Import Options** from the left-hand list. Select **verbosity** in the options that appear and set its numeric counter to **5**. Since this collection takes a while to build, set **maxdocs** to **20**. **Rebuild** the collection and observe how much information is generated.
22. Return the import setting back to the default values: unselect **maxdocs** and **verbosity**.

### **Part III – Fielded Searching**

---

*We now look at adding fielded searching to a collection. Fielded searching is best used for metadata rich collections. In this exercise, we'll use bibliographic data in MARC format. Note that fielded searching is not restricted to bibliographic collections.*

1. Start a new collection called **Beatles Bibliography** which will contain a collection of MARC records from the U.S Library of Congress on the Beatles. Enter the requested information and base it on **New Collection**. Deselect the **Dublin Core** metadata set. There is no need to include any metadata sets because the metadata extracted from the MARC records will appear as extracted metadata.

Since no metadata sets are selected, Greenstone displays a warning message. Click **&ltOK>** to close the popup.

2. In the **Gather** panel, open the *sample\_marc* folder in *workshop\_files* and drag **locbeatles50.marc** into the right-hand pane and drop it there. A popup window asks whether you want to add **MARCPlug** to the collection to process this file. Click **&ltAdd Plugin>**, because this plugin will be needed to process the MARC records.
3. In the **Document Plugins** section of the **Design** panel, remove the plugins **TextPlug** to **NULPlug** (**ZIPPlug**, **GAPLug** and **MARCPlug** remain). It is not strictly necessary to remove these redundant plugins, but it is good practice to

include only plugins that are needed, to avoid unwanted (and unexpected) side effects.

4. Now select **Browsing Classifiers** from within the **Design** panel and **remove** the default classifier for **Source** metadata. In this collection all records are from the same file, so **Source** metadata, which is set to the filename, is not particularly interesting.
5. Switch to the **Create** panel, **build** the collection, and **preview** it. Browse through the **titles a-z** and view a record or two. Try searching—for example, find items that include **John Lennon**.
6. Add an **AZCompactList** classifier for the **Subject** metadata. Select this item from the relevant menu of the **Browsing Classifiers** section of the **Design** panel and click **<Add Classifier>**. In the popup window, select **ex.Subject** as the metadata item, activate the **mingroup** option and set its field to **1**.
7. **Build** the collection and **preview** the result.

#### *Adding fielded searching*

8. In the **Design** panel select **Search Types** from the left-hand list and activate the **Enable Advanced Searches** options. Add a “form” search type by making sure **form** is selected in the **Search Types** pull-down menu, and clicking **<Add Search Type>**. Remove **plain** from the **Currently Assigned Search Types** box by selecting it and clicking **<Remove Search Type>**.
9. **Rebuild** the collection and **preview** the results. Notice that the collection’s home page no longer includes a query box. (This is because the search form is too big to fit here nicely.) To search, you have to click **search** in the navigation bar. Note that the Preferences page has changed to control the advanced searching options.
10. Look at the search form in the collection. There are three fields that can be searched: *text*, *Title* and *Source*. Add some more fields to search on by going back to the Librarian Interface.
11. In the **Design** panel, go to the **Search Indexes** section. Add an index on **Subject** by selecting **ex.Subject** from the **Index Source** drop-down box, and giving it a name in the **Index Name** box, e.g. “Subject”. Add indexes on any other fields that look interesting.
12. **Rebuild** the collection and **preview** the results. Notice the extra fields in the **field** drop-down menu on the search page. You can do quite complicated queries by searching for words in different fields at the same time.

## Part VI – Multimedia collection and UnknownPlug

---

Greenstone has plugins for many file types, but sometimes you may want to include files in a collection for which there is no plugin. Here we will build a collection of multimedia files about the Beatles, which includes some MP3, image and MIDI files.

1. In the Librarian Interface, start a new collection called **Beatles Music**. Use the default settings.
2. Drag the files from the folder *workshop\_files\beatles* into the collection. A series of popups will tell you that no plugins can process the .mid files and that you may have to use **UnknownPlug**. Click <OK> for each popup. A popup will ask if you want to add **MP3Plug**. Click <OK>.
3. In the **Design** panel, add **UnknownPlug** to the list of plugins for the collection. You will need to activate the **process\_extension** option, and set it to “mid” to make it recognize files with extension .mid. Set **file\_format** to “MIDI” and **mime\_type** to “audio/midi”.

**UnknownPlug** is a useful generic plugin. It knows nothing about any given format but can be tailored to process particular document types—like MIDI—based on their filename extension, and set basic metadata.

4. **Build** the collection and **preview** it. All the files, including the MIDI files, are included in the collection, and clicking on the MP3 or unknown icons will enable you to play the audio.
5. Back in the Librarian Interface, go to the **Enrich** panel and look at the metadata. For the MIDI files, *ex.FileFormat* and *ex.MimeType* metadata have been set based on the options for the plugin. *ex.FileFormat* is also set for the other file types. Automatically assigned metadata (for all files) includes *ex.FileSize*, *ex.Source* (the filename), *ex.Title* (based on the filename) and *ex.Plugin*.
6. Add a classifier based on **ex.FileFormat**. In the **Design** panel, add an **AZCompactList**, with **metadata** set to *ex.FileFormat* and **buttonname** set to *Browse*. **Build** and **preview** the collection.

