Scientific data formats

Sebastian Döbel

Arbeitsbereich Wissenschaftliches Rechnen
Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

2016-05-11

Structure

MP3 simular to scientific data formats?

- abstraction of scientific object (MP3: wave to bits)
- fit to your needs (MP3: reduce bandwidth of frequences)

Requirements

- file size
  - what is important to store?
  - meta data
  - precision
  - compression
  - fault tolerance, redundancy
- archivability, backwards compatibility
- portability
- interchangeability
- searchability
- human readable (text based)

Graphs

- traffic system (HVV map)
- state transitions (e.g. electrons)
- FIGURE 1 with mathematical Graph

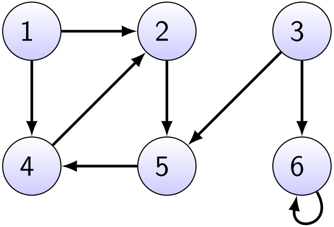

Figure: directed graph

Adjacency lists

- each vertex has list of outgoing neighbours
- sparse graphs: $|E| \ll |V|^2$
- usually neighbour list unsorted
- size $\Theta(V + E)$
- reference: (Cormen u. a., 2007, 531)

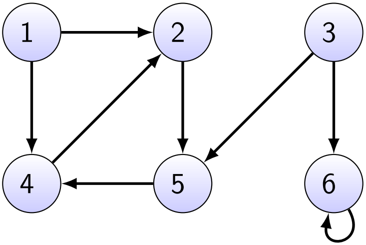

Figure: directed graph

1	$\{2, 4\}$
2	$\{5\}$
3	$\{6, 5\}$
4	$\{2\}$
5	$\{4\}$
6	$\{6\}$

Figure: adjacency list

Adjacency matrix

- $|E| \sim |V|^2$
- faster check if two vertices linked
- $A^{|V| \times |V|}$ with $a_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{else} \end{cases}$
- size (const): $\Theta(V^2)$
- reference: (Cormen u. a., 2007, 531)

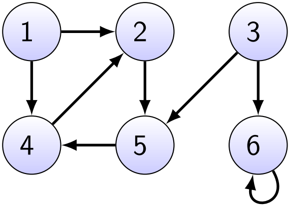

Figure: directed graph

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \end{array}$$

Figure: adjacency matrix

Sparse matrices

- often result of discretization of partial differential equation
- efficient algorithms in calculating
- save only non-null elements with their position
- special case: band matrices

$$\begin{pmatrix}
a_{11} & a_{12} & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\
a_{21} & a_{22} & a_{23} & \ddots & & & & \vdots \\
0 & a_{32} & a_{33} & a_{34} & \ddots & & & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\
\vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\
\vdots & & & \ddots & a_{(n-2)(n-3)} & a_{(n-2)(n-2)} & a_{(n-2)(n-1)} & 0 \\
\vdots & & & & \ddots & a_{(n-1)(n-2)} & a_{(n-1)(n-1)} & a_{(n-1)n} \\
0 & \cdots & \cdots & \cdots & \cdots & 0 & a_{n(n-1)} & a_{nn}
\end{pmatrix}$$

Figure: tridiagonal matrix

Computational life science - FASTA

- ASCII text based format for primary structure of nucleic acid and proteins
- used with FASTA and BLAST algorithm (compare sequences)
- no standard format suffix but usually .fa, .mpfa, .fna, .fsa or .fasta

Format structure

- headline
  - begins with ">"
  - contains unique name and description of the sequence
- comment symbol ";"
- multiple sequences per file possible

```
>gi|2196610|emb|CAB09441.1| cytochrome b, partial (mitochondrion) [Acomys
    ↪ spinosissimus]
MTNIRKTHPLLKIINHAFIDLPAPSNITSWWNFGSLLGICLIIQIITGLFLAMHYTSDTSTAFSSVTHIC
RDVNYGWLIRYLHANGASMFFICLFMHVGRGIYYGSYTYMETSNIGIILLFAVMATAFMGYVLPWGQMSF
WGATVITNLLSAIPYIGTNLVEWIWGGFSVDKATLTRFFAFHFILPFIIAALAMVHLLFLHETGSNNPTG
INSDSDKIPFHPYYTMKDLLGAFILLLTLLALVLFSPDLLGDPDNYTPANPLNTPPHIKPQWYFLFAYAI
LRSIPNKLGGVLALVLSILVLAILPLIHTSKQRSLMFRPISQTLFWILVANLLILTWIGGQPVEHPFIII
GQLASISYFTIILILIPISGLIENKMMKWN
```

