The Abdus Salam
International Centre
for Theoretical Physics

2494-10

Workshop on High Performance Computing (HPC) Architecture and Applications in the ICTP

14 - 25 October 2013

Data Management

S. Cozzini
CNR-IOM DEMOCRITOS, Trieste

Stefano Cozzini

CNR-IOM DEMOCRITOS
Trieste, ITALY

Istituto Officina dei Materiali

Data Management

AIM of this talk

- Frame the problem and the discussion around DATA:
  - What are big data ?
  - Which kind of challenges ahead of us ?

Disclaimer:

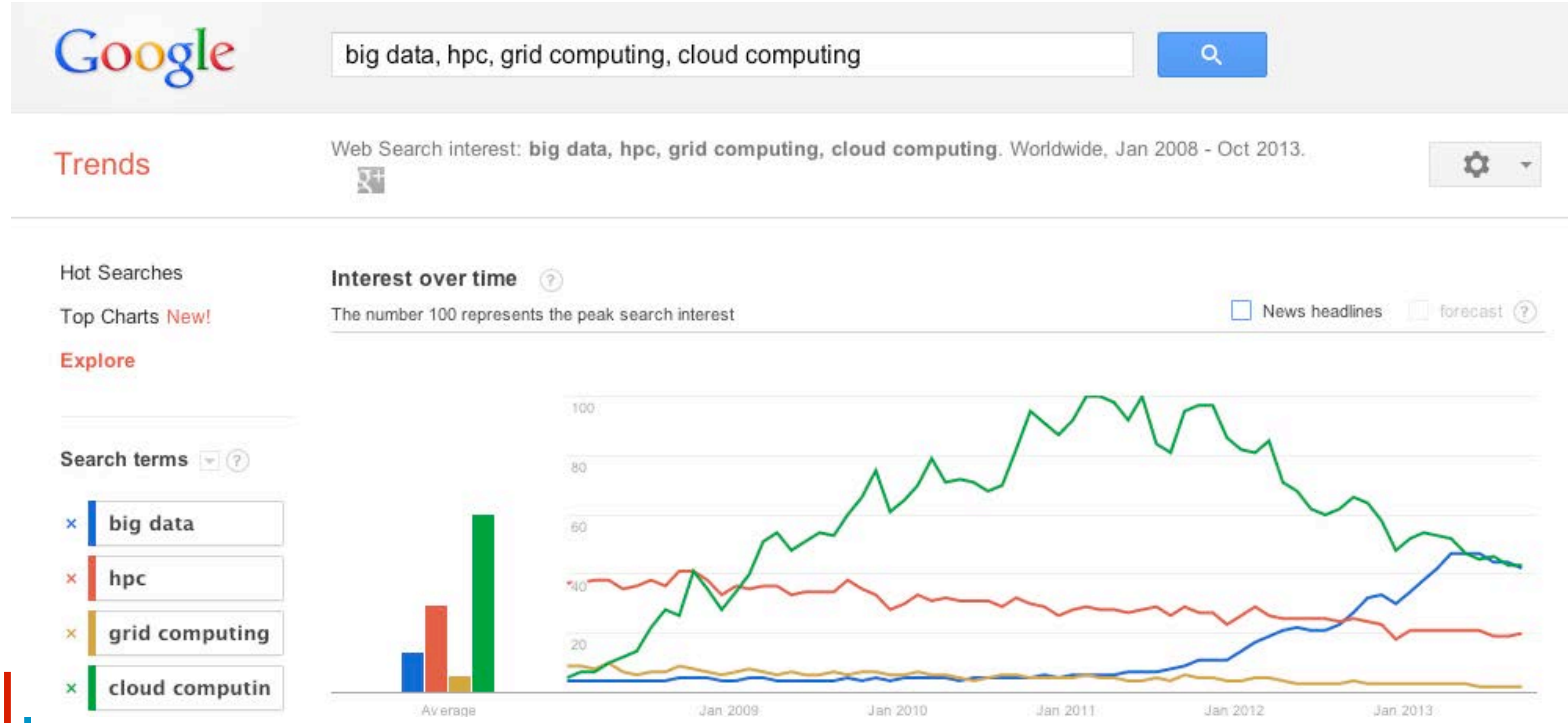
Slides and numbers are taken around: there are a lot of data discussing big data ☺

Agenda

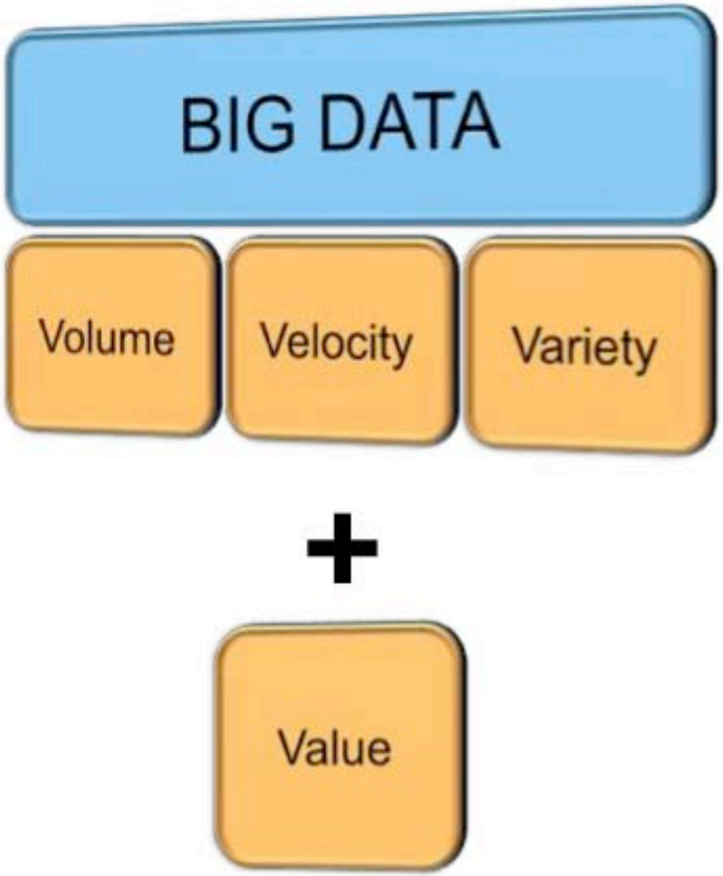
- Introducing big-data problem
- Data intensive science
- Data infrastructures and tools
- Advices to data providers
- Conclusions

INTRODUCTION TO BIG DATA PROBLEM

Big Data: a buzzword..



Big Data: general definition



- Lots of data
- Time critical
- Multiple types (e.g., numbers, text, video)
- Worth something to someone

The 3 V's of big data..

