NetCDF: Data Model, APIs, and Format

Russ Rew
Unidata Program Center
University Corporation for Atmospheric Research

Multidimensional Summit, ESRI, June 8, 2004

www.unidata.ucar.edu

Overview

- What is Unidata?
- What is netCDF?
- Data model
- Programming interfaces
- Format
- Limitations
- GIS connections
- Summary

What is Unidata?

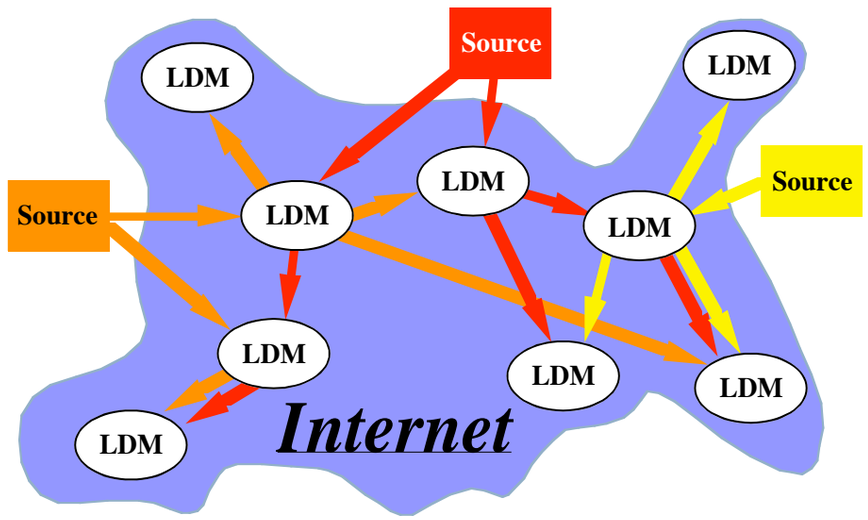
Unidata: A Community Endeavor

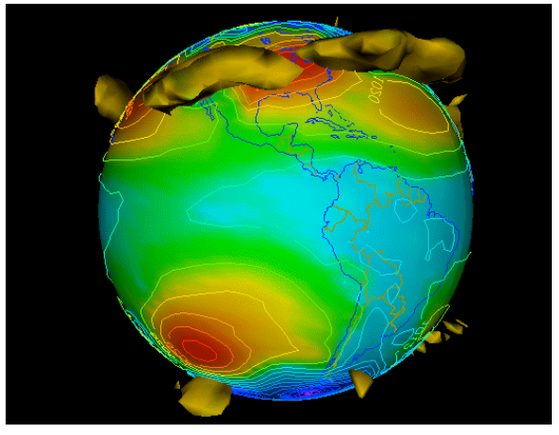
- Community of educators and researchers at 120 universities, 30 other institutions, international in scope
- Mission: *providing data, tools, support, and community leadership for enhanced earth-system education and research*
- Funded primarily by the National Science Foundation

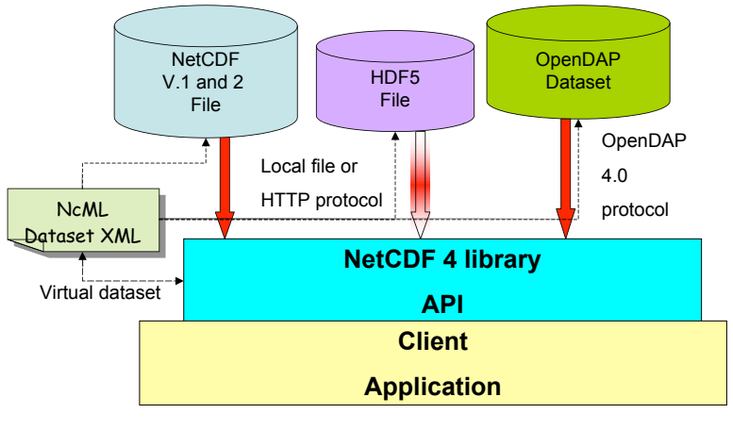
- Atmospheric science community, expanding to oceanography, hydrology, other geosciences
- Unidata Program Center: 25 staff, 15 developers

Unidata Provides ...

- User support, software development, maintenance, and community advocacy
- Near real time data dissemination: Internet Data Distribution among 150 institutions for use in research and education (IDD, LDM)
- Tools for analysis and visualization (Gempak, IDV, McIDAS)
- Data access and collections infrastructure (netCDF, udunits, THREDDS, ...)







What is netCDF?

NetCDF Overview

- ***Model*** for scientific data: variables, dimensions, attributes, coordinates
- ***Libraries*** for data access: C, Fortran, C++, Java, Perl, Python, Ruby, Matlab, IDL, ...
- ***Format*** for portable binary data that supports direct access, metadata, appending new data, shared access

Users need not know anything about format

Some Applications for netCDF

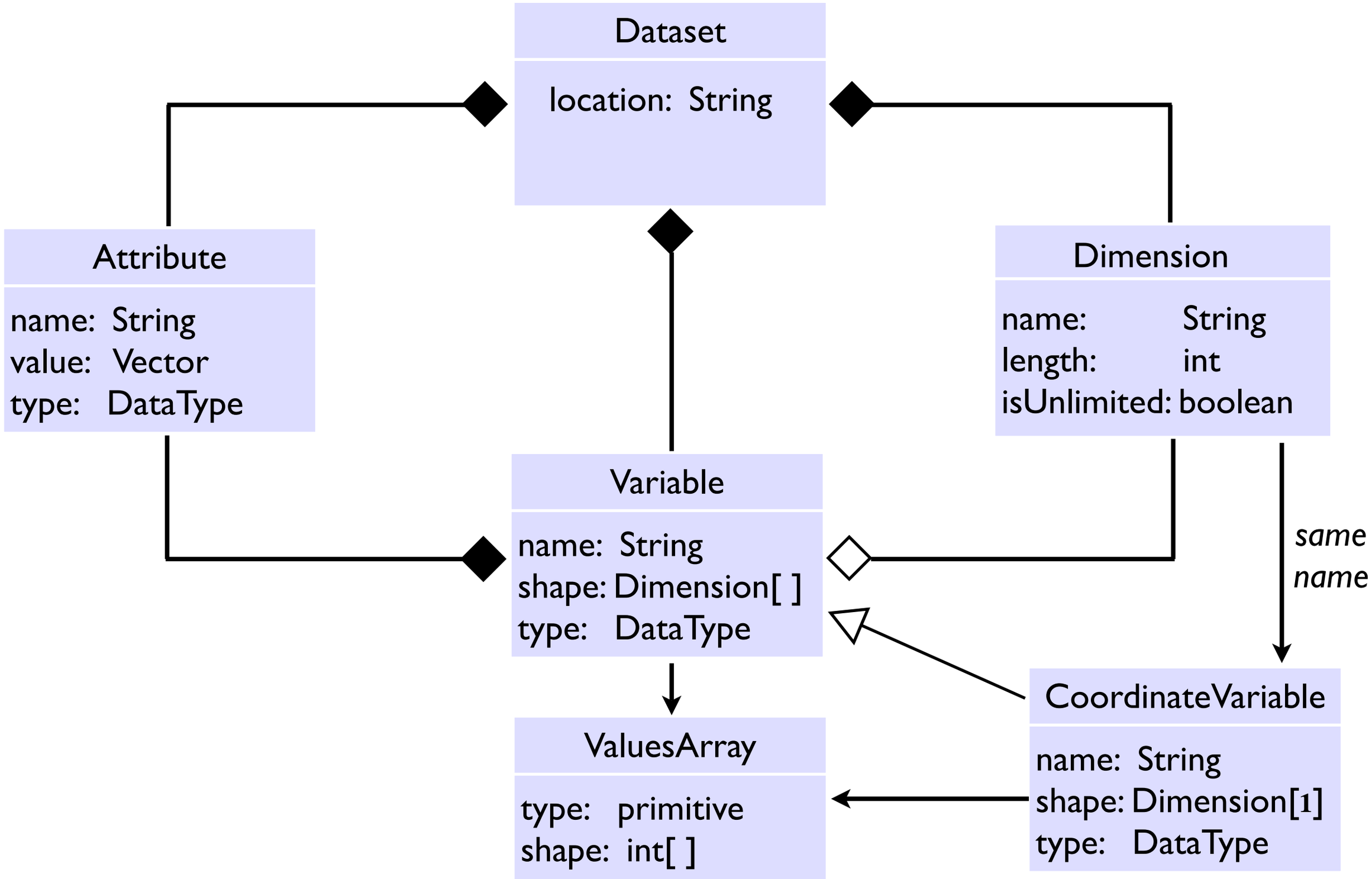
- Gridded output from models: forecast, climate, ocean, atmospheric chemistry, ...
- Observational data: surface, soundings, satellite, lightning, aircraft, radar, ...
- Various scientific data archives
- Chromatography, mass spectrometry, medical imaging, ...
- In general: platform-independent, language-independent interface to insulate data providers from data consumers

NetCDF Data Model

NetCDF Data Model

- Variables: name, shape (list of Dimensions), type, Attributes, values
- Dimensions: name, length
- Attributes: name, type, value(s)
- *Coordinate Variables:* values associated with Dimensions

NetCDF Data Model



netCDF Variables

- Name: e.g. air_temperature
- Type: byte, short, int, float, double, string
- Shape: e.g. (time, height, lon, lat)
- Attributes: e.g. units, long_name, ...
- Values: scalar or multidimensional array

Gridded Data

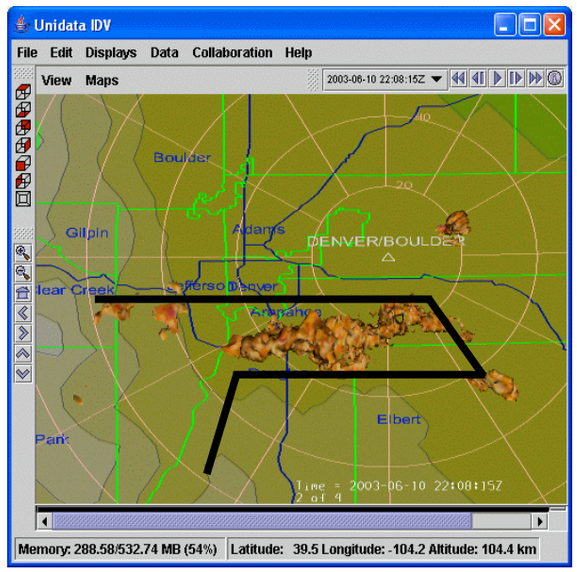
- Examples: model outputs, reanalysis archives
- Typically dependent variables located on multidimensional grid with coordinates, e.g. lat, lon, level, time

```
dimensions:
  lat=18; lon=36; pres=15; time=UNLIMITED;
variables:
  float xwind(time, pres, lat, lon);
    xwind:long_name = “zonal wind”;
    xwind:units = ”m/s”;
  float relHumidity(time, pres, lat, lon);
      relHumidity:units = “percent”;
  double lon(lon);      lon:units = “degrees_E”;
  double lat(lat);      lat:units = “degrees_N”;
  int pres(pres);      pres:units = “millibars”;
  double time(time);  time:units = “days since 1990-1-1 0:0:0”;
```

xwind(n,k,j,i) is associated with lon(i), lat(j), pres(k), time(n)

Trajectory Data

- Examples: observations from moving platforms, e.g. aircraft, soundings, ships
- Typically locations and observations parameterized by time




```
dimensions: time=UNLIMITED;
variables:
  double time(time); // coordinate variable
    time:units = “seconds since 2004-01-01 0:0:0”;
  double lat(time);
  double lon(time);
  double height(time);
  float temperature(time);
  float pressure(time);
  float relHumidity(time);
```

Station Data

- Examples: hourly surface observations from weather stations
- Typically observational variables for each of a set of scattered locations at common times

```
dimensions:
  station = 1000; time = UNLIMITED;
variables:
  double time(time);
    time:units = “hours since 2004-01-01”;
  double lat(stationID);  lat:units = “degrees_N”;
  double lon(stationID);  lon:units = “degrees_E”;
  int elev(stationID);    elev:units = “meters”;
  float temperature(time, stationID);
    temperature:units = ”celsius”;
  float pressure(time, stationID);
     pressure:units = “millibars”;
```

CDL notation and netCDF Utilities

- CDL (Common Data Language) is an editable text version of binary netCDF schema and data
- ncdump and ncgen utilities convert between binary and text versions of netCDF files
- ncgen can also generate programs for creating netCDF files

NetCDF Programming Interfaces

Availability of netCDF

- Freely available source, binaries:
  www.unidata.ucar.edu/packages/netcdf
- Unidata-supported interfaces:
  C, C++, Fortran-77, Fortran-90, Java, Perl
- Platforms:
  Unix, Windows, Mac OS X, ...
- License:
  X Windows/MIT style open source license
  Used in several commercial applications
- Status:
  version 3.5.1 released February 2004
  netCDF-4 under development

netCDF Language Interfaces

Conventions of each language supported by interfaces

- C: core implementation
- C++, Fortran-77, Fortran-90, Perl: call C library
- Java: independent pure-Java implementation
- Python, Ruby, IDL, Matlab, Ada: third-party layers over C library
- Data is language-neutral

Example C netCDF calls

```
#include <netcdf.h>
```

- Open a netCDF file for read-only access

```
err = nc_open("mydata.nc", NC_NOWRITE, &ncid);
```

- Get the variable ID for "windspeed"

```
err = nc_inq_varid(ncid, "windspeed", &varid);
```

- Get nth variable's name, type, shape, # of attributes

```
err = nc_inq_var(ncid, n, name, &typep, &ndimsp,
dimids, &nattsp);
```

- Read a slab of values

```
err = nc_get_vara_float(ncid, varid, start, count,
valarray);
```

Access to Array Sections

- Functions to access subarrays

  single values

  1-D vector of values along any dimension

  2-D cross sections using any 2 dimensions

  ...

  n-dimensional hyperslabs

  entire variable

- One shared dimension may grow

  appendable data

  record-oriented access

Java Interface

- Currently most advanced netCDF interface, used to test new ideas
- Client access to data servers:

  HTTPD

  OPeNDAP/DODS

- Advanced coordinate system support for general and georeferenced coordinates
- Virtual datasets to aggregate, subset, redefine, using NcML (XML representation for netCDF)

NetCDF Format

NetCDF Format

- Purpose: insulate users, applications, and data from machine architectures, format details, additions to data
- Provides data that is

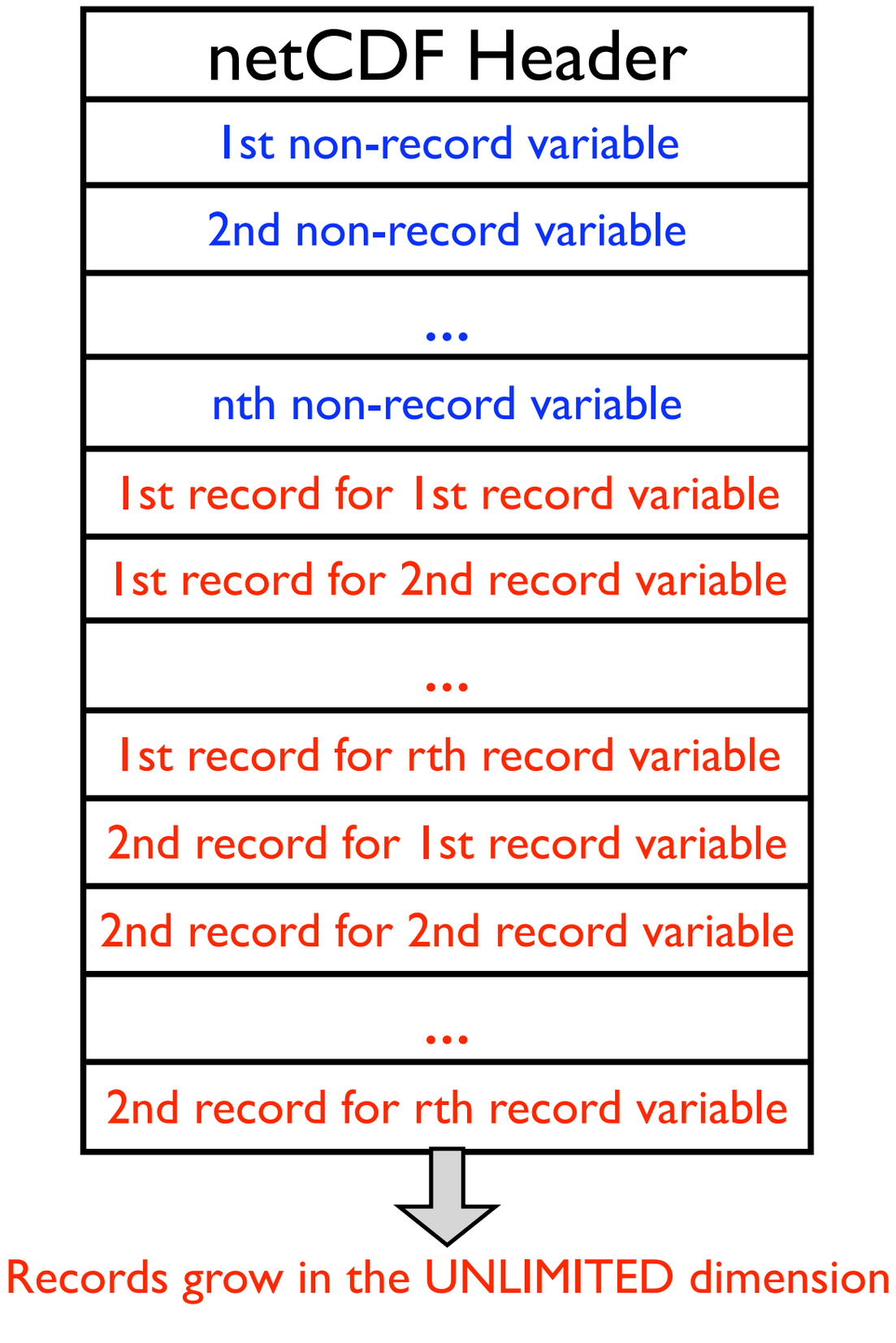
  self-describing

  portable

  directly accessible

  appendable
- Still support original XDR-based format (1988)
- 32-bit file offset limits file size
- New format with 64-bit offsets for huge files

NetCDF File Format



- Header

  Dimensions

  Attributes

  Variable metadata

- Fixed-size variables
- Record variables

  Use UNLIMITED dimension

- XDR-like represenations for all data and metadata

NetCDF Limitations

Some Current netCDF Limitations

- Not a database system: no keyed access, transactions, multiple concurrent writers
- Strings are just character arrays
- No nested data structures
- No “ragged arrays”, variable length records
- Only one growable dimension per dataset
- No built-in compression or packing
- Georeferencing requires agreement on conventions
- ...

GIS Connections

Obstacles to GIS-netCDF Connection

- netCDF data model (finite sampling of real functions) differs from relational, GIS datamodels
- Some netCDF data not spatial
- Much netCDF data multidimensional, > 2D
- Georeferencing for netCDF data uses multiple conventions
- Typical netCDF datasets may be very large
- Large netCDF datasets may be on a remote server

Unidata GIS-netCDF Efforts

- Prototype netCDF to geoTIFF converters
- Parser for georeferencing conventions implemented in Java
- Unidata GeoGrid from georeferenced netCDF variable supported in Java
- GridDataset: collection of GeoGrids in a netCDF dataset with methods for getting volumes and slices
- NcML for wrapping any new conventions around existing data

NetCDF CF Conventions

- **C**limate and **F**orecast metadata conventions

  Metadata for *use*, not *discovery*

  `http://www.cgd.ucar.edu/cms/eaton/cf-metadata/CF-1.0.html`

  Adopted by: PCMDI, PRISM, ESMF, NCAR, Hadley Centre, GFDL, various EU projects

  Developed by volunteers, active email list

- Includes georeferencing conventions for multidimensional coordinate systems

- Includes a "Standard Name Table" for quantities

  Google for "CF standard name table"

  76 of 416 entries (as of June 3, 2004) relate to hydrology

Summary

Summary

- NetCDF is a simple data model, different from Relational or GIS data models
- NetCDF format and interfaces provide a portable way to store and access multidimensional array-oriented data
- Conventions exist for georeferencing netCDF data
- May be a bridge to link GIS and scientific users, applications, and data