Trust Waikato Symphony Orchestra

Digital Library Update Guide

Jessica Turner
jdt11@students.waikato.ac.nz
Computer Science Department
University of Waikato
08/02/2013

Contents

1 Introduction

This guide explains how to maintain and update the Trust Waikato Symphony Orchestra digital library collection. It is assumed that you have a basic understanding of the Greenstone 3 digital library software and know how to import documents into the collection. If you do not have this experience, tutorials can be found on the Greestone 3 wikipedia page: http://wiki.greenstone.org/wiki/index.php/GS3_Tutorial_exercises.
To be able to update the collection you will need the following:

- Scanner (e.g. in the photo copy room)
- OCR software ABBYY FineReader 11 (available on Li Liang's computer)
- Greenstone 3.05 or higher (installed from http://www.greenstone.org/snapshots)
- TWSO Project folder containing existing collection files, material, and OCR programs which is stored on the puka server in the location */greenstone/custom/greenstone3-3.05-showcase/TWSO Project*.
- You may need pdf tools such as Adobe Acrobat if you intend to modify the programmes after they have been scanned.
- Handbrake 0.9.8

TWSO Project Folder

Figure 1 shows the structure of the *TWSO project* folder which contains all the files you need in order to work with the collection.

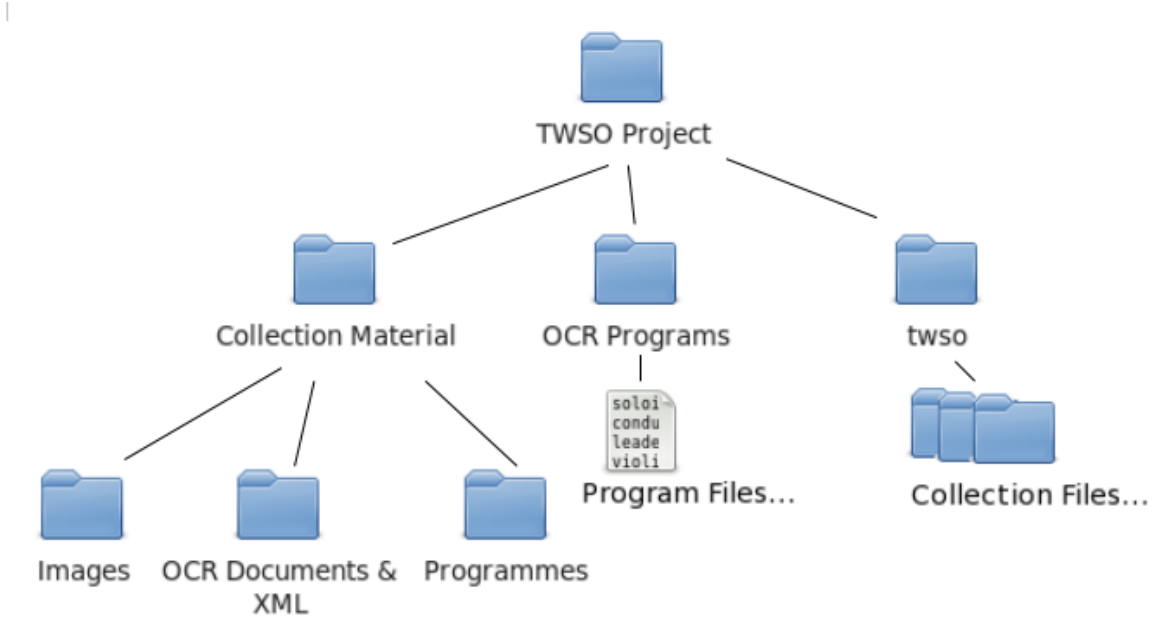

Figure 1: TWSO Project folder structure

The *Collection Material* folder contains raw material that has been included in the collection. *Images* holds the images in the collection. *OCR Documents & XML* contains the OCR files, name extraction files, and metadata. *Programmes* has the original and modified pdf files currently included in the collection.
OCR Programs has the source code for the *Name Finder*, *Master Name Creator* and *Metadata Creator*. It also has the original master name list, roles list and piece name list. The master name list holds the correct spellings of all names included in the collection. The roles file contains all the "roles" found in the programme files; it is used by the *Name Finder* to detect the names within an OCR txt document. The piece name list has the correct spellings and variants of the piece names already included in the collection.
The *twso* folder is the collection folder used by Greenstone 3.05 to build the collection. Once you have Greenstone 3.05 installed on your computer, find the location where the program was installed and navigate to the *collect* folder. This is where all your Greenstone collections are stored. Copy and paste the *twso* folder into this location. The next time you start Greestone you should be able to open the collection in gli for editing. It will have the name *TWSO (twso)*.
Once you have the required programs and files on your computer you can begin to update the Trust Waikato Symphony Orchestra collection.

main.xsl

On the Trust Waikato Symphony website the collection is also embedded in an iframe (http://twso.co.nz/concertsarchive.shtml). In order to achieve this effect the main.xsl file for the collection has been edited to include javascript that modifies the css of the page. The url points to the same location but this javascript detects if it is in an iframe and modifies the layout accordingly. If you wish to modify this code the file is in the *twso* collection folder. The location is: *transform/layouts/main.xsl*. The code you are looking for starts like this:

```
if(window != window.top){
    console.log("iframe");
    var header = document.getElementsByTagName("head");
    var css = document.createElement("link");
    css.setAttribute("rel", "stylesheet");
    css.setAttribute("href", "sites/localsite/collect/twso/style/small.css");
    if(header[0] != null){ header[0].appendChild(css); }
    var classifiers = document.getElementById("results");
    if(classifiers != null){
        classifiers.style.width= "580px";
        ...
```

2 Initial Steps

These are the steps that allow you to add a new programme to the collection and then update the collection on puka.

1. Scan the programme that you wish to add to the collection — see Section 3.

2. Open *gli* and import the file into the appropriate folder in the Gather panel. If the year does not exist create a new folder for it under the Programmes folder. If you are unsure of the date, place the programme in the Unknown folder.
3. Switch to the Enrich panel and enter the performance metadata.
    - **pd.Player:** name of a player in the format: *Last Name, First Name*. If the person has a maiden name: *Last Name (née Maiden Name), First Name*. Change of name: *Last Name (formerly Previous Name), First Name*.
    - **pd.Conductor:** name of the conductor in the same format.
    - **pd.Soloist:** name of a soloist in the same format.
      The above fields can be entered by using the *Name Finder* and *Metadata Creator* — see Sections 4 & 5.
    - **pd.Location:** location of the performance. The format is: *Location One & Location Two & Location Three …*
    - **pd.Date:** used for sorting the program dates using the built in Greenstone date function. The date must be entered in the format: *20110301*. The first four digits are the year, the second two the month, and the last two the day. This data must be entered for the programme to appear under the Dates browsing classifier. If the programme has multiple dates, press *enter* to add a new pd.Date field, and enter the new date in the same way.
    - **pd.formatDate:** used when multiple dates occur, which causes strange default formatting. The format date feature of Greenstone does not display the data in the way we want so we must re-enter it in the correct format. For example, *20111001* would be formatted to *01 October 2011*. We prefer to remove the *0,* so we enter *1 October 2011* into the *pd.formatDate* field. When multiple dates occur, put a slash between the dates, *22/23 November 2009*.
    - **pd.Composer:** composer name. Only the last name of the composer is stored unless duplicates occur (for example, Williams). When this happens, enter the composer name in the same format as players.
    - **pd.Piece:** what was performed at the concert, in the format: *Composer — Name of Piece, Opus Number*. Multiple pieces can be entered by using multiple *pd.Piece* fields. Check the names of the pieces against the ones found in the *correct names.txt* file in the *OCR programs* folder. This helps prevent incorrect spellings and variants. If the name is not there update the file to include it.
      The above fields must be entered manually.
    - **pd.Temp:** if the programme file is only a temporary place holder as a concert took place on this date but no programme was found enter *yes* into this field. If this is a place holder for a video enter *Video* into the field. This changes what icon is displayed instead of the default yellow programme icon.
4. OCR the programme using OCR software. Li Lian's laptop has ABBYY FineReader that can be used. The OCR file will be a word document. Re-save the document as a txt file so that is can be used with the *Name Finder.*
5. Use the *Name Finder* (Section 4) to extract the names from the programme. Check the

spelling of the extracted names by using the *Master Name Creator* (Section 6) to update the master list and look for errors. Update your name list to reflect the correct spellings from the master list. Use the *Metadata Creator* (Section 5) to create the metadata xml file.

6. Import the metadata xml file to the same folder as the programme using the gather panel in gli.
7. Re-build the collection.
8. Check that the programme appears in each index using different metadata values to ensure that the metadata is working.
9. Save and close the collection.
10. Upload the updated collection to puka — see Section 7.

3 How to Scan a file

There are multiple ways to scan a file to a computer. This guide assumes that you are using the scanner on the first floor of G block. In case you are not using this scanner It also gives the requirements of the scanned document.

1. Place your ID card on the swipe area to log in.
2. Press the *scan/fax* button on the keypad to be taken to the scan screen.
3. Press *scan settings* on the touch screen to set up the scanner.
    - Ensure that it is set to scan a multi-page pdf file (this is the default setting).
    - Select *simplex/duplex* and select *2-sided*. This indicates that you will be scanning a multi-paged 2-sided document.
    - Select the *resolution* and set it to standard (200dpi x 100dpi).
    - Select the *scan size*. You have two options: *Custom Size* and *Auto*. *Custom Size* will require you to enter the height and width of the document, and is best for unusual shaped programmes. *Auto* is for regular sized documents (A4 or A5). If it cannot detect the correct size for the pages it will select the largest size available that includes the entire document. This can lead to lots of white space around the pdf, which looks untidy and can lead to more work later requiring pdf tools.
4. Select *Email Me* and then press *start*. The machine will inform you when it is finished scanning each page. You must then turn the page and press *start* again.
5. When you are finished scanning select *Finished* and press *start*. It will then send the document to your registered email address (your waikato email address).

4 Using Name Finder

Inside the *TWSO project* folder is a folder called *OCR Programs*. This contains all the programs required to extract names from the OCR programme files. Before you use the *Name Finder,* edit your OCR programme file. The *Name Finder* assumes that the names will be in the format:

ROLE
Name one
Name two
Name three …

as this is the most common format that I have discovered the OCR program produced. Editing the document in this way will ensure that the *Name Finder* picks up all names, instead of some of them.

Another common problem is that sometimes the program finds roles within the programme notes. You can skip this in the *Name Finder* program, however, I found it more effective to remove the programme notes altogether and put them back when finished, because this saved time looking for names.

1. Copy the *OCR Programs* files to the location where your OCR documents are stored.
2. Open a terminal in that location.
3. To run the program enter the command:
   python namefinder1.py nameoffile
   - *nameoffile* is the name of the OCR programme file that you are going to extract names from.
   - If the file is not directly accessible, enter the path name to the file.
   - You do not need to enter the file extension.
4. The program then looks for names within the file. It is recommended to have the file open so that you can check as you go that names are not missed.
5. You will be presented with dialogue like this:

   THE ORCHESTRA
   Conductor
   LINE: Conductor
   MATCH FOUND CONDUCTOR
   NAME FOUND: Andrew Buchanan-Smart
   NAME TO BE ADDED: Buchanan-Smart, Andrew
   Is this a name? Y|N|E|Hit any other key to continue...

   - *LINE:* what line was found in document.
   - *MATCH FOUND:* which role has been found based on the roles.txt file (in this case, *CONDUCTOR*).
   - *NAME FOUND:* name found on the next line.
   - *NAME TO BE ADDED:* the name the program automatically assumes will be added to the list.
   - You are then provided four options:
     - *Y* = Yes this is a name. This will add the name to the list.
     - *N* = No this is not a name. This will finish looking at this Section of names. This can cause names to be missed if you are not careful.
     - *E* = enter the name. Use this when the name to be added does not show the name in the correct format. You will be presented with the dialogue:

Enter name: .For example names like “Ona de Rooy” will be detected as “Rooy, Ona de”. Provide the correct version of the name. In this example you would type “de Rooy, Ona”.
    - *Hit any key to continue …* = continue looking through the list of names. If there are no more names under this Section the program will resume looking for the next role.
6. When finished the program will print out *Finished writing names to example_name_list.txt*. The file specified is where the names are stored; it will appear in the same directory as the OCR programme file. Check this file to ensure that the program has not made any mistakes.
7. Check the names against the master list and update the file accordingly to reflect the correct spellings. If you cannot find a name, run the *Master Name Creator* to add them to the master list (Section 6).
8. Once you are happy with the names run the *Metadata Creator* to create the xml file that will be imported into the collection (Section 5).

5 Using the Metadata Creator

The *Metadata Creator* is found in the *OCR Programs* folder in the *TWSO project* folder. Copy the *metadatacreator.py* file to the same location as the name list file that has been created by the *Name Finder*.

1. To run the program, enter the command:
   python metadatacreator.py file_name_list.txt programmeToAddMetadataTo
    - *file_name_list.txt* is the file name of the name list you want to turn into metadata.
    - *programmeToAddMetadataTo* is the file name of the pdf programme file that you want the metadata to be created for. You do not need the .pdf extension.
2. The program will process the file and create a document called *filemetadata.xml* where *file* is the name of the original document.
3. Import this xml file into the collection in *gli* using the gather panel. Place it in the same location as the programme file that it applies to; otherwise the data will not work.

Troubleshooting

Conductor as Player

The program will sometimes record a conductor as a player within the xml, causing the conductor’s name to appear under the Players browsing classifier when it should not. This bug has been fixed and tested thoroughly but should it occur again follow these steps to fix the problem.

1. Open the xml file where the problem is occuring.
2. Find the line with the conductor’s name: *<Metadata mode=’accumulate’*

name='pd.Player'>Buchanan-Smart, Andrew</Metadata>

3. Delete this line and save the document.
4. Re-build the collection and the name should have disappeared from the Players classifier.

Two names look the same but have different bookshelves

This occurs because of one of two reasons:

- The name is spelt incorrectly
- There is an extra space somewhere in the name

In order to fix this problem, locate the name in the xml file and spell it correctly or remove the space.

Metadata doesn't appear

If you have run the *Metadata Creator* twice on the same file it sometimes creates errors in the existing file, as you already have information in it. Delete the file and run the program again.

XML file is fine but metadata doesn't appear

Open the xml file and check that the file name, *<FileName>example\.pdf</FileName>,* is correct. Save the file. If this is fine then check that the metadata file is in the same location as the file specified in the *<FileName>* attribute. Re-build the collection and the data should appear.

6 Using Master Name Creator

The *Master Name Creator* is used to build a master list of all the names that have been included in the collection so that you can check for correct spellings. This file can be found in the *TWSO project* folder under *OCR Programs*. Before you run the program to create a new master list, be sure to keep an old version as back up to ensure that you never lose the list.

1. Open a new terminal and enter the command:
   python masternamecreator2.py example_list.txt example_list1.txt example_list2.txt...
2. The program will process the files you have specified and add the names to the master list. Duplicates are not included.
3. Check the master list to ensure that new entries were not added. This indicates that all the spellings of your names in the list are correct. If new entries do occur check that these are new names and not incorrectly spelt ones.
4. If an incorrectly spelt name occurs, remove the incorrect version from the master list and update it accordingly in the name list.

7 Upload a collection to Puka

1. Zip the collection as a tar.gz file.
    - Open a new terminal in the directory of the file that you want to zip.
    - Enter the command:
      tar –cvzf file-name.tar.gz file-name
2. Open a new terminal.
3. SSH into puka with the command:
   ssh puka
4. To update the collection you must have nzdl permissions. To do this enter the command:
   sudo su – nzdl
5. Locate the greenstone directory by using:
   cd /greenstone/custom/greenstone3-3.05-showcase/
6. Now stop the tomcat server. This is to prevent the redirect url from going to a cached version of the site instead of the updated version.
   source gs3-setup.sh
   ant stop
7. Locate the collection directory:
   cd web/sites/localsite/collect/
8. Check that you are in the correct directory by using the command:
   ls
9. If the collection is already on puka, deactivate the existing collection if you have not previously stopped the server in step 6. Re-name it by using the following commands:
    - In a web browser, enter the url where *collection-name* is the name of the collection you want to deactivate.
      http://www.nzdl.org/greenstone3-3.05-showcase/library?a=s&sa=d&st=collection&sn=collection-name
    - Re-name the file, where *myfile* is the name of your file.
      mv myfile myfile.orig
8. Copy the file from your machine to puka.
      scp -r username@machine:/path/to/file/filename.tar.gz .
    - *username* is your username.
    - *machine* is the machine you are working on.
    - */path/to/file/* is the path to the file you want to copy.
    - *filename* is the name of your file you want to copy.
    - The dot indicates that you want to copy the file into the current directory.
    - Dialogue will then appear to tell you how long it will take to copy the file.
9. Now untar the file and use *ls* to ensure that the file has unzipped properly:
   tar -xvzf filename.tar.gz
10. cd back to the *greenstone3-3.05-showcase* directory using:
    cd ../../../../..
11. Restart the server:
    ant start

12. Navigate to a web browser and enter the url below to activate the collection.
    http://www.nzdl.org/greenstone3-3.05-showcase/library?a=s&sa=a&st=collection&sn=collection-name
13. Navigate to your collection using the default link:
    http://www.nzdl.org/greenstone3-3.05-showcase/library/collection/collection-name/page/about
12. Logout of nzdl and puka by typing:
    logout

Rebuild a Collection on Puka

These steps are provided if you wish to update the collection directly on your computer without using gli.

1. Navigate to the greenstone3-3.05-showcase installation.
   cd /greenstone/custom/greenstone3-3.05-showcase
2. Setup the environment.
   source setup.sh
3. Re-build the collection.
   import.pl -collectdir web/sites/localsite/collect twso
   buildcol.pl -collectdir web/sites/localsite/collect twso
4. Restart the server.
   ant restart or
   ant stop
   ant start
5. Now check to ensure that collection has been update accordingly in a web browser.

8 OCR Programs folder

Inside the *TWSO project* folder is the *OCR Programs* folder. It contains seven files.

correct names.txt

This file stores the correct names for the pieces. It is used to avoid variants when new programmes are added. If you find a piece that is not within this file, add it to the file under the correct composer's name for future reference.

masternamecreator2.py

This is the python source code that is used to update the master list.

master_name_list.orig.txt

Original master list backup.

master_name_list.txt

This is the file that the *Master Name Creator* will update when you run it.

metadatacreator.py

Python source code used to create metadata.

namefinder1.py

Python source code used to create a specific name list for a specified programme.

roles.txt

This contains the names of all the roles ever found in the programmes currently in the collection. The *Name Finder* uses this file to locate names within the OCR document. If you find that the program is not picking up a particular name, check this file to make sure that the role exists. If it doesn't, add the new role to the list and run the program again to check that the role is now being picked up by the program.

9 Collection Material folder

This folder contains all the raw collection material that is included in the *twso* collection. It has has the original pdfs, images, OCR documents, name lists, and metadata.

Images

This folder has all the images that are included in the collection, including icons and the background gradient.

OCR Documents & XML

Here you will find the original OCR word documents, txt documents, name lists, and metadata xml files organized by year.

Programmes

In this folder you will find the pdf documents that have been included in the collection. The original documents at 600dpi are all named *file.orig.pdf*, while the 150dpi optimized pdf files are called *file.pdf*.

10 Adding a Video to the collection

Before you add a video to the collection it is recommended that you read the jwplayer "Working with Playlists" support documentation

(http://www.longtailvideo.com/support/jw-player/28842/working-with-playlists). This will provide you with all the information that you need to understand how the jwplayer works with playlists. Also note that all videos must be in .mp4 web optimized format to allow them to work with the player in all different types of browsers and to use streaming. You can convert your videos or rip them from DVDs using the *Handbrake 0.9.8* software which is free to download from the internet (http://handbrake.fr/downloads.php). If you need help using *Handbrake 0.9.8* please refer to their wiki (https://trac.handbrake.fr/wiki).

1. Add the .mp4 video to the *videos* folder in the *twso* collection folder. The reason the video is added here and not imported into the collection is to prevent long wait times when building the collection.
2. SSH into puka as seen in section 7 and locate the collect folder. Ensure that you stop the server.
3. *cd* into the *videos* folder in the *twso* collection and copy the video you want into this location using:
   scp -r username@machine:/path/to/file/video-name.mp4 .
4. Start the server again.
5. Open *gli* and go to the *Format* panel. Select *Format Features* from the menu. Select *CL7* from the classifier format menu.
6. Locate the jwplayer embed code and add the new video to the list following the instructions on the jwplayer site and in the same format as the previous videos.
   jwplayer("myElement").setup({
   playlist: [{
   image: "sites/localsite/collect/twso/images/sunset_0003.jpg",
   duration: '5826',
   sources: [{
   file:"http://www.nzdl.org/greenstone3-3.05-showcase/sites/localsite/collect/twso/videos/sunset.mp4" }],
   title: "Sunset Symphony Orchestra",
   description: "Hamilton Gardens Summer Festival, 25 February 2007" } ...
   - *image*: a display image that is shown before the video is played.
   - *duration*: the length of the video in seconds.
   - *sources*: array with one or more media sources for this playlist item.
   - *file:* url to video location. Only absolute urls work so include the url location of the video on puka instead of in the localhost folder as it will not work on the server otherwise.
   - *title*: name of the concert.
   - *description*: location and date the concert was performed.
7. Preview the collection to ensure that the video is working correctly.
8. See section 7 to update the collection.

Troubleshooting

For troubleshooting with the jwplayer please go to “Troubleshooting Your Setup” on the jwplayer

website (http://www.longtailvideo.com/support/jw-player/28840/troubleshooting-your-setup/). The jwplayer is very picky and often the main reason for a video not working is that the file location is wrong or there is a syntax error. The most common errors I found was leaving out a comma or typing the path name wrong. Ensure that this is not the problem before consulting the “Troubleshooting Your Setup” guide.