- Velocity

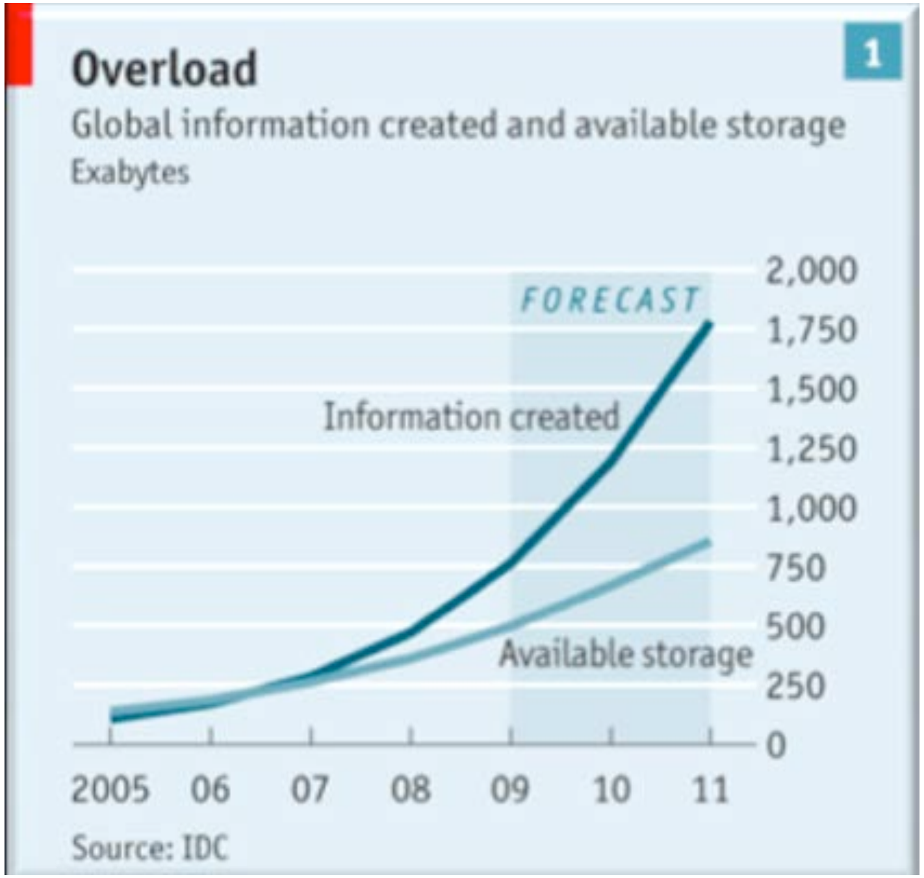
  Data are produced at speed higher than the speed you are able to move/analyze and understand them..

- Variety
  - Data range from simulation to remote sensing information, from instruments to market analysis etc..
  - datasets come in a variety of data formats and span a variety of metadata standards
- Volume
  - The amount of data will increase of factor 61 in the next 10 years
  - The amount of data is estimated to exceed the size of available data infrastructure to store them by 60%.

[from The 2011 IDC Digital Universe Study]

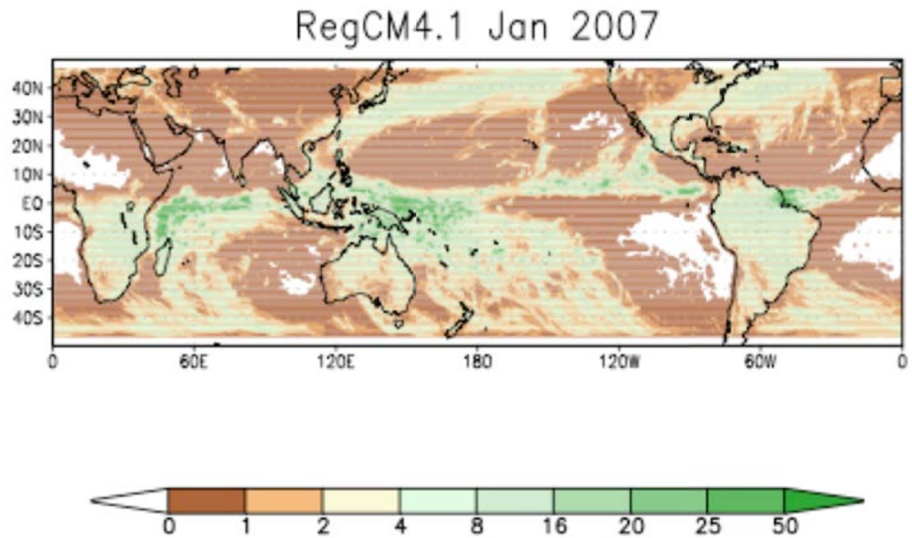
Data Deluge

- 1.25 trillion gigabytes generated in 2010
  - # digital bits > # stars in the universe
  - growing by a factor of 10 every 5 years
  - fueled by rapid growth of digital sensing/multimedia
- Total data generated > total storage



Data Deluge in science

- An example: climate change with RegCM4
  - Output generated is:
    - X 32bit variables for X*Z*Y point of the domain every T hours of simulation for N years of simulation
  - Larger example currently studied at ICTP:
    - Equator Belt :
      - Number of variables: 50
      - domain: 832x250x18 points
      - Frequency: 3 hours
      - Length : 150 years



50 * 832 * 250 * 18 * 6 * 365 * 150 ^ 4 = 2.459808 $\times 10^{14}$!!! ~250 TB !!!

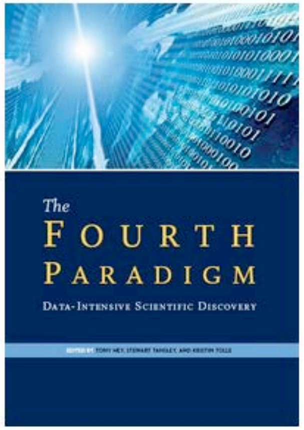
DATA INTENSIVE SCIENCE

Data-intensive science

- A "fourth paradigm" after experiment, theory, and computation..

The Fourth Paradigm: Data-Intensive Scientific Discovery

Presenting the first broad look at the rapidly emerging field of data-intensive science



Increasingly, scientific breakthroughs will be powered by advanced computing capabilities that help researchers manipulate and explore massive datasets.

The speed at which any given scientific discipline advances will depend on how well its researchers collaborate with one another, and with technologists, in areas of eScience such as databases, workflow management, visualization, and cloud computing technologies.

In *The Fourth Paradigm: Data-Intensive Scientific Discovery*, the collection of essays expands on the vision of pioneering computer scientist Jim Gray for a new, fourth paradigm of discovery based on data-intensive science and offers insights into how it can be fully realized.

Download

- Full text, low resolution (6 MB)
- Full text, high resolution (93 MB)
- By chapter and essay

Purchase from Amazon.com

- Paperback
- Kindle version

In the news

- Sailing on an Ocean of 0s and 1s *(Science* Magazine)
- A Deluge of Data Shapes a New Era in Computing *(New York Times)*
- A Guide to the Day of Big Data *(Nature)*

Critical praise for *The Fourth Paradigm*

It involves collecting, exploring, visualizing, combining, subsetting, analyzing, and using huge data collections

Challenges & Requirements

Challenges:

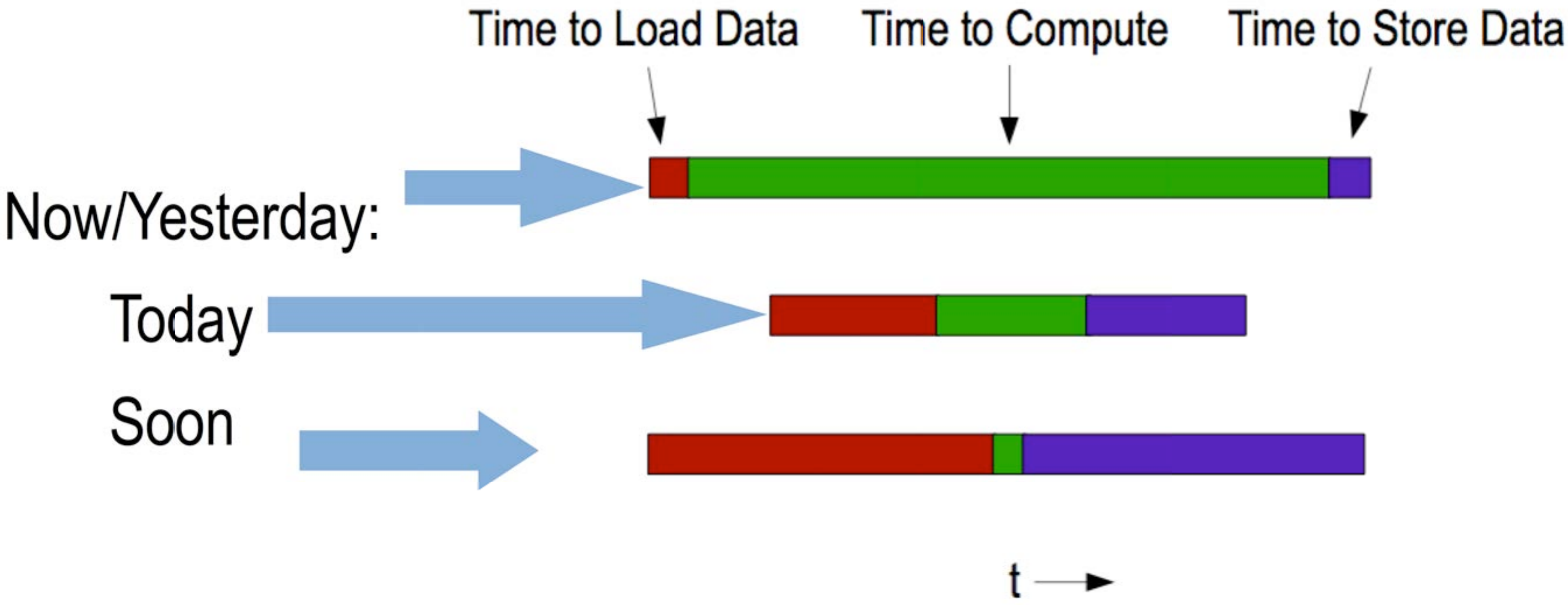
- Deluge of observational data, “exaflood” of simulation model outputs
- Need for collaboration among groups, disciplines, communities
- Finding insights and discoveries in a “Sea of Data”

Requirements:

- New tools, techniques, and infrastructure
- Standards for interoperability
- Institutional support for data stewardship, curation

Two important concepts for HPC

- You can only compute as fast as you can move it
- HPC workflow will be soon be bounded by the speed of the storage system..

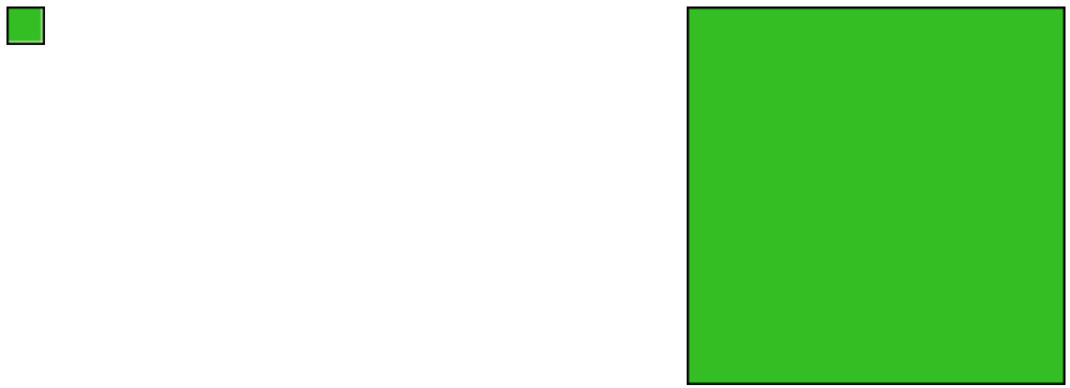


IOPS vs FLOPS

- HPC is today *compute-centric*
- scientific computing needs data accessibility rather than computing speed

computing 1 calculation ≈ 1 picojoule

moving 1 calculation ≈ 100 picojoule



Source: IDC Direction 2013

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

WHAT IS YOUR ROLE ?

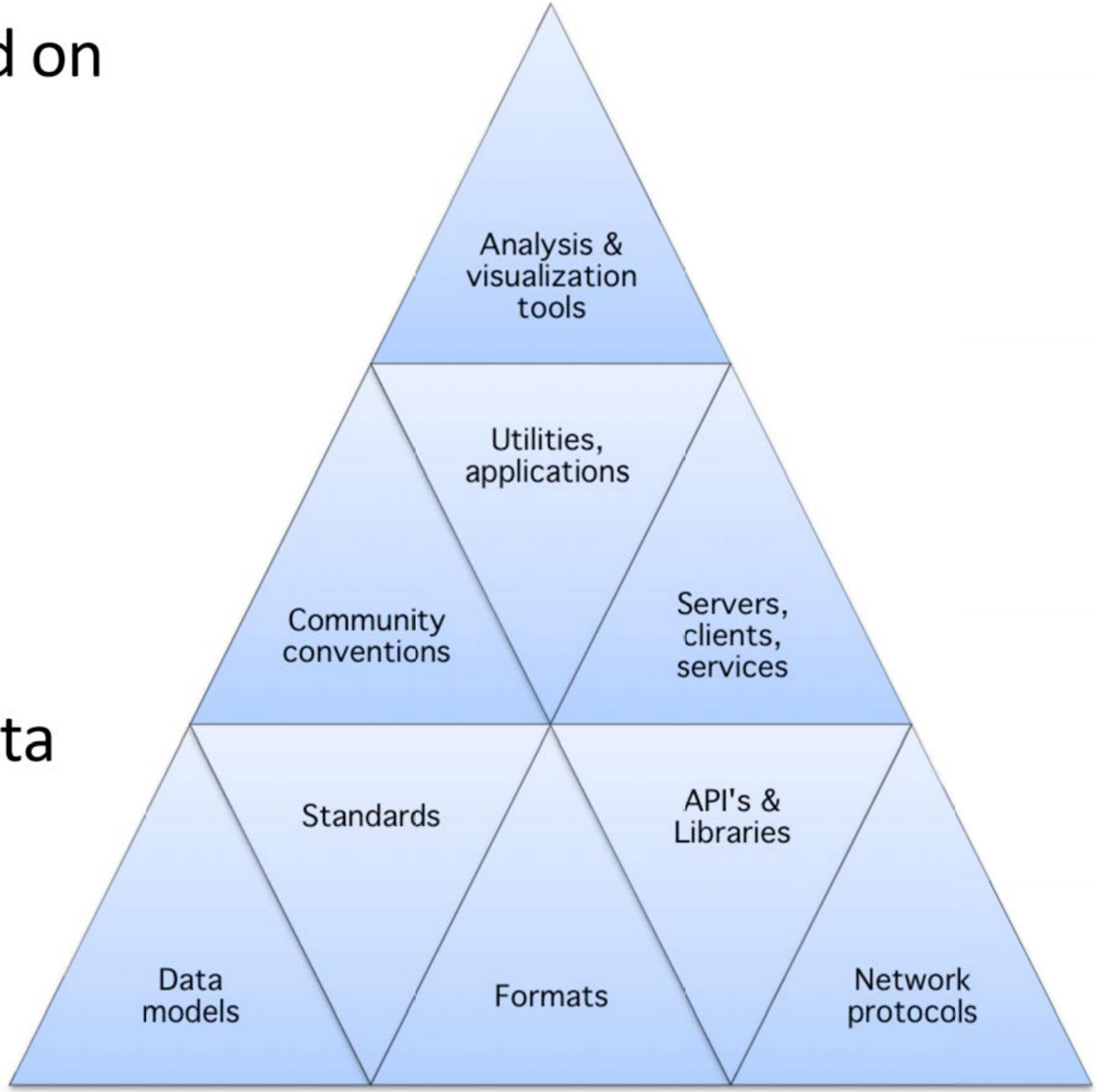
Scalability and “Big Data”

- What’s the big deal about big data?
  - aren’t more and faster computers and larger disks the solution?
- The missing parts:
  - Network
    - What about data transfer ?
  - Software
    - What about the software ?
  - People
    - Who is maintaing the infrastructure ?

DATA INFRASTRUCTURES

Infrastructure for sharing scientific Data

- Applications depend on lower layers
- Sharing requires agreements
  - formats
  - protocols
  - conventions
- Data needs metadata



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), fooML, …
- 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: v 13.9, multidimensional array-based database

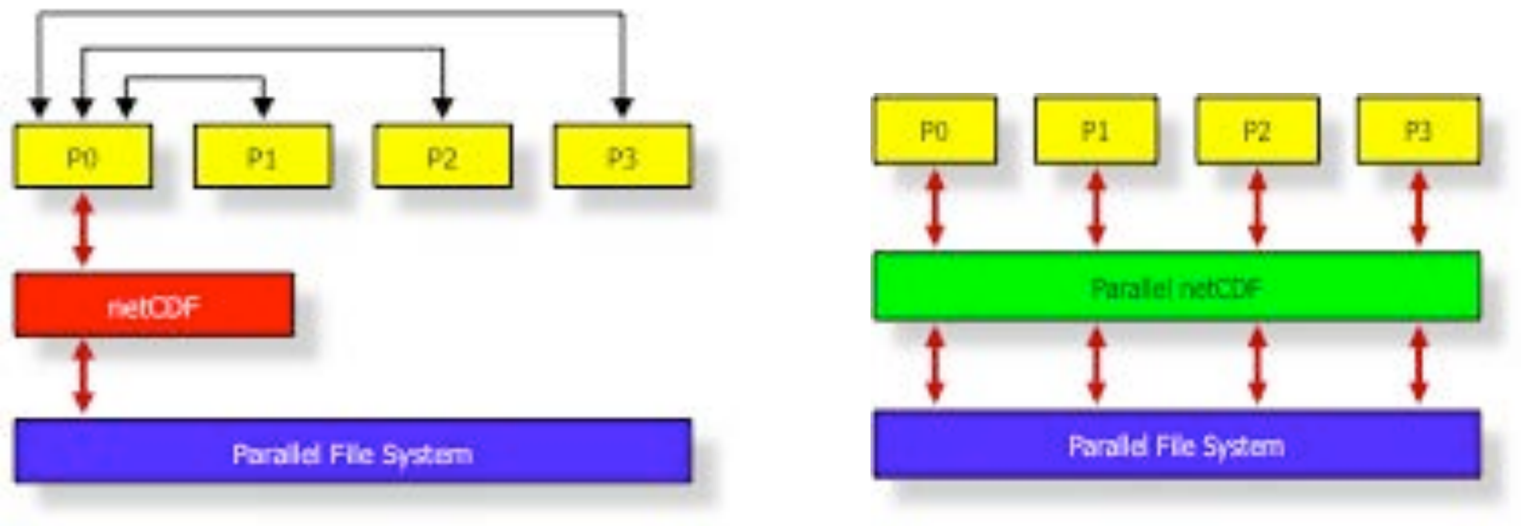
High level libraries

- Provide an appropriate abstraction:
  - Multidimensional datasets
  - Typed variables
  - Attributes
- Self-describing, structured file format
- Map to middleware interface
  - > Encourage collective I/O
- Provide optimizations that middleware cannot (has to be generic)

Data management libraries : netcdf

- NetCDF: Network Common Data Format from Unidata:
- Data Model:
  - Collection of variables in single file
  - Typed, multidimensional array variables
  - Attributes on file and variables

From netcdf to parallel netcdf



HDF5

- Hierarchical Data Format from NCSA
- Data Model:
  - Hierarchical data organization in single file
  - Typed, multidimensional array storage
  - Attributes on dataset, data
- Features:
  - C, C++, and Fortran interfaces
  - Portable data format
  - Optional compression
  - Data reordering (chunking)
  - Noncontiguous I/O (memory and file) with hyperslabs
- Note
  - NetCDF-4 allows some interoperability with HDF5.

Data infrastructure crucial aspects

Large datasets:

Deal with large datasets and large data rates from experiments

Reliability :

Increase the level of QoS and SLA, e.g. increasing reliability by replicating data sources and increasing accessibility by copying source to several places

Accounting

Allow monitoring and checking of resource usage Integration Provide the same set of services that are understandable (compatible) between domains

Interoperation

Interoperation through common standard schemes

Access

Broadband data access Allow transparent and secure remote access to data

Source: e-irg blue paper on data management 2012

http://www.e-irg.eu/images/stories/dissemination/e-irg-blue_paper_on_data_management_v_final.pdf

Data infrastructure crucial aspects

Data preservation

Allow long-term availability of data

High quality

Quality of data to enable advanced and cross-disciplinary access and enrichment operations

Economic justification

As the scientific community is operating on increasingly larger datasets and want to preserve the information concerned, the infrastructure provided should have a clear roadmap of technology exchange and backwards compatibility.

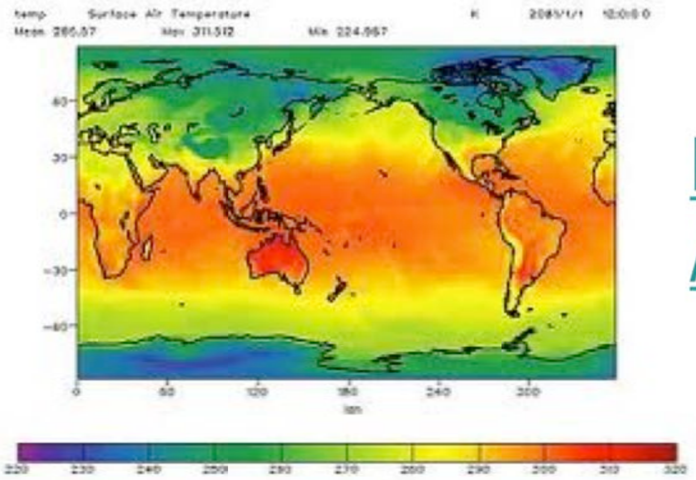
Access control

Provide the infrastructure to allow for fine-grained access control

Publishing scientific data: advice to data providers

Slides/ Advices by Russ Rew from Unidata

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

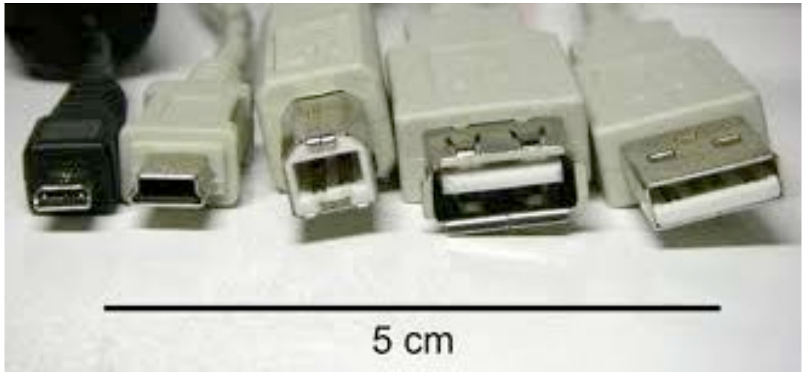
Provide easy access to metadata

- 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

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

CONCLUSIONS

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:

Data are not just for science..

- High-end commercial analytics pushing up into academia.
- The journey from science data to industry& commerce can be relatively short…
- ..and plenty of ethical legal and societal implications

IEEE SECURITY & PRIVACY

Big Data, Big Brother, Big Money

July-Aug. 2013 (vol. 11 no. 4)

pp. 85-89

Michael Lesk, Rutgers University

DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/MSP.2013.81

ABSTRACT

Government snooping, recently publicized, is now using the same data sources that corporations use to watch us. The same records used by federal agencies to search for terrorists are used, with fewer controls, by corporations searching for customers.

ADDITIONAL INFORMATION

Index Terms:
Surveillance,Government agencies,Surveillance,Data mining,Marketing and sales,Analytical models,marketing analytics,surveillance,data mining

Citation:
Michael Lesk, "Big Data, Big Brother, Big Money," *IEEE Security & Privacy*, vol. 11,
4, pp. 85-89, July-Aug. 2013, doi:10.1109/MSP.2013.81

August 06, 2013

Big Data Meets Big Brother: Inside the Utah Data Center

Thomas Parent

The widely publicized leaks from the National Security Agency (NSA) have provided a fascinating glimpse into the covert data collection activities of the U.S. Government. Among other things, these leaks have revealed the enormous volume of phone records currently being collected, stored and analyzed.

This begs the question: "Where does all of this Big Data go?" Starting this fall, it's likely that a good chunk of it will be going to Utah!

While the NSA operates many data facilities, none compare to the new $1.5 Billion data center scheduled to open this September in Bluffdale, Utah. Vanee Vines, an NSA spokeswoman, has provided the following information: