Unidata

Providing data services, tools, & cyberinfrastructure leadership
that advance Earth system science, enhance educational opportunities, & broaden participation

Data-Intensive Science and Scientific Data Infrastructure

Russ Rew, UCAR Unidata

ICTP Advanced School on High Performance and Grid Computing

13 April 2011

UCAR COMMUNITY PROGRAMS

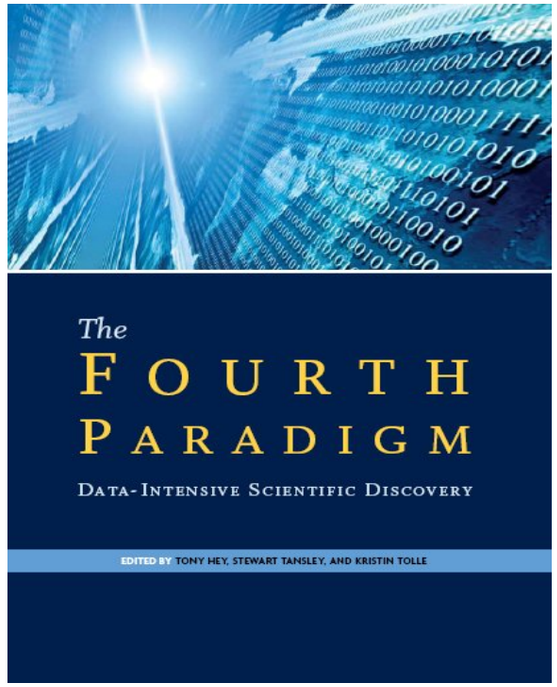
unidata

Overview

- Data-intensive science
- Publishing scientific data

Data-intensive science

- A “fourth paradigm” after experiment, theory, and computation
- Involves collecting, exploring, visualizing, combining, subsetting, analyzing, and using huge data collections
- Challenges include
  - Deluge of observational data, “exaflood” of simulation model outputs
  - Need for collaboration among groups, disciplines, communities
  - Finding insights and discoveries in a “Sea of Data”
- Data-intensive science requires
  - New tools, techniques, and infrastructure
  - Standards for interoperability
  - Institutional support for data stewardship, curation



