

An Overview of the HDF5 Technology Suite and its Applications

Seminar Supercomputer: Forschung und Innovation

Liviana Franke

Fachbereich Informatik
Fakultät für Mathematik, Informatik und Naturwissenschaften
Universität Hamburg

2022-06-21

Gliederung

- 1 Überblick
- 2 Datenformat & Modell
- 3 Bibliothek
- 4 Zusammenfassung
- 5 Literatur

Motivation

Beispiel: Bioinformatik, BioHDF

- DNA Sequenzen (GACT / GACU)
- Gespeichert als textbasiertes Format (Strings)
- Steigende Rohdatenmengen von MB zu GB pro Tag
- Datenformate wie TABIX:
 - Nutzt Kompression und Indexierung
 - Nutzen der Daten benötigt diese vollständig im RAM
 - Erweitern der Daten benötigt oftmals Kopieren der Daten
 - Enthält keine zusätzlichen, beschreibenden Daten
- BioHDF als Kollaboration von Geospiza und der HDF Group

Die HDF5 Technology Suite

- **Hierarchical Data Format 5** der HDF Group
- Beta-Veröffentlichung in 1998
- Kombination aus Datenformat, -modell und Bibliothek
- Fundamental anders als HDF4 (ehemals HDF Release 4.x)
- HDF5 ist selbstbeschreibend und hierarchisch:

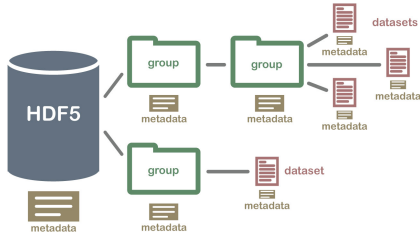


Abbildung: Hierarchische Struktur von HDF5, [Was], Abb. 2

Ziele von HDF5

- Unbegrenzte Diversität an Datentypen
- Flexibles und effizientes I/O
- Unterstützung für *Parallel I/O*
- Verarbeitung von sehr großen und komplexen Daten
- Portabilität und Erweiterbarkeit um Funktionalität
- Schnittstelle soll unabhängig von Fortschritten in der Speichertechnologie sein

Eine Alternative: netCDF

- **Network Common Data Form** von Unidata (UCAR)
- Portables, selbstbeschreibendes Datenformat
- Nicht auf dynamische Restrukturierung/Vergrößerung ausgelegt
- Direkt-Zugriff auf Daten unabhängig von Anzahl an Variablen
- Schnittstelle unabhängig von Speicher-Repräsentation
- Zwei Modelle (klassisch, erweitert)

Alternative: netCDF

Beispiel einer klassischen netCDF Datei

▼ Variables	
▶ longitude	
▶ latitude	
▶ time	
▶ u10	
▶ v10	
▼ t2m	
Type	short
Dimensions	(time, latitude, longitude)
▼ Attributes	
scale_factor	0.00160418343620771
add_offset	261.823019929766
_FillValue	-32767s
missing_value	-32767s
units	K
long_name	2 metre temperature
▶ Data values	
▼ Dimensions	
longitude	1440
latitude	721
time	1

Abbildung: Struktur von netCDF, [Rus], Abb. 1

Modelle von netCDF

- Klassisches Modell:
 - Variablen sind begrenzt auf $2^{32} - 4 \text{ B} < 4 \text{ GiB}$
 - Anzahl an Variablen ist System- und Versionsabhängig
 - Maximal eine unlimitierte Array-Dimension pro Datei
 - Begrenzte Auswahl an Datentypen
 - Parallel I/O via PnetCDF
- Erweitertes Modell (netCDF4):
 - Verwendet speziell strukturierte HDF5 Dateien
 - Ermöglicht unlimitiert große Variablen
 - Fügt Option für benutzerdefinierte Typen hinzu
 - Parallel I/O via HDF5 Bibliothek
 - Hierarchische Struktur via HDF5

Unterscheidung

Daten-/Dateiformat:

- Konkret, (datei)systemspezifisch
- Standardisiert Repräsentation im Speicher
- Beispiel: PNG vs. JPG (lossless vs. lossy)

Datenmodell:

- Abstrakt, konzeptuell
- Standardisiert Relationen zwischen Daten
- Beispiel: Relationale Datenbank Management Systeme (RDBMS)

Multigraph

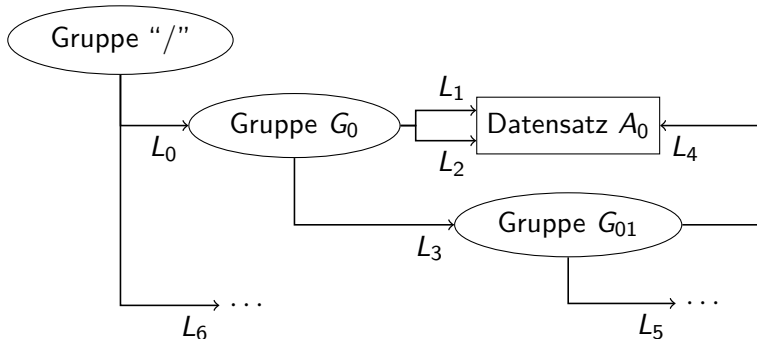


Abbildung: Selbsterstellte Multigraph-Darstellung des HDF5 Datenmodells.

Datensätze

- Multidimensionale Array-Variablen
- Bestehen aus *HDF5 Dataspace* und *HDF5 Datatype*
- *HDF5 Dataspace* bestimmt Rang, Größe und Maximalgröße
- *HDF5 Datatype* bestimmt Datentyp der Array-Elemente
- Speicheranordnung unabhängig von anderen HDF5 Datensätzen

Speicheranordnung

- *Contiguous*:
 - Elemente als einfache Sequenz
 - Ermöglicht nahezu konstante Zugriffszeit
- *Chunked*:
 - Elemente verteilt auf Chunks fixer Größe
 - Keine Begrenzung der Anzahl der Elemente pro Dimension
 - Filter-Unterstützung (Kompression, Prüfsummen, ...)
- *Compact*:
 - Elemente als Teil der Metadaten des Datensatzes
 - Nur für kleine Datensätze (≤ 64 KB)

Contiguous

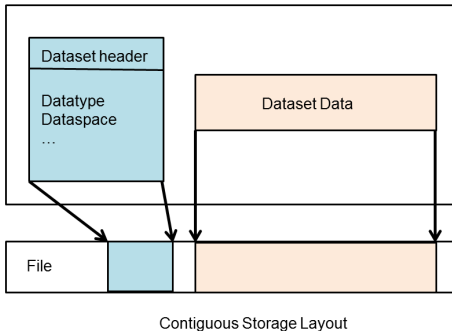


Abbildung: *Contiguous* Speicheranordnung, [Groa], Abb. 2.

Chunked

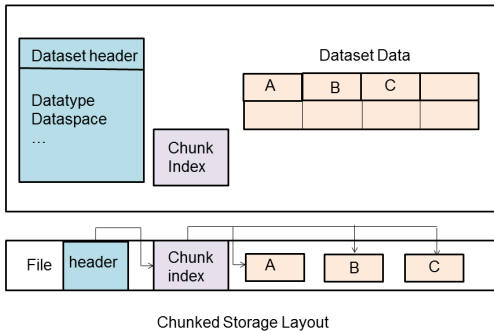


Abbildung: *Chunked* Speicheranordnung, [Groa], Abb. 4.

Gruppen

- Vergleichbar mit Verzeichnissen
- Gruppieren (mindestens null) *HDF5 Information Items*
- Bedeutung einer Gruppe ist anwendungsbasiert:
 - G ist Kontext von A , B und C
 - A , B , C gehören zu G
- *HDF5 Root Group* “/” existiert immer
- Hierarchie durch eindeutige Links

Links

- Benannte 1 : 1-Links von Gruppe zu *HDF5 Information Item*
- Zyklen sind explizit erlaubt
- Persistenz von Daten durch Erhöhung deren Referenzzählers

Link Typ	Verknüpfung via	Ref. Zähler Änderung
Hard/Persistent	Adresse/ID	Ja, erhöht
Soft/Symbolisch	Interner Pfad	Nein
Extern	Ex-/Interner Pfad	Nein
Benutzerdefiniert	Benutzerdefiniert	Benutzerdefiniert

Tabelle: HDF5 Link Typen, basierend auf [Fol+11], Tabelle 1.

Datentypen

- Integer, Floating-Point, String, Bitfield, Opaque, Compound, Reference, Enum, Sequence, Array
- Datentypen haben weitere Parameter z. B. für Präzision, Vorzeichen, etc.
- Compound, Sequence & Array unterstützen beliebige Kombinationen sowie Nestung der obigen Datentypen
- Compound & Opaque ermöglichen deutlich komplexere Datentypen

Attribute

- Dienen zur Annotation von *HDF5 Information Items*
- Müssen einzigartig im Sichtbereich des Items sein
- Haben ebenfalls *HDF5 Dataspace & -type*
- Gedacht für kleine Daten (z. B. Datum eines logischen Zeitpunkts)

Schnittstellen

- HDF5 library API (high-level, Vgl. ODBC)
- Core Library geschrieben in C
- Unterschiedlich starke Language-Bindings für diverse Sprachen (FORTRAN, C++, Java, Python, etc.)
- Erlaubt das Erstellen, Bearbeiten und Auslesen

Standardanwendungen

Anwendungen zur manuellen Arbeit mit HDF5 Dateien

- Konvertierung (`h5import`, `h5dump`, `h52gif`, ...)
- Lesen / Schreiben (`h5diff`, `h5dump`, `h5ls`, `h5copy`, ...)
- Statistiken (`h5perf_serial`, `h5fc`, ...)
- GUI Anwendung: HDFView

Erweiterbarkeit

- Datenmodell lässt Erweiterungen der Funktionalität zu
- Externe Filter (z. B. andere Kompressionsalgorithmen)
- Schnelle Indexierung für bestimmte Datentypen oder Array-Größen
- Erweiterte Thread-Sicherheit über globales Semaphor hinaus

Exkurs: Parallel HDF5

- Parallel I/O:
 - Paralleles Dateisystem (z. B. Lustre) für parallele I/O Anfragen
 - Verwendung des Dateisystems durch mehrere, parallele Prozesse
- PHDF5 ist Parallel I/O Schnittstelle für HDF5
- Parallele und serielle HDF5-Dateien sind kompatibel
- PHDF5 stellt ein Dateiabbild für alle Prozesse bereit
- PHDF5 nutzt Kombination aus *Collective* und *Independent*

Exkurs: Parallel HDF5 (*cont.*)

- *Collective*
 - Kombination vieler kleiner Datenänderungen als große Anfrage
 - Synchronisation mithilfe eines *MPI Communicators*
- *Independent*
 - Keine Koordination mit anderen Prozessen
 - Eher langsam bei vielen kleinen, fragmentierten I/O Anfragen

Zusammenfassung

- HDF5 nutzt hierarchische Struktur für modulare Dateien
- Stellt konsistentes Datenmodell zur Verfügung
- Schnittstelle ist unabhängig von technologischen Entwicklungen
- Ermöglicht Verarbeitung großer und komplexer Datenmengen
- Effiziente und skalierbare Datenverwaltung
- Unterstützt Parallel I/O und Multithreading
- Erweiterbar (z. B. Indexierung)

Literatur I

- [Atma] University Corporation for Atmospheric Research. *NetCDF users Guide: File Structure and Performance*. URL: https://docs.unidata.ucar.edu/nug/current/file_structure_and_performance.html (besucht am 12.06.2022).
- [Atmb] University Corporation for Atmospheric Research. *NetCDF: FAQ*. URL: <https://docs.unidata.ucar.edu/netcdf-c/current/faq.html> (besucht am 07.06.2022).
- [Cod90] E. F. Codd. *The Relational Model for Database Management: Version 2*. USA: Addison-Wesley Longman Publishing Co., Inc., 1990.
- [Fol+11] Mike Folk u. a. „An overview of the HDF5 technology suite and its applications“. In: März 2011, S. 36–47.
- [Groa] The HDF Group. *Dataset Storage Layout*. URL: <https://confluence.hdfgroup.org/display/HDF5/Dataset+Storage+Layout> (besucht am 07.06.2022).
- [Grob] The HDF Group. *Introduction to HDF5*. URL: <https://confluence.hdfgroup.org/display/HDF5/Introduction+to+HDF5> (besucht am 27.05.2022).
- [Groc] The HDF Group. *Introduction to HDF5*. URL: https://web.mit.edu/fwtools_v3.1.0/www/H5.intro.html (besucht am 27.05.2022).
- [Grod] The HDF Group. *Introduction to Parallel HDF5*. URL: <https://confluence.hdfgroup.org/display/HDF5/Introduction+to+Parallel+HDF5> (besucht am 11.06.2022).
- [Hay11] David C. Hay. *Data Model Patterns*. USA: Dorset House Publishing Co., Inc., 2011.

Literatur II

- [Mas+10] Christopher Mason u. a. „Standardizing the Next Generation of Bioinformatics Software Development with BioHDF (HDF5)“. In: *Advances in experimental medicine and biology* 680 (Jan. 2010), S. 693–700.
- [Rus] Iain Russell. *Part 5 - NetCDF*. URL: <https://confluence.ecmwf.int/display/METV/Part%5C%205%5C%20-%5C%20NetCDF> (besucht am 18. 06. 2022).
- [Was] Leah A. Wasser. *Hierarchical Data Formats - What is HDF5?* URL: <https://www.neonscience.org/resources/learning-hub/tutorials/about-hdf5> (besucht am 18. 06. 2022).