Customization of this collection will be done in **Lab 5**.

## Part V – Your custom collection

---

Enhance your own collection by using some of the features you have learnt about in this lab.

## Part VI – Extra work

---

*Partitioning the full-text index based on metadata values.*

Here you will split the full text index into parts based on the Category metadata. This enables users to restrict their searches to individual categories of documents.

1. In the GLI, open up the **TudorX** collection.
2. Switch to the **Design** panel, and click <**Partition Indexes**>. This feature is disabled because you are operating in *Librarian Mode*. Switch to *Library Systems Specialist* mode (**File**→**Preferences...**→**Mode**).
3. Ensure that the **Define Filters** tab is selected (the default). Define a subcollection filter with name **monarchs** that matches against **usp.Category**, and type **Monarchs** as the regular expression to match with. Click <**Add Filter**>. This filter includes any file whose **usp.Category** metadata contains the word *Monarchs*.
4. Define another filter, **relatives**, which matches **usp.Category** against the word **Relatives**. Define a third and fourth, **citizens** and **others**, which matches it against the words **Citizens** and **Others** respectively.
5. Having defined the subcollection filters, we partition the index into corresponding parts. Click the <**Assign Partitions**> tab. Select the first filter and give it the name **citizens**; click <**Add Partition**>. Repeat for the other three subcollections, naming their partitions, **monarchs**, **others** and **relatives**.

The order they appear in the **Assigned Subcollection Partitions** list is the order they will appear in the drop down menu on the search page. You can change the order by using the **Move Up** and **Move Down** buttons.

6. **Build** and **preview** the collection.
7. The search page includes a pulldown menu that allows you to select one of these partitions for searching. For example, try searching the *relatives* partition for *mary*, and then search the *monarchs* partition for the same thing.
8. To allow users to search the collection as a whole as well as each subcollection individually, return to the **Partition Indexes** section of the **Design** panel and select the **Assign Partitions** tab. Type **all** into the **Partition Name** and select all four subcollections by checking their boxes. Click <**Add Partition**>.
9. To ensure that the *all* index appears first in the list on the reader's web page, use the <**Move Up**> button to get it to the top of the list. Then **build** and **preview** the collection.
10. Search for a common term (like *the*) in all five index partitions, and check that the numbers add up.
11. Return to *Librarian* mode, using **Preferences** (on the *File* menu).

#### *Exploding a metadata database*

13. Open the **Beatles Bibliography** collection.

14. Go to the **Enrich** panel and try to see the metadata. It doesn't appear! This is because the metadata is associated with records inside the file, not the file itself.

Metadata file types, such as MARC, CDS/ISIS, BibTex etc can be imported into Greenstone, but their metadata cannot be viewed in the GLI. To edit any metadata, you need to go back to the program that created the file.

Greenstone provides a new way of exploding a metadata database so that each record appears as an individual document, with viewable and editable metadata. This process is irreversible: once this step has been done, the database is deleted, and can no longer be used in its original program.

15. Before exploding the database, we need to add a metadata set to the collection. This is so the metadata from the database can be added to it. Switch to the **Design** panel, and choose **Metadata Sets** from the left hand list. There will only be one metadata set in the **Available Metadata Sets** list: extracted metadata. To add a new set, click <**Add Metadata Set**>. A popup window shows you a list of the metadata set files that are available. Select **dublin.mds** and click <**Add Metadata Set**>.
16. In the **Gather** panel, you may notice that the MARC database has a different coloured icon to other files. This green icon indicates that it is a metadata database that can be exploded. Right-click on the icon and choose **Explode metadata database** from the menu. A new window opens, containing options for the exploding process. A description of each option can be obtained by hovering the mouse over the option briefly. In standard usage these options are not required.
17. Click <**Explode**> to start the exploding process. This may take a short while, depending on the size of the database.
18. During the exploding process the GLI will ask you how to deal with the metadata it finds in the MARC records. For each metadata field you have the option of adding it to a metadata set in the collection (**Add**), mapping it to an existing metadata element (**Merge**), or ignoring it (**Ignore**). **Add** will not be available if an element of the same name already exists in the target metadata set. For each element, you should **Add** if it doesn't already exist, and **Merge** if it does.
19. Once the exploding has finished, the MARC database file will have been deleted and a folder created in its place. This folder contains an entry for each record in the original database. The metadata for each of these records can be viewed and edited by switching to the **Enrich** pane.
20. To rebuild this collection we need to change the list of plugins. The MARC file is no longer present; it has been replaced by a series of empty (.nul) files. In the **Document Plugins** section of the **Design** panel, remove **MARCPlug** and add **NULPlug** (use the default configuration).
21. **Rebuild** and **preview** the collection.