Large Scale Data Management (Spring 2022) Report

Exploration and Analysis of the Protein Data Bank (PDB)

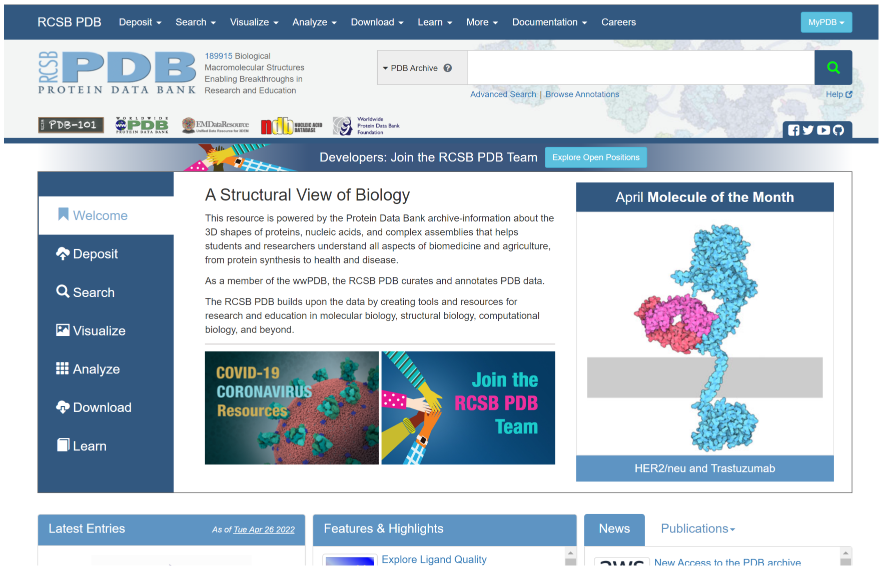
Ethan Payne

May 4, 2022

1 Introduction

1.1 The RCSB PDB

The Research Collaboration for Structural Biology Protein Data Bank (RCSB PDB) is a database containing 189,915 proteins, nucleic acids, and protein-nucleic acid complexes as of April 27 2022. The January 1, 2020 snapshot of the PDB Core Archive is 575 GB in size and has since grown. The database is accessible though its website rcsb.org, and the individual entries are submitted by scientists around the world, including chemists, biologists, and biochemists. Entries are submitted in the form of X-ray crystallography, NMR spectroscopy, or cryo-electron microscopy data, but the PDB website offers access to a multitude of additional features for analysis and visualization, including the Mol* viewer for 3D visualization of structures, structure alignment, and protein symmetry analysis tools among others.



189915 Biological Macromolecular Structures Enabling Breakthroughs in Research and Education

A Structural View of Biology

This resource is powered by the Protein Data Bank archive-information about the 3D shapes of proteins, nucleic acids, and complex assemblies that helps students and researchers understand all aspects of biomedicine and agriculture, from protein synthesis to health and disease.

As a member of the wwPDB, the RCSB PDB curates and annotates PDB data.

The RCSB PDB builds upon the data by creating tools and resources for research and education in molecular biology, structural biology, computational biology, and beyond.

Figure 1: The RCSB PDB homepage directs users toward its suite of tools for querying and visualizing the database, and displays other convenient news and curation features. (https://www.rcsb.org/)

1.2 This Project

1.2.1 Goals and Scope

While accessing and querying the PDB through the rcsb.org website is simple enough on an individual basis, this project will attempt to simulate a larger-scale analysis of the structures contained in the database which would otherwise not be convenient or feasible to perform through the web interface. As a prerequisite step, this project will explore the various means of accessing the PDB with the end goal of downloading a small-scale subset and prototyping some machine-learning analysis. Due to the exploratory nature of this project, the list of specific goals and milestones is subject to change according to what will end up being feasible within the timeframe (see section 2, Implementation).

1.2.2 Project Importance

This project is significant because it involves a real world example of a large scale heterogeneous database depended on by scientists and industries around the world. Furthermore, the RCSB is actively hiring Scientific/Bioinformatics Software Developers, Unit Computing Managers, Researchers, and Web Developers, while many other institutions and companies have positions requiring candidates who can effectively navigate and utilize the database. This project and this report will give students a brief glimpse into the PDB system, perhaps inspiring or enabling them to pursue career opportunities at the intersection of data science and bioinformatics.

The broader goal of this project of facilitating large-scale data analysis on the PDB dataset is also of significant scientific importance. Modern machine learning techniques may reveal patterns and insights which would otherwise go undiscovered by domain scientists who are limited by their lack of familiarity with more advanced computational data analysis methods. Bringing the PDB dataset into the repertoire of data science and machine learning students will foster collaboration between the sciences, and will hopefully lead to a more thorough and productive analysis of biomolecular data.

1.2.3 Target Audience

The target audience of this report is any CS student with an interest in the physical sciences, especially biology and biochemistry. The PDB database serves as not only a good technical challenge in terms of data management, but also exposes students to an example of highly valued domain-specific data.

2 Implementation

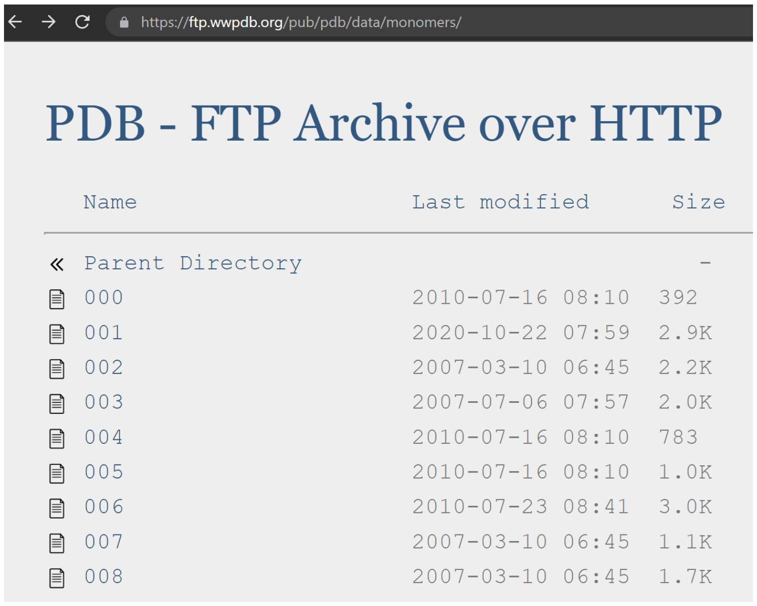
2.1 Data Aquisition

The PDB has its own .pdf file format for describing molecular structures. Unfortunately the format was developed may years ago, and so is cumbersome to use, being in essence a column-delineated text file without delimiter. Additionally, different versions of the .pdb format are described by different data dictionaries, making reliable and maintainable support of the format difficult. Some R and Python packages (such as biopython) exist for parsing PDB files into dataframes, but even these are at times inconsistent and have their own learning curves. Transitioning the .pdb file format to a more structured paradigm may greatly improve its usability for data scientists and researchers. Some other data types within the PDB are stored in the .xml format, so it is not unreasonable to hope that the archaic .pdb text format could be phased out at some point in favor of something more suitable for large scale analysis.

```
RESIDUE   00H     33
CONECT      N12    3 C13  HN12 HN1A
CONECT      C13    4 N12  C14  P21  H13
CONECT      C14    4 C13  C15  H14  H14A
CONECT      C15    3 C14  C16  C20
CONECT      C16    3 C15  C17  H16
CONECT      C17    3 C16  C18  H17
CONECT      C18    3 C17  C19  H18
CONECT      C19    3 C18  C20  H19
CONECT      C20    3 C15  C19  H20
CONECT      P21    4 C13  O22  O23  C24
CONECT      O22    2 P21  HO22
CONECT      O23    1 P21
CONECT      C24    4 P21  C25  H24  H24A
CONECT      C25    4 C24  C26  H25  H25A
CONECT      C26    3 C25  O27  OXT
CONECT      O27    1 C26
CONECT      HN12   1 N12
CONECT      HN1A   1 N12
CONECT      H13    1 C13
CONECT      H14    1 C14
CONECT      H14A   1 C14
CONECT      H16    1 C16
CONECT      H17    1 C17
CONECT      H18    1 C18
CONECT      H19    1 C19
CONECT      H20    1 C20
CONECT      HO22   1 O22
CONECT      H24    1 C24
CONECT      H24A   1 C24
CONECT      H25    1 C25
CONECT      H25A   1 C25
CONECT      OXT    2 C26  HXT
CONECT      HXT    1 OXT
END
HET    00H             33
HETNAM     00H 3-[(R)-[(1R)-1-amino-2-phenylethyl](hydroxy)phosphoryl]
HETNAM   2 00H propanoic acid
FORMUL      00H    C11 H16 N1 O4 P1
```

Figure 2: An example .pdb file for residue 00H. We can notice that this file contains atom connectivity information, but does not specify atom coordinates. We must assume that the coordinate .pdb files are located in some other location within the database. (https://ftp.wwpdb.org/pub/pdb/data/monomers/00H)

36,959 .pdb files from the PDB FTP http archive site monomer subdirectory were downloaded using the BeautifulSoup web crawler package in Python (https://www.crummy.com/software/BeautifulSoup/bs4/doc/) for use as prototyping subset to develop this project. The monomer subdirectory was deemed a good candidate, because it contained small and uncompressed .pdb files which could be directly downloaded in full, as opposed to larger subdirectories of structure files which must first be uncompressed from .gz archive files to obtain the much larger .pdb files. These monomers also offer a highly controlled context for experimentation, since each has far fewer atoms and bonds than other larger structures, therefore simplifying the analysis task. Datatypes other than .pdb files were initially considered for analysis in this project, but accessing and interpreting these other heterogeneous file types introduced too much complexity for a project of this scale, especially considering domain knowledge requirements. Ideally, we would develop a pipeline to automatically parse and load .pdb files into a semi-structured DBMS for subsequent querying and analysis, but this endeavor in itself would require a significant time investment and is thus also out of scope for this project.


https://ftp.wwpdb.org/pub/pdb/data/monomers/

PDB - FTP Archive over HTTP

Name	Last modified	Size
« Parent Directory		-
000	2010-07-16 08:10	392
001	2020-10-22 07:59	2.9K
002	2007-03-10 06:45	2.2K
003	2007-07-06 07:57	2.0K
004	2010-07-16 08:10	783
005	2010-07-16 08:10	1.0K
006	2010-07-23 08:41	3.0K
007	2007-03-10 06:45	1.1K
008	2007-03-10 06:45	1.7K

Figure 3: The monomers subdirectory of the PDB http archive site used for data aquisition (https://ftp.wwpdb.org/pub/pdb/data/monomers/)

After all monomer files are downloaded, the connectivity information for each file is parsed into a python dictionary, and the list of all such dictionaries is converted to a pandas dataframe which is saved to a file using pickle as our processed data repository. Although this approach using pandas is not remotely as scalable as a true DBMS, it is similar enough for the purposes of this project and provides sufficient functionality for the intended analysis prototyping.

2.2 Functionality

This project implements the following functionality in a python notebook (.ipynb) as a stand-alone application:

1. Download all residue files from the monomer subdirectory from of the PDB http archive site `https://ftp.wwpdb.org/pub/pdb/data/monomers/` to a local directory using the BeautifulSoup webcrawler package
2. Parse these raw PDB files into a pandas dataframe and save the parsed database (using pickle) to a file
3. Query the database using pandas dataframe queries
4. Compute some simple statistics on the processed database (average number of bonds per atom in each residue)
5. Plot (using plotly) the computed statistics for visualization of the monomers database

The following additional functionalities were additionally tentatively planned, but were not implemented due to time constraints or complexity of implementation (evolving project scope):

- Perform additional statistical and machine learning analyses on the parsed data
  - clustering
  - principal component analysis (PCA)
  - graph similarity analysis

  - etc.

- Download data from the PDB using the API rather than a web crawler
- Insert the parsed monomer data into a stand-alone DBMS such as MongoDB
- Deploy and display the generated data and plots to a web or mobile application (Django)

2.3 Testing

We generate the following plots from the loaded database using plotly:

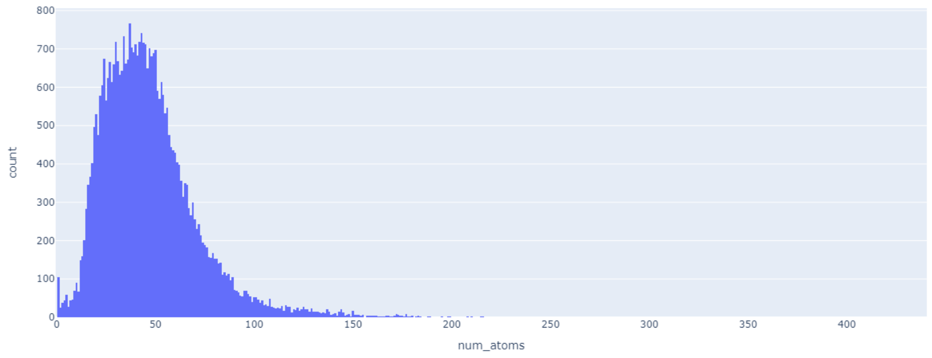


Figure 4: Histogram of total number of atoms per residue. We can observe that the data appear to follow a truncated normal distribution, which makes sense since no residue can have a negative number of atoms, and many monomers share similar structural features.

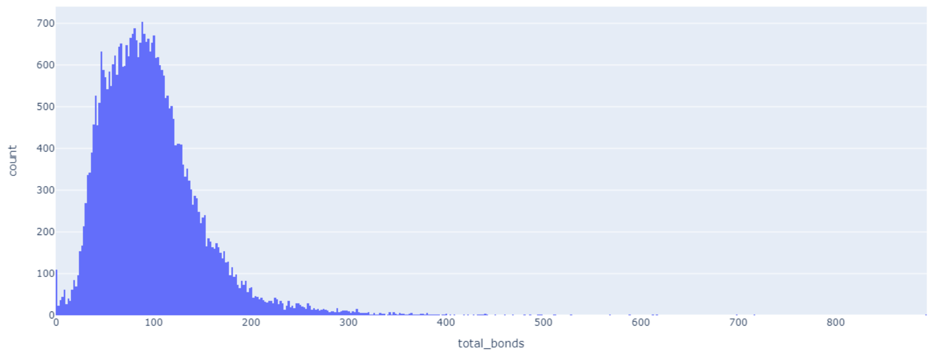


Figure 5: Histogram of total number of bonds per residue. This distribution appears extremely similar to the distribution of the number of atoms for similar reasons.

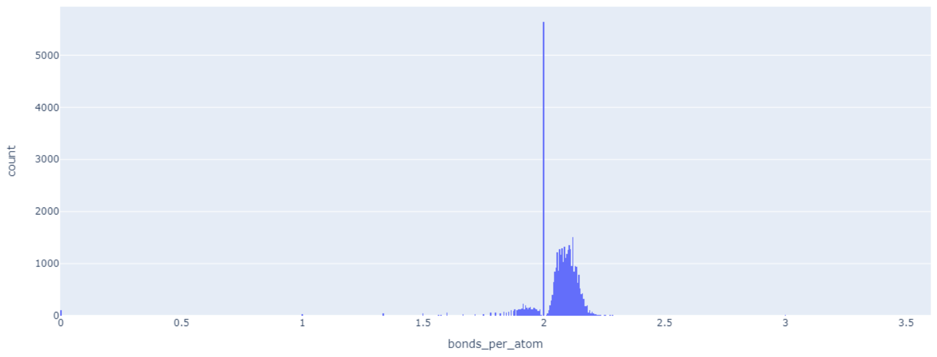


Figure 6: Histogram of average number of bonds atoms per residue. The wide range of values suggests that there may be some outlier monomers. Whether all of these are true outliers or whether some are due to erroneous input files is difficult to determine.

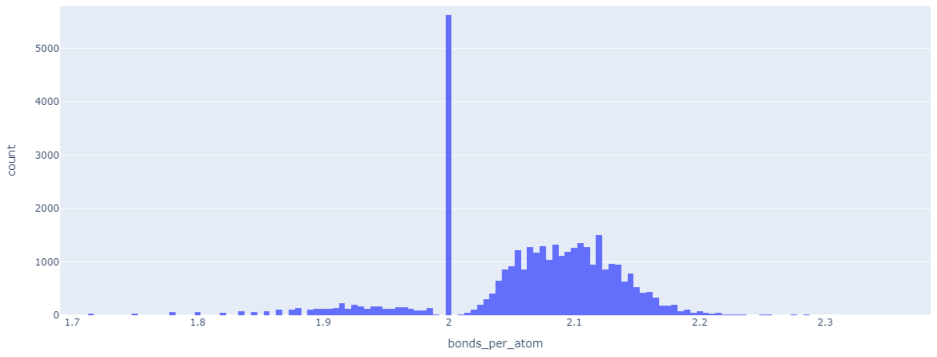


Figure 7: Histogram of average number of bonds atoms per residue (detail). We can observe that while there appears to be a normal distribution centered at approximately 2.1 bonds per atom, there is also a large number of residues which have nearly exactly 2 bonds per atom. These may be very small monomers, or monomers which have some particular structure in common.

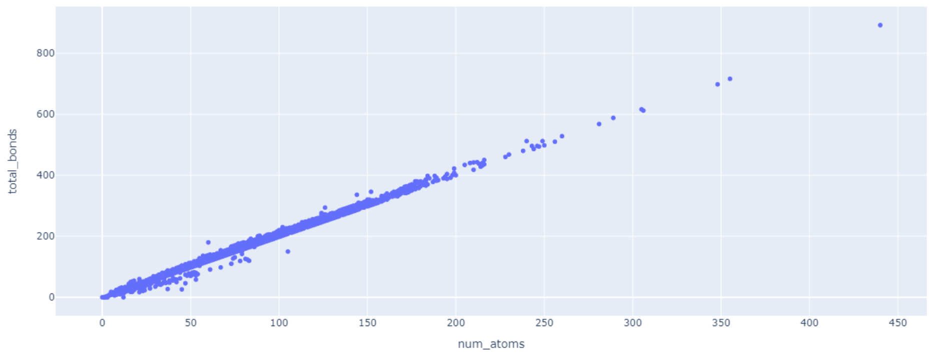


Figure 8: Scatter Plot of number of bonds against number of atoms. We can note the strong linear trend, which is unsurprising since most monomers (and most biomolecules in general) are comprised of the same few component atoms which each have physical restrictions on the number of bonds they can form.

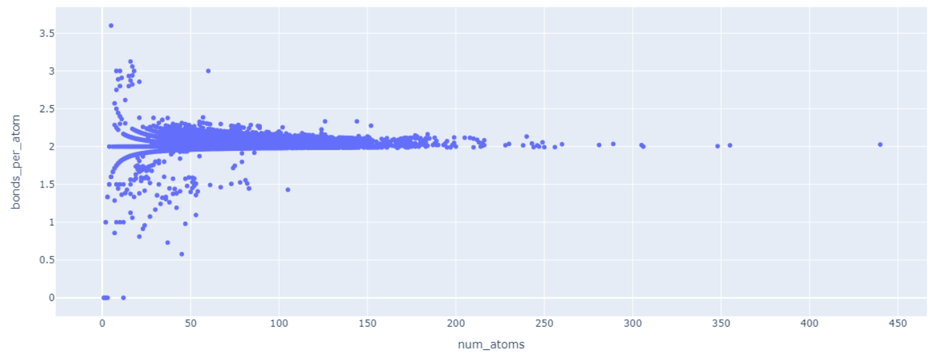


Figure 9: Scatter Plot of average bonds per atom against number of atoms. It is clear that most residues, independent of total size, have an average of roughly 2 bonds per atom. As we might expect, the variability from this average decreases as the number of atoms increases. We can also note what appears to be some distinct patterns for small residues. These may represent monomers which are identical except for one or two atom substitutions.

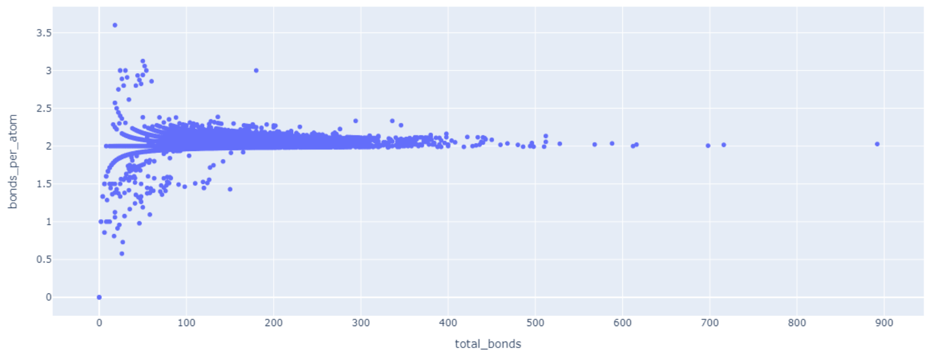


Figure 10: Scatter Plot of average bonds per atom against number of bonds. Unsurprisingly, this trend is extremely similar to the plot of bonds per atom against the total number of atoms, since there is a strong linear relationship between the number of bonds and the number of atoms.

The above plots make semantic sense considering the physical restrictions and patterns that monomers follow, and we could further extend these analysis by investigating the same trends stratified by atomic element. To extend this same idea to larger structures and not just monomers, we can consider similar plots but with number of residues and residue connectivity as an additional variable. Very quickly we can see that this analysis becomes quite complex considering the hierarchical nature of the data.

2.4 Verification/Validation

We had to account for the following data verification considerations:

- Download only the desired .pdb files when using the BeautifulSoup web crawler
- Ensure correct data fields are populated and that the generated plots pass a semantic sanity-checks

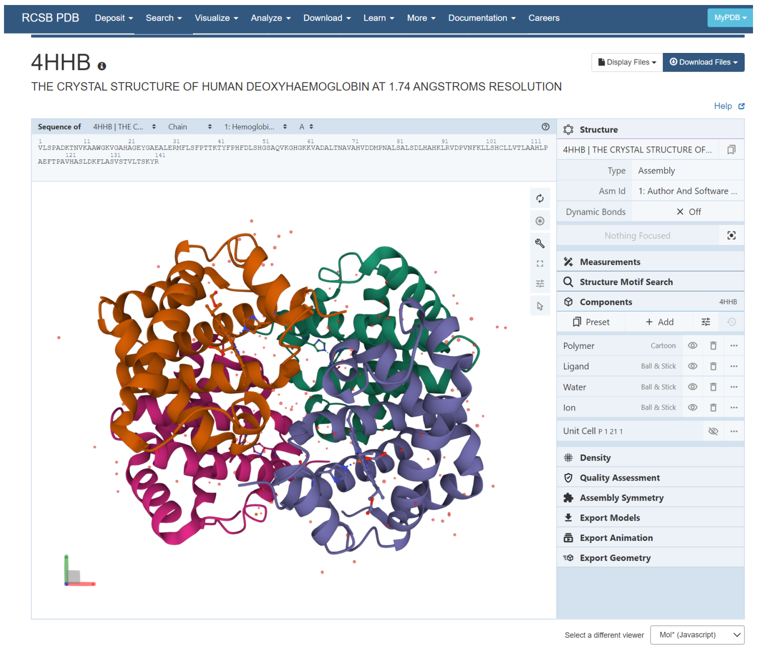
Since the validation goals for this project were limited due to the limited scope of the project, we can consider as an exercise the validation and verification requirements of the greater RCSB database itself.

- Verifying structures and spectra submitted by researchers
  - Verify user accounts before allowing submission (check for user credentials through associated institution or publication records)
  - Checking submitted files for formatting and corruption
  - Verifying that submitted structures (and spectra, etc.) are backed up by peer-reviewed research in order to prevent frivolous submission of unsynthesized or unobserved structures
- Cross-validating duplicate submissions for the same structure
- Correctly annotating files according to structure properties and published results
- Verifying the functionality of rcsb.org itself
  - Verify that all files appear correctly in query/search results
  - Verify accuracy of plots, visualizations, and other displayed analyses
  - Verify the functionality of the Search and Data APIs
  - Maintain network security
    * Detect malicious traffic such as DDOS
    * Detect attempted access to restricted data
    * **https**://www.rcsb.org/

2.5 Challenges

2.5.1 PDB Size and Complexity

Even when using the relatively user-friendly interface on the PDB webpage, interpreting the file structure of the database proved difficult due to the large amount of data heterogeneity (Proteins and Nucleic Acids, but also sub-components of these along with spectra and other supplementary instrumental analysis and metadata files) coupled with the high level of domain knowledge required even for successful use of the search query interface. The PDB website is also clearly designed around individual highly-trained domain users and not general large-scale data delivery, and thus it was difficult to identify an entry point for direct download of the database independent of any specific analysis tool or search functionality. Even though I had significantly more domain knowledge than the average CS student and had used the database during my undergraduate research, I still felt overwhelmed and did not initially know how to select a reasonable subset for download. Luckily, the archived version of the database presented a simpler directory interface, which I was able to use to identify a subset of structures which I was conceptually familiar with for download.


4HHB

THE CRYSTAL STRUCTURE OF HUMAN DEOXYHAEMOGLOBIN AT 1.74 ANGSTROMS RESOLUTION

Sequence of 4HHB | THE C... Chain 1: Hemoglobi... A

```
1         11        21        31        41        51        61        71        81        91        101       111
VLSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNAVAHVDDMPNALSALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLP
    121       131       141
AEFTPAVHASLDKFLASVSTVLTSKYR
```

4HHB | THE CRYSTAL STRUCTURE OF...

Type: Assembly

Asm Id: 1: Author And Software ...

Dynamic Bonds: Off

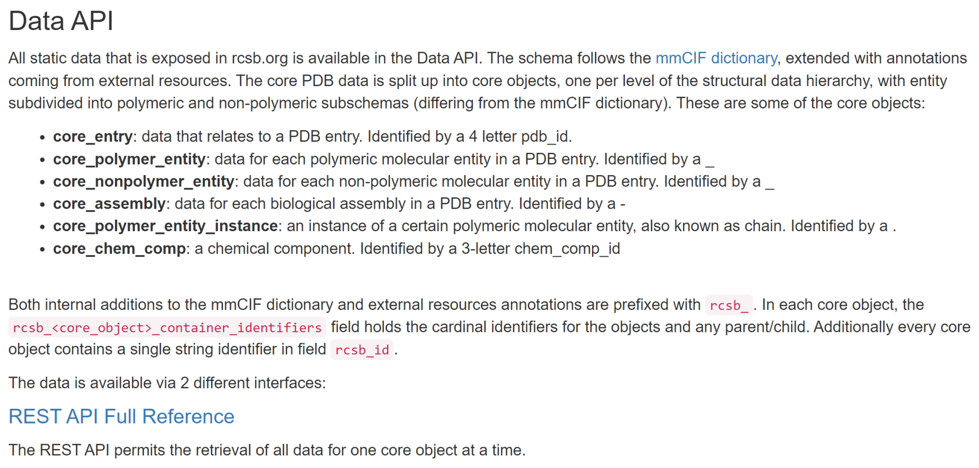
Polymer	Cartoon	
Ligand	Ball & Stick	
Water	Ball & Stick	
Ion	Ball & Stick	
Unit Cell P 1 21 1		

Select a different viewer: Mol* (Javascript)

Figure 11: The Mol* 3D viewer is one of many tools offered for visualizing and analyzing the PDB database. I had originally hoped to embed the Mol* viewer within my own application for this project, but that goal ended up being significantly beyond the scope of what I was able to achieve. (https://www.rcsb.org/3d-view/4hhb/1)

2.5.2 The PDB API

While the original goal of the project was to use the RCSB REST API to download subsets of the full dataset, this goal proved to be more challenging than anticipated. Initial attempts to access the database using the Data API and Search API provided by RCSB were unsuccessful. The high level of domain knowledge required for effective use of these APIs precluded the feasibility of attaining sufficient familiarity with the APIs during the course of a single semester. Additionally, it appears that changes were made to the API system within the last few years, so the tutorials and examples provided on the rcsb.org website were no longer compatible with the new system. Overall, this initial grappling with the APIs slowed the initial exploration and data acquisition phases of the project and resulted in a scaling-back of subsequent scope and expectations.


Data API

All static data that is exposed in rcsb.org is available in the Data API. The schema follows the mmCIF dictionary, extended with annotations coming from external resources. The core PDB data is split up into core objects, one per level of the structural data hierarchy, with entity subdivided into polymeric and non-polymeric subschemas (differing from the mmCIF dictionary). These are some of the core objects:

- **core_entry**: data that relates to a PDB entry. Identified by a 4 letter pdb_id.
- **core_polymer_entity**: data for each polymeric molecular entity in a PDB entry. Identified by a _
- **core_nonpolymer_entity**: data for each non-polymeric molecular entity in a PDB entry. Identified by a _
- **core_assembly**: data for each biological assembly in a PDB entry. Identified by a -
- **core_polymer_entity_instance**: an instance of a certain polymeric molecular entity, also known as chain. Identified by a .
- **core_chem_comp**: a chemical component. Identified by a 3-letter chem_comp_id

Both internal additions to the mmCIF dictionary and external resources annotations are prefixed with `rcsb_`. In each core object, the `rcsb_<core_object>_container_identifiers` field holds the cardinal identifiers for the objects and any parent/child. Additionally every core object contains a single string identifier in field `rcsb_id`.

The data is available via 2 different interfaces:

REST API Full Reference

The REST API permits the retrieval of all data for one core object at a time.

Figure 12: The Data API was difficult to interpret without a deep study of the full reference and a review of pertinent domain knowledge, and the provided example code was out-of-date. (https://www.rcsb.org/docs/programmatic-access/web-services-overview)

3 Conclusions

3.1 The PDB

This project was extremely enlightening and allowed me to both expand my familiarity with the PDB database, and identify areas where the RCSB could improve their user experience to further facilitate academic research. In particular, the PDB API should be better documented, with clear examples to facilitate programmatic large-scale data analysis. Additionally, even though the PDB database is primarily intended for use by domain specialists, ensuring accessibility to outside data scientists would have a very positive impact on the speed and diversity of research. For this reason I believe that the PDB should develop a crash-course in the associated domain knowledge for biochemical structures and their corresponding data structures in the PDB database.

3.2 Project Functionality

While not all functionality goals made it into the final implementation, I expected at the onset of the project that this may end up being the case due to the complexity of the PDB database. Much time was required during the initial exploratory phase of the project, even just to determine a viable point-of-entry for programmatically accessing and downloading the data. However, now that the problem of data acquisition has been relatively solved and some preliminary analyses have been performed, the additional project functionality goals (other than the use of the PDB API) seem very feasible if this project were to be extended during a future semester.

3.3 Ideas for Future Work

It is my personal belief that the PDB database and this general approach to exploring and downloading subsets for analysis represents a very well-suited project for a course in large scale data management. While CS students are fairly often exposed to big data sources in the form of social media and google image data, it is not as common that they have such direct access with an actual actively-used scientific database. During my exploration of the PDB database, I have identified the following avenues for extension of this project into future assignments or projects:

- Develop a pipeline for parsing and loading .pdb files into a DBMS
- Identify all the different file formats and data types employed by the RCSB

- Plot the distribution of filesizes, file formats, number of subcomponents, and other metadata for the entire PDB database or a subset
- Develop an application using MapReduce to query and analyze the PDB in parallel
- Use API calls to embed PDB query results into another web or mobile application
- Extend the structural analyses performed in this project to consider additional variables, such as atomic element and graph-connectivity features.
- Extend the analyses of this project to include atom coordinate data in addition to connectivity information.

For an extreme challenge worthy of publication in high-level scientific journals:

- incorporate the physical properties of structures (atomic potentials, masses, charges, simulation data etc.) into the analysis results
- combine/synthesize an analysis of nucleic acid structure with other biological data to identify trends or effects in certain species
- perform a comprehensive analysis of any specific structures which are of particular interest to domain scientists

4 References

1. http://www.rcsb.org/
2. H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, P.E. Bourne. (2000) The Protein Data Bank Nucleic Acids Research, 28: 235-242.