FAIR Data Principles, Data formats & metadata standards, and Provenance

Ahmed Eleish

Data Science – ITWS/CSCI/ERTH 4350/6350 Module 3, September 17, 2025

Tetherless World Constellation
Rensselaer Polytechnic Institute

Reading Assignments from last week

Metadata and Provenance Management [16]
W3 PROV Overview [19]

Contents

- Data Management Review
- FAIR data principals
- Data formats
- Metadata standards, conventions,
- Provenance

Data Management

- Creation of logical collections
- Physical data handling
- Interoperability support
- Security support
- Data ownership
- Metadata collection, management and access
- Persistence
- Knowledge and information discovery
- Data distribution and publication

Data Management

- Creation of logical collections

e.g. a catalogue mapping data objects to files/database views with standardized naming conventions.

- Physical data handling

e.g. storing the data in relational database tables with foreign key constraints linking the tables, backups monthly placed in a “backup” directory.

- Interoperability support

e.g. backups exported as CSVs, metadata follow naming convention from a controlled vocabulary.

Data Management

- Security support

e.g. data stored on a network shared storage device with access and permissions managed through Microsoft Active Directory.

- Data ownership

e.g. data products are assigned to specific team member for quality control and metadata injection.

- Metadata collection, management and access

e.g. metadata associated with lab experiments are recorded in plain text file using the LINCS Phase II Extended Metadata Standards.

Data Management

- Persistence

e.g. Data package will be submitted to RPI Institute Archives and Special Collections for archival, archival package must conform to the OAIS standard.

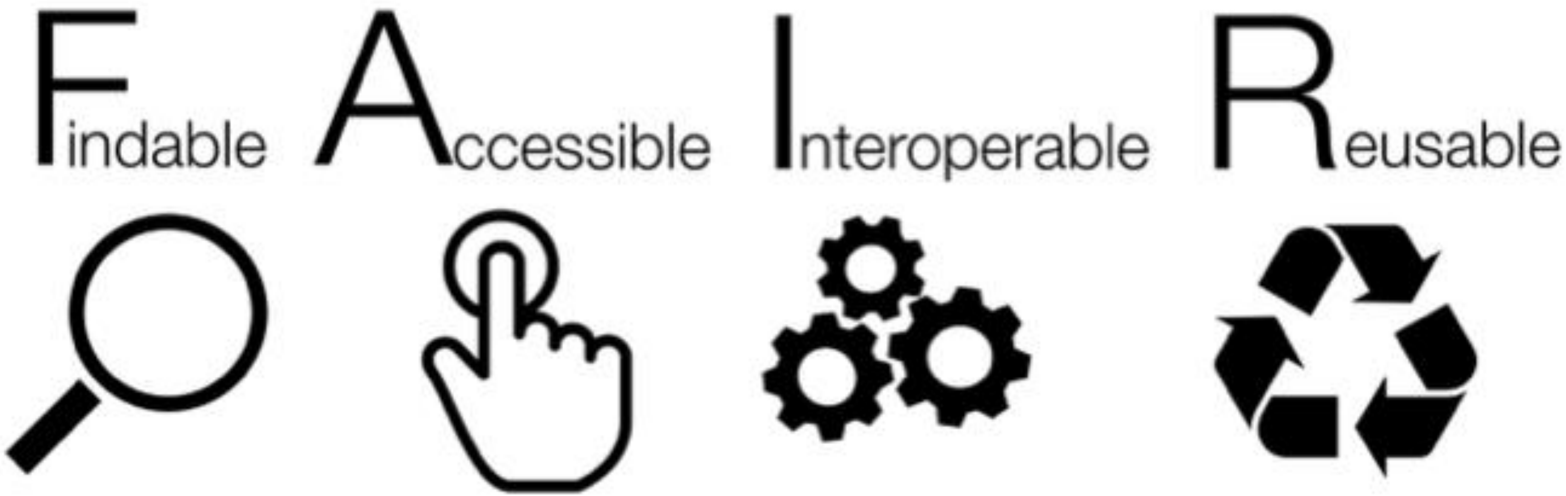
- Knowledge and information discovery

e.g. An interactive visualization allows researchers to examine a portion of the data before downloading the entire dataset OR statistics computed from the dataset are published along with the data files.

- Data distribution and publication

e.g. Dataset is stored on a content management system with version control that sends notifications to users when data changes.

FAIR

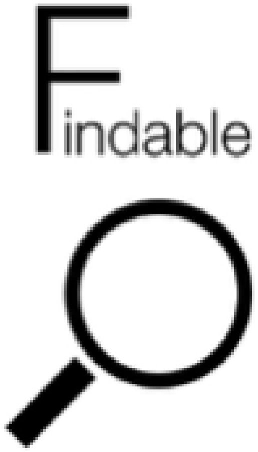


https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Findable

- The first step in (re)using data is to find them. **Metadata and data should be easy to find for both humans and computers**.

- **Machine-Readable** metadata are essential for automatic **Discovery** of datasets and services, so this is an essential component of the *FAIRification* process.



Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit:
https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Findable

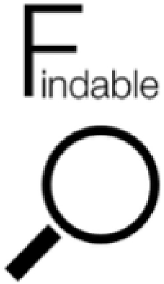
• The first step in (re)using data is to **find** them. Metadata and data should be easy to find for both humans and computers. **Machine readable metadata** are essential for **automatic discovery** of datasets and services, so this is an essential component of the FAIRification process.

• *F1. (Meta)data are assigned a globally unique and persistent identifier*

• *F2. Data are described with rich metadata (defined by R1 in Reusable below)*

• *F3. Metadata clearly and explicitly include the identifier of the data they describe*

• *F4. (Meta)data are registered or indexed in a searchable resource*



Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Accessible

Once the user finds the required data, they need to know how they can be accessed, possibly including: *authentication and authorization.*

Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Accessible

Once the user finds the required data, they need to know how they can be accessed, possibly including:
Authentication and Authorization.

- *A1. (Meta)data are retrievable by their identifier using a standardized communications protocol*
    - *A1.1 The protocol is open, free, and universally implementable*
    - *A1.2 The protocol allows for an authentication and authorization procedure, where necessary*
- *A2. Metadata are accessible, even when the data are no longer available*

Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Interoperable

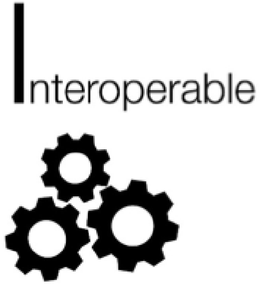
The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Interoperable

The data usually need to be integrated with other data. In addition, the data need to interoperate with applications or workflows for analysis, storage, and processing.

- *I1. (Meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.*
- *I2. (Meta)data use vocabularies that follow FAIR principles*
- *I3. (Meta)data include qualified references to other (meta)data*



Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Reusable

- The *ultimate goal* of FAIR is to optimize the reuse of data.
- To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.

Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Reusable

- *The ultimate goal of FAIR is to optimize the reuse of data.*
- *To achieve this, metadata and data should be well-described so that they can be replicated and/or combined in different settings.*

- *R1. Meta(data) are richly described with a plurality of accurate and relevant attributes*
- *R1.1. (Meta)data are released with a clear and accessible data usage license*
- *R1.2. (Meta)data are associated with detailed provenance*
- *R1.3. (Meta)data meet domain-relevant community standards*

Resource/Reference: https://www.go-fair.org/fair-principles/
Image Credit: https://en.wikipedia.org/wiki/FAIR_data#/media/File:FAIR_data_principles.jpg

Data Formats

Data Formats

- We will cover some (not all)
  - ASCII, UTF-8, ISO 8859-1
  - Self-describing formats
  - Table-driven
  - Markup languages and other web-based
  - Databases
  - Graphs
  - Unstructured

ASCII

- American Standard Code for Information Interchange
- http://www.webopedia.com/TERM/A/ASCII.html
- Table of characters
- http://www.webopedia.com/quick_ref/asciicode.asp
- ISO-8859-1 (aka ISO Latin 1) is a superset of ASCII – used on the web to represent ‘non- ASCII’ characters
- Non-printing characters

Example – good or bad?

1749 01 58.0
1749 02 62.6
1749 03 70.0
1749 04 55.7
1749 05 85.0
1749 06 83.5
1749 07 94.8
1749 08 66.3
1749 09 75.9
1749 10 75.5
1749 11 158.6
1749 12 85.2
1750 01 73.3
1750 02 75.9
1750 03 89.2
1750 04 88.3
1750 05 90.0
1750 06 100.0

Example – good or bad?

```
1749 01  58.0
1749 02  62.6
1749 03  70.0
1749 04  55.7
1749 05  85.0
1749 06  83.5
1749 07  94.8
1749 08  66.3
1749 09  75.9
1749 10  75.5
1749 11 158.6
1749 12  85.2
1750 01  73.3
1750 02  75.9
1750 03  89.2
1750 04  88.3
1750 05  90.0
1750 06 100.0
1750 07  85.4
1750 08 103.0
1750 09  91.2
1750 10  65.7
1750 11  63.3
```

```
                    MONTHLY MEAN SUNSPOT NUMBERS
==========================================================
========================
Year   Jan  Feb  Mar  Apr  May  Jun  Jul  Aug  Sep  Oct  Nov  Dec
----------------------------------------------------------------
1749   58.0 62.6 70.0 55.7 85.0 83.5 94.8 66.3 75.9 75.5 158.6
85.2
1750   73.3 75.9 89.2 88.3 90.0 100.0 85.4 103.0 91.2 65.7 63.3
75.4

1751   70.0 43.5 45.3 56.4 60.7 50.7 66.3 59.8 23.5 23.2 28.5 44.0
1752   35.0 50.0 71.0 59.3 59.7 39.6 78.4 29.3 27.1 46.6 37.6 40.0
1753   44.0 32.0 45.7 38.0 36.0 31.7 22.0 39.0 28.0 25.0 20.0  6.7
1754    0.0  3.0  1.7 13.7 20.7 26.7 18.8 12.3  8.2 24.1 13.2  4.2
1755   10.2 11.2  6.8  6.5  0.0  0.0  8.6  3.2 17.8 23.7  6.8 20.0

1756   12.5  7.1  5.4  9.4 12.5 12.9  3.6  6.4 11.8 14.3 17.0  9.4
1757   14.1 21.2 26.2 30.0 38.1 12.8 25.0 51.3 39.7 32.5 64.7 33.5
1758   37.6 52.0 49.0 72.3 46.4 45.0 44.0 38.7 62.5 37.7 43.0 43.0
1759   48.3 44.0 46.8 47.0 49.0 50.0 51.0 71.3 77.2 59.7 46.3 57.0
1760   67.3 59.5 74.7 58.3 72.0 48.3 66.0 75.6 61.3 50.6 59.7 61.0

1761   70.0 91.0 80.7 71.7 107.2 99.3 94.1 91.1 100.7 88.7 89.7
46.0
1762   43.8 72.8 45.7 60.2 39.9 77.1 33.8 67.7 68.5 69.3 77.8 77.2
1763   56.5 31.9 34.2 32.9 32.7 35.8 54.2 26.5 68.1 46.3 60.9 61.4
1764   59.7 59.7 40.2 34.4 44.3 30.0 30.0 30.0 28.2 28.0 26.0 25.7
```

Example – good or bad?

```
% #0   Date-time:      9/12/2006 4:07:21 PM
% #1   Recorder:       7T0271
% #2   File type:      1
% #3   Columns:        3
% #4   Channels:       1
% #5   Field separation:     0
% #6   Decimal point:  1
% #7   Date def.:      0     1
% #8   Time def.:      0
% #9   Channel 1:      Temperature(<B0>C)     Temp(<B0>C)    3     1
% #11  Reconvertion:   0
% #19  Line color:     1     2     3     4
1      30 07 06 15 30 00      22.712
2      30 07 06 15 31 00      22.673
3      30 07 06 15 32 00      22.635
4      30 07 06 15 33 00      22.609
5      30 07 06 15 34 00      22.558
6      30 07 06 15 35 00      22.532
7      30 07 06 15 36 00      22.494
8      30 07 06 15 37 00      22.468
9      30 07 06 15 38 00      22.442
10     30 07 06 15 39 00      22.430
11     30 07 06 15 40 00      22.404
```

Where is the data?
Where is the provenance?

Making ASCII more useful

- Delimited: Comma Separated Values (CSV) or Tab Separated Values (TSV)

  – Improves parsing
  – How to handle special characters? (UTF-8 + double quotes)

- Moving them in/out of "Excel"

Data in “data structures”

- JSON – JavaScript Object Notation json.org/example

```
{"menu": {
   "id": "file", "value": "File", "popup": {
      "menuitem": [
         {"value": "New", "onclick": "CreateNewDoc()"},
         {"value": "Open", "onclick": "OpenDoc()"},
         {"value": "Close", "onclick": "CloseDoc()"}
      ]
   }
}}
```

Data in “data structures”

- JSON – JavaScript Object Notation json.org/example

```
{"menu": {
   "id": "file", "value": "File", "popup": {
      "menuitem": [
         {"value": "New", "onclick": "CreateNewDoc()"},
         {"value": "Open", "onclick": "OpenDoc()"},
         {"value": "Close", "onclick": "CloseDoc()"}
      ]
   }
}}
```

The same text expressed as XML:

```
<menu id="file" value="File">
  <popup>
    <menuitem value="New" onclick="CreateNewDoc()" />
    <menuitem value="Open" onclick="OpenDoc()" />
    <menuitem value="Close" onclick="CloseDoc()" />
  </popup>
</menu>
```

Data in "applications"

- Increasing trend in storing data in application files, e.g. *.xls (Excel), *.mat (Matlab), *.sav (IDL - Interactive Data Language), ...

- What are the advantages?
  – Ready to use
  – Data structures are provided

- What problems?
  – Data structures may not match the underlying data representation (model), i.e. information and data may be lost (e.g. float instead of double)
  – Format versions
  – Interoperability – can it be read by another app?

Free Form

- 20+ years ago there was an attempt to provide a templated (almost table driven) approach
- Good homework assignment (ungraded) when you are bored – find out why it was created and what happened to it
- Search “Esperanto”

Spreadsheets

- e.g. Excel – import data, Save As csv

Volcanic_of2007-1114_table6.xls

	A	B	C	D	E	F	G	H
1	Date	SO2 (t/d)	SD (t/d)	WS (m/s)	WD (degrees)	N	Location	Code
2	10/01/02	660	380	9.2	9	6	L	C
3	16/01/02	610	180	8.9	14	7	L	C
4	1/02/02	1710	--	3.5	70	1	U	C
5	4/02/02	1050	270	5.5	54	3	U	C
6	11/02/02	1170	310	10.3	30	7	L	B
7	21/02/02	950	150	7.9	30	6	L	C
8	25/02/02	1280	240	11.3	30	6	L	C
9	4/03/02	720	120	5.2	40	6	UL	C
10	18/03/02	1010	250	14.7	30	7	L	A
11	2/04/02	1150	200	8.3	355	5	L	B
12	2/05/02	980	220	5.6	34	5	U	C
13	16/05/02	1240	260	6.1	34	7	U	B
14	20/05/02	1510	200	7.2	70	3	U	C
15	29/05/02	900	250	5	80	7	U	C
16	30/05/02	860	230	4.7	56	4	U	C
17	4/06/02	1020	430	6	48	8	U	B
18	12/06/02	1170	380	7	25	7	U	C

Documentation?

Volcanic_of2007-1114_table6.xls

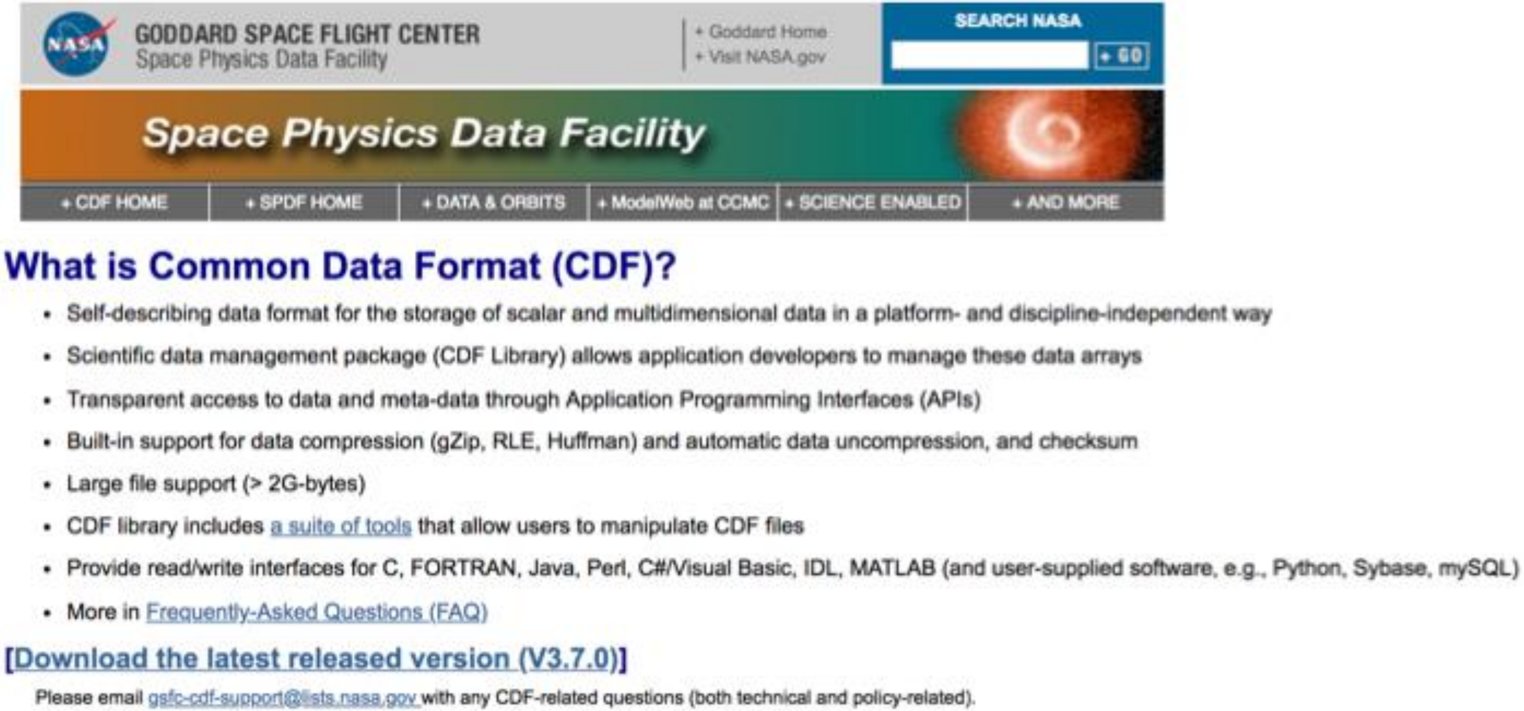
	A	B	C	D	E	F	G	H	I	J	K
218	20/12/06	1830	830	5.7	33	6	U	C			
219	29/12/06	1680	630	4.4	80	6	U	C			
220											
221	Table 6.										
222	Kilauea east rift zone SO2 emission rates - vehicle-based										
223											
224											
225	Location Codes: (see fig. 1)										
226	U- Above the 180° turn at Holei Pali (upper Chain of Craters Road)										
227	L- Below Holei Pali (lower Chain of Craters Road)										
228	UL-individual traverses were made both above and below the 180° turn at Holei Pali										
229	H- Highway 11										
230											
231											
232	Data Quality Codes:										
233	A - BEST QUALITY DATA -usually with strong, steady, well constrained wind conditions, and a compact, consistent plume shape.										
234	(15.7% of data)										
235	B - GOOD QUALITY DATA - usually with moderately consistent plume shape and location of plume on road.										
236	Collected under moderately strong, uniform winds, with good constraint on wind speed and direction.										
237	(40.7% of data)										
238	C - ACCEPTABLE DATA - may have variable plume location and shape. Wind speed and direction may be										
239	variable or poorly constrained. Some runs may measure a partial plume, and result in a minimum										
240	emission rate. Measurements with instrument inconsistencies are included in this category.										
241	(43.5% of data)										
242	Abreviations: t/d=metric tonne (1000 kg)/day, SD=standard deviation, WS=wind speed, WD=wind direction east of true north, N=number of t										
243											
244	[1]SO2 measurements by FLYSPEC										
245	[2]Reported SO2 measurements prior to this date are by COSPEC; those from this date onward are by FLYSPEC.										
246											

Sheet1 Sheet2 Sheet3

Some data formats we will see

- CDF - Common Data Format
- netCDF - Network Common Data Format
- HDF4 - Hierarchical Data Format 4
- HDF5 - Hierarchical Data Format 5
- HDEOS - Hierarchical Data Format Earth Observing System

CDF (Common Data Format)


GODDARD SPACE FLIGHT CENTER
Space Physics Data Facility

+ Goddard Home
+ Visit NASA.gov

SEARCH NASA
+ GO

Space Physics Data Facility

+ CDF HOME | + SPDF HOME | + DATA & ORBITS | + ModelWeb at CCMC | + SCIENCE ENABLED | + AND MORE

What is Common Data Format (CDF)?

- Self-describing data format for the storage of scalar and multidimensional data in a platform- and discipline-independent way
- Scientific data management package (CDF Library) allows application developers to manage these data arrays
- Transparent access to data and meta-data through Application Programming Interfaces (APIs)
- Built-in support for data compression (gZip, RLE, Huffman) and automatic data uncompression, and checksum
- Large file support (> 2G-bytes)
- CDF library includes a suite of tools that allow users to manipulate CDF files
- Provide read/write interfaces for C, FORTRAN, Java, Perl, C#/Visual Basic, IDL, MATLAB (and user-supplied software, e.g., Python, Sybase, mySQL)
- More in Frequently-Asked Questions (FAQ)

[Download the latest released version (V3.7.0)]

Please email gsfc-cdf-support@lists.nasa.gov with any CDF-related questions (both technical and policy-related).

https://cdf.gsfc.nasa.gov/

CDF (Common Data Format)

- **The Common Data Format(CDF) is a self-describing data format for the storage and manipulation of scalar and multidimensional data in a platform- and discipline-independent fashion**

- Although CDF has its own internal self-describing format, it consists of more than just a data format. **CDF is a scientific data management package (known as the "CDF Library")** which allows programmers and application developers to manage and manipulate scalar, vector, and multi-dimensional data arrays

https://cdf.gsfc.nasa.gov/

CDFML

- *The CDF office realized that scientific progress is often impeded by the lack of, or excessive multiplicity of, available standards for data formats and structures and/or data format translators. In a bid to facilitate and promote data sharing with other data formats, the CDF office has decided to adopt Extensible Markup Language (XML) as a basis for establishing interoperability with other scientific data formats and created CDF Markup Language (CDFML) to describe CDF data and metadata.*

Hierarchical Data Format 4 (HDF4)

• At its lowest level, HDF is a physical file format for storing scientific data

• At its highest level, HDF is a collection of utilities and applications for manipulating, viewing, and analyzing data in HDF files

• Between these levels, HDF is a software library that provides high-level APIs and a low-level data interface

https://www.hdfgroup.org/solutions/hdf4/

Hierarchical Data Format 5 (HDF5)

- HDF5 is a data model, library, and file format for storing and managing data. It supports an unlimited variety of datatypes, and is designed for flexible and efficient I/O and for high volume and complex data.

- HDF5 is portable and is extensible, allowing applications to evolve in their use of HDF5.

- The HDF5 Technology suite includes tools and applications for managing, manipulating, viewing, and analyzing data in the HDF5 format.

- VERY complex API

http://www.hdfgroup.org/HDF5/

HDF5View

- A Quick Look at the HDF5 File Format Using HDFView

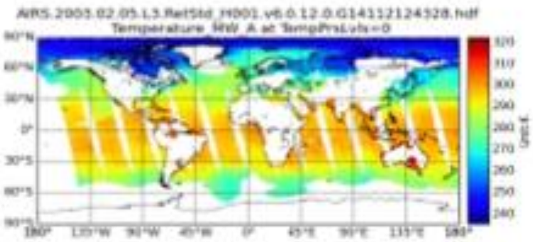
Take a look at this example found on YouTube on HDF5 Viewer:

https://www.youtube.com/watch?v=q14F3WRwSck

HDFEOS

- A variant of HDF for the Earth Observing System (EOS)

http://hdfeos.org/



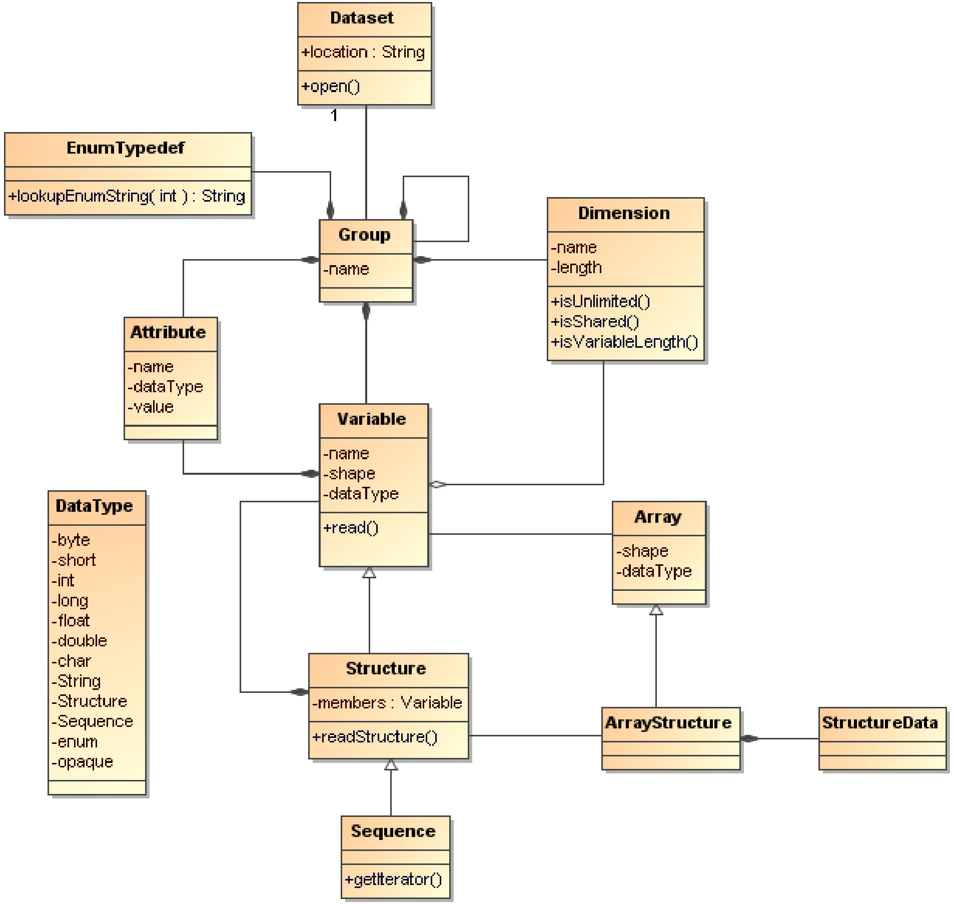
Common Data Model

- Combines netCDF and HDF into one model, and API
- Uses the underlying HDF format representation but uses the netCDF v4 API
- Simplifies access

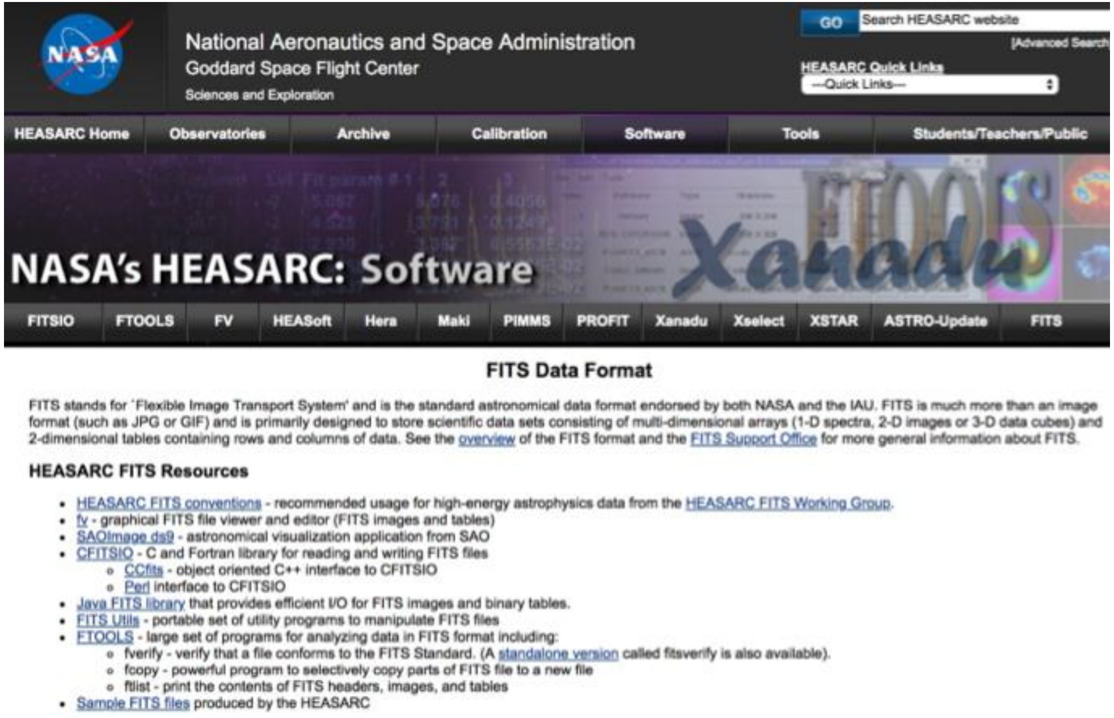
- Latest: https://downloads.unidata.ucar.edu/netcdf-java/

- CDM overview https://docs.unidata.ucar.edu/netcdf-java/current/userguide/common_data_model_overview.html
(note the “data model”)

Common Data Model
Data Access Layer
Object Model



FITS (Flexible Image Transport System)



National Aeronautics and Space Administration
Goddard Space Flight Center
Sciences and Exploration

GO | Search HEASARC website | [Advanced Search]

HEASARC Quick Links

HEASARC Home | Observatories | Archive | Calibration | Software | Tools | Students/Teachers/Public

NASA's HEASARC: Software

FITSIO | FTOOLS | FV | HEASoft | Hera | Maki | PIMMS | PROFIT | Xanadu | Xselect | XSTAR | ASTRO-Update | FITS

FITS Data Format

FITS stands for 'Flexible Image Transport System' and is the standard astronomical data format endorsed by both NASA and the IAU. FITS is much more than an image format (such as JPG or GIF) and is primarily designed to store scientific data sets consisting of multi-dimensional arrays (1-D spectra, 2-D images or 3-D data cubes) and 2-dimensional tables containing rows and columns of data. See the overview of the FITS format and the FITS Support Office for more general information about FITS.

HEASARC FITS Resources

- HEASARC FITS conventions - recommended usage for high-energy astrophysics data from the HEASARC FITS Working Group.
- fv - graphical FITS file viewer and editor (FITS images and tables)
- SAOImage ds9 - astronomical visualization application from SAO
- CFITSIO - C and Fortran library for reading and writing FITS files
  - CCfits - object oriented C++ interface to CFITSIO
  - Perl interface to CFITSIO
- Java FITS library that provides efficient I/O for FITS images and binary tables.
- FITS Utils - portable set of utility programs to manipulate FITS files
- FTOOLS - large set of programs for analyzing data in FITS format including:
  - fverify - verify that a file conforms to the FITS Standard. (A standalone version called fitsverify is also available).
  - fcopy - powerful program to selectively copy parts of FITS file to a new file
  - ftlist - print the contents of FITS headers, images, and tables
- Sample FITS files produced by the HEASARC

FITS

- FITS stands for ‘Flexible Image Transport System’ and is the standard astronomical data format endorsed by both NASA and the IAU.

- FITS is much more than an image format (such as JPG or GIF) and is primarily designed to store scientific data sets consisting of multi-dimensional arrays (1-D spectra, 2-D images or 3-D data cubes) and 2-dimensional tables containing rows and columns of data.

- FITS includes many APIs

https://heasarc.gsfc.nasa.gov/docs/heasarc/fits.html

TIFF/GeoTIFF

• Tagged Image File Format 24-bit support

• GeoTIFF is a public domain metadata standard which allows georeferencing information to be embedded within a TIFF file.

•The potential additional information includes projections, coordinate systems, ellipsoids, datums, and everything else necessary to establish the exact spatial reference for the file.

•The GeoTIFF format is fully compliant with TIFF 6.0, so software incapable of reading and interpreting the specialized metadata will still be able to open a GeoTIFF file.

http://www.libtiff.org/

Binary Universal Form for the Representation of meteorological data (BUFR)

https://public.wmo.int/en

https://en.wikipedia.org/wiki/BUFR

BUFR

- Binary Universal Form for the Representation of meteorological data (BUFR) is a binary data format maintained by the World Meteorological Organization

- The latest version is BUFR Edition 4

- BUFR Edition 3 is also considered current for operational use

https://public.wmo.int/en

GriB

- **General Regularly-distributed Information in Binary form**

- GRIB (GRIdded Binary) is a mathematically concise data format commonly used in meteorology to store historical and forecast weather data

- See Wikipedia page for more details

https://en.wikipedia.org/wiki/GRIB

Resource Description Framework (RDF)

- http://www.w3.org/RDF/ - Resource Description Framework
  - Read the introduction and overview

- Many tools, and very good language support

- RDF is the foundation of 'data on the web',
see ~~www.linkeddata.org~~ RIP :'(

•JSON-LD (JSON for Linked Data)

We cover this more in a later class (maybe)..

Metadata Standards

Dublin Core

• Dublin Core Metadata Initiative(DCMI) is an open organization engaged in the development of interoperable online metadata standards that support a broad range of purposes and business models.

– ISO Standard 15836-2003 of February 2003

– ANSI/NISO Standard Z39.85-2007 of May 2007

– IETF RFC 5013 of August 2007

• Metadata terms - http://dublincore.org/documents/dcmi-terms/

Date/Time

- ISO 8601 specifies numeric representations of date and time.
  – helps to avoid confusion in international communication due to different national notations
  – increases the portability of computer user interfaces

The international standard date notation is **YYYY-MM-DD**
where YYYY is the year in the usual Gregorian calendar, MM is the month of the year between 01 (January) and 12 (December), and DD is the day of the month between 01 and 31.

The international standard notation for the time of day is **hh:mm:ss**

Good read: http://www.cl.cam.ac.uk/~mgk25/iso-time.html

- In XML encodings, see xsd:datetime
  – http://www.w3.org/TR/NOTE-datetime
  – http://www.w3.org/TR/xmlschema-2/

Advantages of ISO 8601

Advantages of the ISO 8601 standard date notation compared to other commonly used variants:

• easily readable and writeable by software (no ‘JAN’, ‘FEB’, ... table necessary)
• easily comparable and sortable with a trivial string comparison
• language independent
• can not be confused with other popular date notations
• consistency with the common 24h time notation system, where the larger units (hours) are also written in front of the smaller ones (minutes and seconds)
• strings containing a date followed by a time are also easily comparable and sortable (e.g. write “1995-02-04 22:45:00”)
• the notation is short and has constant length, which makes both keyboard data entry and table layout easier
• identical to the Chinese date notation, so the largest cultural group (>25%) on this planet is already familiar with it :-)
• date notations with the order “year, month, day” are in addition already widely used e.g. in Japan, Korea, Hungary, Sweden, Finland, Denmark, and a few other countries and people in the U.S. are already used to at least the “month, day” order
• a 4-digit year representation avoids overflow problems after 2099-12-31

Reference: https://www.cl.cam.ac.uk/~mgk25/iso-time.html

Spatial representation

- **ISO 19115:2003 defines the schema required for describing geographic information and services**

- It provides information about the identification, the extent, the quality, the spatial and temporal schema, spatial reference, and distribution of digital geographic data.

- ISO 19115:2003 is applicable to:

– the cataloguing of datasets, clearinghouse activities, and the full description of datasets
– geographic datasets, dataset series, and individual geographic features and feature properties.

ESML (Earth Science Markup Language)

- Earth science data is archived and distributed in many different formats varying from character format, packed binary, "standard" scientific formats to self-describing formats. This heterogeneity results in data-application interoperability problems for scientific tools. The Earth Science Markup Language (ESML) is an elegant solution to this problem

- https://www.itsc.uah.edu/home/projects/esml-earth-science-markup-language

CSML (Climate Science Markup Language)

• Climate Science Markup Language & Climate Data Markup Language
The Climate Data Markup Language (CDML) is the markup language used to represent metadata in CDMS (Community Data Management System)
The Community Data Management System is an object-oriented data management system, specialized for organizing multidimensional, gridded data used in climate analysis and simulation

• **CDMS is implemented as part of the Climate Data Analysis Tool**

[CDAT](https://cdat.llnl.gov/),
https://cdms.readthedocs.io/en/latest/manual/cdms_6.html

• CSML is a standards-based data model and GML (Geography Markup Language) application schema for atmospheric and oceanographic data with associated software tools developed at the Rutherford Appleton Laboratory.

More markup languages

• GML - Geography Markup Language – developed as a way to standardize geographic representations (to facilitate interoperability) ISO 19136:2007

– Stores data and metadata

– Because it focuses on coordinates, is important as representing structural elements, such as points, lines, polygons used in a specific discipline

– Features application schema to represent roads, rivers, etc.

– https://schemas.opengis.net/gml/3.3/

Markup languages

• **KML–Keyhole Markup Language**– developed as an interlingua for a specific application, i.e. Google Earth

– Currently stores data and metadata
– XML tag and nesting provides for embedding structure and associations between metadata and data
– Uses other markup languages, e.g. GML
• Currently, KML 2.3 (as of 2015, August) utilizes certain geometry elements derived from GML 2.1.2. These elements include point, line string, linear ring, and polygon.

– Can contain links (external) to other content

What is KML ?

• **KML is a file format used to display geographic data in an Earth browser such as Google Earth**.

• **You can create KML files to pinpoint locations, add image overlays, and expose rich data in new ways**.

• KML is an international standard that maintained by the Open Geospatial Consortium, Inc. (OGC).

KML is an XML language focused on geographic visualization, including annotation of maps and images. Geographic visualization includes not only the presentation of graphical data on the globe, but also the control of the user's navigation in the sense of where to go and where to look.

https://developers.google.com/kml/documentation/kml_tut

Who uses KML ?

- Casual users

- You can use KML to plan trips, share location data with friends, or record hikes you've been on.

- Scientists

- Scientific data, such as natural resource maps, or geographic trends, are easily shared as a KML file.

- Non-Profits

- KML files can be used to highlight problems and advocate change.

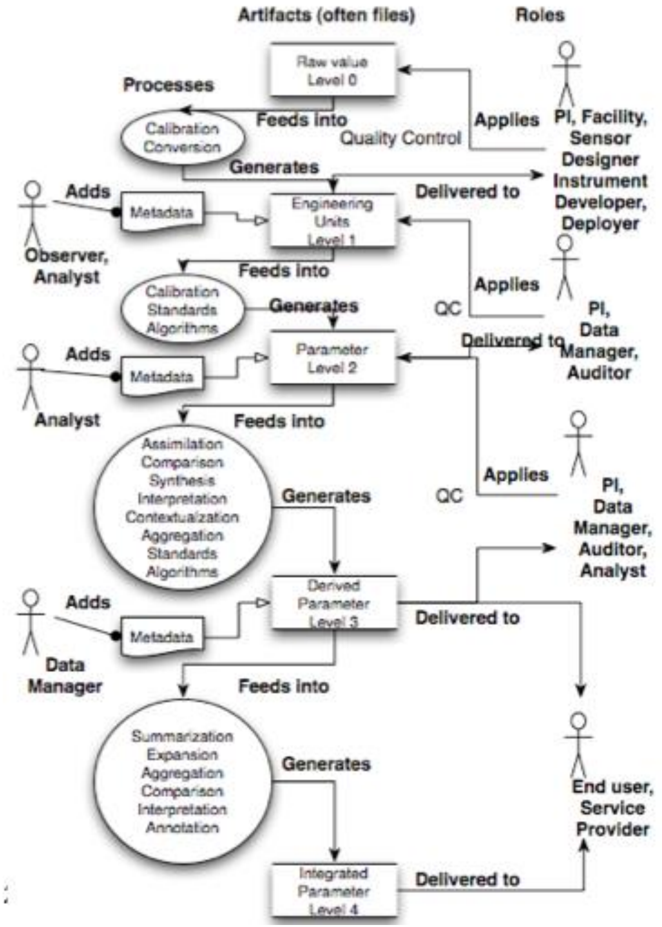
https://developers.google.com/kml/documentation/kml_tut

Provenance (again)

Provenance – metadata in a given context – think this way

- Who?
- What?
- Where?
- Why?
- When?
- How?

- Provenance in this data pipeline
- Provenance is metadata in *context*
- What context? – Who you are?
– What you are asking?
– What you will use the answer for?



Summary and Considerations

• What is in common about the data and metadata formats?

• Many choices for both – what are the key criteria for choosing?

– Faithful representation of structure of data
– Accurate representation of metadata with no (or minimal) loss of *information*

Class 3 Reading Materials

Required:

Metadata Encoding and Transfer Standard - METS [24]

Choose 2:

Data formats: netCDF [21]
Spatial Data Transfer Standard GIS format [22]
HDF5 TUTORIAL: Learning HDF5 with HDFVIEW [23]
Open Archives Initiative - Protocol for Metadata Harvesting - OAI-PMH [25]
Keyhole Markup Languge - KML Tutorial [26]
Earth Science Markup Language - ESML [27]
HDF5View User's Guide [28]
HDF5 files in Python [29]

Email summaries to eleisa2@rpi.edu or submit to LMS

Thanks!

Work on assignment 1!