Title

The Pfizer Crystal Structure Database: An essential tool for structure-based design at Pfizer

Permalink
https://escholarship.org/uc/item/3wt6x14g

Journal
Journal of Computational Chemistry, 43(15)

ISSN
0192-8651

Authors
Gehlhaar, Daniel K
Luty, Brock A
Cheung, Philip P
et al.

Publication Date
2022-06-05

DOI
10.1002/jcc.26862

Received: 18 January 2022 | Revised: 7 March 2022 | Accepted: 23 March 2022

DOI: 10.1002/jcc.26862

SOFTWARE NOTE

Journal of COMPUTATIONAL CHEMISTRY WILEY

The Pfizer Crystal Structure Database: An essential tool for structure-based design at Pfizer

Daniel K. Gehlhaar[1] | Brock A. Luty[2] | Philip P. Cheung[3] | Andy H. Litman[1] | Robert M. Owen[4] | Peter W. Rose[5]

[1]Pfizer, Inc., San Diego, California, USA

[2]Independent Consultant, Poway, California, USA

[3]Refactor BioSciences, San Diego, California, USA

[4]Pfizer Ltd, Cambridge, UK

[5]Structural Bioinformatics Laboratory, San Diego Supercomputer Center, San Diego Supercomputer Center, La Jolla, California, USA

Correspondence
Daniel K. Gehlhaar, Pfizer, Inc., 10777 Science Center Drive, San Diego, CA 92121, USA.
Email: dan.gehlhaar@pfizer.com

Abstract

Pfizer's Crystal Structure Database (CSDB) is a key enabling technology that allows scientists on structure-based projects rapid access to Pfizer's vast library of in-house crystal structures, as well as a significant number of structures imported from the Protein Data Bank. In addition to capturing basic information such as the asymmetric unit coordinates, reflection data, and the like, CSDB employs a variety of automated methods to first ensure a standard level of annotations and error checking, and then to add significant value for design teams by processing the structures through a sequence of algorithms that prepares the structures for use in modeling. The structures are made available, both as the original asymmetric unit as submitted, as well as the final prepared structures, through REST-based web services that are consumed by several client desktop applications. The structures can be searched by keyword, sequence, submission date, ligand substructure and similarity search, and other common queries.

KEYWORDS
database, protein-ligand interactions, structure preparation, X-ray crystallography

1 | INTRODUCTION

Pfizer has a long history of successfully prosecuting structure-based targets to yield marketed drugs. Essential to this effort has been the availability of high-quality structural data, most notably utilizing protein X-ray crystallography. There has been explosive growth in the number of protein X-ray structures available in the public domain, through the Protein Data Bank (PDB),[1] which has been driven by technological advances in molecular biology, crystallization techniques, data collection methods, and computational refinement methods, as well as the increased availability of high-intensity radiation sources. These technologies have driven a similar explosive growth within Pfizer, resulting in an ever-increasing number of novel and proprietary protein crystal structures being solved by Pfizer scientists worldwide.

Making these crystallographic structures available to project scientists in an efficient and timely fashion is a significant informatics and scientific challenge. While flat files deposited by crystallographers into a shared file system may work for small teams over a short period of time, this methodology cannot scale to a global network of collaborating scientists who access the data across research sites distributed over many time zones, across projects, and who need immediate access to structures solved many years prior. Given the timescales involved in the drug design industry, it is also common for a project team to require access to structures solved by crystallographers who are no longer employed with Pfizer. In addition, with a flat-file-based methodology, it is very easy to misplace essential data files such as intensity data, refinement, and scaling logs, which are essential for publication, and also helpful for establishing an intellectual property timeline.