Roles in Data-intensive Science

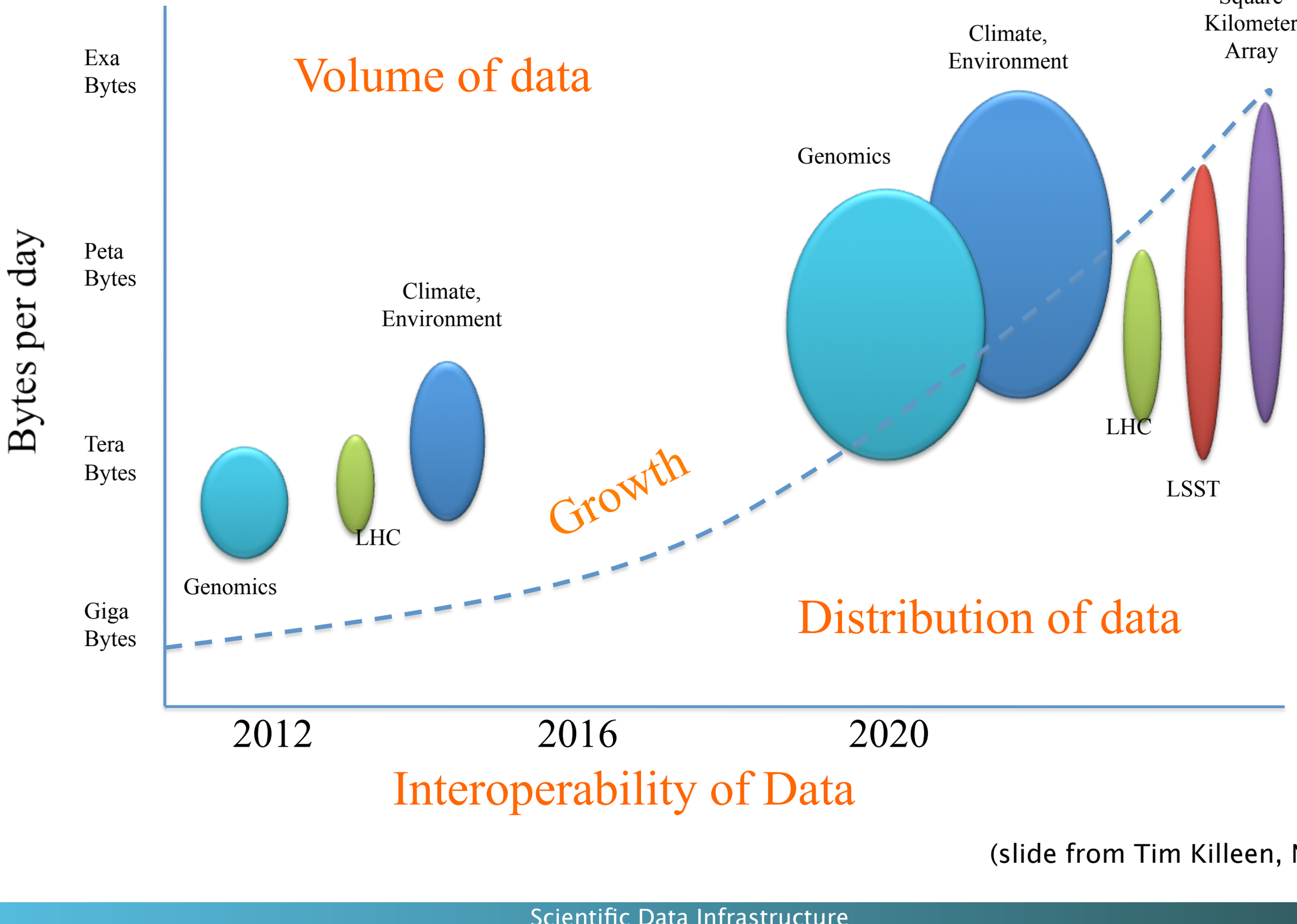
- **Scientists/researchers:** acquire, generate, analyze, check, organize, format, document, share, publish research data
- **Data users:** access, understand, integrate, visualize, analyze, subset, and combine data
- **Data scientists:** develop infrastructure, standards, conventions, frameworks, data models, Web-based technologies
- **Software developers:** develop tools, formats, interfaces, libraries, services
- **Data curators:** preserve data content and integrity of science data and metadata in archives
- **Research funding agencies, professional societies, governments:** encourage free and open access to research data, advocate elimination of most access restrictions

Growth in data from sensor systems

According to Science article [2011-02-11, Baraniuk]:

- Majority of data generated each year now comes from sensor systems
- Amount generated passed storage capacity in 2007
  - in 2010 the world generated 1250 billion gigabytes of data
  - generated data growing at 58% per year
  - storage capacity growing at 40% per year
- We generate more scientific sensor data than we can process, communicate, or store (e.g. LHC)