Listing 1: cytochrome b of Southern African spiny mouse (Acomys spinosissimus) (german: Zwergstachelmaus); taken from (miceFasta)

Translation of Codes

- codes represents IUB/IUPAC standard

Code	Meaning
A	**A**denine
G	**G**uanine
T	**T**hymine
K	K = GT (**K**etone)
-	gap of indeterminate length

Table: extract of nucleic acid codes

reference: Tao

Code	Meaning
A	alanine
G	glycine
T	threonine
K	lysine
X	any
*	translation stop
-	gap of indeterminate length

Table: extract of amino acid codes, 25 acids + 3 special codes

Compression

- very big files
- general purpose tools like gzip fall short
- many algorithms proposed to compress genomic data $\Longrightarrow$ MFCompress
- in comparison to gzip 50% additional compression but also computation time
- highly redundant datasets 8-fold of gzip possible

reference: Pinho und Pratas (2013)

Applications

- GenomeTools & libgenometools (ZBH)
- Vmatch (ZBH)
- SeqAn
- FASTA/FASTQ parser in C[1]
- huge databases
  - National Center for Biotechnology Information (NCBI)
  - European Bioinformatics Institute (EBI)

[1] http://lh3lh3.users.sourceforge.net/parsefastq.shtml

FITS

- Flexible Image Transport System
- standardized 1981, last release 2008
- designed for long-term archival storage $\implies$ backwards compatibility - "Once FITS, always FITS"
- $\implies$ used in Vatican Apostolic Library

Format structure

- HDU = header + data
- header: image card (80 character fixed-length ASCII strings) with key-value-pairs (size, origin, coordinates) and maybe comments
- data in multidimensional tables (variable length columns supported)
- also for non-image data e.g. spectra, data cubes
- contains several extensions e.g. x-ray and infra-red
- references supported
- internal storage sometimes in heaps

Applications

- CFITSIO (Fortran, C)
- Aladin
- Detect the Dark Ages (LEDA) (24TB/day)[2]
- image viewer
  - GIMP
  - Photoshop
  - XnView
  - irfanView

references: Schwarzburg (2005); Price u. a. (2015); Kayser (2012); fitsNasa

[2](see Price u. a., 2015)

GRIB

- GRIdded Binary (version 1), General Regularly-distributed Information in Binary form (version 2)
- standardized by World Meteorological Organization's (WMO) Commission for Basic Systems (CBS)
- store temperature, rainfall, wave height, ...

Format

- grid discretization of room
- binary 2d-matrix
- collection of self-contained, independent records (messages)
- records contains sections
  - start and stop sequence
  - meta data e.g. origin, time, ...
  - dimension of grid, projection type
  - data
    - version 1: scaled to integer
    - version 2: compressed
- optional inventory: "table of contents" with user meta data and positions

Tools

- *wgrib* like typical UNIX filter
- *degrib* creates indices for faster access
- *dkrz_readgrib*[3]
- GUI: GRIBview[4]

references: Scherer (2009)

[3] http://mms.dkrz.de/pdf/klimadaten/static/Pingo/post/post.dkrzgrib.html
[4] http://www.theyr.com

Applications

- scientific applications
  - PINGO post-processing package (DKRZ)
- common applications
  - zyGrib
  - web site: PassageWeather.com
  - Android app: Marine Weather | SailGrib Free

Climate & geo sciences - NetCDF

- Network Common Data Format
- set of software libraries and self-describing, machine-independent data formats
- open standard maintained by University Corporation for Atmospheric Research (UCAR)
- originally based on NASA's Common Data Format but not compatible anymore
- version 4 allows use of HDF5

- self-contained, platform independent, binary
- dimensions
  - contain name and size
  - only one size unlimited (dataset dimension)
  - measurands e.g. time, length, ...
- varibales
  - array of values with same type
  - contain name, datatype, shape
  - coordinate variable: one dimensional variable with same name as dimension
- attributes
  - meta data
  - used in variables and global
- conventions
  - standards for specific use case
  - compare files from different sources
  - e.g. Climate and Forecast (CF), Cooperative Ocean / Atmosphere Research Data Service (COARDS)

reference: netCdfArcGis

Applications

- libraries for C, C++, Fortran, Java
- third party: Perl, Python, MATLAB/Octave
- ArcGIS

HDF5

- Hierarchical Data Format
- standardized by National Centre for Supercomputing Applications (NCSA) (1988), now developed by HDF Group
- binary file
- backward compatibility
- huge platform support
  - official: C, C++, Fortran, Java
  - third-party: Go, Python, R, MATLAB (Scilab, Octave), Mathematica, ERLANG, Perl, LabVIEW, ...
- contains datasets and groups $\implies$ "datasystem"

Objects

- dataset: singe value or table of any dimension
- group: has name, attributes and contains groups and datasets
- attribute: any information for user e.g. simulation parameter
- meta data: information about content e.g. size of datatype (api)

HDF5Viewer

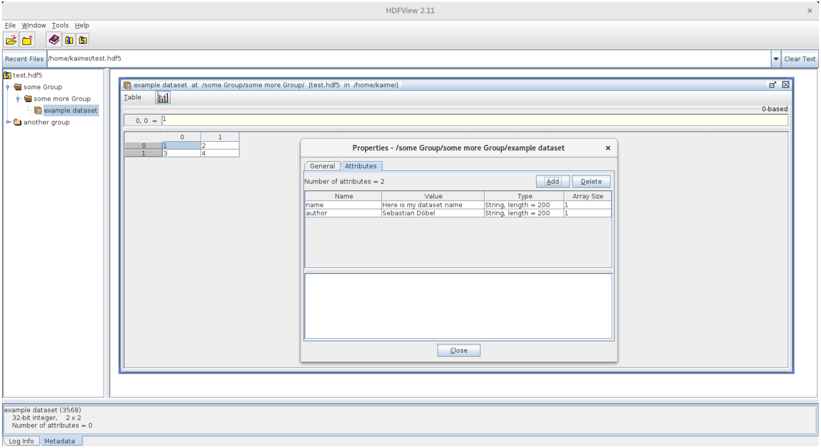

Figure: Screenshot from HDF5Viewer

Modules

- separated in modules
- modules not independent
- e.g.
  - H5: library functions
  - H5A: annotation interface
  - H5F: file interface
  - H5G: group interface
  - H5Z: compression interface
  - ...

Datatypes I

- own implementation of datatypes $\Longrightarrow$ portability
- pre-defined and derived types
- native: determine how standard types (atomic, composited) are stored
- standard types: atomic and composited types

Datatypes II

- atomic types
  - integer, float
  - string
  - reference
  - bitfield
  - date
  - time
  - opaque
- composited types
  - array
  - variable length
  - enumeration
  - compound datatypes

Datatypes III

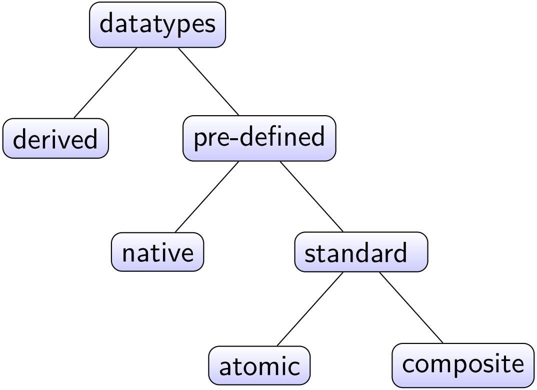

Figure: hierarchy of datatypes

Datatypes IV

- pre-defined types create big bases
- possible to derive $\Longrightarrow$ change:
  - precision
  - size
  - character set
  - binary representation and interpretation
- create own application specific datatype

Optimization I

- overide system methods e.g. cache usage
- parallel IO
  - possible to read and write parallel in same datasets by multiple processes
  - file copy per process
  - uses MPI in background $\Longrightarrow$ parallel file system
  - user has to use *H5Pcreate* instead of *H5Fcreate*
  - usage only possible with C and Fortran
- compression
  - different algorithms implemented (performance vs. size)
  - possible to implement own algorithms

Optimization II

- module *table*: any number of entries with same structure
  - unique number of types
  - unique size of types)
- module *packet table*:
  - different type sizes
  - possible to save strings with different lengths

Tools

- post processing of already written data
- converting HD4 to HD5 (*h4toh5*, (*h5toh4*))
- import & export e.g.
  - export dataset of an image to GIF (*h52gif*)
  - import GIF file into HDF5 file (*gif2h5*)
  - create printable version (*h5dump*)
- split & merge files (*h5repart*)
- copy and compression (*h5repack*)
- compare two files (*h5diff*, *ph5diff*)
- performance tests (*h5perf*, *h5perf_serial*)
- GUI: *HDFView* (Java-binding)

references: Kirchhart (2009)

Applications

- FLASH (Hamburg Observatory)
- RAMS
- GNU Octave / MATLAB, Mathematica
- ParaView
- for more see (hdf5Software)

reference: hdf5Software; HdfGroup

Summary

- sciences already have specialized data formats
- own format
  - what to store (data, meta data, . . .)
  - text vs. binary based
- HDF5 provides fast, powerful storage interface for general problems
- HDF5 is support by many platforms
- migrating data format not easy

References I

[fitsNasa] http://fits.gsfc.nasa.gov/. – Last visited 28.04.2016

[miceFasta] *cytochrome b of Southern African spiny mouse (Acomys spinosissimus) in FASTA format*. http://www.ncbi.nlm.nih.gov/protein/2196610?report=fasta. – Last visited 28.04.2016

[HdfGroup] *Introduction to HDF5 by HDF Group*. https://www.hdfgroup.org/HDF5/doc/H5.intro.html. – Last visited 09.05.2016

[netCdfArcGis] *Overview on netCDF-Files*. http://desktop.arcgis.com/de/arcmap/10.3/manage-data/netcdf/a-quick-tour-of-netcdf-data.htm. – Last visited 09.05.2016

References II

[hdf5Software] *Software using HDF5*. https://www.hdfgroup.org/tools5desc.html. – Last visited 09.05.2016

[Cormen u. a. 2007] Cormen, Thomas H. ; Leiserson, Charles E. ; Rivest, Ronald ; Stein, Clifford ; Molitor, Paul: *Algorithmen - Eine Einführung*. Bd. 2. Oldenburg : Oldenbourg Wissenschaftsverlag GmbH, 2007

[Kayser 2012] Kayser, Rainer: Astro-Dateiformat für Vatikanische Bibliothek. In: *astronews.com* (2012), January

[Kirchhart 2009] Kirchhart, Lukas: *Hierarchical Data Format vs. Textbasierte Datenformate*. Aachen, RWTH Aachen, seminar paper, December 2009

References III

[Pinho und Pratas 2013] PINHO, Armando J. ; PRATAS, Diogo: MFCompress: a compression tool for FASTA and multi-FASTA data. In: *BIOINFORMATICS APPLICATIONS NOTE* (2013)

[Price u. a. 2015] PRICE, D. C. ; BARSDELL, B. R. ; GREENHILL, L. J.: HDFITS: porting the FITS data model to HDF5. In: *Astronomy and Computing* 12 (2015), October, S. 212–220

[Scherer 2009] SCHERER, Roman: *Konzeption und Implementierung einer Community-Plattform für Surfer*. Berlin, Humboldt-Universität zu Berlin, diploma thesis, 2009

[Schwarzburg 2005] SCHWARZBURG, Stefan: *Eine Software zur Echtzeitanalyse von experimentellen Daten im Flexible Image Transport System (FITS)*, Eberhard Karls Universität Tübingen, diploma thesis, June 2005

References IV

[Tao] TAO, Tao: http://www.ncbi.nlm.nih.gov/staff/tao/tools/tool_lettercode.html. – Last visited 09.05.2016

[Voß 2008] VOSS, Peter: *Ein automatisches Softwaresystem zur statistischen Auswertung von Radiookkultationsdaten und zusätzlicher meteorologischer Modelldaten*. Neubrandenburg, Hochschule Neubrandenburg, Bachelor thesis, September 2008