RESEARCH TOOLS 2011

LECTURE 18

2011-Nov-01
Kurt Schwehr
http://schwehr.org

UNH CCOM/JHC
BAG, HDF5 and XML

http://creativecommons.org/licenses/by-nc-sa/3.0/
Research Tools Class 18: BAGs, HDF5, XML, matplotlib

2011-Nov-01
Kurt Scheweir
UNH CCOM/JHC
http://scheweir.org

researchtools@ubuntu:~
File Edit View Search Terminal Help

researchtools@ubuntu:~$ sudo ntpdate ntp.ubuntu.com
[sudo] password for researchtools:
1 Nov 11:01:20 ntpdate[9668]: step time server 91.189.94.4 offset 414636.735846
sec
researchtools@ubuntu:~$ date
Tue Nov 1 11:01:33 EDT 2011
researchtools@ubuntu:~$ mkdir -p class/18
researchtools@ubuntu:~$ cd cl

---

Logging in as 'kurtvm'... done
# Lectures:

<table>
<thead>
<tr>
<th>No.</th>
<th>Date</th>
<th>Title/Notes</th>
<th>Audio</th>
<th>Present</th>
<th>Video</th>
<th>Blog</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>2011-11-01</td>
<td>BAGs, HDF5 and XML</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>2011-10-27</td>
<td>QGIS and GDAL for Raster data</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>2011-10-25</td>
<td>Python: matplotlib 2 - graphics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vid 16</td>
<td>2011-10-24</td>
<td>Python part 9: Simple signals</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vid 15</td>
<td>2011-10-23</td>
<td>Python part 8: Matplotlib</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>2011-10-20</td>
<td>Python: Matplotlib part 1</td>
<td>mp3</td>
<td>pdf key</td>
<td></td>
<td>comment</td>
</tr>
<tr>
<td>14</td>
<td>2011-10-18</td>
<td>Python: parsing GPS data</td>
<td>mp3</td>
<td>pdf key</td>
<td></td>
<td>comment</td>
</tr>
<tr>
<td>Vid 14</td>
<td>2011-10-18</td>
<td>Python part 7 - more reusable code</td>
<td>mp3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rd 1</td>
<td>2011-10-13</td>
<td>NOAA - Making Waves: Building a Better Geoid</td>
<td>mp3</td>
<td>txt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vid 13</td>
<td>2011-10-15</td>
<td>Python part 6 - parsing GPS data</td>
<td></td>
<td>pdf key</td>
<td>YouTube H264</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2011-10-13</td>
<td>Python: if, while, functions, classes</td>
<td></td>
<td>pdf key</td>
<td></td>
<td>comment</td>
</tr>
<tr>
<td>Vid 12</td>
<td>2011-10-13</td>
<td>Python part 5 - while loops</td>
<td></td>
<td>pdf key</td>
<td>YouTube m4v</td>
<td></td>
</tr>
<tr>
<td>Vid 11</td>
<td>2011-10-12</td>
<td>Python part 4 - if, modules, command line args</td>
<td></td>
<td>pdf key</td>
<td>YouTube H264</td>
<td></td>
</tr>
<tr>
<td>2011-10-11</td>
<td>No class - UNH Monday Schedule</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Navigation Surface (DOWNLOAD [2.56M]) paradigm is a design for a databased alternative to traditional methods of representing bathymetric data. It aims to preserve the highest level of detail in every bathymetric dataset and provide methods for their combination and manipulation to generate multiple products for both hydrographic and non-hydrographic purposes. The advantages of the method over traditional schema are such that a number of commercial vendors have adopted the technology. However, this means that there is a strong requirement for a method to communicate results in a vendor neutral technology. The Open Navigation Surface (ONS) project was designed to fill this gap by implementing a freely available source-code library to read and write all of the information required for a Navigation Surface.

The Navigation Surface concept requires that in addition to estimation of depth, we must also estimate the uncertainty associated with the depth. In order to make the system suitable to support Safety of Navigation applications, we also require a means to over-ride any automatically constructed depth estimates with a "Hydrographer Privilege", essentially a means to specify directly the depth determined by a human expert being the most significant in the area (irrespective of any statistical evidence to the contrary). Finally, we must provide data on the data, or metadata, which describes all aspects of the data's life from methods of processing methods, geospatial extents to responsible party. The ONS project provides means to implement all of these requirements in a portable, platform neutral, vendor neutral format.
NOS Hydrographic Survey Data

**New** Hydrographic Survey Rap Sheets display survey summaries and locations, cross references, statistics, comprehensive platforms and correctors, processing and archive information. Survey types include Hydrographic, Field Edit, Reconnaissance, Lake, and Non-NOS sources.

