Building Digital Library Collections
with Greenstone

One-day Workshop

Ian H. Witten   ihw@cs.waikato.ac.nz

http://nzdl.org
http://greenstone.org

Course material prepared by

Greenstone Digital Library Project
University of Waikato, New Zealand

Workshop materials

❖ Workbook
  – introductory material
  – Greenstone blurb
  – lecture slides
  – tutorial exercises

❖ CD-ROM: UNESCO Greenstone Digital Library Software: Tutorial Edition
  – Greenstone software (v2.60), documented example collections, complete interface and full documentation in English/French/Spanish/Russian
  – As distributed by UNESCO, but with additional material:
    ▪ 40 language interfaces
    ▪ Tutorial exercises
    ▪ Sample files for tutorial exercises

❖ Greenstone manuals on the CD-ROM
  – *User's Guide* (user.pdf, 90pp)
  – *Installer's Guide* (install.pdf, 36pp)
  – *From Paper To Collection* (paper.pdf, 30pp)
  – *Developer's Guide* (develop.pdf, 113pp)

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Example

Humanity Development Library

for sustainable development and basic human needs

- ❖160,000 pages
- ❖30,000 images
- ❖800 books
- ❖430 magazines
- ❖340 kg
- ❖US$20,000

- ❖CD-ROM
- ❖US$1
- ❖Win3.1x upward
- ❖Stand-alone
- ❖*and* intranet server
- ❖Web browser user interface

Global Help Project, Antwerp (+ UN agencies)

What we wanted

ALA 2002

Greenstone turns a ragtag menagerie of documents in various formats into an easy-to-use collection that can run on a standalone laptop in a Ugandan village's information center

What we wanted

- ❖ "Collections" of digital material
- ❖ Individualized, depending on metadata etc
- ❖ Up to several Gb of text …
- ❖ … + associated images, movies, whatever
- ❖ Fully searchable
- ❖ Served on WWW, or published on removable media
- ❖ Run anywhere, on any computer
- ❖ Fully internationalized
- ❖ Non-exclusive: documents and metadata in any format
- ❖ Non-prescriptive: standard and non-standard metadata

What we got: Greenstone

Access ❖ Accessible via any Web browser
❖ Server runs on anything (all Windows + Unix + Mac)
❖ Collections can be published on CD-ROM/DVD
❖ Trivial to install
❖ GUI interface for building and publishing collections

Searching/ ❖ Collection-specific
browsing ❖ Full-text and fielded search
❖ Flexible browsing facilities
❖ Metadata-based (Dublin Core recommended)
❖ Creates all access structures automatically

Extensible ❖ Plugins — new document, metadata formats
❖ Classifiers — new metadata browsers

Multi-* ❖ Multilingual: Documents *and* interfaces
❖ Multimedia: image, video, audio collections exist
❖ Multiformat: Documents *and* metadata

UNESCO: Distributing Greenstone DL software

Sustainable development

*"Give a man a fish, feed him for a day
Teach a man to fish, feed him for life"*

Greenstone software on CD-ROM

❖ **GNU licensed**
❖ **Fully documented ... in English/French/Spanish/Russian**
❖ **Language interfaces ... Arabic Chinese Czech ... Thai Turkish**
❖ **Unix/Windows/Mac OS-X**
❖ **Trivial to install**
❖ **GUI interface for gathering, enriching, building ...**
❖ **Serve collections on Web or write them to CD-ROM**
❖ **Document formats: HTML, Word, PDF, PS, plain text, e-mail**
❖ **Metadata formats: XML, DC, OAI, MARC, ...**

download from *http://greenstone.org*

Greenstone facts

Distribution	❖ Open source: Gnu GPL
	❖ Distributed via SourceForge since: Nov
	❖ Average downloads: 5000/month sinc
	❖ Humanitarian CD-ROMs produced: 30-
	❖ Distribution for each one: 5000/year
International	❖ Languages for interface: 38
	❖ Languages for full software + manuals
	❖ Countries represented on email lists: 6
	❖ UNESCO training courses in:
	Bangalore, Almaty, Dakar, Suva, ..
UN Agencies	❖ UNESCO, Paris ("Information for All" p
	❖ FAO, Rome (Info Management Resour
	❖ UNU, Japan (CD-ROM collections of U
Technical centers	❖ University of Waikato, New Zealand
	❖ Indian Institute of Sciences, Bangalore
	❖ University College, London
	❖ University of Cape Town, South Africa
	❖ University of Lethbridge, Canada

Arabic
Armenian
Bosnian
Croatian
Deutsch (German)
English
Español (Spanish)
Farsi
Finnish
Français (French)
Galician
Georgian
Hebrew
Hindi
Indonesian
Italiano
Kannada
Māori
Nederlands (Dutch)
Serbian
Thai
Turkish
Ukrainian
português–BR (Brasil)
português–PT (Portugal)
Česky (Czech)
Ελληνικά (Greek)
Русско (Russian)
Қазақ (Kazakh)
日本語 (Japanese)
汉语 (Chinese)

Sample collections

U.S.		
	Auburn University	Alabama
	University of California at Riverside	California
	Hawaiian Electronic Library	Hawaii
	University of Chicago Library	Illinois
	University of Illinois	Illinois
	Illinois Wesleyan University	Illinois
	Washington Research Library Consortium	Maryland
	Detroit Public Library	Michigan
	New York Botanical Garden	New York
	ibiblio project, University of North Carolina	North Carolina
	LeHigh University, Pennsylvania	Pennsylvania
	Texas A&M University	Texas
International		
	Argentina Human Rights Commission	Argentina
	Tasmania State Library	Australia
	Peking University Digital Library	China
	Gresham College, London	England
	University of Applied Sciences, Stuttgart	Germany
	Association of Indian Labour Historians, Delhi	India
	Indian Institute of Management, Kozhikode	India
	Indian Institute of Science, Bangalore	India
	Vimercate Public Library, Milan, Italy	Italy
	Netherlands Institute for Scientific Info Services	Netherlands
	Philippine Government Information Network	Philippines
	Mari El Republic, Russia	Russia
	Slavonski Brod Public Library, Slovenia	Slovenia
	Vietnam National University	Vietnam
	Welsh Books Council	Wales

Standards

Metadata ❖ Can use any metadata set, Dublin Core supplied
❖ Plugins for

XML	**Refer**
MARC	**OAI**
CDS/ISIS	**METS (subset)**
ProCite	**DSpace**
BibTex	

❖ METS can be used as Greenstone's internal representation

Serving ❖ Web
❖ Can publish Greenstone collections on CD-ROM
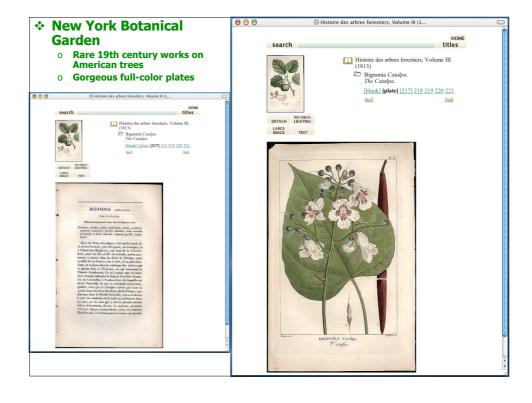❖ Can publish Greenstone collections on OAI
❖ Export collections to METS
❖ Export collections to DSpace (*ready for DSpace's batch import program*)

Documents ❖ Plugins for

PDF	**ZIP**	**Images**
PostScript	**Excel**	**(any format: GIF, JPEG, TIFF …)**
Word, RTF	**PPT**	**MP3**
HTML	**Email**	**Ogg Vorbis**
Plain text	**Source code**	**UnknownPlug**
Latex		**(e.g. for audio, MPEG, Midi)**

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

❖ **New York Botanical Garden**
  o **Rare 19th century works on American trees**
  o **Gorgeous full-color plates**

Chinese documents
(pictures of text)
+ Chinese interface

Peking University Library

Chinese
(Chinese & English
interfaces)

Classic Chinese literature

French

UNESCO, Paris

Spanish

PAHO, WHO

Russian

Mari El Republic
http://gov.mari.ru/gsdl

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Tutorial exercise #3
Installing Greenstone
(local library version)

Greenstone: Platforms

❖ Operating system:
  – Windows (any version)
  – Linux (any version)
  – Unix (most versions, e.g. Solaris)
  – Mac OS X (some problems with GLI interface)
❖ Restrictions:
  – No longer runs under Windows 3.1/3.11
  – For Librarian interface (GLI), need Java—
    which is no longer supported on Windows 95
❖ Disk space
  – 50 MB for a binary installation
  – 215 MB for the example collections (optional)
  –   5 MB for online documentation
  – 25 MB for "export to CD" function

Local library vs Web library

❖ **Local library: stand-alone**
  – Serves collections on a standalone PC …
  – And on others on the same network
  – Includes built-in Web server

❖ **Web library: uses external web server**
  – Apache, Microsoft PWS/IIS
  – Need to configure web server (geeky job?)

❖ **Windows: Both local library and web library**
  – All versions: (3.1, 3.11,) 95, 98, NT, 2000, ME, XP
  – Binaries supplied
  – Normally use local library (else must set up server)
  – Web library works with Microsoft PWS, IIS

❖ **Unix, Mac OS/10: Web library only**
  – Use Apache (or other web server)
  – Linux binaries supplied
  – Tested on SUN Solaris, Mac OS/10
  – Need GDBM (standard on Linux)

❖ **Greenstone developed on Linux**

Documentation and Help

Manuals on the CD-ROM (gsdl/docs)
  – *From Paper To Collection* (paper.pdf, 30pp)

    Scanners and scanning, OCR, 3 examples – from 1,000 to 100,000 pages, Creating an electronic collection

  – *Installer's Guide* (install.pdf, 36pp)

    Versions of Greenstone, installation procedure, Greenstone collections, setting up the web server, configuring your site, personalizing your installation

  – *User's Guide* (user.pdf, 90pp)

    Overview of Greenstone, using Greenstone collections, the collector, administration, software features, glossary of terms

  – *Developer's Guide* (develop.pdf, 113pp)

    Understanding the collection building process, getting the most out of your collections, the Greenstone runtime systems, configuring your Greenstone site

Documentation and Help

❖ *Installer's Guide* (install.pdf)
   – Versions of Greenstone, installation procedure,
   – Greenstone collections, setting up the web server
   – Configuring your site, personalizing your installation

❖ *FAQ*: sections on
   – Obtaining Greenstone
   – Installing Greenstone
   – Running Greenstone

❖ *Tutorial exercises*
   – #3 Installing Greenstone (on Windows)
   – #4 Updating a Greenstone installation

Documentation and Help

❖ greenstone.org
   – Examples
   – support
     (+ join mailing list)
   – Configuration files for
     nzdl.org collections
❖ FAQ
   – General info section
❖ nzdl.org
   – Documented
     example collections

Example collections

development library subset
The DLS collection has the same structure as the Greenstone demo collection. It's fairly complex, and if you're just starting out you might prefer to look at some other collections first (e.g. msword and pdf demonstration, or the greenstone archives, or the simple image collection).

msword and pdf demonstration
This contains a few documents in PDF, MSWord, RTF, and Postscript formats, demonstrating the ability to build collections from documents in different formats. The collection configuration file is very simple.

greenstone archives
A collection of email messages from the Greenstone mailing list archives, this uses the Email plugin, which parses files in email formats. The collection configuration file is very simple.

bibliography collection
With about 4,000 bibliography entries, this collection incorporates a form-based search interface that allows fielded searching. It is fairly complex.

bibliography supplement
This tiny collection of 10 bibliography entries illustrates the "supercollection" facility which searches several collections together, seamlessly. It operates together with the Bibliography collection, and its configuration file is almost the same.

