LAB 2:

Greenstone: Adding metadata - and using it

2.1. A collection of Word and PDF files

You will need some source files like those in the sample_files → Word_and_PDF folder.

Start a new collection called **reports** (**File → New...**) and base it on **-- New Collection --**.

Copy all the files .doc, .rtf, .pdf and .ps files from *sample_files → Word_and_PDF → Documents* into the collection. There are 9 files in all: you can select multiple files by clicking on the first one and shift-clicking on the last one, and drag them all across together. (This is the normal technique of multiple selection.)

3. Switch to the **Create** panel, and **build** and **preview** the collection.

Viewing the extracted metadata

4. Again, this collection contains no manually assigned metadata. All the information that appears—title and filename—is extracted automatically from the documents themselves. Because of this the quality of some of the title metadata is suspect.

5. Back in the Librarian Interface, click the **Enrich** tab to view the automatically extracted metadata. You will need to scroll down to see the extracted metadata, which begins with "ex.".

6. Check whether the **ex.Title** metadata is correct for some of the documents by opening them. You can open a document from the Librarian Interface by double clicking on it.

7. The extracted Title metadata for some documents is incorrect. For example, the Titles for *pdf01.pdf* and *word03.doc* (the same document in different formats) have missed out the second line. The Title for *pdf03.pdf* has the wrong text altogether. The PostScript documents (*cluster.ps* and *langmodl.ps* do not have extracted titles: what appears in the *Titles* list is just the first few characters of the document).

Manually adding metadata to documents in a collection

1. In the **Enrich** panel, manually add Dublin Core **dc.Title** metadata to those documents which have incorrect **ex.Title** metadata. Select *word03.doc* and double-click to open it. Copy the title of this document ("Greenstone: A comprehensive open-source digital library software system") and return to the Librarian Interface. Scroll up or down in the metadata table until you can see **dc.Title**. Click in the value box and paste in the metadata.

2. Now add **dc.Creator** information for the same document. You can add more than one value for the same field: when you press **Enter** in a metadata value field, a new empty field of the same type will be generated. Add each author separately as **dc.Creator** metadata.

3. Close the document (in Microsoft Word) when you have finished copying metadata from it. External programs opened when viewing documents must be closed before building the collection, otherwise errors can occur.

4. Next add **dc.Title** and **dc.Creator** metadata for a few of the other documents.

5. You will notice as you add more values, they appear in the **Existing values for ...** box below the metadata table. If you are adding the same metadata value to more than one document, you can select it from this list. For example, *pdf01.pdf* and *word03.doc* share the same Title; and many documents have common authors.

6. Repeat the exercise for all the Documents in this collection.

*If you build and preview your collection at this point, you will see that the **Titles** list now shows your new Titles. However, the **dc.Creator** metadata is not displayed. You need to alter the collection design to use this metadata.*

Document Plugins

6. In the Librarian Interface, look at the **Document Plugins** section of the **Design** panel, by clicking on this in the list to the left. Here you can add, configure or remove plugins to be used in the collection. There is no need to remove any plugins, but it will speed up processing a little. In this case we have only Word, PDF, RTF, and PostScript documents, and can remove the **ZIPPlugin**, **TextPlugin**, **HTMLPlugin**, **EmailPlugin**, **PowerPointPlugin, ExcelPlugin, ImagePlugin**, **ISISPlugin** and **NULPlugin** plugins. To delete a plugin, select it and click **<Remove Plugin>**. **GreenstoneXMLPlugin** is required for any type of source collection and should not be removed.

Search indexes

7. The next step in the **Design** panel is **Search Indexes**. These specify what parts of the collection are searchable (e.g. searching by title and author). Delete the **ex.Source** index, which is not particularly useful, by selecting it and clicking **<Remove Index>**.

8. By default the titles index (**dc.Title,ex.Title**) includes **dc.Title** and **ex.Title**. Searching this index will search both **dc.Title** and **ex.Title** metadata. If you want to restrict searching to just the manually added **dc.Title** metadata, edit this index and deselect **ex.Title** from the list of metadata..

9. You can add indexes based on any metadata. Add a new index based on **dc.Creator** by clicking **<New Index>**. Select **dc.Creator** in the list of metadata, and click **<Add Index>**.

