The netCDF-4 data model and format

Russ Rew, UCAR Unidata

NetCDF Workshop

25 October 2012

UCAR COMMUNITY PROGRAMS

unidata

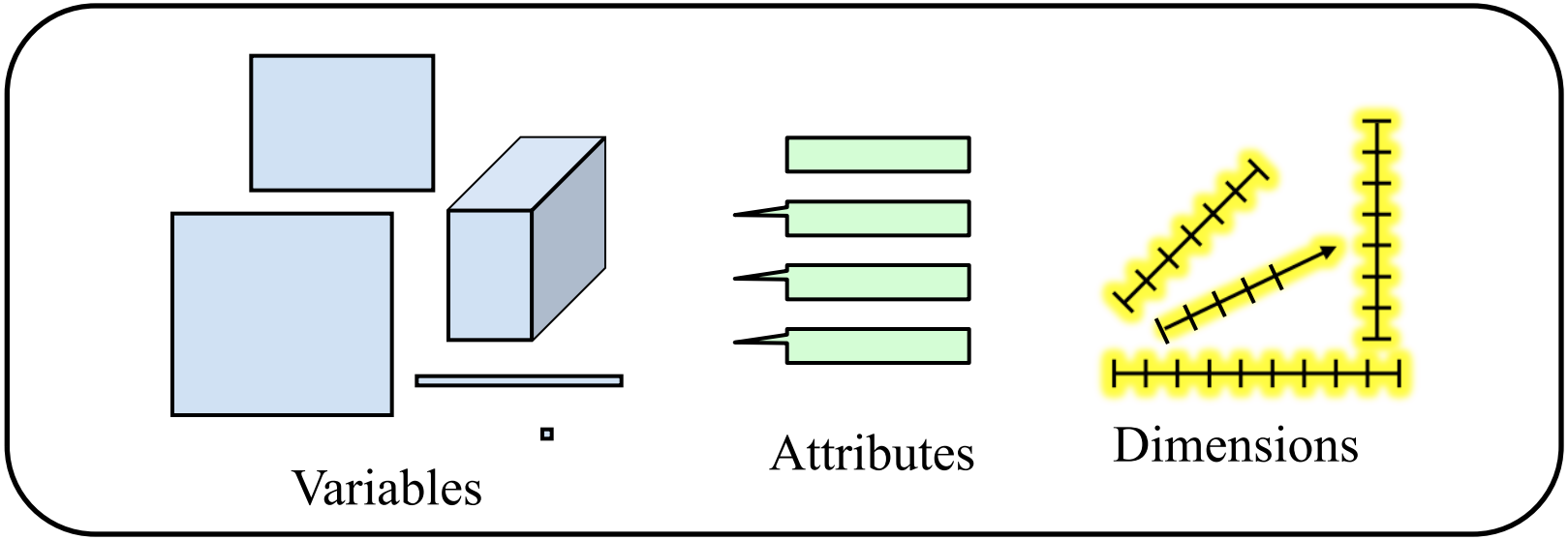
NetCDF data models, formats, APIs

- **Data models** for scientific data and metadata
  - *classic:* simplest model -- dimensions, variables, attributes
  - *enhanced:* more powerful model -- adds groups, types, nesting
- **File formats** for portable data
  - Array-oriented scientific data and metadata
  - Formats: classic, 64-bit offset, netCDF-4, netCDF-4 classic model
  - Formats make data self-describing, portable, direct access, appendable, extensible, sharable, archivable
- **Application programming interfaces (APIs)**
  - C, Java, Fortran, C++
  - Python, Ruby, Perl, MATLAB, IDL, … (3rd party APIs)

Together, the data models, file formats, and APIs support the creation, access, and sharing of scientific data

The netCDF "classic" data model

- A netCDF file has named ***variables***, ***attributes***, and ***dimensions***.
- Variables are for data, attributes are for metadata (data about data)
- Dimensions are for specifying shapes of variables
- Attributes may apply to a whole file or to a single variable
- Variables may share dimensions, indicating a common grid.
- One dimension may be of unlimited length.
- Each variable or attribute has 1 of 6 types: char, byte, short, int, float, double



The netCDF classic data model, in UML

NetCDF Data has

- **Variables** *(eg temperature, pressure)*
- **Attributes** *(eg units)*
- **Dimensions** *(eg lat, lon, level, time)*

Each variable has

- Name, shape, type, attributes
- N-dimensional array of values

Each attribute has

- Name, type, value(s)

Each dimension has

- Name, length

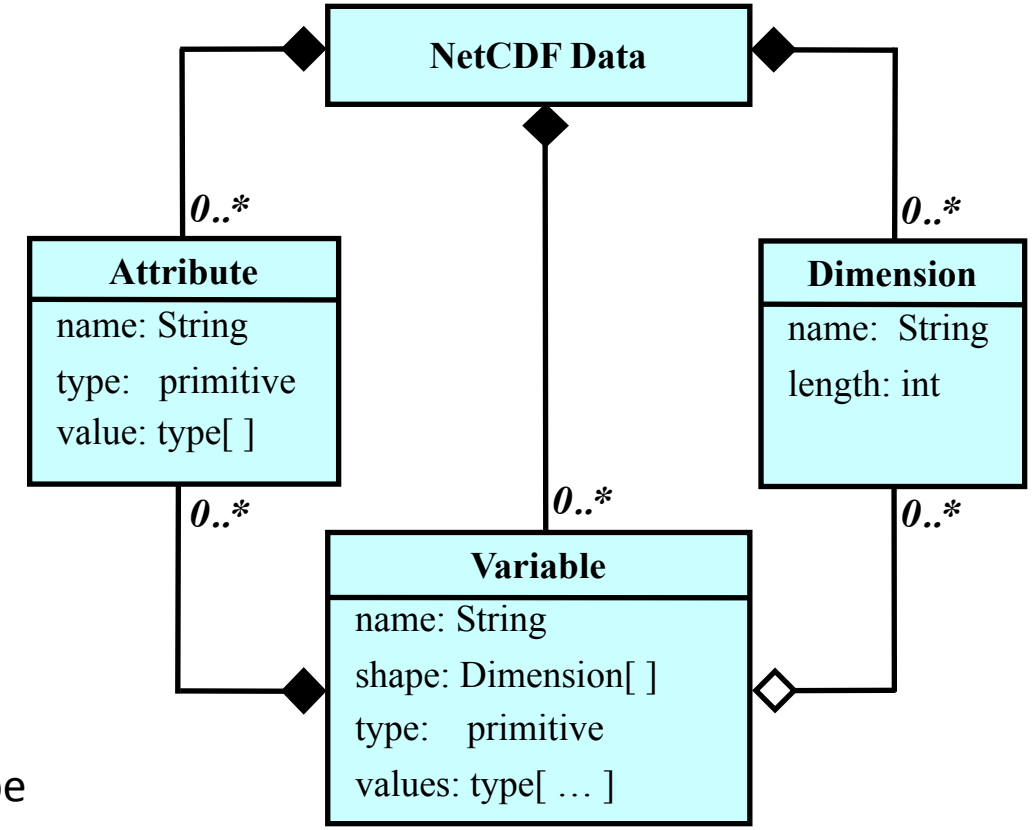
Variables *may share* dimensions

- Represents shared coordinates, grids

Variable and attribute values are of type

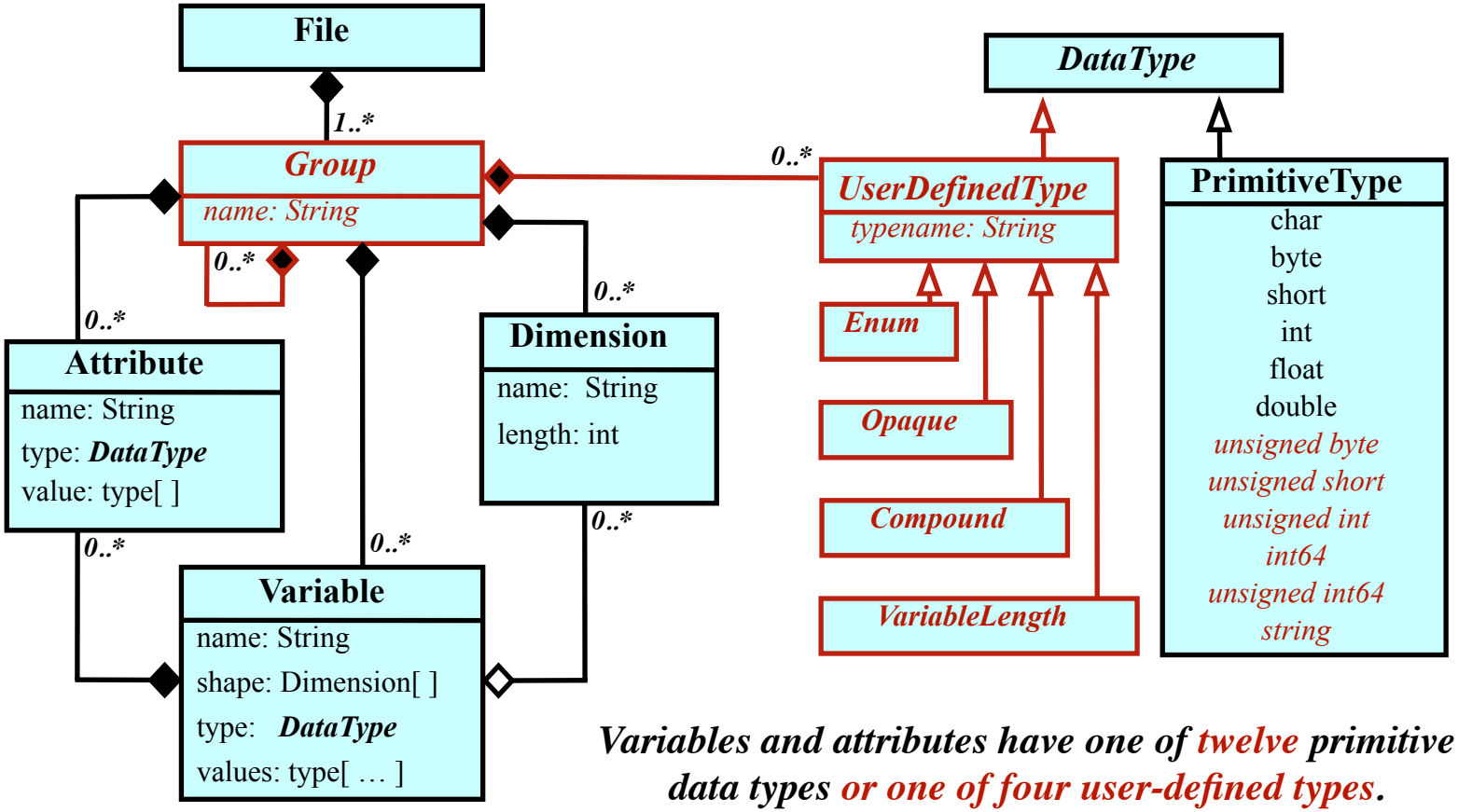
- Numeric: 8-bit **byte**, 16-bit **short**, 32-bit **int**, 32-bit **float**, 64-bit **double**
- Character: arrays of **char** for text

UML = Unified Modeling Language



The netCDF-4 enhanced data model

A file has a top-level unnamed group. Each group may contain one or more named subgroups, user-defined types, variables, dimensions, and attributes. Variables also have attributes. Variables may share dimensions, indicating a common grid. One or more dimensions may be of unlimited length.



Variables and attributes have one of twelve primitive data types or one of four user-defined types.

NetCDF *classic data model*

Strengths

- ✓ Data model simple to understand and explain
- ✓ Can be efficiently implemented
- ✓ Representation good for gridded multidimensional data
- ✓ Shared dimensions useful for coordinate systems
- ✓ Generic applications easy to develop

Limitations

- Small set of primitive types
- Data model is flat, limited to multidimensional arrays, (name, value) pairs
- Flat name space not ideal for organizing many data objects
- Lacks nested structures, variable-length types, enumerations

NetCDF *enhanced data model*

Strengths

- ✓ Simpler than HDF5, with similar representational power
- ✓ Adds shared dimensions to HDF5 data model
- ✓ Continues support for existing data, software, and conventions
- ✓ Adds real Strings and unsigned integer types
- ✓ Provides nested structures: hierarchical groups, recursive data types
- ✓ Independent features permit incremental adaptation, adoption

On the other hand ...

- More complex than classic data model
- More effort required to develop general tools and applications
- Adoption proceeding slowly
- Hence, no comprehensive experience-based best practices or conventions yet

NetCDF *classic formats*

Strengths

- ✓ Simple to understand and explain
- ✓ Supported by many applications
- ✓ Standard used in many archives, data projects
- ✓ Mature conventions and best practices have evolved

Limitations

- No support for efficient compression
- Only one growable dimension
- Schema changes can be costly
- Portable representation favors big-endian platforms

New NetCDF Binary Format

- Before the netCDF-4 project, there were two binary formats: classic and 64-bit offset
- NetCDF-4.0 introduced a new binary format: netCDF-4/HDF5
- It is an HDF5 file, with some additional metadata
- It is read by netCDF code just like any other netCDF file

NetCDF-4 *formats*

- Use HDF5 as a storage layer
- Provide performance advantages of HDF5
  - Compression
  - Chunking
  - Efficient schema changes
- Useful for very large or complex data
- Suitable for high-performance computing

NetCDF-4 classic-model format

netCDF-3

- Compatible with existing applications
- Simplest data model and API

netCDF-4
classic model

- Uses classic API for compatibility
- Uses netCDF-4/HDF5 storage for compression, chunking, performance
- To use, just recompile, relink

netCDF-4

- Not compatible with some existing applications
- Enhanced data model and API, more complex, powerful

Commitment to Compatibility

To ensure future access to existing data archives, Unidata is committed to compatibility of:

- **Data access**: new versions of netCDF software will provide read and write access to previously stored netCDF data.
- **Programming interfaces**: C and Fortran programs using documented netCDF interfaces from previous versions will work without change with new versions of netCDF software.
- **Future versions**: Unidata will continue to support both data access compatibility and program compatibility in future netCDF releases.

NetCDF standards endorsements

- *2009-02:* NASA Earth Science Data Systems (ESDS) Standards Process Group endorsed **netCDF classic and 64-bit offset formats** as appropriate for NASA Earth Science data.
- *2010-03:* Integrated Ocean Observing System (IOOS) Data Management and Communications (DMAC) Subsystem endorsed **netCDF with Climate and Forecast (CF) conventions** as a preferred data format.
- *2010-09:* Steering Committee of the US Federal Geographic Data Committee (FGDC) officially endorsed **netCDF** as a Common Encoding Standard.
- *2011-04:* Open Geospatial Consortium (OGC) endorsed "**OGC Network Common Data Form (NetCDF) Core Encoding Standard version 1.0**" as an OGC standard.
- *2011-11:* NASA ESDS Standards Process Group endorsed **NetCDF-4/HDF-5 File Format**, as a NASA Recommended Standard.
- *2012-10:* the Open Geospatial Consortium (OGC) approved the **NetCDF Enhanced Data Model Extension Encoding Standard**, making netCDF-4 an OGC