MARC example
Based on some MARC records from the Library of Congress, this is a simple collection (and does not allow form-based searching).

Example collections

OAI example
Using the Open Archive Protocol and the Import-From feature, this retrieves metadata from an archive and builds a collection from the records. In this case they are images, so both the OAI and Image plugins are used.

simple image collection
This very basic image collection contains no text and no explicit metadata -- which makes it rather unrealistic. The configuration file is about as simple as you can get.

authentication and formatting demo
With the same material as the original Greenstone demo collection, this shows off two independent features: non-standard document formatting, and controlled access to the documents via user authentication.

garish demo
This collection also contains the same material as the Greenstone demo. Its appearance has been altered to show how the pages generated can be set out differently. It relies on a non-standard macro file that is supplied with Greenstone.

CDS/ISIS example
This collection is built from a CDS/ISIS database of about 150 bibliography entries. It uses the ISISPlug plugin, which reads the standard ISIS .mst and .fdt files and converts them to Greenstone metadata.

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Tutorial exercise #5
Small collection of HTML files

Invoke GLI: build a small collection of HTML files
❖**Gather**
❖**Create**
❖**Look at extracted metadata**
❖**Set up shortcut in the Librarian interface**

Tutorial exercise #6
Collection of Word and PDF files

Gather
Create
Enrich: **Manually add metadata (dc.Title, dc.Creator)**
Design: **Brand with image**
       Remove plugins
       Both Form and Plain search
       Indexes: text, title, author
       Browsers: title (AZList), author (AZCompactList)
Create
Write to CD-ROM

Tutorial exercise #8
Simple image collection

Gather
Create
Enrich: **Add Dublin Core metadata set**
       Add dc.Title metadata
Design: **Change format to display titles; re-build**
       Change thumbnail size; re-build
Enrich: **Add dc.Description metadata**
Design: **Change format to display description**
       Add browser for descriptions (AZList)
       Add searchable index based on descriptions

<div style="border:1px solid">

Other tutorial exercises

#9　　**A large collection of HTML files—Tudor**
Looking at different views of the files in the Gather and Enrich panels

#10　**Exporting a collection to CD-ROM**
Very simple to do!

#11　**Pointing to documents on the web**

#13　**Enhanced collection of HTML files**
Adding hierarchically-structured metadata and a Hierarchy classifier
Partitioning the full-text index based on metadata values
Adding a hierarchical phrase index (PHIND)
Controlling the building process

#16/17 Looking at/building a multimedia collection
Manually correcting metadata; Browsing by media type
Suppressing dummy text; Using AZCompactList rather than AZList
Making bookshelves show how many items they contain
Using UnknownPlug; Cleaning up a title browser using regular expressions
Using non-standard macro files; Using different icons for different media types
Changing the collection's background image
Building a full-size version of the collection; Adding an image collage browser

#18　**Scanned image collection**
Grouping documents by series title and displaying dates within each group
Suppressing dummy text

</div>

<div style="border:1px solid">

Tutorial exercise #9, 11, 13
Large HTML collection — Tudor

#9　**A large collection of HTML files—Tudor**
Looking at different views of files in Gather/Enrich panels

#11 **Pointing to documents on the web**

#13 **Enhanced collection of HTML files**
Adding hierarchically-structured metadata
Adding a Hierarchy classifier
Partitioning the full-text index based on metadata values
Adding a hierarchical phrase index (PHIND)
Controlling the building process

</div>
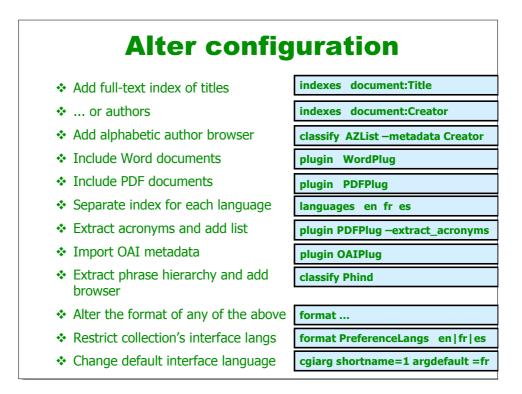
Tutorial exercise #16, 17
Building a multimedia collection

❖ **Manually correcting metadata**
❖ **Browsing by media type**
❖ **Suppressing dummy text**
❖ **Using AZCompactList rather than AZList**
❖ **Making bookshelves show how many items they contain**
❖ **Using UnknownPlug**
❖ **Cleaning up a title browser using regular expressions**
❖ **Using non-standard macro files**
❖ **Using different icons for different media types**
❖ **Changing the collection's background image**
❖ **Building a full-size version of the collection**
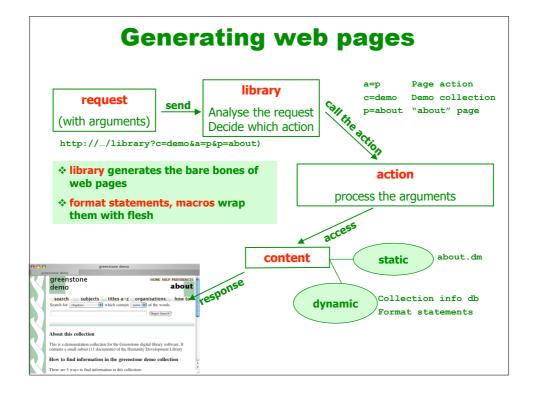❖ **Adding an image collage browser**

Tutorial exercise #18
Scanned image collection

❖ **Grouping documents by series title and displaying dates within each group**
❖ **Suppressing dummy text**

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Collection configuration file

❖ **name, icon, etc**
❖ **description**
❖ **email of creator**
❖ **search indexes**
❖ **plugins**
❖ **classifiers**

how to format

   ❖ **documents**
   ❖ **query results**
   ❖ **classifiers**

```
creator      sjboddie@cs.waikato.ac.nz
maintainer   sjboddie@cs.waikato.ac.nz
public       true
beta         true


indexes      section:text section:Title document:text
defaultindex section:text

plugin       GAPlug
plugin       ArcPlug
plugin       RecPlug

classify     Hierarchy -hfile sub.txt -metadata Subject -sort Title
classify     HDLList -metadata Title
classify     Hierarchy -hfile org.txt -metadata Organization -sort Title
classify     List -metadata Howto

format       SearchVList "<td valign=top>[link][icon][/link]</td>
             <td>{If}{[parent(All': '):Title],[parent(All': '):Title]: }
             [link][Title][/link]</td>"
format       CL4VList    "<br>[link][Howto][/link]"
format       DocumentImages true
format       DocumentText "<h3>[Title]</h3>\\n\n<p>[Text]"

collectionmeta collectionname  "greenstone demo"
collectionmeta collectionextra "This is a demonstration collection for the
             Greenstone digital library software.\nIt contains a small
             subset (11 books) of the Humanity Development Library"
collectionmeta iconcollectionsmall "/gsdl/collect/demo/images/demosm.gif"
collectionmeta iconcollection      "/gsdl/collect/demo/images/demo.gif"
collectionmeta .section:Title      "section titles"
collectionmeta .document:text      "entire books"
collectionmeta .section:text       "chapters"
```

Alter configuration

❖ Add full-text index of titles	**indexes document:Title**
❖ … or authors	**indexes document:Creator**
❖ Add alphabetic author browser	**classify AZList –metadata Creator**
❖ Include Word documents	**plugin WordPlug**
❖ Include PDF documents	**plugin PDFPlug**
❖ Separate index for each language	**languages en fr es**
❖ Extract acronyms and add list	**plugin PDFPlug –extract_acronyms**
❖ Import OAI metadata	**plugin OAIPlug**
❖ Extract phrase hierarchy and add browser	**classify Phind**
❖ Alter the format of any of the above	**format …**
❖ Restrict collection's interface langs	**format PreferenceLangs en|fr|es**
❖ Change default interface language	**cgiarg shortname=1 argdefault =fr**

Generating web pages

a=p   Page action
c=demo   Demo collection
p=about   "about" page

request
(with arguments)

→ **send** →

library
Analyse the request
Decide which action

call the action

`http://…/library?c=demo&a=p&p=about)`

❖ **library generates the bare bones of web pages**

❖ **format statements, macros wrap them with flesh**

action
process the arguments

access

content

response

static   `about.dm`

dynamic   `Collection info db`
                   `Format statements`

greenstone demo

greenstone demo

greenstone
demo   **about**   HOME HELP PREFERENCES

search   subjects   titles a–z   organisations   how to
Search for [chapters] which contain [some] of the words

[   ] [Begin Search]

About this collection

This is a demonstration collection for the Greenstone digital library software. It
contains a small subset (11 documents) of the Humanity Development Library

How to find information in the greenstone demo collection

There are 5 ways to find information in this collection:

Tutorial exercise #14
Learning about formats and macros

❖ **Experimenting with format statements**
❖ **Collection-specific macros**
❖ **General macros**
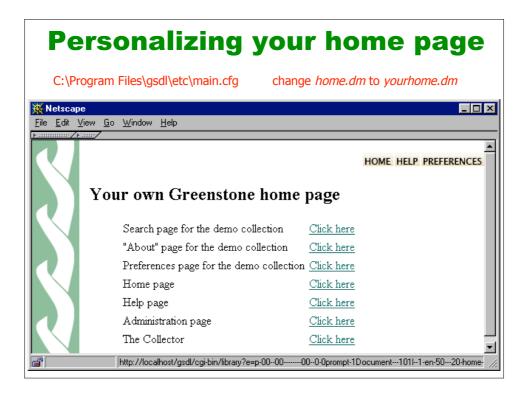
Customizing with macros

– let you customize presentation
– present pages in different languages
– print variables into the page text
   (e.g. number of search hits)

❖ Macro files
   – stored in gsdl/macros folder
   – named *.dm
   – each file defines one or more "packages"
   – loaded on startup
      (note difference between Local and Web Library)
   – listed in etc/main.cfg

❖ Collection-specific macros
   – put in gsdl/collect/mycol/macros/extra.dm
   – Or include argument [c=collectionname] for each macro

❖ A "package" is a group of macros

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page ⬅
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Personalizing your home page

C:\Program Files\gsdl\etc\main.cfg   change *home.dm* to *yourhome.dm*

Netscape

File   Edit   View   Go   Window   Help
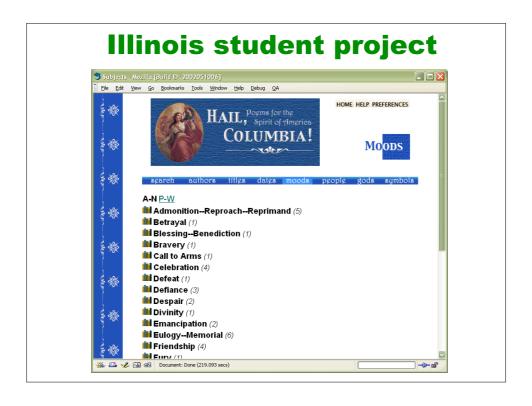
HOME HELP PREFERENCES

Your own Greenstone home page

Search page for the demo collection	Click here
"About" page for the demo collection	Click here
Preferences page for the demo collection	Click here
Home page	Click here
Help page	Click here
Administration page	Click here
The Collector	Click here

http://localhost/gsdl/cgi-bin/library?e=p-00-00---------00-0-0prompt-1Document---101l--1-en-50---20-home-

```
package home
_content_ {
<h2>Your own Greenstone home page</h2>
<ul>
<table>
<tr valign=top><td>Search page for the demo collection<br></td>
    <td><a href="_httpquery_&c=demo">Click here</a></td></tr>

<tr><td>"About" page for the demo collection</td>
    <td><a href="_httppageabout_&c=demo">Click here</a></td></tr>

<tr><td>Preferences page for the demo collection</td>
    <td><a href="_httppagepref_&c=demo">Click here</a></td></tr>

<tr><td>Home page</td>
    <td><a href="_httppagehome_">Click here</a></td></tr>

<tr><td>Help page</td>
    <td><a href="_httppagehelp_">Click here</a></td></tr>

<tr><td>Administration page</td>
    <td><a href="_httppagestatus_">Click here</a></td></tr>

<tr><td>The Collector</td>
    <td><a href="_httppagecollector_">Click here</a></td></tr>

</table>
</ul>
}
```

yourhome.dm

Macros used in home.dm

httppagehome	name of the home page
httppagehelp	… the help page
httppagestatus	… the administration page
httppagecollector	… the Collector page
httpquery&c=demo	search page for the demo collection
httppageabout&c=demo	about page for the demo collection
httppagepref&c=demo	preferences page for the demo collection
_content_{ … }	defines a macro called _content_ contains HTML, but '{', '}', '\', and '_' must be escaped with a backslash
_header_{ … }	HTML page header (contains squirly bar)
_footer_{ … }	HTML page footer
main.cfg	contains list of macros, replace home.dm by yourhome.dm and put it in the macros directory

Macro syntax

- ❖ Names begin and end with underscore
- ❖ Content is defined using curly brackets
  - _macroname_ [Argument] { content of the macro ...}
- ❖ Content can be
  - – plain text
  - – HTML
  - – links to Java applets and JavaScript
  - – other macro names
- ❖ can contain conditional statements
  - – _If_ ...
- ❖ can take arguments
  - – a=p, l=de etc.

---

about.dm


```
package about

##################################
# about page content
##################################

_pagetitle_ {_collectionname_}

_content_ {
<center>
_navigationbar_
</center>
_query:queryform_
<p>_iconblankbar_
<p>_textabout_
_textsubcollections_
<h3>_help:textsimplehelpheading_</h3>
_help:simplehelp_
}

_textabout_ {
<h3>_textabcol_</h3>
_Global:collectionextra_
}
```

Customization hints

❖ To change the look and feel of Greenstone
  – edit the *base* and *style* packages

❖ To change the query page
  – edit *query.dm*

❖ To change the Preferences page
  – edit *pref.dm*

Different interface languages

english.dm

```
#--------------------------------------------------
# text macros
#--------------------------------------------------

_textimageabout_ {About page}
_textimagehelp_ {Help page}
_textimagepref_ {Preferences page}
_textimagegreenstone_ {Greenstone Digital Library Software}

_textimagesearch_ {Search for specific terms}
_textimageCreator_ {Browse alphabetical list of authors}
_textimageTitle_ {Browse alphabetical list of titles}
_textmatches_ {Matches }
_textbeginsearch_ {Begin Search}

_texticonsmalltext_ {View this section of the text}
_texticondetach_ {Open this page in a new window}
_texticonexpandtoc_ {Expand table of contents}
_texticonexpandtext_ {Display all text}
_texticonhighlight_ {Highlight search terms}

_textlanguage_ {Interface language: }
_textsimplemode_ {simple query mode}
_textadvancedmode_ {advanced query mode (allows boolean searching u
```

chinese.dm

```
#------------------------------------------------------
# text macros
#------------------------------------------------------

_textimageabout_ [l=zh] {主页}
_textimagehelp_ [l=zh] {帮助页}
_textimagepref_ [l=zh] {选项页}
_textimagegreenstone_ [l=zh] {绿石数字化图书馆软件}

_textimagesearch_ [l=zh] {特定范围检索}
_textimageCreator_ [l=zh] {按作者名字的阿拉伯字母顺序浏览 }
_textimageTitle_ [l=zh] {按题名的阿拉伯字母顺序浏览}
_textmatches_ [l=zh] {匹配 }
_textbeginsearch_ [l=zh] {开始查询}

_texticonsmalltext_ [l=zh] {  浏览该部份文本}
_texticondetach_ [l=zh] {  打开一个新的窗口显示本页}
_texticonexpandtoc_ [l=zh] {  扩展书录表}
_texticonexpandtext_ [l=zh] {  显示所有文本}
_texticonhighlight_ [l=zh] {  查询项以粗体显示}

_textlanguage_ [l=zh] {界面语言：}
_textsimplemode_ [l=zh] {简单查询方式}
_textadvancedmode_ [l=zh] {高级查询方式(可以使用！、&、|和括号来布尔查询)}
```

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Illinois student project

No structural changes

Strong branding through images

Illinois student project

Illinois student project

Illinois student project

Illinois student project

"How to build" web site

One customized plugin

JavaScript to tweak formatting

Single collection *is* site

"How to build" web site

"How to build" web site

Zespri innovation library

Embedded in larger site (frames)

Structural variation to macros

Referer mechanism (security)

Zespri innovation library

Zespri innovation library

Archives of Indian labour

Login security to whole site

Home page customized

Archives of Indian labour

Archives of Indian labour

Gresham College archive

Embedded in larger site (frames)

Blended with other technologies

Issued on CD-ROM

Gresham College archive

Gresham College archive

The NY Botanical Gardens

Single collection *is* site

Supplementary information provided through static pages

Redesign of *about collection* page

The NY Botanical Gardens

Lehigh University

Single collection *is* site

Strongly branded

Search feature embedded into essentially static site

Lehigh University

Lehigh University

Lehigh University

Lehigh University

Aladin digital library system

Greenstone is component of larger digital library system

Access to browse redefined

Aladin digital library system

Aladin digital library system

Aladin digital library system

Aladin digital library system

Aladin digital library system

Aladin digital library system

Aladin digital library system

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Tutorial exercises

#15 A bibliographic collection
Adding fielded searching
Branding the collection with an image

#19 Open Archives Initiative (OAI) collection
Tweaking the presentation with format statements

#20 Downloading over OAI

#21 Exporting a collection as METS

#22 Moving a collection from DSpace to Greenstone
Adding indexing and browsing capabilities to match DSpace's

#23 Moving a collection from Greenstone to DSpace
Using Greenstone from the command line

Building a collection from MARC records

Sample record

```
001 978640
005 19950323092224.3
008 850731s1984    nz b          f000 0 eng
035    $9 (DLC)   85186085
906    $a 7 $b cbc $c orignew $d 3 $e ncip $f 1
010    $a    85186085
040    $a DLC $c DLC $d DLC
043    $a t------ $a u-nz---
050 00 $a G877 $b .A58 1984
082 00 $a 998/.9 $2 19
245 00 $a Antarctica and New Zealand.
260    $a Wellington, N.Z. : $b Ministry of Foreign Affairs, $c [1984]
300    $a 48 p. : $b maps ; $c 29 cm.
490 1  $a Information bulletin / Ministry of Foreign Affairs $x 0111-8315 ; $v
          no. 8 (Aug. 1984)
500    $a Cover title.
651  0 $a Antarctica.
651  0 $a New Zealand.
710 1  $a New Zealand. $b Ministry of Foreign Affairs
830  0 $a Information bulletin (New Zealand. Ministry of Foreign Affairs) ;
$vo.
          8.
991    $b c-GenColl $h G877 $i .A58 1984 $t Copy 1 $w BOOKS
```

001 Control number uniquely identifying the record
005 Date and time that the record was last modified
008 Fixed fields
010 Library of Congress control number
015 National Bibliographic number
035 System control number
040 Cataloging source
050 Library of Congress classification
082 Dewey classification
100 Main entry -- personal name
245 Title
260 Imprint: place of publication, publisher, date
300 Physical description
500 General note
504 Bibliography note
650 Subject entry
906 Tags in the 900 range are reserved for local use,
985 and are used by vendors, systems, or individual
991 libraries to exchange additional data

Tutorial exercise #15
Bibliographic collection

❖ **Adding fielded searching**
❖ **Branding a collection with an image**

Dublin Core Metadata Initiative
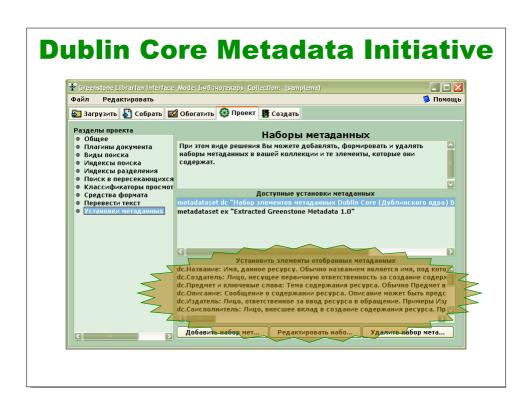
A metadata element set designed to facilitate discovery of electronic resources
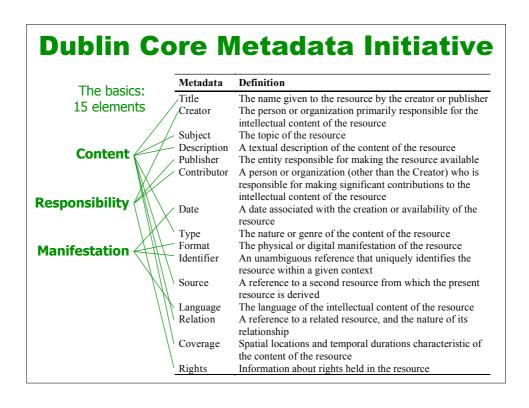
❖ Originally intended for non-specialist use, such as author-generated description of Web resources

❖ Upon publication, saw rapid uptake worldwide

❖ Used by people and for applications way beyond its original scope

– In particular formal resource description communities such as museums, libraries, government agencies and commercial organizations.

Key characteristics

❖ Simplicity

❖ Semantic interoperability

❖ International consensus

❖ Extensibility

Dublin Core Metadata Initiative

Dublin Core Metadata Initiative

Dublin Core Metadata Initiative

The basics:
15 elements

Content

Responsibility

Manifestation

Metadata	Definition
Title	The name given to the resource by the creator or publisher
Creator	The person or organization primarily responsible for the intellectual content of the resource
Subject	The topic of the resource
Description	A textual description of the content of the resource
Publisher	The entity responsible for making the resource available
Contributor	A person or organization (other than the Creator) who is responsible for making significant contributions to the intellectual content of the resource
Date	A date associated with the creation or availability of the resource
Type	The nature or genre of the content of the resource
Format	The physical or digital manifestation of the resource
Identifier	An unambiguous reference that uniquely identifies the resource within a given context
Source	A reference to a second resource from which the present resource is derived
Language	The language of the intellectual content of the resource
Relation	A reference to a related resource, and the nature of its relationship
Coverage	Spatial locations and temporal durations characteristic of the content of the resource
Rights	Information about rights held in the resource

Dublin Core Metadata Initiative: Unqualified

```
<head>
  <META NAME="DC.Creator"   CONTENT="State Services Commission">
  <META NAME="DC.Publisher" CONTENT="State Services Commission">
  <META NAME="DC.Rights"    CONTENT="http://www.ssc.govt.nz/copyright.htm">
  <META NAME="DC.Title"     CONTENT="E-government">
  <META NAME="DC.Subject"   CONTENT="public administration">
  <META NAME="DC.Identifier" CONTENT="http://www.sc.govt.nz/index.html">
  <META NAME="DC.Description" CONTENT="This page provides access to
     information about the New Zealand government's electronic government
     programme.">
  <META NAME="DC.Language"  CONTENT="en">
  <META NAME="DC.Coverage"  CONTENT="New Zealand">
  <META NAME="DC.Date"      CONTENT="2001-04-26">
  <META NAME="DC.Type"      CONTENT="home page">
  <META NAME="DC.Format"    CONTENT="text/html">
</head>
```

Unqualified Dublin Core

Sample XML record (Open Archives compliant)

```
<oai_dc>
  <dc:creator>State Services Commission</dc:creator>
  <dc:publisher>State Services Commission</dc:publisher>
  <dc:rights>http://www.ssc.govt.nz/copyright.htm</dc:rights>
  <dc:title>E-government</dc:title>
  <dc:subject>public administration</dc:subject>
  <dc:identifier>http://www.sc.govt.nz/index.html</dc:identifier>
  <dc:description>
     This page provides access to information about the New Zealand
     government's electronic government programme.
  </dc:description>
  <dc:language>en</dc:language>
  <dc:coverage>New Zealand</dc:coverage>
  <dc:date>2001-04-26</dc:date>
  <dc:type>home page</dc:type>
  <dc:format>text/html</dc:format>
</oai_dc>
```

Unqualified Dublin Core

"Basic" Dublin Core (unqualified) can be *too* general.
The standard defines *qualified* Dublin Core, which enables more precise information to be provided.

Two broad classes of qualifiers:

❖ **Element Refinement**
– Makes the meaning of an element narrower or more specific.
– A refined element shares the meaning of the unqualified element, but with a more restricted scope.
– Can therefore still be handled as unqualified if desired by ignoring qualifier.

❖ **Encoding Scheme**
– Identify schemes that aid in the interpretation of an element value.
– Includes controlled vocabularies and formal notations or parsing rules.
– Does not necessarily have to imply a machine-readable schema

Qualified Dublin Core

```
<head>
  <META NAME="DC.Creator"
    CONTENT="corporateName=State Services Commission">
  <META NAME="DC.Publisher"  SCHEME=AGLS Agent"
    CONTENT="corporateName=State Services Commission">
  <META NAME="DC.Rights"     SCHEME="URI"
    CONTENT="http://www.ssc.govt.nz/copyright.htm">
  <META NAME="DC.Title"        CONTENT="E-government">
  <META NAME="DC.Subject"     SCHEME="SONZ" CONTENT="public administration">
  <META NAME="DC.Identifier" SCHEME="URI"
    CONTENT="http://www.sc.govt.nz/index.html">
  <META NAME="DC.Description"
    CONTENT="This page provides access to information about the New Zealand
        government's electronic government programme.">
  <META NAME="DC.Language"        SCHEME="RFC3066"   CONTENT="en">
  <META NAME="DC.Coverage.jurisdiction"         CONTENT="New Zealand">
  <META NAME="DC.Date.modified" SCHEME="ISO8601"  CONTENT="2001-04-26">
  <META NAME="DC.Type.category" SCHEME="NZGLS"     CONTENT="Document">
  <META NAME="DC.Type.Text"     SCHEME="NZGLS"     CONTENT="home page">
  <META NAME="DC.Type.aggregationLevel"          CONTENT="collection">
  <META NAME="DC.Format"          SCHEME="IMT"       CONTENT="text/html">
</head>
```

"Encoding scheme" qualifier (includes controlled vocab)

"Element refinement" qualifier

Dublin Core crosswalk

Greenstone's MARC to Dublin Core mapping
based on LOC's MARC ⇒ DC crosswalk

```
### Creator        ###
# 720 = "Uncontrolled Name"
#   with $e=author
720^e -> Creator
# 100 = "Personal Name"
100-> Creator
# 110 = "Corporate Name"
110-> Creator

### Source
# Greenstone has its own Source
# metadata...
# 786 = "Data Source Entry"
786 -> MarcSource

### Title        ###
# 245 = "Title Statement"
245 -> Title
```

```
### Subject        ###
# 650 = "Subject: Topical Term"
650 -> Subject
# 653 = "Index Term: Uncontrolled"
653 -> Subject

### Publisher      ###
# 260 = "Publication, etc"
260 -> Publisher

### Relation       ###
# 787 = "Nonspecific Relationship Note"
787 -> Relation

### Rights         ###
# 540 = "Reproduction Note"
540 -> Rights
...
```

Dublin Core crosswalk

Greenstone's MARC to Dublin Core mapping:

```
### Description  ###
# 520 = "Summary, Note"
520 -> Description

### Type ###
# 655 = "Index Term - Genre/Form"
655 -> Type

### Identifier    ###
# 024 = "Standard Identifier/number"
# greenstone uses its own Identifier metadata
024 -> MarcIdentifier

# 856 = "Electronic Location"
856 -> URL

### Language      ###
# Greenstone has its own Language metadata...
546 -> MarcLanguage
```

❖ Just a text file
❖ Obviously we should use XSLT …
❖ … as we do for MODS (see later)

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Open Archives Initiative

OAI Protocol for Metadata Harvesting

❖ Provides an application-independent interoperability framework based on *metadata harvesting*.

❖ Two classes of participants:

– *Data Providers* expose metadata

– *Service Providers* use metadata harvested from data providers as a basis for building value-added services.

Renaming of client-server model to emphasize interaction driven by client.

Open Archives Initiative

6 verbs

OAI Request	Description
Identify	Return both fixed format and domain - specific descriptions.
ListSets	Returns the repository 's classification hierarchy.
ListIdentifiers	Returns a list of document identifiers.
ListMetadataFormats	Returns th e metadata formats supported by the repository in general or for a specific document.
GetRecord	Returns the repository item specified by the document identifier in the requested format.
ListRecords	Returns a list of repository items in the requeste d format.

The Greenstone OAI server

❖ Runs as a CGI program called *oaiserver*
– Greenstone installation
`http://.../cgi-bin/library`
– OAI server

OAI 1.1
OAI 2.0

`http://.../cgi-bin/oaiserver?verb=Identify`
`http://.../cgi-bin/oaiserver?verb=ListSets`
`http://.../cgi-bin/oaiserver?verb=ListIdentifiers&set=xxx`
`http://.../cgi-bin/oaiserver?verb=ListIdentifiers&set=xxx&metadataPrefix=oai_dc`
`http://.../cgi-bin/oaiserver?verb=ListRecords&set=xxx&metadataPrefix=oai_dc`
`http://.../cgi-bin/oaiserver?verb=GetRecord&identifier=xxx&metadataPrefix=oai_dc`

❖ Requires a full webserver (not "local library" version)

❖ Configuration file: `etc/oai.cfg` in the Greenstone filespace
– repository name and version (OAI 1.1 or 2.0)
– collections to be made accessible to OAI clients
– metadata mapping file into DC (server only supports DC)

Open Archives Initiative: Sample

```
<?xml version="1.0" encoding="UTF-8"?>
<OAI-PMH >
  <responseDate>2005-05-05T02:44:29Z</responseDate>
  <request verb="GetRecord" identifier="oai:celebration:addams/hullhouse"
      metadataPrefix="oai_dc">http://digital.library.upenn.edu/webbin/OAI-celebration
  </request>
    <GetRecord>
      <record>
        <header>
          <identifier>oai:celebration:addams/hullhouse</identifier>
          <datestamp>2005-01-03</datestamp>
        </header>
        <metadata>
          <oai_dc:dc  >
            <dc:title>Twenty Years at Hull-House, With Autobiographical Notes</dc:title>
            <dc:contributor>Addams, Jane</dc:contributor>
            <dc:subject>Social settlements—Illinois—Chicago—History.</dc:subject>

    <dc:identifier>http://digital.library.upenn.edu/women/addams/hullhouse/hullhouse.html<
    /dc:identifier>
            …
            <dc:date>1998-03-10</dc:date>
          </oai_dc:dc>
        </metadata>
      </record>
    </GetRecord>
</OAI-PMH>
```

Tutorial exercise #20, #19
OAI collection

Using OAI-PMH, build a Greenstone collection (called oai-e) based on metadata exported from the JCDL01 OAI server rocky.dlib.vt.edu.

1. **Use Greenstone's importfrom.pl command**
2. Use Greenstone's import.pl and buildcol.pl commands to build a service provider based on the exported metadata.

OAI Collection: acquisition

Using `importfrom.pl` acquire metadata from the external data provider:

```
gsdl% importfrom.pl oai-e
OAI Acquire: from rocky.dlib.vt.edu/~jcdlpix/cgi-
   bin/OAI1.1/jcdlpix.pl
Requesting list of identifiers ...
... Done.
Downloading metadata record for oai:JCDLPICS:200101dla1.oai
Getting document
   http://rocky.dlib.vt.edu/~jcdlpix/pictures/200104dla/01dla1.jpg
Downloading metadata record for oai:JCDLPICS:200101dla2.oai
Getting document
   http://rocky.dlib.vt.edu/~jcdlpix/pictures/200104d1a/01dla2.jpg
Downloading metadata record for oai:JCDLPICS:200101dla3.oai
Getting document
   http://rocky.dlib.vt.edu/~jcdlpix/pictures/200104dla/01dla3.jpg
…
Number of documents processed: 81
```

OAI Collection: acquisition

Excerpts from Greenstone collection configuration file.
Used by `importfrom.pl`, `import.pl` and `buildcol.pl`

```
acquire OAI -src rocky.dlib.vt.edu/~jcdlpix/cgi-bin/OAI1.1/
   jcdlpix.pl –getdoc
#...

plugin  OAIPlug -input_encoding iso_8859_1 -default_language en
plugin  ImagePlug -screenviewsize 300
plugin  GAPlug
plugin  ArcPlug
plugin  RecPlug -use_metadata_files -show_progress
#...

classify AZCompactList -metadata Subject -doclevel top
classify AZCompactList -metadata Description -buttonname Captions
#...

format VList "<td>[link][thumbicon][/link]</td> \
             <td valign=middle><i>[Description]</i></td>"
```
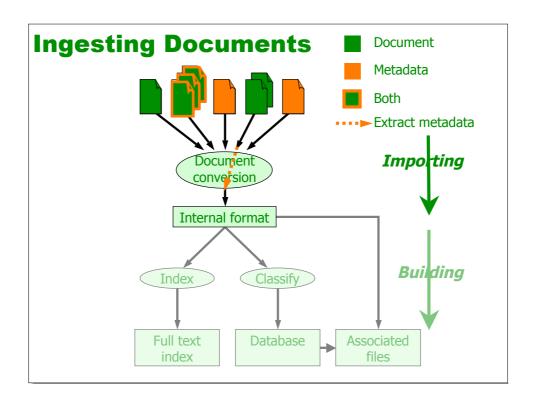
Tutorial exercise #20, #19
OAI collection

Using OAI-PMH, build a Greenstone collection (called oai-e) based on metadata exported from the JCDL01 OAI server rocky.dlib.vt.edu.

1. Use Greenstone's importfrom.pl command
2. **Use Greenstone's import.pl and buildcol.pl commands to build a service provider based on the exported metadata.**

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Ingesting Documents

- Document
- Metadata
- Both
- ▸ Extract metadata

Importing

Document
conversion

Internal format

Index   Classify

Full text
index   Database   Associated
files

Building

Ingesting Documents

- Document
- Metadata
- Both
- ▸ Extract metadata

Structural
stuff

Document
conversion

Internal format
- Metadata included
- Section-level
- Links to external files

■ Plain text, PDF, PostScript, source code, image formats, …

■ HTML, WORD, Email, TIFF images, Gutenberg text, …

■ XML metadata format, BibTex, Refer, OAI, USMARC, BRS, …

▸ Identify language, extract acronyms, image thumbnails …

Traverse directory structure, split files, ZIP archives, …

Plugin Architecture: import.pl

Top level directory

Traverse directory hierarchy

WORD — *file accepted*

unZIP

PDF — *file accepted*

MARC — *file accepted*

Warn and ignore

language

keyphrases

Internal format file

Metadata and Document Parsing: LOM

View example collection based on Learning Object Metadata (LOM) sample records
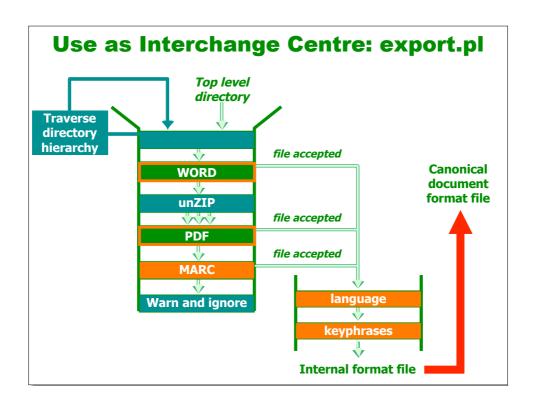
❖ Demonstration of working with sub-fields

IEEE LTSC LOM (*Learning Technology Standards Committee*)

```
<lom>
    <classification> …
    <general> …
    <lifecycle> …
    <educational> …
    <rights> …
    <technical> …
```

Example is representative—
    can do similar things with Refer, BibTeX, ISIS, etc.

Metadata and Document Parsing: LOM

```
<lom                    xmlns="http://linux2.commons.ucalgary.ca/rob_text/ims1_2_1"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://linux2.commons.ucalgary.ca/rob_text/ims1_2_1.xsd">
<classification>
 <keyword>                      <lifecycle>                    <educational>
  <langstring>                   <version>                      <context>
    Arts                         </version>                      <source>
  </langstring>                  <status>                         <langstring>
 </keyword>                       <langstring>                       3D Animation/Maya
 <purpose>                         Finished                         </langstring>
  <langstring>                    </langstring>                    </source>
    Discipline                   </status>                        </context>
  </langstring>                  <contribute>                     <language>
 </purpose>                       <centity>                         English
</classification>                  <vcard>                          </language>
                                    Dalvin Chung Trieu            </educational>
<general>                          </vcard>
 <language>                       </centity>                      <rights>
   English                        <date>                           <description>
 </language>                       <datetime>                        <langstring>
 <title>                             2002-02-01                      Copyright 2002 Vancouve r
  <langstring>                     </datetime>                       Film School
    A Teddy Bear Picnic           <description>                     </langstring>
  </langstring>                     <langstring>                    </description>
 </title>                          Born and raised in B.C.         <copyrightandotherrestrict>
 <keyword>                          Dalvin grew up having a          <langstring>
  <langstring>                     ... video gaming industry.         yes
    teddy bear                     </langstring>                    </langstring>
  </langstring>                    </description>                  </copyrightandotherrestric>
 </keyword>                       </date>                          </rights>
 <keyword>                        <role>
  <langstring>                     <langstring>                    <technical>
    run away                        Director/Creator               <size>
  </langstring>                     </langstring>                    2053707
 </keyword>                        </role>                          </size>
 <coverage>                       </contribute>                    <duration>
 </coverage>                     </lifecycle>                        1:38
 <description>                                                     </duration>
 This film is about a ...                                          <location>
 now he has to get back!                                           http://.../DalvinT.mov
  </langstring>                                                    </location>
 </description>                                                    <format>
</general>                                                             video/quicktime
                                                                   </format>
                                                                 </technical></lom>
```
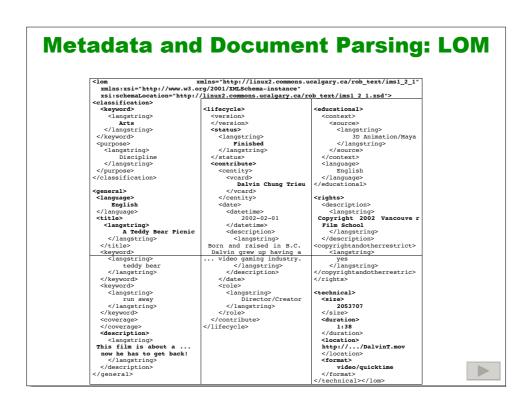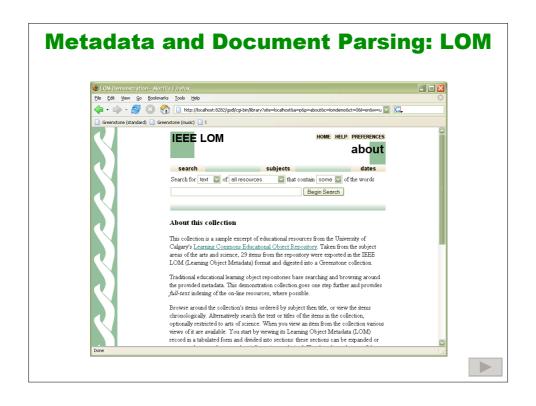
Metadata and Document Parsing: LOM

Use as Interchange Centre: export.pl

METS–Metadata Encoding and Transmission Standard

❖ Aim: encode metadata for any digital material
  – electronic texts, still images, digitized video, sound files
❖ Specifies a *framework* that works with other XML standards
❖ Coherent overall structure for encoding all metadata
  – descriptive, administrative, structural
  used to describe digital library objects
❖ Metadata either
  – embedded within METS structure
  – or held in external files that METS references
❖ Internal identifiers link all the object's components

extends the "conventional" use of the term metadata to encompass document structure and content as well

METS–Metadata Encoding and Transmission Standard

❖ Origins in Making of America II, late 1990s.

"... to create a DL object standard by encoding descriptive, administrative and structural metadata, along with primary content, inside a DL object"

❖ Led to an XML DTD that specified data elements and encoding for objects

– text and still image content files
– including diaries, journals, photographs, correspondence

❖ 2001 workshop to discuss future of the DTD

– Outcome: produce new version for library community
– Recast the DTD as an XML schema

❖ 2001: first draft of METS for review

❖ 2002–2003: successive versions – now version 1.3

METS

7 top-level sections:

❖ METS Header

❖ Descriptive Metadata

❖ Administrative Metadata

❖ File Section

– lists the individual files that comprise the digital object

❖ Structural Map

– describes overall structure of object

❖ Structural Links

❖ Behavior

METS Structural map

```
<structMap TYPE="PHYSICAL">
  <div ID="title1.div.1"  LABEL="Chapter 1">
    <div ID="title.1.div.1.1" LABEL="Section 1">
      <fptr FILEID="title1.image.1"/>
      <fptr FILEID="title1.image.2"/>
    </div>
    <div ID="title.1.div.1.2" LABEL="Section 2">
      <fptr FILEID="title1.image.3"/>
    </div>
  </div>
  <div ID="title1.div.2"  LABEL="Chapter 2">
    [Contents of Chapter 2 omitted]
  </div>
</structMap>
```

METS File groups

❖ Each file is referenced by a file element, e.g.

```
<file ID="FILE003" MIMETYPE="image/jpeg">
  <FLocat LOCTYPE="URL">
    http://dlib.nyu.edu/tamwag/beame.jpg
  </FLocat>
</file>
```

❖ <FLocat> gives file's physical location as URL

❖ ID used elsewhere in the METS file to reference it

❖ Different manifestations of the same resource may be
  – grouped by <fileGrp> … <fileGrp>
  – referenced together if necessary

METS Descriptive metadata

❖ held in <dmdSec> sections
❖ structural map references <dmdSec> using ID attribute
❖ descriptive metadata can be either be
  – held in external files referenced from within the METS
  – or embedded directly with it

```
<dmdSec ID="dmd001">
  <mdRef LOCTYPE="URN" MIMETYPE="application/xml" MDTYPE="EAD"
    LABEL="Berol Collection Finding Aid">urn:x-nyu:fales1735</mdRef>
</dmdSec>
<dmdSec ID="dmd002">
  <mdWrap MIMETYPE="text/xml" MDTYPE="DC" LABEL="Dublin Core
Metadata">
    <dc:title>Alice's Adventures in Wonderland</dc:title>
    <dc:creator>Lewis Carroll</dc:creator>
    <dc:date>between 1872 and 1890</dc:date>
    <dc:publisher>McCloughlin Brothers</dc:publisher>
    <dc:type>text</dc:type>
  </mdWrap>
</dmdSec>
```

METS: Remaining parts

Administrative Metadata

❖ Like descriptive, but recorded in <amdSec>
  elements (embedded or external)
  – technical information on files that comprise an object
  – information on IP rights
❖ Can use any metadata scheme
  – METS editorial board has recommendations

Behavior

❖ How to render object components for the user, e.g.
  ❖specific software packages to be used
  ❖particular parameters to be used
  (Late addition to the METS standard)

Tutorial exercise #21
Export a collection as METS

This exercise is mistitled. To export your collection as METS, just go to the File menu, choose Export, and select METS.

The exercise shows you how to use METS as the internal representation for a Greenstone collection, as a substitute for Greenstone's own (and older) GA format.

▶

METS: Example

One folder for each document:

```
HASH0101.dir/
HASH0102.dir/
HASH0103.dir/
HASH010c.dir/
...
```

Sample document folder (HASH0101.dir):

```
1500s.gif
docmets.xml
doctxt.xml
```

Sample docmets.xml (as before):

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<mets:mets OBJID="HASH0101533fd4a2d9174eaa65ef:2">
  <mets:dmdSec ID="DM1" GROUPID="1">
    <mets:mdWrap MDTYPE="OTHER" OTHERMDTYPE="gsdl3" ID="gsdl1">
      <mets:xmlData>
        <gsdl3:Metadata name="lastmodified">1117427985</gsdl3:Metadata>
        <gsdl3:Metadata name="gsdlsourcefilename">ab.html</gsdl3:Metadata>
        ...
</mets:mets>
```

METS: Issues

❖ Lots of people generate METS

❖ Not so many people read it in

❖ Difficult to handle all possible METS files

– e.g. can include any metadata standard with an XML manifestation—even ones that haven't been devised yet!

❖ Subsets have been institutionalized as METS "profiles"

– e.g. Greenstone profile
http://www.loc.gov/standards/mets/profiles/00000006.xml

❖ Two systems may apparently use METS but fail to interoperate meaningfully

You have to read the fine print

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

MODS – Metadata Object Description Schema

A contemporary form of MARC library metadata, expressed in XML
   (the MARC Standards Office helped define MODS)

❖ Fairly easy to convert MARC metadata to MODS
   – … though not a straightforward one-to-one mapping
   – >30 years between the inception of MARC and MODS
   – MODS includes some additional concepts

❖ MARCXML is a separate standard that gives an exact 1-1 representation of MARC in XML
   – also administered by MARC Standards Office at LoC

Development history:
   – MODS listserv provides an avenue for user input
   – Version 1.0 (draft): trial use for six months in 2002
   – Version 2.0: Feb 2003
   – Version 3.0: issued for review Sept 2003
   – released Dec 2003

MODS Overview

❖ 19-top level elements for describing objects
❖ 64 further sub-elements under these
❖ Covers standard bibliographic facets, e.g.
   – titles
   – names of creators and contributors
   – subject and classification numbers
❖ Plus provision for
   – physical description
   – access restrictions
   – genre
   – etc
❖ Extension mechanism: can incorporate other XML documents (e.g. related objects)
❖ Includes facilities for authority control, and is fully granular in the level of descriptions it provides.

1. titleInfo
2. name
3. classification
4. language
5. originInfo
6. identifier
7. accessCondition
8. genre
9. subject
10. abstract
11. physicalDescription
12. relatedItem
13. recordInfo
14. targetAudience
15. typeOfResource
16. tableOfContents
17. note
18. location
19. extension

MODS: Sample

```
<mods>
  <titleInfo>
    <title>Sound and fury :</title>
    <subTitle>the making of the punditocracy /</subTitle>
  </titleInfo>
  <name type="personal">
    <namePart>Alterman, Eric</namePart>
    <role>
      <text>creator</text>
    </role>
  </name>
  <typeOfResource>text</typeOfResource>
  <genre authority="marc">bibliography</genre>
  <originInfo>
    <place>
      <code authority="marc">nyu</code>
      <text>Ithaca, N.Y</text>
    </place>
    <publisher>Cornell University Press</publisher>
    <dateIssued>c1999</dateIssued>
    <dateIssued encoding="marc">1999</dateIssued>
```
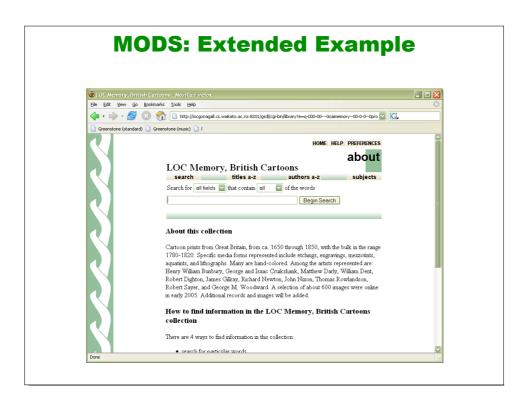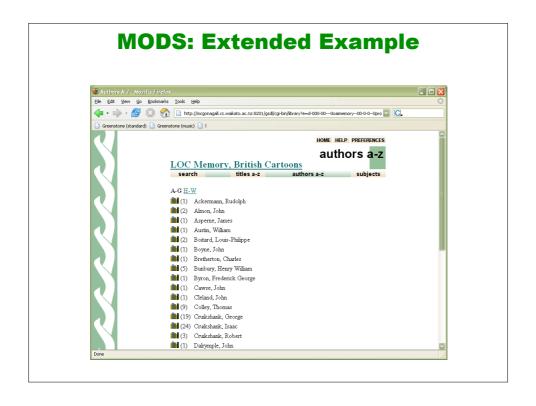
MODS: Sample

```
…
    <issuance>monographic</issuance>
  </originInfo>
  <language authority="iso639-2b">eng</language>
  <physicalDescription>
    <form authority="marcform">print</form>
    <extent>vii, 322 p. ; 23 cm.</extent>
  </physicalDescription>
  <note type="statement of responsibility">Eric Alterman.</note>
  <note>
    Includes bibliographical references (p. 291-312) and index.
  </note>
  <subject authority="lcsh">
    <topic>Journalism</topic>
    <topic>Political aspects</topic>
    <geographic>United States.</geographic>
  </subject>
…
```

MODS: Sample

```
…
  <classification authority="lcc">PN4888.P6 A48
   1999</classification>
  <classification edition="21"
   authority="ddc">071/.3</classification>
  <identifier type="isbn">0801486394 (pbk. : acid-free, recycled
   paper)</identifier>
  <identifier type="lccn">99042030</identifier>
  <recordInfo>
    <recordContentSource>DLC</recordContentSource>
    <recordCreationDate encoding="marc">990730</recordCreationDate>
    <recordChangeDate
   encoding="iso8601">20000406144503.0</recordChangeDate>
    <recordIdentifier>11761548</recordIdentifier>
  </recordInfo>
</mods>
```
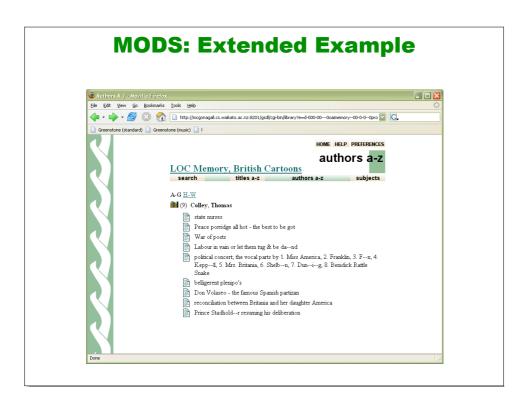
Extended example
MODS
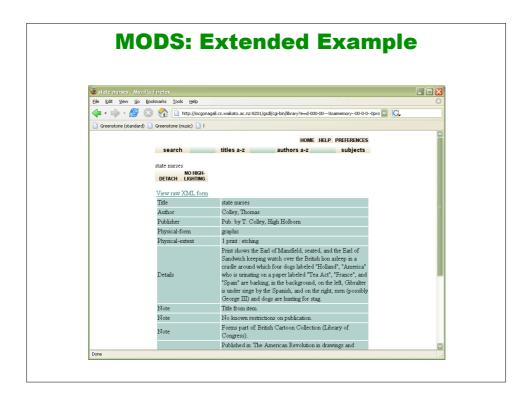
Using MODS records, build a Greenstone digital library collection based on the Library of Congress' British Cartoons collection (part of the American Memory project).
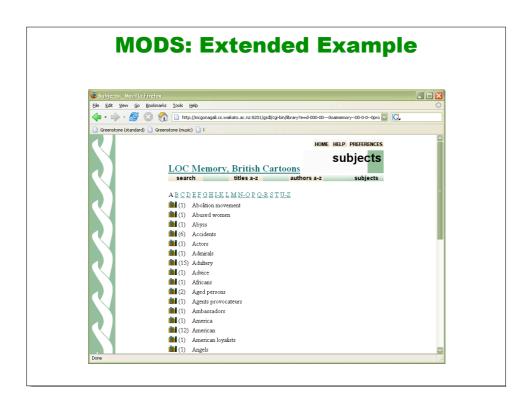
❖ MODS metadata records exported from LOC's OAI server
❖ Greenstone configured to support:
  – Browse by title, author, and subject
  – Search by title, subject and all fields
  – Display summary of MODS record nicely formatted
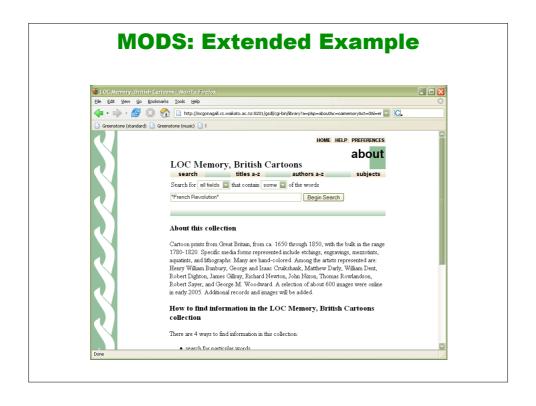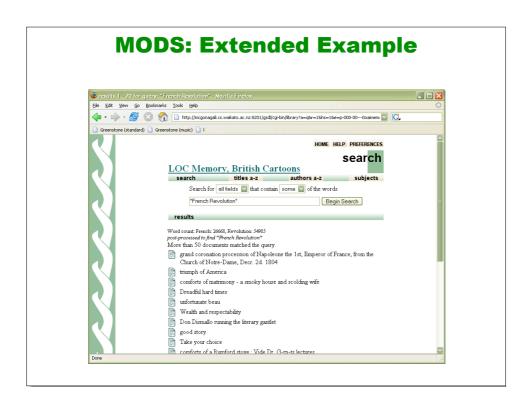 – Hyperlink to original resource

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

MODS: Extended Example

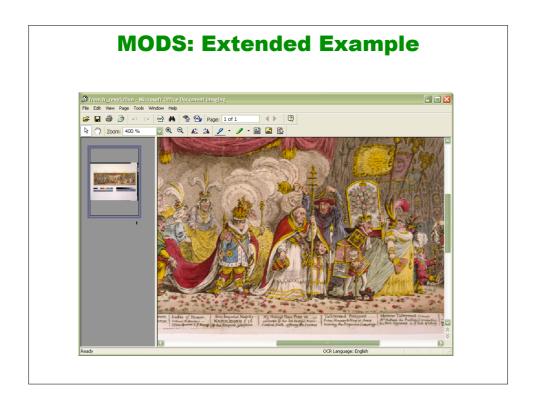## MODS: Extended Example
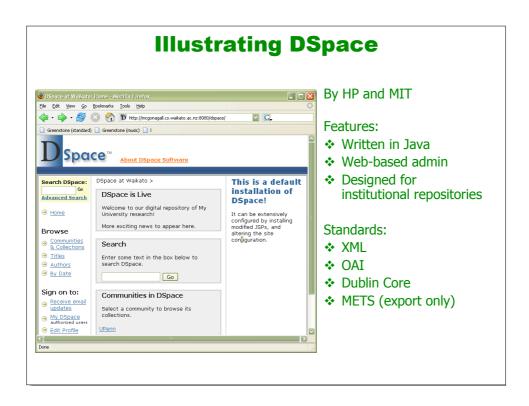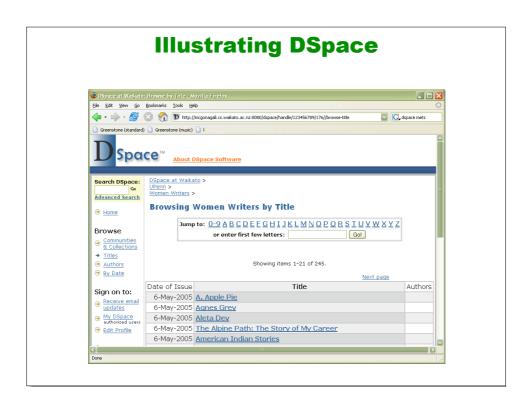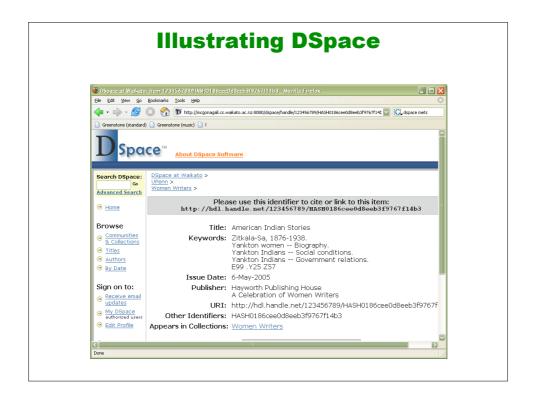

MODS: Extended Example
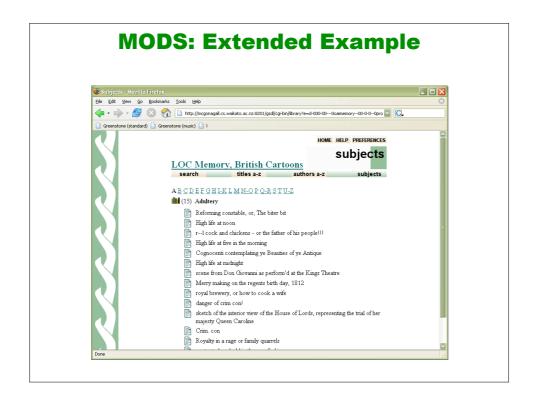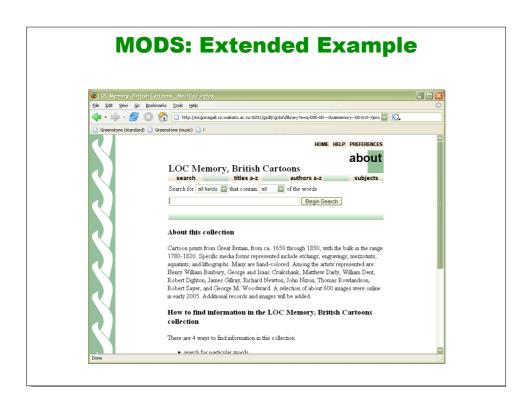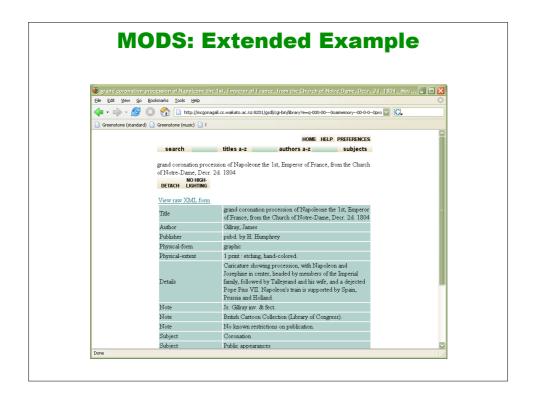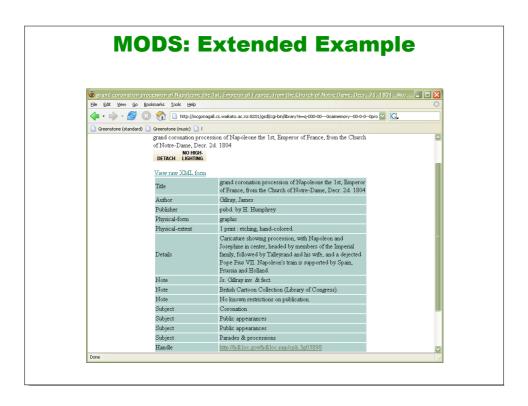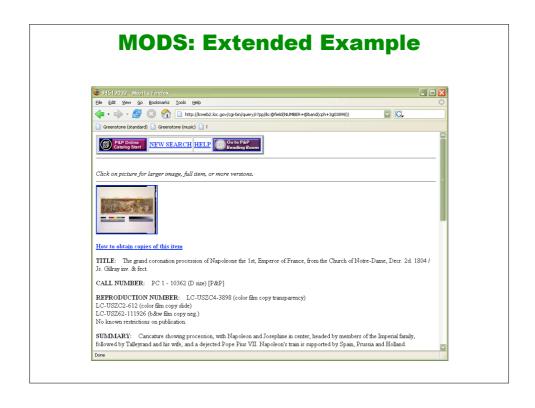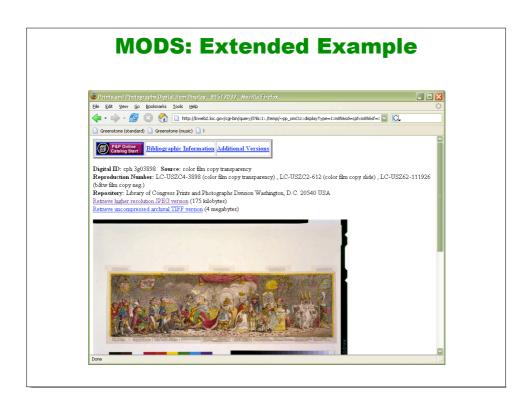
MODS: XSLT mapping

MODS mapping into Greenstone using XSLT
(specify as argument to OAIPlug):

```
...
<xsl:template match="mods:topic">
  <dc:Subject><xsl:value-of select="." /></dc:Subject>
</xsl:template>

<xsl:template match="mods:titleInfo">
  <dc:Title>
    <xsl:value-of select="mods:title" />
    <xsl:if test="mods:subTitle">
      <xsl:text>: </xsl:text>
      <xsl:value-of select="mods:subTitle" />
    </xsl:if>
  </dc:Title>
</xsl:template>
...
```
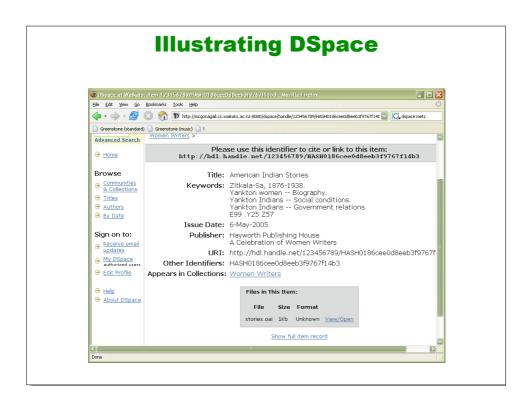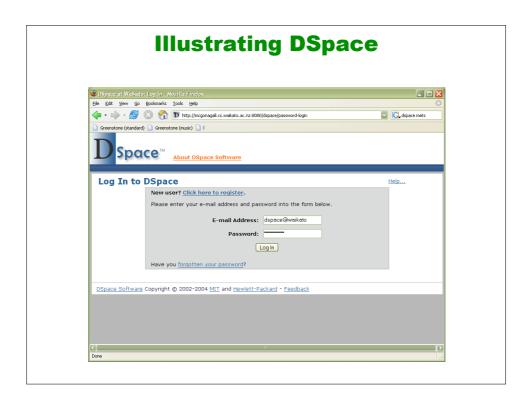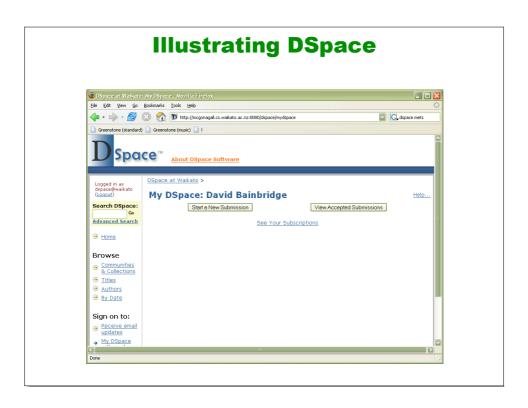
Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion

Illustrating DSpace

By HP and MIT

Features:
- ❖ Written in Java
- ❖ Web-based admin
- ❖ Designed for institutional repositories

Standards:
- ❖ XML
- ❖ OAI
- ❖ Dublin Core
- ❖ METS (export only)

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace

Illustrating DSpace
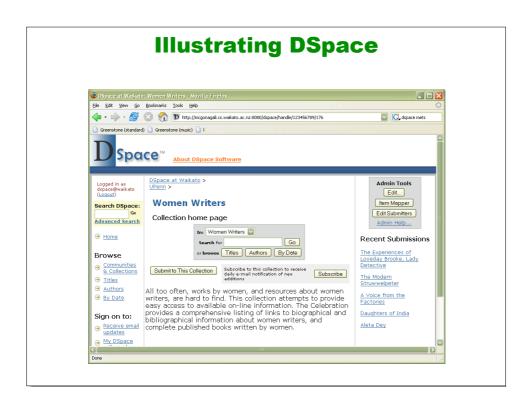

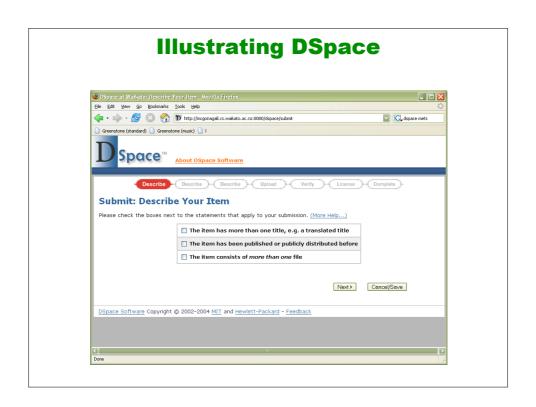# Illustrating DSpace

and so on ...

Tutorial exercise #22, #23
Greenstone – DSpace conversion

1. **Move a collection from DSpace to Greenstone**
   Configure it with comparable indexing and browsing capabilities to the original

2. **Move a collection from Greenstone to DSpace**
   Configure it in the same way

DSpace to Greenstone

Input documents, one folder for each:

    1   2   3   4   5

❖ Sample document folder (1):

    contents dancingH.doc dancingH.pdf dublin_core.xml

❖ Document manifest in contents:

```
dancingH.doc     bundle:ORIGINAL
dancingH.pdf     bundle:ORIGINAL
```

❖ Dublin Core:

```
<dublin_core>
  <dcvalue element="date" qualifier="accessioned">2005-01-10T02:48:06Z</dcvalue>
  <dcvalue element="identifier" qualifier="uri">
    http://hdl.handle.net/123456789/40
  </dcvalue>
  <dcvalue element="title" qualifier="none">
    A history of dance music from E to P
  </dcvalue>
  <dcvalue element="language" qualifier="iso">en</dcvalue>
  <dcvalue element="type" qualifier="none">Working Paper</dcvalue>
  ...
</dublin_core>
```
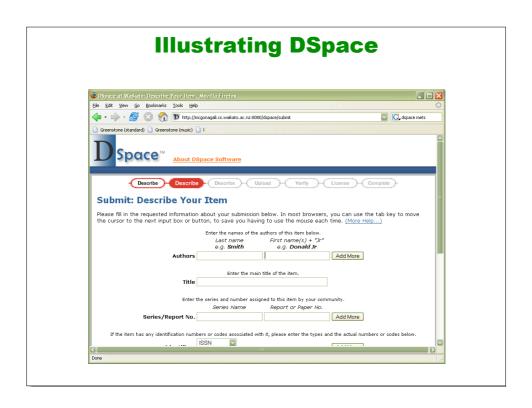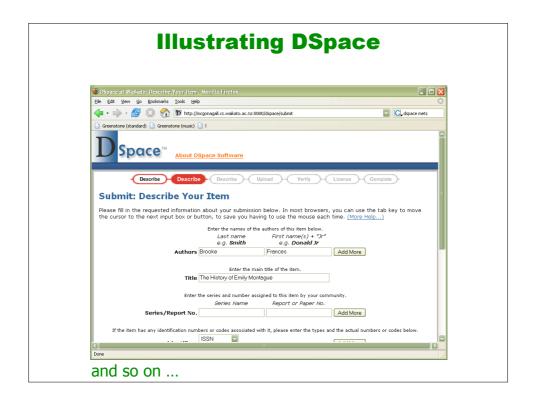
Tutorial exercise #22, #23
Greenstone – DSpace conversion

1. Move a collection from DSpace to Greenstone
    Configure it with comparable indexing and
    browsing capabilities to the original

2. **Move a collection from Greenstone to DSpace**
    Configure it in the same way

▶

DSpace to Greenstone

Input documents, one folder for each:

    1   2   3   4   5

❖ Sample document folder (1):

  `contents  dancingH.doc  dancingH.pdf  dublin_core.xml`

❖ Document manifest in `contents`:

```
dancingH.doc    bundle:ORIGINAL
dancingH.pdf    bundle:ORIGINAL
```

❖ Dublin Core:

```
<dublin_core>
  <dcvalue element="date" qualifier="accessioned">2005-01-10T02:48:06Z</dcvalue>
  <dcvalue element="identifier" qualifier="uri">
    http://hdl.handle.net/123456789/40
  </dcvalue>
  <dcvalue element="title" qualifier="none">
    A history of dance music from E to P
  </dcvalue>
  <dcvalue element="language" qualifier="iso">en</dcvalue>
  <dcvalue element="type" qualifier="none">Working Paper</dcvalue>
  ...
</dublin_core>
```

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services ⬅

Concluding discussion

Fedora

University of Virginia Library and Cornell University

Features:
- ❖ Written in Java
- ❖ Admin application
- ❖ Document versions
- ❖ Designed to provide foundation to repository

Standards:
- ❖ SOAP, XML, OAI, Dublin Core, METS
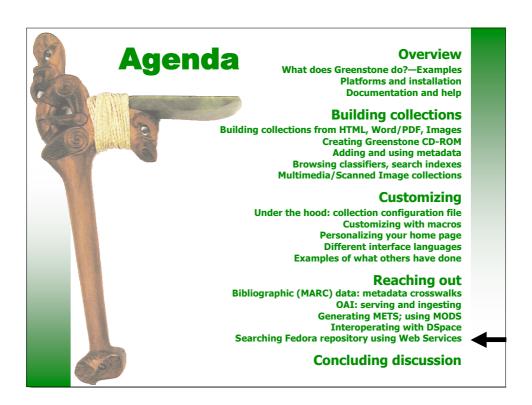
Web Services

❖ Allows any piece of software to communicate with a standardized XML messaging system.
❖ They're self-contained, modular applications that can be
  – described
  – published
  – located
  – invoked over a network (generally the Web)
❖ Client-server communication is "loosely coupled," leading to flexible implementation
    (*Unlike CORBA, DCOM, etc.*)
❖ Ideal for DL service-level interoperability

Web Services: Three key components

❖ Standard way to represent data
  – XML for data representation
  – XML Schema to define data types
❖ Common, extensible, message format
  – Simple Object Access Protocol (SOAP)
  – lightweight protocol for information exchange
  – an extensible message format
  – conventions for representing remote procedure calls (RPCs)
  – HTTP bindings
❖ Common, extensible, service description language
  – Web Services Description Language (WSDL)
  – XML-based contract language
  – provides a way to document the messages a Web Service accepts/generates.

Extended example
Web services and Fedora

❖ **Demonstrate Greenstone searching a Fedora repository using Web Services**

❖ **Implemented using Greenstone 3:**
  – **a research framework that is backwards compatible with Greenstone 2**
  – **uses METS as its internal representation**
  – **implemented in Java**
  – **service based**
  – **supports SOAP communication**

Web Services: Greenstone to Fedora

Web Services: Greenstone to Fedora

Agenda

Overview
What does Greenstone do?—Examples
Platforms and installation
Documentation and help

Building collections
Building collections from HTML, Word/PDF, Images
Creating Greenstone CD-ROM
Adding and using metadata
Browsing classifiers, search indexes
Multimedia/Scanned Image collections

Customizing
Under the hood: collection configuration file
Customizing with macros
Personalizing your home page
Different interface languages
Examples of what others have done

Reaching out
Bibliographic (MARC) data: metadata crosswalks
OAI: serving and ingesting
Generating METS; using MODS
Interoperating with DSpace
Searching Fedora repository using Web Services

Concluding discussion