Data challenges: gigabytes to exabytes



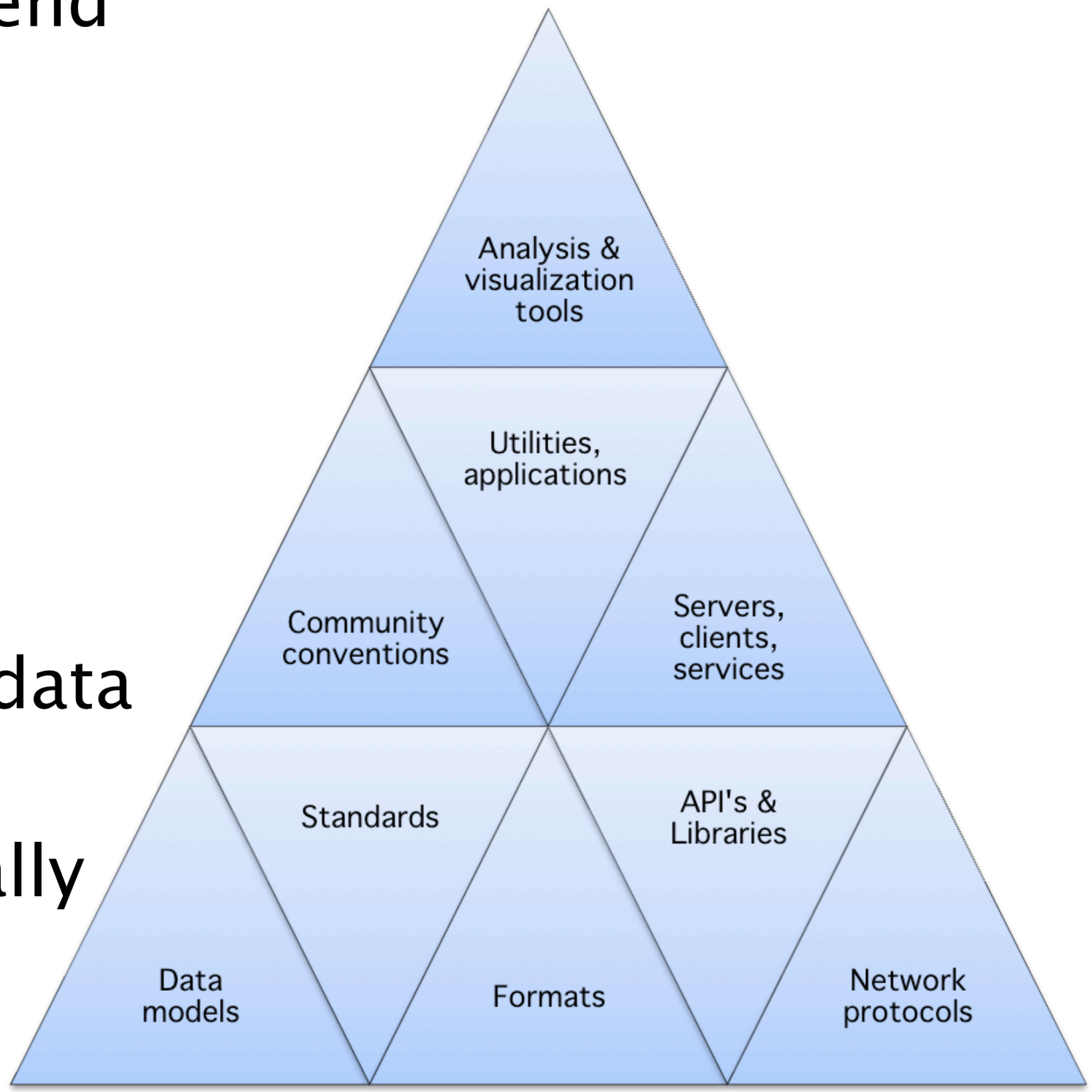
(slide from Tim Killeen, NSF)

Scalability and “Big Data”

- What’s the big deal about big data?
  - aren’t more and faster computers and larger disks the solution?
- I/O access and bandwidth can’t keeping up with computing speed
- Too big to transfer, must move processing to data
- Sensors and models can generate huge datasets easily
- Making huge datasets accessible and useful is difficult
- Other problems: discovery, curation, provenance, organization, integrity, ...

Infrastructure for sharing scientific data

- Applications depend on lower layers
- Sharing requires agreements
  - formats
  - protocols
  - conventions
- Data needs metadata
- Is all this infrastructure really necessary?



Why not use binary I/O?

```
real  :: a(len), b(len)

write (nunit, rec=14) a
read  (nunit, rec=14) b
```

Simple, but …

- Not portable
- Lacks metadata for use, discovery
- Not usable by general analysis and visualization tools
- Inaccessible from other programming languages, for example reading Fortran binary data from Java or C/C++

Why not use formatted I/O?

```
real   :: a(len), b(len)

write (nunit, '(10f10.3') a
read  (nunit, '(10f10.3') b
```

Simple, but …

- Inefficient for large datasets (time and space)
- Sequential, not direct ("random") access
- Lacks metadata for use, discovery
- Not usable by general analysis and visualization tools

Why not use relational databases?

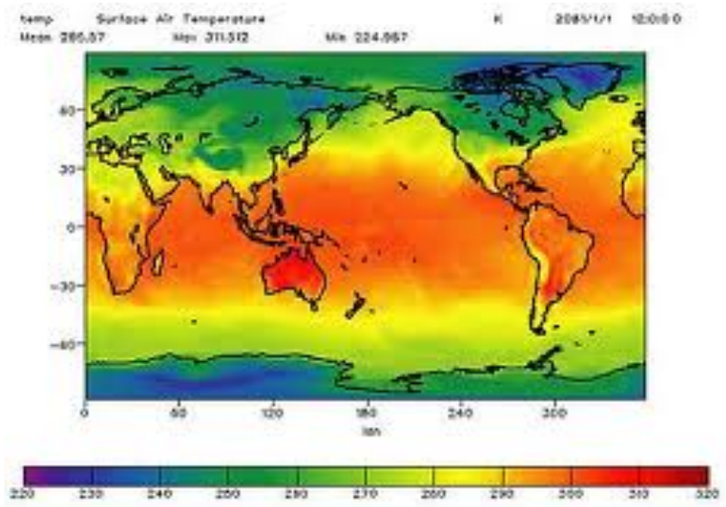
- Data model may not be appropriate
  - no direct support for multidimensional arrays
  - tables and tuples are wrong abstractions for model output, coordinate systems
- Tools: lacking for analysis and visualization
- Portability: difficult to share, publish, preserve, cite, database contents
- Performance
  - database row orientation slows access by columns
  - transactions unnecessary for most scientific use
- But sometimes databases are ideal, e.g. virtual observatories

Other alternatives for scientific data

- XML, YAML, JSON, CSV, other text notations
  - Require parsing
  - Sequential, not direct access
  - Inefficient for huge datasets
  - Conversions between text and binary can lose precision
- Discipline-specific: FITS (astronomy), GRIB (meteorology), XMDF (hydrology, meshes), *foo*ML, …
- General-purpose, for scientific data:
  - CDF: historically one of the first, used in NASA projects
  - netCDF: widely used, simplest data model
  - HDF5: most powerful, most complex data model
  - SciDB: coming soon, multidimensional array-based database

Publishing scientific data: advice to data providers

Don’t just provide pictures, provide data



http://www.some-archive.org/id3456/my-results/

- So your research can be reused by others in future research and analyses
- So your plots can be duplicated and integrated with other data
- So users can choose their favorite display and analysis software for your data
- So corrections to data are practical
- So your results have a longer shelf life

Don’t just make data available interactively

- Programs need access to data, not just humans
- Accessing lots of data by mouse clicks or display touching is difficult and slow
- Provide bulk access for large datasets
- Anticipate need for programs to access data remotely

Support efficient access to small subsets of data

- Database queries should return only requested data
- Don’t provide only huge files with all the data, that discourages reuse
- Remote access is faster for small subsets
- Interactive visualization integrating data from multiple sources is practical with small subsets
- Some problems require a little data from many places, not a lot of data from one place

Provide easy access to metadata

- More metadata is usually better
- Make it easy to add more metadata later
- Keep metadata with the data, if practical
- Support discovery metadata, so your data can be found
- Support use metadata, so your data can be understood
  - coordinate systems
  - units

Strive for interoperability

- Data should be portable now
- Data should be portable to the future
- Don't optimize packaging or format for specific data or application
- Valuable scientific data is written once, read many times

Support standards

- If available, use them
- If not, help develop them
- If possible, help maintain them

Summary: What Data Producers Should Provide

- Data (not just visualizations)
- Useful metadata (not just data)
- Remote access (not just physical copies or local access)
- Convenient granularities of access (not too large or too small)
- Program access (not just for interactive users)
- Standard formats (not machine-, application-, or language-specific; but what about discipline-specific?)
- Organization for users and readers (not just what's most convenient for provider)

But scientists want to do science ...

- ... not data management
- Valuable scientific data must be acquired, organized, accessed, visualized, distributed, published, and archived
- How can scientists do all this and still have time to do science?
  - graduate students?
  - data managers, curators, stewards, ...?
  - database systems?
  - general purpose scientific data infrastructure?
- Standards supported by open source software may help:

But scientists want to do science ...

- ... not data management
- Valuable scientific data must be acquired, organized, accessed, visualized, distributed, published, and archived
- How can scientists do all this and still have time to do science?
  - graduate students?
  - data managers, curators, stewards, ...?
  - database systems?
  - general purpose scientific data infrastructure?
- Standards supported by open source software may help: