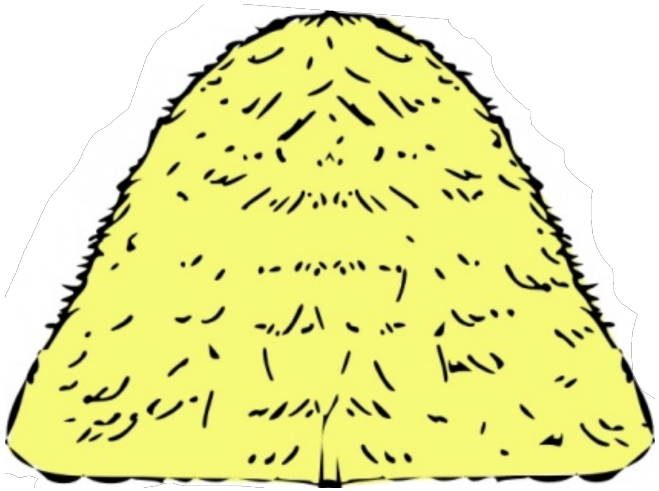
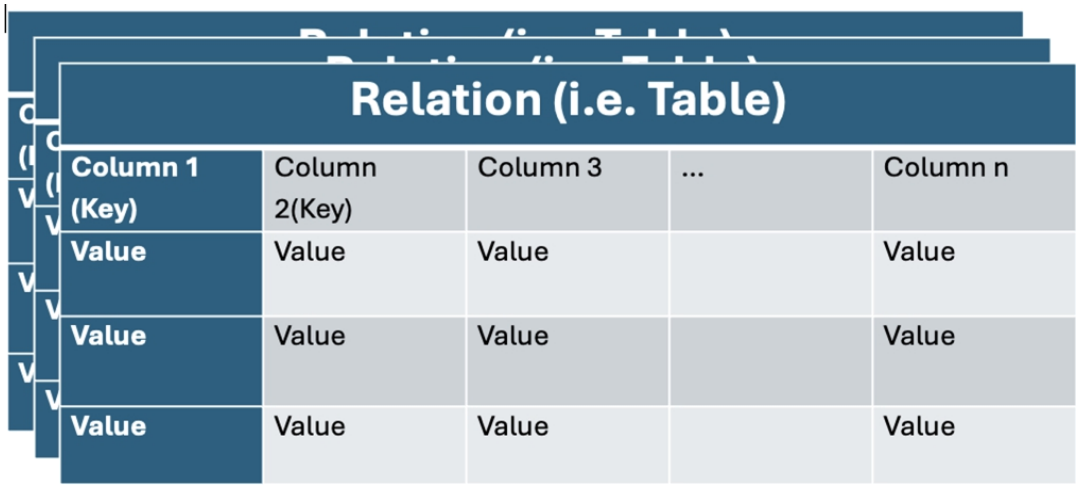
Finding Molecular Needles in Structural Biology Data and MetaData Haystacks The Importance of Relational Databases





Relation (i.e. Table)

Column 1 (Key)	Column 2(Key)	Column 3	...	Column n
Value	Value	Value		Value
Value	Value	Value		Value
Value	Value	Value		Value

Herbert J. Bernstein

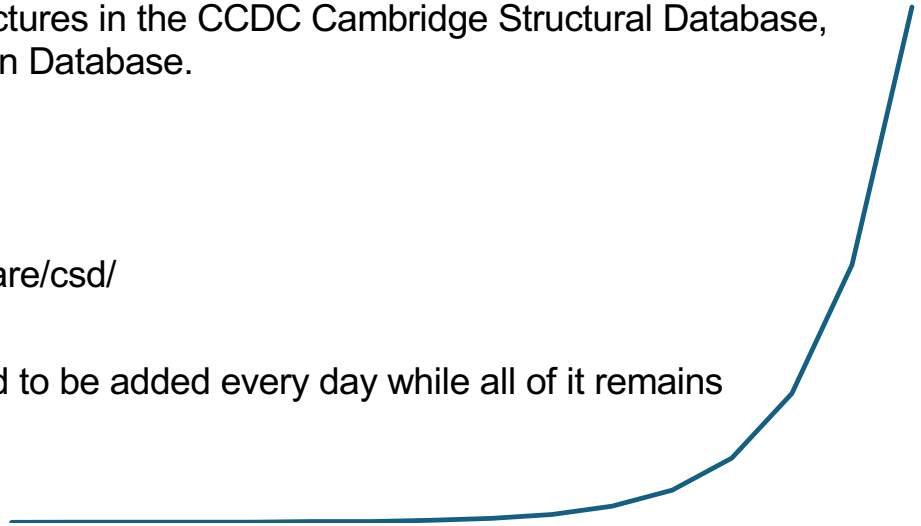
Fresh Pond Research Institute
c/o NSLS-II Brookhaven National Laboratory, Upton, NY 11973

Work supported in part by DIALS National Resource (R24GM154040)

Introduction

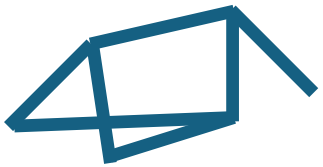
- Structural biology depends on Data and Metadata
- Metadata is the information that helps us to understand the data and how it is organized
- Data may be determined by theory, by calculations, by observations and by experiments, and much data is derived from other data.
- As of 23 March 2025, there were 233,249 structures in the RCSB Protein Data Bank as well as 1,068,577 computed structures models, over 1.25 million structures in the CCDC Cambridge Structural Database, and 523,157 structures in the Crystallographic Open Database.
- Growing exponentially for over half a century
- See

  https://www.rcsb.org
  https://www.ccdc.cam.ac.uk/solutions/software/csd/
  http://www.crystallography.net/cod/
- Very large numbers of datasets with metadata need to be added every day while all of it remains searchable.



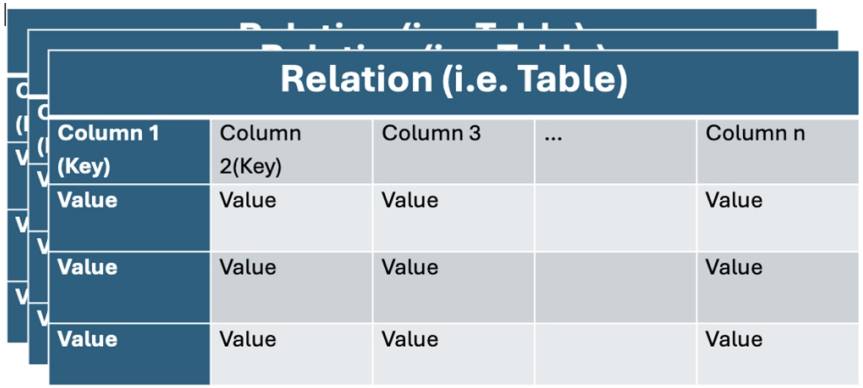
Databases

- People have kept organized sets of data throughout written history, but the growth of computers, modern electronics and networking after World War II made computerized databases an essential part of modern life, especially for doing science. See the Wikipedia article on Databases https://en.wikipedia.org/wiki/Database.
- A fundamental change in our understanding of how to manage databases came from E. F. Codd in 1970, "**insisting that applications should search for data by content, rather than by following links.** The relational model employs sets of ledger-style tables, each used for a different type of entity. Only in the mid-1980s did computing hardware become powerful enough to allow the wide deployment of relational systems (DBMSs [Database Management Systems] plus applications)" [see Wikipedia].
- **Despite many attempts to find non-relational alternatives, for large dynamic databases with multiple readers and writers, as of 2025 there is no reliable alternative to relational databases.**


Relational Databases

- In a relational database, all the data stored in organized into relations (tables). Each relation contains all the data for a given category of data, such as atomic sites
- Each column of a relation identifies a particular set of data, such as atom x-coordinates.
- Each row of a table (a tuple) contains all the values for a relevant instance of data.
- **The only way to find a particular tuple is by the values in columns identified as keys of the relation.** There may be more than one key column for a relation, but there need not be more than one key column.



- For macromolecular data the Nucleic Acid Database was an early example of a relational database [Berman, H.M., Olson, W.K., Beveridge, D.L., Westbrook, J., Gelbin, A., Demeny, T., Hsieh, S.H., Srinivasan, A.R. and Schneider, B., 1992. The nucleic acid database. A comprehensive relational database of three-dimensional structures of nucleic acids. Biophysical journal, 63(3), p.751] which **John Westbook used as the model for a relational version of the Protein Data Bank and the basis for the macromolecular Crystallographic Information File [Westbrook, J.D. and Bourne, P.E., 2000. STAR/mmCIF: an ontology for macromolecular structure. Bioinformatics, 16(2), pp.159-168.]**

Example from mmCIF Category

Here is a small extract of a relation from the mmCIF PDB entry for pdb_00007kty ”Data clustering and dynamics of chymotrypsinogen average structure”, PDB DOI: https://doi.org/10.2210/pdb7KTY/pdb\
The column headings are on the left.

```
•loop_
•_atom_site.group_PDB
•_atom_site.id
•_atom_site.type_symbol
•_atom_site.label_atom_id
•_atom_site.label_alt_id
•_atom_site.label_comp_id
•_atom_site.label_asym_id
•_atom_site.label_entity_id
•_atom_site.label_seq_id
•_atom_site.pdbx_PDB_ins_code
•_atom_site.Cartn_x
•_atom_site.Cartn_y
•_atom_site.Cartn_z
•_atom_site.occupancy
•_atom_site.B_iso_or_equiv
•_atom_site.pdbx_formal_charge
•_atom_site.auth_seq_id
•_atom_site.auth_comp_id
•_atom_site.auth_asym_id
•_atom_site.auth_atom_id
•_atom_site.pdbx_PDB_model_num
```

```
•ATOM   1    N N   . CYS A 1 1   ? 54.77900 13.88900 -10.49600 1.000 54.17000  ? 1   CYS A N   1
•ATOM   2    C CA  . CYS A 1 1   ? 53.44100 13.97400 -9.94100  1.000 53.26000  ? 1   CYS A CA  1
•ATOM   3    C C   . CYS A 1 1   ? 52.67600 12.66700 -10.17300 1.000 51.63000  ? 1   CYS A C   1
•ATOM   4    O O   . CYS A 1 1   ? 52.99500 11.91100 -11.08800 1.000 58.07000  ? 1   CYS A O   1
•ATOM   5    C CB  . CYS A 1 1   ? 52.68200 15.16000 -10.56300 1.000 51.28000  ? 1   CYS A CB  1
•ATOM   6    S SG  . CYS A 1 1   ? 52.39900 15.07300 -12.37200 1.000 59.95000  ? 1   CYS A SG  1
•ATOM   7    N N   . GLY A 1 2   ? 51.66300 12.40800 -9.35300  1.000 49.81000  ? 2   GLY A N   1
•ATOM   8    C CA  . GLY A 1 2   ? 50.72200 11.32300 -9.61800  1.000 49.06000  ? 2   GLY A CA  1
•ATOM   9    C C   . GLY A 1 2   ? 51.23500 9.91200  -9.40100  1.000 53.21000  ? 2   GLY A C   1
•ATOM   10   O O   . GLY A 1 2   ? 50.50300 8.95400  -9.69700  1.000 53.42000  ? 2   GLY A O   1
```

Alternatives to Relations

- There are many alternative formats for structural biology data.
- NeXus/HDF5 provides a tree-structured alternative in which data is organized into nested groups and datasets [Bernstein, H.J., Förster, A., Bhowmick, A., Brewster, A.S., Brockhauser, S., Gelisio, L., Hall, D.R., Leonarski, F., Mariani, V., Santoni, G. and Vonrhein, C., 2020. Gold Standard for macromolecular crystallography diffraction data. IUCrJ, 7(5), pp.784-792.]
- However, if the data involved is ever going to be used in large databases, it is important that any new NeXus/HDF5 presentation of data have a translation to CIF and therefore to relational databases.
- The major step involved is called Normalization.

Normalization

- Very large, logically flat relations mean that any multiwriter database has to have much larger portions locked during writes than if relations are made as small in terms of number of columns and as local in terms of data motion as possible.
- This is not just important for timing; it also greatly reduces the chances of database corruption (“loss of referential integrity”) during database updates.
- This warning was ignored by many (including the author) during the 1970s, but by the1990s it was clear that Codd was right.
- For the current HDRMX project for managing metadata changes for new faster detectors and multimodal experiments, the time to do the normalized versions is now, before too much data is generated.
- imgCIF and mmCIF are already well-normalized.
- The basic idea is simply good modular programming practice.