The National Ocean Service (NOS) Hydrographic Data Base (NOSHDB) and Hydrographic Survey Meta Data Base (HSMDB), both maintained by NOS and NGDC, provide extensive survey coverage and Federal Geographic Data Committee (FGDC) with Remote Sensing Extensions (RSE) metadata of the coastal waters and Exclusive Economic Zone (EEZ) of the United States and its territories. The NOSHDB contains digitized data from smooth sheets of hydrographic surveys completed between 1837 and 1965, and from survey data acquired digitally on NOS survey vessels since 1965.

Over 76 million soundings from over 6,600 surveys are included in the NOSHDB. NGDC has developed an interactive map and data discovery tool, the Hydrographic Survey Data Map Service, and a text-based search engine, the Hydrographic Survey Text Search, which enable rapid data discovery and download. The NOSHDB data with search and retrieval software are also available on a DVD-ROM or CD-ROM set.

**Data products from NOS surveys**, including BAG files, Descriptive Reports, smooth sheet images, survey data images, textual gridded data, and geo-referenced sidescan sonar mosaics, FGDC/RSE metadata, and survey statistics are available for download from NGDC.

**Bathymetric Attributed Grid (BAG)** is a non-proprietary file format for storing and exchanging bathymetric data developed by the Open Navigation Surface Working Group. BAG files are gridded, multi-dimensional bathymetric data files and is the standard NOS hydrographic data file for public release. Current versions of the BAG file contain position...
The National Ocean Service (NOS) Hydrographic Data Base (NOSHDB) and Hydrographic Survey Meta Data Base (HSMDB), both maintained by NOS and NGDC, provide extensive survey coverage and Federal Geographic Data Committee (FGDC) with Remote Sensing Extensions (RSE) metadata of the coastal waters and Exclusive Economic Zone (EEZ) of the United States and its territories. The NOSHDB contains digitized data from smooth sheets of hydrographic surveys completed between 1837 and 1965, and from survey data acquired digitally on NOS survey vessels since 1965.

Over 76 million soundings from over 6,600 surveys are included in the NOSHDB. NGDC has developed an interactive map and data discovery tool, the Hydrographic Survey Data Map Service, and a text-based search engine, the Hydrographic Survey Text Search, which enable rapid data discovery and download. The NOSHDB data with search and retrieval software are also available on a DVD-ROM or CD-ROM set.

Data products from NOS surveys, including BAG files, Descriptive Reports, smooth sheet images, survey data images, textual gridded data, and geo-referenced sidescan sonar mosaics, FGDC/RSE metadata, and survey statistics are available for download from NGDC.

Bathymetric Attributed Grid (BAG) is a non-proprietary file format for storing and exchanging bathymetric data developed by the Open Navigation Surface Working Group. BAG files are gridded, multi-dimensional bathymetric data files and is the standard NOS hydrographic data file for public release. Current versions of the BAG file contain position and depth grid data, as well as position and uncertainty grid data, and the metadata specific to that BAG file, providing end users information about the source and contents of the BAG file. Please visit the Open Navigation Surface Working Group for additional information on BAG files.

NGDC has developed the capability to output data from BAG files as American Standard Code for Information Interchange (ASCII) xyz files. Empty space in BAG files is not included in xyz output files. On NGDC’s web site, individual files, or entire surveys spanning multiple BAG files, can be selected, extracted, processed to xyz format, and zipped for shipment using the new conversion capability. Each conversion also creates an Extensible Markup Language (XML) metadata file.

Descriptive Reports (DRs) are approved metadata documents relevant to survey level hydrographic data and are available in Portable Document Format (PDF) form. The DR serves as a narrative document that describes the conditions under which the hydrographers performed the work and discusses various factors affecting the adequacy.
Class 18: BAGs, HDF5, and XML

Introduction

Bathymetric Attributed Grid (BAG)

Bags were started here at CCOM.

- http://www.opennavsurf.org/ (hosted at CCOM)

I am not sure how to go from the above to this:

- http://surveys.ngdc.noaa.gov/mgg/
- http://surveys.ngdc.noaa.gov/mgg/NOS/coast/H12001-H14000/H12263/BAG/

```bash
# BEGIN_SRC sh
cd ~/class/18
mkdir bags
cd bags
Descriptive
hydrograph
- 18-bag-hdf-xml.org Top L22 (Org)
```
<table>
<thead>
<tr>
<th>Name</th>
<th>Last modified</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent Directory</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H12263_MB_1m_MLLW_1of4.bag.gz</td>
<td>25-May-2011 18:31</td>
<td>3.6M</td>
<td></td>
</tr>
<tr>
<td>H12263_MB_2m_MLLW_2of4.bag.gz</td>
<td>25-May-2011 18:39</td>
<td>1.7M</td>
<td></td>
</tr>
<tr>
<td>H12263_MB_4m_MLLW_3of4.bag.gz</td>
<td>25-May-2011 18:43</td>
<td>52M</td>
<td></td>
</tr>
<tr>
<td>H12263_MB_8m_MLLW_4of4.bag.gz</td>
<td>25-May-2011 18:46</td>
<td>6.2M</td>
<td></td>
</tr>
<tr>
<td>H12263_MB_8m_MLLW_combined.bag.gz</td>
<td>25-May-2011 18:50</td>
<td>17M</td>
<td></td>
</tr>
</tbody>
</table>

NGDC home | privacy, disclaimers, and copyright notice | questions

Website of the US Department of Commerce / NOAA / NESDIS / National Geophysical Data Center
Research Tools Class 18:

2011-Nov-01
Kurt Schwehr
UNH CCOM/JHC
http://schwehr.org

* Bathymetric Attributed Grid (BAG)

Bags were started here at CCOM.

researchtools@ubuntu: ~/class/18

100%[==================================] 3,793,797 1.38M/s in 2.6s

2011-11-01 11:15:51 (1.38 MB/s) - `H12263_MB_1m_MLLW_1of4.bag.gz' saved [3793797 /3793797]

researchtools@ubuntu:~/class/18$ wget http://surveys.ngdc.noaa.gov/mgg/NOS/coast/H12001-H14000/H12263/BAG/H12263_MB_8m_MLLW_combined.bag.gz
Resolving surveys.ngdc.noaa.gov... 140.172.184.18
Connecting to surveys.ngdc.noaa.gov|140.172.184.18|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 17890717 (17M) [application/x-gzip]
Saving to: `H12263_MB_8m_MLLW_combined.bag.gz'
100%[==================================] 17,890,717 907K/s in 19s

2011-11-01 11:16:10 (906 KB/s) - `H12263_MB_8m_MLLW_combined.bag.gz' saved [17890717 /17890717]

researchtools@ubuntu:~/class/18$

# 9f5775098df7e168969268a59e8f141742a99c89f7d749e16f47eb1662e29f H12263_MB_1m
# aaa4ef417849d6144744bc26a1e975b99f2ac41df88f6c057989ca3fb1329e3e H12263_MB_8m

--- 18-bag-hdf-xml.org 83% L35 [#,r] (Org) --
menu-bar edit copy
Research Tools Class 18:

2011-Nov-01
Kurt Schwehr
UNH CCOM/JHC
http://schwehr.org

* Bathymetric Attributed Grid (BAG)

Bags were started here at CCOM.

2011-11-01 11:16:10 (906 KB/s) - `H12263_MB_8m_MLLW_combined.bag.gz' saved [1789 0717/17890717]

researchtools@ubuntu:~/class/18$ ls -l

total 21192
-rw-r--r-- 1 researchtools researchtools 8880 2011-11-01 10:35 18-bag-hdf-xm

researchtools@ubuntu:~/class/18$ gunzip *.gz

researchtools@ubuntu:~/class/18$ ls -l

total 3363708
-rw-r--r-- 1 researchtools researchtools 8880 2011-11-01 10:35 18-bag-hdf-xm.xml
-rw-r--r-- 1 researchtools researchtools 3391351952 2011-05-25 20:31 H12263_MB_1
m_MLLW_1of4.bag
-rw-r--r-- 1 researchtools researchtools 53067176 2011-05-25 20:50 H12263_MB_8
m_MLLW_combined.bag
researchtools@ubuntu:~/class/18$
Hierarchical Data Format (version 5) data

Hierarchical Data Format (version 5) data

Hierarchical Data Format (version 5) data

Hierarchical Data Format (version 5) data

Hierarchical Data Format (version 5) data
Err... what is a Hierarchical Data Format??? (HDF) It is a container for data of all different sorts of format.

And this one will not load in qgis

* Another try at a bag

http://surveys.ngdc.noaa.gov/mgg/NOS/coast/H10001-H12000/H11703/BAG/

```bash
# BEGIN_SRC sh
wget http://surveys.ngdc.noaa.gov/mgg/NOS/coast/H10001-H12000/H11703/BAG/H11703_H12263_MB_1m_MLLW_1of4_bag.gz
# H11703_5m_Combined_MLLW_5of5_bag.gz: gzip compressed data, was "H11703_5m_Combi"
# END_SRC

Nope! But I have a snapshot of NGDC's bag archive from May 2010.

```bash
# BEGIN_SRC sh
wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Combined_5m.bag
wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Office_5m.bag
```

gdalinfo --version
# GDAL 1.6.3, released 2009/11/19

gdalinfo H11703_Office_5m.bag
# END_SRC

# BEGIN_EXAMPLE
Driver: HDF5/Hierarchical Data Format Release 5
Files: H11703_Office_5m.bag
Size is 512, 512
Coordinate System is `
researchtools@ubuntu:~/class/18/bags

researchtools@ubuntu:~/class/18/bags$ wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Office_5m.bag.bz2
--2011-11-01 11:22:45-- http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Office_5m.bag.bz2
Resolving vislab-ccom.unh.edu... 192.168.3.3
Connecting to vislab-ccom.unh.edu[192.168.3.3]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1474546 (1.4M) [application/x-bzip2]
Saving to: `H11703_Office_5m.bag.bz2'
100%[==========================================] 1,474,546 5.08M/s in 0.3s
2011-11-01 11:22:45 (5.08 MB/s) `H11703_Office_5m.bag.bz2' saved [1474546/1474546]
researchtools@ubuntu:~/class/18/bags$ ls -l
total 3369416
-rw-r--r-- 1 researchtools researchtools 4381819 2011-10-27 09:44 H11703_Combined_5m.bag.bz2
-rw-r--r-- 1 researchtools researchtools 1474546 2011-10-27 09:44 H11703_Office_5m.bag.bz2
-rw-r--r-- 1 researchtools researchtools 3391351952 2011-05-25 20:31 H12263_MB_1m_MLLW_1of4.bag
-rw-r--r-- 1 researchtools researchtools 53067176 2011-05-25 20:50 H12263_MB_8m_MLLW_combined.bag
researchtools@ubuntu:~/class/18/bags$ bunzip2 *.bag.bz2

wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Combined_5m.bag.bz2
wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Office_5m.bag.bz2

gdalinfo --version
# GDAL 1.6.3, released 2009/11/19

gdalinfo H11703_Office_5m.bag
#+BEGIN_EXAMPLE
Driver: HDF5/Hierarchical Data Format Release 5
Files: H11703_Office_5m.bag
Size is 512, 512
Coordinate System is `
researchtools@ubuntu:~/class/18/bags

```
-rw-r--r-- 1 researchtools researchtools 53067176 2011-05-25 20:50 H12263_MB_8m_MLLW_combined.bag
researchtools@ubuntu:~/class/18/bags$ gdalinfo --version
GDAL 1.6.3, released 2009/11/19
researchtools@ubuntu:~/class/18/bags$ gdalinfo H11703.Office_5m.bag
Driver: HDF5/Hierarchical Data Format Release 5
Files: H11703.Office_5m.bag
Size is 512, 512
Coordinate System is ''
Metadata:
  BAG_root:Bag Version=1.0.0
Subdatasets:
  SUBDATASET 1_NAME=HDF5://H11703.Office_5m.bag://BAG_root/elevation
  SUBDATASET 1_DESC=[1434x2004] //BAG_root/elevation (32-bit floating-point)
  SUBDATASET 2_NAME=HDF5://H11703.Office_5m.bag://BAG_root/uncertainty
  SUBDATASET 2_DESC=[1434x2004] //BAG_root/uncertainty (32-bit floating-point)
Corner Coordinates:
  Upper Left  ( 0.0, 0.0)
  Lower Left  ( 0.0, 512.0)
  Upper Right (512.0, 0.0)
  Lower Right (512.0, 512.0)
  Center      (256.0, 256.0)
researchtools@ubuntu:~/class/18/bags$
```

```
wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703_Combined_5m.bag
wget http://vislab-ccom.unh.edu/~schwehr/rt/examples/old-bags/H11703.Office_5m.bag

gdalinfo --version
# GDAL 1.6.3, released 2009/11/19

gdalinfo H11703.Office_5m.bag
#+END_SRC

#+BEGIN_EXAMPLE
Driver: HDF5/Hierarchical Data Format Release 5
Files: H11703.Office_5m.bag
Size is 512, 512
Coordinate System is ''
```
The version of gdal that comes with Ubuntu 11.04 is just too old to read bags.

```sh
#BEGIN_SRC sh
gdalinfo --formats | egrep -i 'bag|hdf'
# HDF4 (ro): Hierarchical Data Format Release 4
# HDF4Image (rw+): HDF4 Dataset
# HDF5 (ro): Hierarchical Data Format Release 5
# HDF5Image (ro): HDF5 Dataset
#END_SRC
```

Using gdal from fink on Mac OSX:
The version of gdal that comes with Ubuntu 11.04 is just too old to read bags.

```sh
#BEGIN_SRC sh
gdalinfo --formats | egrep -i 'bag|hdf'
# HDF4 (ro): Hierarchical Data Format Release 4
# HDF4Image (rw+): HDF4 Dataset
# HDF5 (ro): Hierarchical Data Format Release 5
# HDF5Image (ro): HDF5 Dataset
#END_SRC

Using gdal from fink on Mac OSX:

```sh
#BEGIN_SRC sh
gdalinfo --version
# GDAL 1.8.1, released 2011/07/09
snipe:BAG schwehr$ gdalinfo --formats | egrep -i 'hdf|bag'
# BAG (ro): Bathymetry Attributed Grid
# HDF5 (ro): Hierarchical Data Format Release 5
# HDF5Image (ro): HDF5 Dataset
#END_SRC

* Bags from hdf5 tools

```sh
#BEGIN_SRC sh
sudo apt-get install hdfs-tools
#END_SRC

```sh
#BEGIN_SRC sh
h5ls H11703_Office_5m.bag
h5ls --recursive H11703_Office_5m.bag
h5ls -d H11703_Office_5m.bag/BAG_root/metadata | head
h5ls --simple -d H11703_Office_5m.bag/BAG_root/metadata | head
```
NAME
h5ls - Prints information about a file or dataset.

SYNOPSIS
h5ls [OPTIONS] file [OBJECTS...]

DESCRIPTION
h5ls prints selected information about file objects in the specified format.

OPTIONS
Manual page h5ls(1) line 1
researchtools@ubuntu:~/class/18/bags

researchtools@ubuntu:~/class/18/bags$ man h5ls
researchtools@ubuntu:~/class/18/bags$ h5ls H11703.Office_5m.bag
BAG_root              Group
researchtools@ubuntu:~/class/18/bags$ apt-cache search h

h5ls H11703.Office_5m.bag
h5ls --recursive H11703.Office_5m.bag
h5ls -d H11703.Office_5m.bag/BAG_root/metadata | head
h5ls --simple -d H11703.Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | grep '' | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | grep '' | cut -f2 -d" | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | grep '' | cut -f2 -d" | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | grep '' | cut -f2 -d" | head
h5ls --string --simple -d H11703.Office_5m.bag/BAG_root/metadata | grep '' | cut -f2 -d" | head
emacsclient --no-wait H11703.Office_5m.xml

### END_SRC

In emacs, we need to make this more readable:

M-x replace-string > < RET > C-q C-j <
In emacs, we need to make this more readable:

```
M-x replace-string > < <RET> > C-q C-j
```
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <
NAME
h5ls - Prints information about a file or dataset.

SYNOPSIS
h5ls [OPTIONS] file [OBJECTS...]

DESCRIPTION
h5ls prints selected information about file objects in the specified format.

OPTIONS
-h or -? or --help Print a usage message and exit.
-a or --address Print addresses for raw data.
-d or --data Print the values of datasets.
-e or --errors Show all HDF5 error reporting.
-f or --full Print full path names instead of base names.
-g or --group Show information about a group, not its contents.
-l or --label Label members of compound datasets.
-r or --recursive List all groups recursively, avoiding cycles.
-s or --string Print 1-bytes integer datasets as ASCII.
-S or --simple Use a machine-readable output format.
-wN or --width=N Set the number of columns of output.
-v or --verbose Generate more verbose output.
-V or --version Print version number and exit.
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <
```
researchtools@ubuntu:~/class/18/bags

h5ls -d H11703_Office_5m.bag/BAG_root/metadata | head
metadata
Data: (0) "<", "?", "x", "m", "l", ",", "v", "e", "r", "s", "i", "o", "n", ",", ",", ",", "1", ",", "0",
(18) ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", "t",
(34) ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", "t",
(52) ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", "t",
(70) "d", "g", ",", "w", "g", "", ",", "o", "r", "g", "", ",", "s", "m", "X", "M", ",", "L",
(88) ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", "t", "t", "p", ":" 
(106) "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w",
researchtools@ubuntu:~/class/18/bags

h5ls -d H11703_Office_5m.bag/BAG_root/metadata | grep """" | head
(0) "<", "?", "x", "m", "l", ",", "v", "e", "r", "s", "i", "o", "n", ",", ",", ",", "1", ",", "0",
(18) ",", "t", "a", "d", "a", "t", "a", ",", "t",
(34) ",", "t", "t", ",", ",", "t", "t", "t",
(52) ",", "t", "t", "t", "t", "t", "t", "t", "t", "t",
(70) "d", "g", ",", "w", "g", "", ",", "o", "r", "g", "", ",", "s", "m", "X", "M", ",", "L",
(88) ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", ",", "t", "t", "p", ":" 
(106) "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w", "w",

h5ls --simple -d H11703_Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep """" | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep """" | cut -f2 -d" |

emacsclient --no-wait H11703_Office_5m.xml

#:END_SRC

In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <
```
researchtools@ubuntu:~/class/18/bags

```bash
researchtools@ubuntu:~/class/18/bags$ h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep "" | head
```

```
h5ls --simple -d H11703_Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep "" | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep "" | cut -f2 -d" | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep "" | cut -f2 -d" | head
h5ls --string --simple -d H11703_Office_5m.bag/BAG_root/metadata | grep "" | cut -f2 -d" | head
```

```
emacsclient --no-wait H11703_Office_5m.xml

#END_SRC

In emacs, we need to make this more readable:

```
M-x replace-string > < <RET> > C-q C-j <
```

---

18-bag-hdf-xml.org 51% L136 [#,#] (Org)
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <

Got that? C-q says to take the next character litterally. Then C-j is a new line (aka move the move down a line)

Now we want to change the indenting of the XML metadata file.

- Go to the beginning of the file: C-
- Start marking a region: C-space
- Go to the end of the file: C->
- Ask emacs to indent all: M-x indent-region
  - or press C-M-\n
Not the most fun way to get the metadata out of a bag!

*Bags from python*
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <

Got that? C-q says to take the next character litterally. Then C-j is a new line (aka move the move down a line)

Now we want to change the indenting of the XML metadata file.

- Go to the beginning of the file: C-
- Start marking a region: C-space
- Go to the end of the file: C-
- Ask emacs to indent all: M-x indent-region
  - or press C-M-

Not the most fun way to get the metadata out of a bag!

* Bags from python
In emacs, we need to make this more readable:

M-x replace-string > < <RET> > C-q C-j <

Got that? C-q says to take the next character literally. Then C-j is a new line (aka move the move down a line)

Now we want to change the indenting of the XML metadata file.

- Go to the beginning of the file: C-<
- Start marking a region: C-space
- Go to the end of the file: C->
- Ask emacs to indent all: M-x indent-region
  - or press C-M-

Not the most fun way to get the metadata out of a bag!

* Bags from python
In emacs, we need to make this more readable:

```
x replace-string > < <RET> C-q C-j <
```

Got that? C-q says to take the next character literally. Then C-j is a new line (aka move the move down a line).
  <smXML:BAG_DataIdentification>
    <citation>
      <smXML:CI_Citation>
        <title>BAG%20file%20created%20from%20N:\OPR0322KR07\Surveys\H11703\Caris\Fieldsheets\H11703_Office\H11703_Office_5m_Final.hns</title>
        <date>
          <smXML:CI_Date>
            <date>2008-05-09</date>
            <dateType>creation</dateType>
          </smXML:CI_Date>
        </date>
        <citedResponsibleParty>
          <smXML:CI_ResponsibleParty>
            <individualName>Chief,%20Pacific%20Hydrographic%20Branch</individualName>
            <organisationName>NOAA,%20NOS,%20COOS,%20Hydrographic%20Surveys%20Division</organisationName>
            <positionName>Chief,%20Pacific%20Hydrographic%20Branch</positionName>
          </smXML:CI_ResponsibleParty>
        </citedResponsibleParty>
      </smXML:CI_Citation>
    </citation>
  </smXML:BAG_DataIdentification>
</identificationInfo>

Mark activated
</contact>
<dateStamp>2008-05-09</dateStamp>
<metadataStandardName>ISO%2019115</metadataStandardName>
<metadataStandardVersion>2003</metadataStandardVersion>
</smXML:MD_Metadata>%0a
<?xml version="1.0"?>
<smXML:MD_Metadata>
  <md:abstract>
  </md:abstract>
</smXML:MD_Metadata>
18-bag-hdf-xml.org
18 baggage
Gnu Emacs
scratch
messages
next buffer C-x C-right
previous buffer C-x C-left
select named buffer C-x b
list all buffers C-x C-b
individualname Chief, Pacific Hydrographic Branch
organisationname NOAA, NOS, COASTAL HYDROGRAPHIC SURVEYS, DIVISION
positionname Chief, Pacific Hydrographic Branch
pointofcontact role
historicalarchive status
topiccategory elevation
extent

In emacs, we need to make this more readable:

\[ M-x \text{replace-string} > < \text{RET} > C-q C-j < \]

Got that? C-q says to take the next character literal. Then C-j is a new line (aka move the move down a line).

Now we want to change the indenting of the XML metadata file.

- Go to the beginning of the file: C-<
- Start marking a region: C-space
- Go to the end of the file: C->
- Ask emacs to indent all: M-x indent-region
  - or press C-M-

Not the most fun way to get the metadata out of a bag!

*Bags from python*

See also: [https://github.com/schwehr/bag-py/](https://github.com/schwehr/bag-py/)

```sh
#BEGIN_SRC sh
sudo apt-get install python-h5py # Only need to do this once!
ipython
#END_SRC

#BEGIN_SRC python
bag = h5py.File('H11703_Office_5m.bag')
bag.values()
bag.items()
root = bag['/BAG_root']
root.values()
root.items()
root.keys()
```

metadata = bag['/BAG_root/metadata']
```bash
researchtools@ubuntu:~/class/18/bags$ ls -l
total 3408604
-rw-r--r-- 1 researchtools researchtools 22967808 2011-10-27 09:44 H11703_Combined_5m.bag
-rw-r--r-- 1 researchtools researchtools 23003248 2011-10-27 09:44 H11703_Office_5m.bag
-rw-r--r-- 1 researchtools researchtools 5260 2011-11-01 11:47 H11703_Office_5m.xml
-rw-r--r-- 1 researchtools researchtools 3391351952 2011-05-25 20:31 H12263_MB_1m_MLLW_1of4.bag
-rw-r--r-- 1 researchtools researchtools 53067176 2011-05-25 20:50 H12263_MB_8m_MLLW_combined.bag
researchtools@ubuntu:/class/18/bags$ emacsclient --no-wait H11703_Office_5m.xml
researchtools@ubuntu:/class/18/bags$ sudo apt-get install python-h5py
[sudo] password for researchtools:
Reading package lists... Done
Building dependency tree
Reading state information... Done
python-h5py is already the newest version.
0 upgraded, 0 newly installed, 0 to remove and 37 not upgraded.
researchtools@ubuntu:/class/18/bags$
```

See also: [https://github.com/schwehr/bag-py/](https://github.com/schwehr/bag-py/)

```bash
#BEGIN_SRC sh
sudo apt-get install python-h5py  # Only need to do this once!
ipython
#END_SRC

#BEGIN_SRC python
bag = h5py.File('H11703_Office_5m.bag')
for values in bag.values():
    print('items()')
```

---

18-bag-hdf-xml.org  63% L178  [#,r] (Org)------

menu-bar edit copy

backup

video

snippets
In [2]: h5py.File('H11703_Combined_5m.bag')
H11703_Office_5m.bag
H11703_Office_5m.bag
H11703_Office_5m.bag
H11703_Office_5m.xml
H12263_MB_8m_MLLW_combined.bag
H12263_MB_1m_MLLW_lof4.bag

In [2]: h5py.File('H11703_Combined_5m.bag')
H11703_Office_5m.bag
H11703_Office_5m.bag
H11703_Office_5m.xml

In [2]: h5py.File('H11703_Combined_5m.bag')
H11703_Office_5m.bag
H11703_Office_5m.bag
H11703_Office_5m.xml

In [2]: h5py.File('H11703_Combined_5m.bag')
H11703_Office_5m.bag
H11703_Office_5m.bag
H11703_Office_5m.xml

# END_SRC

--- 18-bag-hdf-xml.org 63% L178 [#,$] (Org)---

menu-bar edit copy

backup

video

snippets
In [2]: h5py.File('H11703_Office_5m.xml')
H11703_Office_5m.xml
H12263_MB_8m_MLLW_combined.bag
H11703_Office_5m.xml

In [2]: h5py.File('H11703_Office_5m.xml')
H11703_Office_5m.xml
H11703_Office_5m.xml

In [2]: h5py.File('H11703_Office_5m.xml')
Out[2]: <HDF5 file "H11703_Office_5m.xml" (mode r+, 21.9M)>

In [3]: bag = h5py.File('H11703_Office_5m.xml')

In [4]: whos
Variable   Type       Data/Info
----------- ----------- ------------------------
bag         File       <HDF5 file "H11703_Office...5m.xml" (mode r+, 21.9M)>
h5py        module     <module 'h5py' from '/usr...on2.7/h5py/_init__pyc'>

In [5]:

```python
#BEGIN_SRC python
bag = h5py.File('H11703_Office_5m.xml')
bag.values()
bag.items()
#END_SRC
```
In [5]: bag.values()
Out[5]: [<HDF5 group "/BAG_root" (4 members)>]

In [6]: bag.items()
Out[6]: [('BAG_root', <HDF5 group "/BAG_root" (4 members)>)]

In [7]: root = bag['/BAG_root']

In [8]: root.items()
Out[8]: [('elevation', <HDF5 dataset "elevation": shape (1434, 2004), type "<f4">),
        ('metadata', <HDF5 dataset "metadata": shape (5097,), type "|S1">),
        ('tracking_list', <HDF5 dataset "tracking_list": shape (0,), type "|V19">),
        ('uncertainty', <HDF5 dataset "uncertainty": shape (1434, 2004), type "<f4">)]

In [9]: metadata = bag['/BAG_root/metadata']

In [10]: type(metadata)
Out[10]: <class 'h5py.highlevel.Dataset'>

In [11]:

bag.items()
root = bag['/BAG_root']
root.values()
root.items()
root.keys()

metadata = bag['/BAG_root/metadata']
type(metadata)
metadata[0]
metadata.value

metadata_txt = ''.join(metadata)
metadata_txt[:80]
In [10]: <class 'h5py.highlevel.Dataset'>
Out[11]: '<

In [12]: metadata.value
Out[12]:
array(['<', '?', 'x', '...', '>', '

In [13]: metadata_txt = ''.join(metadata.value)

In [14]: metadata_txt[:100]
Out[14]: '<?xml version="1.0"?>

In [15]: elevation = bag['/BAG_root/elevation'].value

In [16]:

metadata_txt[:80]

elevation = bag['/BAG_root/elevation'].value
elevation.shape
elevation.min()
elevation.max()

from matplotlib import pyplot
pyplot.plot(elevation)
pyplot.show()  # YUCK!
pyplot.imshow(elevation)  # Better but not great
#+END_SRC

Can we make the plot more useful?
```python
In [14]: metadata_txt[:100]
Out[14]: '<?xml version="1.0"?>
   <smXML:MD_Metadata xmlns:smXML="http://metadata.dgiwg.org/smXML" xmlns:xlink=""

In [15]: elevation = bag['/BAG_root/elevation'].value

In [16]: type(elevation)
Out[16]: <type 'numpy.ndarray'>

In [17]: elevation.shape
Out[17]: (1434, 2004)

In [18]: elevation.min()
Out[18]: -107.07632

In [19]: elevation.max()
Out[19]: 1000000.0

In [20]: from matplotlib import pyplot
In [21]: pyplot.plot(elevation)
```

Can we make the plot more useful?

---

18-bag-hdf-xml.org 70% L200 [#,r] (Org)
In [22]:

In [23]: pyplot.show()

In [24]: pyplot.imshow(elevation)

from matplotlib import pyplot
pyplot.plot(elevation)
pyplot.show() # YUCK!
pyplot.imshow(elevation) # Better but not great
# END_SRC

Can we make the plot more useful?
Can we make the plot more useful?

```python
from matplotlib import pyplot

pyplot.plot(elevation)
pyplot.show()  # YUCK!
pyplot.imshow(elevation)  # Better but not great

# END_SRC
```
In [29]: import numpy

In [30]: numpy.N
   numpy.NINF    numpy.NZERO    numpy.NaN

In [30]: numpy.NAN?
Type:   float
Base Class:   <type 'float'>
String Form:    nan
Namespace:    Interactive
Docstring:
float(x) -> floating point number

   Convert a string or number to a floating point number, if possible.

In [31]: numpy.NAN
Out[31]: nan

In [32]:
   # BEGIN_SRC python
   import numpy # for NAN aka "not a number"
   for x in range(elevation.shape[0]):
       for y in range(elevation.shape[1]):
           if elevation[x,y] > 0:
               elevation[x,y] = numpy.NAN
   # wait a while... this isn't fast
   
   pyplot.figure(2)
   pyplot.imshow(elevation)

   e_data = elevation.reshape(elevation.size)
**In [30]:** numpy.N

numpy.NAN  numpy.NINF  numpy.NZERO  numpy.NaN

**In [30]:** numpy.NAN?

* Type:  
  float

* Base Class:  
  <type 'float'>

* String Form:  
  nan

* Namespace:  
  Interactive

* Docstring:  
  float(x) -> floating point number

  Convert a string or number to a floating point number, if possible.

**In [31]:** numpy.NAN

**Out[31]:** nan

**In [32]:** for x in range(elevation.shape[0]):
  ....:     for y in range(elevation.shape[1]):
  ....:         if elevation[x,y] > 0:
  ....:             elevation[x,y] = numpy.NAN

```python
import numpy # for NAN aka "not a number"

for x in range(elevation.shape[0]):
    for y in range(elevation.shape[1]):
        if elevation[x,y] > 0:
            elevation[x,y] = numpy.NaN

# wait a while... this isn't fast

pyplot.figure(2)
pyplot.imshow(elevation)
```

```
... 18-bag-hdf-xml.org  73% L212  [#] (Org)
```
**imshow**

Display the image in *X* to current axes. *X* may be a float array, a uint8 array or a PIL image. If *X* is an array, *X* can have the following shapes:

* MxN -- luminance (grayscale, float array only)
* MxNx3 -- RGB (float or uint8 array)
* MxNx4 -- RGBA (float or uint8 array)
import numpy
for x in range(elevation.shape[0]):
    for y in range(elevation.shape[1]):
        if elevation[x,y] > 0:
            elevation[x,y] = numpy.NAN

e_data = elevation.reshape(elevation.size)