Browsing classifiers

10. The **Browsing Classifiers** section adds "classifiers," which provide the collection with browsing functions. Go to this section and observe that Greenstone has provided two *List* classifiers, based on **dc.Title;Title** and **ex.Source** metadata. These correspond to the *Titles* and *Filenames* buttons on the collection's access bar.

Remove the **ex.Source** classifier by selecting it and clicking **<Remove Classifier>**.

12. Now add an **AZCompactList** classifier for **dc.Creator**. Select **AZCompactList** from the **Select classifier to add:** drop-down list and click **<Add Classifier...>**. A popup window **Configuring Arguments** appears. Select **dc.Creator** from the **metadata** drop-down list and click **<OK>**.

    AZCompactList is like **List**, except that values that appear multiple times in the hierarchy are automatically grouped together and a new node, shown as a bookshelf icon, is formed.

13. Switch to the **Create** panel, and **build** and **preview** the collection.

14. Check that all the facilities work properly. There should be three full-text indexes, called *_texttext_*, *Titles*, and *Creators*. The *Titles* list should display all the document Titles. The *Creators* list should show one bookshelf for each author you have assigned as **dc.Creator**, and clicking on that bookshelf should take you to all the documents they authored..

Renaming the search indexes

15. The default display text for the indexes in the drop-down list on the search page contains the content of the index. Now we will change this display text to make it nicer. Go to the **Format** panel by clicking its tab. This panel is split into several sections, each controlling some aspect of collection presentation.

16. Fro versions before 2.82, select **Search** in the left hand list. This section allows you to modify what text is displayed for the drop-down lists in the search form (indexes, subcollections, levels etc). Set the **Display text** for the **dc.Title,Title** index to be "titles", and that for the **dc.Creator** index to be "creators". Preview the collection by clicking the **Preview Collection**. The search form should display the new text.

Classifying on multiple metadata

17. For versions before 2.82, the new *Titles* list shows only those documents which have been assigned **dc.Title** metadata. For many documents, extracted Titles may be fine, and it is impractical to add the same metadata again as **dc.Title**. Fortunately there is a way we can use both metadata types in one classifier: specify a list of metadata names in the classifier.

18. In the **Browsing Classifiers** section of the **Design** panel, select the **AZList** for **dc.Title** in the **Assigned Classifiers** box and click **<Configure Classifier...>**. Note you can achieve the same result by double clicking on the classifier.

19. In the **metadata** field, type ",ex.Title" after the "dc.Title"—i.e. make it read

    ```
    dc.Title,ex.Title
    ```

20. If you have already done the **Enhanced Word document handling** exercise, some of the documents will have extracted ex.Creator metadata, and some will have dc.Creator. To use both of these in the Creators classifier, make a similar change to the **AZCompactList**: make the **metadata** field read `dc.Creator,ex.Creator`.

Branding a collection with an image

21. Switch back to the **Format** panel. The first section **General** appears. This allows you to modify the values you provided when defining the collection, if desired. You can also brand the collection using a suitable image.

Click on the **<Browse...>** button associated with **URL to 'about page' image:**, and browse to the image *sample_files → Word_and_PDF → wrdpdf.gif* on your computer. When you select this image, Greenstone automatically generates an appropriate URL for the image. **Preview** the collection: you should see the new image at the top left of the page.

2.2. Formatting the Word and PDF collection

In this exercise, we play around with the format statements in the Word and PDF collection.

1. Open the **reports** collection in the Librarian Interface and go to the **Format Features** section of the **Format** panel.

Tidying up the default format statement

2. In this part of the exercise, we make the format statement simpler without changing the resulting display.

   Greenstone's default format statement is complex because it is designed to produce something reasonable under almost any conditions, and also because for practical

reasons it needs to be backwards compatible with legacy collections. For this collection, we don't need all of the complexity.

Make sure that the **VList** format statement is selected in the list of formats.

The default **VList** format statement looks like the following:

```
<td valign="top">[link][icon][/link]</td>
<td
valign="top">[ex.srclink]{Or}{[ex.thumbicon],[ex.srcicon]}[ex./srclink
]</td>
<td valign="top">[highlight]
{Or}{[dc.Title],[exp.Title],[ex.Title],Untitled}
[/highlight]{If}{[ex.Source],<br><i>([ex.Source])</i>}</td>
```

This format statement is the default used for any vertical list, such as search results, classifiers, and document table of contents.

`{Or}{[ex.thumbicon],[ex.srcicon]}` chooses *ex.thumbicon* metadata if its there, otherwise chooses *ex.srcicon* metadata. If neither are present, nothing is displayed. For this collection there is no *ex.thumbicon* metadata so the choice is not needed.

Replace `{Or}{[ex.thumbicon],[ex.srcicon]}` with `[ex.srcicon]`.

There is no *exp.Title* metadata, so remove that element from `{Or}{[dc.Title],[exp.Title],[ex.Title],Untitled}`.

The resulting format statement looks like the following:

```
<td valign=top>[link][icon][/link]</td>
<td valign=top>[ex.srclink][ex.srcicon][ex./srclink]</td>
<td valign=top>[highlight]
{Or}{[dc.Title],[ex.Title],Untitled}[/highlight]
{If}{[ex.Source],<br><i>([ex.Source])</i>}</td>
```

Preview the collection to make sure the display hasn't changed. You shouldn't notice any difference when looking at search results, classifiers etc.

Linking to Greenstone version or original version of documents

3. For collections with documents that undergo a conversion process during importing (e.g. Word, PDF, PowerPoint documents, but not text, HTML documents), the original file is stored in the collection along with the converted version. The default **VList** format statement links to both versions:

   `[link][icon][/link]` links to the Greenstone HTML version, while `[srclink][srcicon][/srclink]` links to the original.

Choose **SearchVList** in **Format Features** by selecting **Search** from the **Choose Feature** drop down list, and **VList** from the **Affected Component** list. Click **<Add Format>** to add the **SearchVList** format statement into the list of assigned formats. Experiment with removing either of the two links from the format statement.

To see the results of your changes, preview the collection and do a search. You are making changes to **SearchVList**, which means the changes will only apply to search results.

Storing and displaying the original allows users to see the correct format, but requires the user to have the relevant program installed. It also increases the size of the collection. The Greenstone version can be viewed in a browser, but may not look as nice.

Making bookshelves show how many items they contain

4. Next, we'll customize the format for the *Creators* list. Classifier bookshelves have only a few pieces of metadata to display: `[ex.Title]` and `[numleafdocs]`. Whatever metadata the classifier has been built on, the bookshelf label is always stored as `[ex.Title]`. This is why a Creator is printed out for each bookshelf even though `[dc.Creator]` is not specified in the format statement. `[numleafdocs]` is only defined for bookshelves, so this metadata can be used in an `{If}` statement to make bookshelves and documents display differently in the list.

   Make each bookshelf in the Creator classifier show how many entries it contains. In the **Format Features** section of the **Format** panel, select the **CL2 AZCompactList** classifier which is based on **dc.Creator** metadata from the **Choose Feature** drop down list, and **VList** from the **Affected Component** list. Click the **<Add Format>** button to add this format into the list of assigned formats. Note that it gets added as **CL2VList** in this list: it is the **VList** format for the second (**CL2**) classifier.

   Append the following text to the bottom of the format statement:

   ```
   {If}{[numleafdocs],<td><i>([numleafdocs])</i></td>}
   ```

   Preview the collection. Click on the *Creators* list and notice that the bookshelves now display how many documents they contain.

   This revised format statement has the effect of specifying in brackets how many items are contained within a bookshelf. Since only bookshelves define `[numleafdocs]`, only they will display this. By modifying **CL2VList** instead of **VList**, the change will only apply to the second classifier (Creators).

Displaying multi-valued metadata

5. Next we modify the document entries in the Creator classifier to display all authors. Back in **Format Features**, select the **CL2VList** format in the list of assigned formats. After `{If}{[ex.Source],<br>` in the format statement, add `[sibling:dc.Creator]`.

   `[ex.Source]` is not defined for bookshelves, so can also be used to differentiate bookshelves and documents.

   The resulting format statement looks like:

```
<td valign=top>[link][icon][/link]</td>
<td valign=top>[ex.srclink][ex.srcicon][ex./srclink]</td>
<td valign=top>[highlight]
{Or}{[dc.Title],[ex.Title],Untitled}[/highlight]
{If}{[ex.Source],<br>[sibling:dc.Creator]
<i>([ex.Source])</i>}</td>
{If}{[numleafdocs],<td><i>([numleafdocs])</i></td>}
```

   This will display the Greenstone link, the link to the original, then the Title. For bookshelves, it will also display how many documents the bookshelf contains. For documents, it will display all the Authors (Creators), and the source document. `[sibling:dc.Creator]` displays all the Creator metadata for the document, separated by a space (" "), while `[dc.Creator]` displays only the first author. Preview the *Creators* list and make sure that all authors are displayed for documents.

6. You can change the separator between the authors. Modify the format statement, and replace `[sibling:dc.Creator]` with `[sibling(All'<br/>'):dc.Creator]`. This will add a new line after each author (`<br/>` specifies a line break in HTML). Preview the *Creators* list.

   If you have done exercise **Enhanced Word document handling**, the collection will have both dc.Creator and ex.Creator metadata. To display both, you can use

```
[sibling:dc.Creator] [sibling:ex.Creator]
```

   To display dc.Creator if it is present, otherwise display ex.Creator, use

```
{Or}{[sibling:dc.Creator],[sibling:ex.Creator]}
```

Advanced multi-valued metadata

7. You may notice that **AZCompactList** has two options after the **metadata** option: **firstvalueonly** and **allvalues**. Manually added metadata can be used to replace or enhance automatically extracted metadata, and these options control exactly which pieces of metadata a document is classified by.

   For example, say we have two documents. Document 1 has four Creators specified (dc.Creator = dcA, dc.Creator = dcB, ex.Creator = exA, ex.Creator = exB), while document 2 has three (ex.Creator = exA, ex.Creator = exB, ex.Creator = exC). The

following table shows which metadata values each document is classified by, for the different classifier options:

AZCompactList options	Document 1	Document 2
-metadata dc.Creator,ex.Creator	dcA, dcB	exA, exB, exC
-metadata dc.Creator,ex.Creator -firstvalueonly	dcA	exA
-metadata dc.Creator,ex.Creator -allvalues	dcA, dcB, exA, exB	exA, exB, exC

8. Now we set the **firstvalueonly** option for the *Creators* classifier. Switch to the **Browsing Classifiers** section of the **Design** panel, select the **AZCompactList** for **dc.Creator** metadata in the **Assigned Classifiers** box and click **<Configure Classifier...>**. Select the **firstvalueonly** option.

9. **Rebuild** and **preview** the collection. Now the *Creators* list classifies documents based on the first author appearing in the **dc.Creator** metadata.

10 If you set the **metadata** field of **AZCompactList** to `dc.Creator,ex.Creator` in the A collection of Word and PDF files exercise, now the *Creators* list will classify based on the first author appearing in either the **dc.Creator** metadata or the **ex.Creator** metadata.

2.3 Building and searching with different indexers

Greenstone supports three indexers **MG**, **MGPP** and **Lucene**. **MG** is the original indexer used by Greenstone which is described in the book **"Managing Gigabytes"**. It does section level indexing and compression of the source documents. **MG** is implemented in C.

MGPP is re-implementation of **MG** that provides word-level indexes and enables proximity, phrase and field searching. **MGPP** is implemented in C++ and is the default indexer for new collections.

Lucene (http://lucene.apache.org/) is java-based full-featured text indexing and searching system developed by Apache. It provides a similar range of search functionality to MGPP with the addition of single-character wildcards and range searching. It was added to Greenstone to facilitate incremental collection building, which **MG** and **MGPP** can't provide.

Build with Lucene

1. Start a new collection (**File → New...**) called **Demo Lucene** and base it on the **Greenstone demo (demo)** collection, fill out its fields appropriately.

2. In the **Gather** panel, click **Documents in Greenstone Collections** and click **Greenstone demo (demo)**, it will show the documents in the **Greenstone demo** collection. Drag all 11 folders underneath *Greenstone demo (demo)* into the new collection

3. Go to the **Enrich** panel, look at the metadata that associated with each directory. Go to the **Search Indexes** section in the **Design** panel. The **MG indexer** is in use because the original **Greenstone Demo** collection, which this collection is based on, uses **MG indexer**.

4. Click the **Change...** button at the right top corner of the panel. A new window will pop up for selecting the Indexers. After selecting an indexer, a brief description will appear in the box below. Select Lucene and click **OK**. Please note that the **Assigned Indexes** has changed accordingly

5. **Build** and **preview** the collection

Search with Lucene

6. Lucene provides single letter and multiple letter wildcards and range searching. The query syntax could be quite complicated (for more information please see http://lucene.apache.org/java/docs/queryparsersyntax.html). Here we will learn how to use the wildcards while constructing queries.

7. ***** is a multiple letter wildcard. To perform a a multiple letter wildcard search, append ***** to the end of the query term. For example, *econom** will search for words like *econometrics*, *economist*, *economical*, *economy*, which have the common part *econom* but different word endings.

8. To perform a single letter wildcard search, use **?** instead. For example, search for *economi??* will only match words that have two and only two letters left after *economi*, such as *economist*, *economics*, and *economies.*

9. Please note that stopwords are used by default with Lucene indexer, so search for words like *the* will match 0 document. There is also a message on the search page saying that such words are too common and were ignored.

Build with MGPP

10. Start a new collection called **Greenstone Demo MGPP** and also base it on the **Greenstone demo (demo).**

11. In the **Gather** panel, drag all the 11 folders from → *Greenstone demo (demo)* into the new collection.

12. Go to the **Search Indexes** section in the **Design** panel, click the **Change...** button and select **MGPP**. Click **OK**. Check the **Assigned Indexes** has changed accordingly.

13. There are three options at the bottom of the panel — **Stem**, **Casefold** and **Accent fold**. Notice that **Stem** and **Casefold** are enabled. Once an option is enabled, it will also appear in the collection's **PREFERENCES** page.

14. In the **Indexing Levels** section, also select **section.**

15. **Build** and **preview** the collection.

    Search with MGPP

16. MGPP supports stemming and casefolding. By default search in collections built with MGPP indexer is set to **whole word must match** and **ignore case differences**. So search *econom* will return 0 document. Search for *fao* and *FAO* return the same result — 78 word counts and 9 matched documents.

    Go to the **PREFERENCES** page by click the **PREFERENCES** button at the top right corner. You can see that the **Word endings:** option is set to **whole word must match** and the **Case differences:** option is set to **ignore case differences**

17. Sometimes we may want to ignore word endings while searching so as to match different variations of the term. Go to the **PREFERENCES** page and change the **Word endings:** option from **whole word must match** to **ignore word endings**. Click the **set preferences** button. Click **Search**. This time try search for *econom* again, 9 documents are found.

    Please note that word endings are determined according to the third-party stemming tables incorporated in Greenstone, not by the user. Thus the searches may not do precisely what is expected, especially when cultural variations or dialects are concerned. Besides, not all languages support stemming, only English and French have steming at the moment.

    Go to the **PREFERENCES** page and change back to **whole word must match** to avoid confusion later on. Click the **set preferences** button.

18. Sometimes we may want to search the exact term, that is, differentiate the upper cases from lower cases. Set the **Case differences:** option from **ignore case differences** to **upper/lower case must match**. Click the **set preferences** button. Click **Search**. Now try search for *fao* and *FAO* respectively this time, notice the difference in the results?

    Go back to the **PREFERENCES** page and change the **Case differences:** option back to **ignore case differences** to avoid confusion later on. Click **set preferences** button.

    Use search mode hotkeys with query term

    *MGPP have several hotkeys to set search modes for a query term. These hotkeys explicitly set the **Word endings:** option and the **Case differences:** option for the query being constructed*

19. **#s** and **#u** are hotkeys for the **Word endings:** option. Appending **#s** to a query term will specifically enable the **ignore word endings** function. For example, try search for *econom#s*, 7 documents are found, which is the same as in step 17. Remember that we have set it back to **whole word must match**. This means using hotkeys will override the current preference settings.

20. Appending **#u** to a query term will explicitly set the current search to **whole word must match.**

    Note that using hotkeys will only affect that query term. That is, hotkeys are used per term. For example, if a query expression contains more than one terms, some terms can have hotkeys and others not, and the hotkeys can be different for different terms. This provides a fine-grained control of the query, whereas changing settings in the **PREFERENCES** page will affect the query as a whole

21. Hotkeys **#i** and **#c** control the case sensitivity. Appending **#i** to a query term will explicitly set the search to **ignore case differences** (i.e. case insensitive).

22. On the contrary, appending **#c** will specifically turn off the casefolding, that is, **upper/lower case must match**. For example, search for *fao#c* returns 0 document.

23. Finally, the hotkeys can also be used in combination. For example, you can append *#uc* to a query term so as to match the whole term (without stemming) and in its exact form (differentiate upper cases and lower cases).

A quick reference of the search mode hotkeys in MGPP

```
Word endings:

#s    ignore word endings

#u    whole word must match

Case differences:

#i    ignore case differences

#c    upper/lower case must match
```

****