Another issue that must be addressed is that the result of the X-ray crystallographic experiment—the asymmetric unit of the crystal lattice—is not usable for drug design. These structures lack hydrogen atoms and often have missing residues or side chains, due to poor electron density. In addition, there is a fundamental issue that is intuitive to crystallographers but is often not realized by downstream scientists such as chemists and computational chemists. Associated with every asymmetric unit (and space group) is a set of operators (crystallographic symmetry operations) that defines how to take the asymmetric unit and fully populate the unit cell and thus the infinite lattice. Crystallographers only solve the unique atoms in the asymmetric unit, and chemists often do not understand that a given water molecule or ligand, for example, is actually present not just in the location given in the raw asymmetric unit file, but in a number of other symmetry-related locations as defined by the symmetry operations. The same issue exists, naturally, for the protein chains themselves. Although a given asymmetric unit may appear to contain a binding site that is fully exposed to solvent, for example, crystallographic symmetry may place another chain directly proximal to this binding site, and the ligand bound to that site may well be making interactions with the neighboring chain. Ignoring these packing and symmetry issues can easily result in misinformed design decisions by project teams.

One more issue that must be addressed is that a large set of crystal structures, especially those from a single protein, or from a family of related proteins (e.g., a set of kinases), can in theory be mined as an aggregate to yield emergent information not present when seen as a collection of individual structures. Examples include specificity analysis, which is vitally important for kinase projects or other highly related protein families; analysis of protein conformations; knowledge-based potentials; ligand and scaffold hopping; hybrid ligand design; and the like. These types of analyses can only be done practically if residues and chains are consistently numbered, and each set of related structures shares a common coordinate reference frame (especially with respect to the binding site). These sorts of transformations are very tedious to do in an ad hoc fashion, especially for thousands of protein structures. In addition, different scientists may perform these transformations slightly differently, resulting in results that are not directly comparable.[2]

Pfizer has invested a considerable effort over the last two decades to build a comprehensive software platform to resolve all of these issues and give project scientists access to high-quality structural data in a very efficient manner. The result of this effort is the Pfizer Crystal Structure Database (CSDB), an informatics platform that accepts X-ray crystallographic asymmetric units and homology models from project scientists or the PDB, performs extensive cleanup and structure preparation, and delivers the resulting structures directly to project scientists. The complete system will be described in detail in this manuscript.

To date, the CSDB contains nearly 20,000 asymmetric units which generate over 30,000 prepared structures. Of these, approximately 12,000 structures are proprietary, with the remainder being imported from the PDB. Several structures per day on average are loaded into the CSDB and are accessed by scientists at a rate of several hundred structures per day.

We believe that the CSDB constitutes one of the largest and most diverse protein crystal structure databases in existence and is a key competitive advantage for Pfizer. With the recent entry of various commercial products in this space,[3–5] we felt it important to document this history of CSDB and its impact on drug design at Pfizer. The ease of use and attention to what is most important for scientists, such as having turn-key structures that are immediately ready for use by computational and medicinal chemists, eliminating the need for scientists to interpret crystallographic symmetry, and automatically overlaying related structures based on their binding sites, makes the CSDB essential for structure-based design at Pfizer.

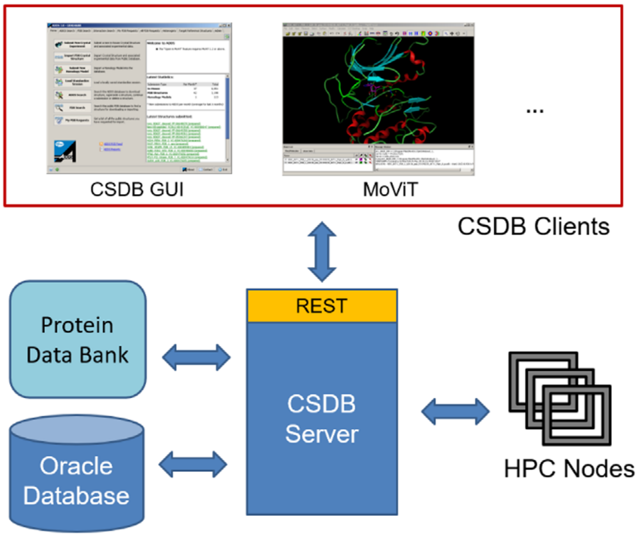
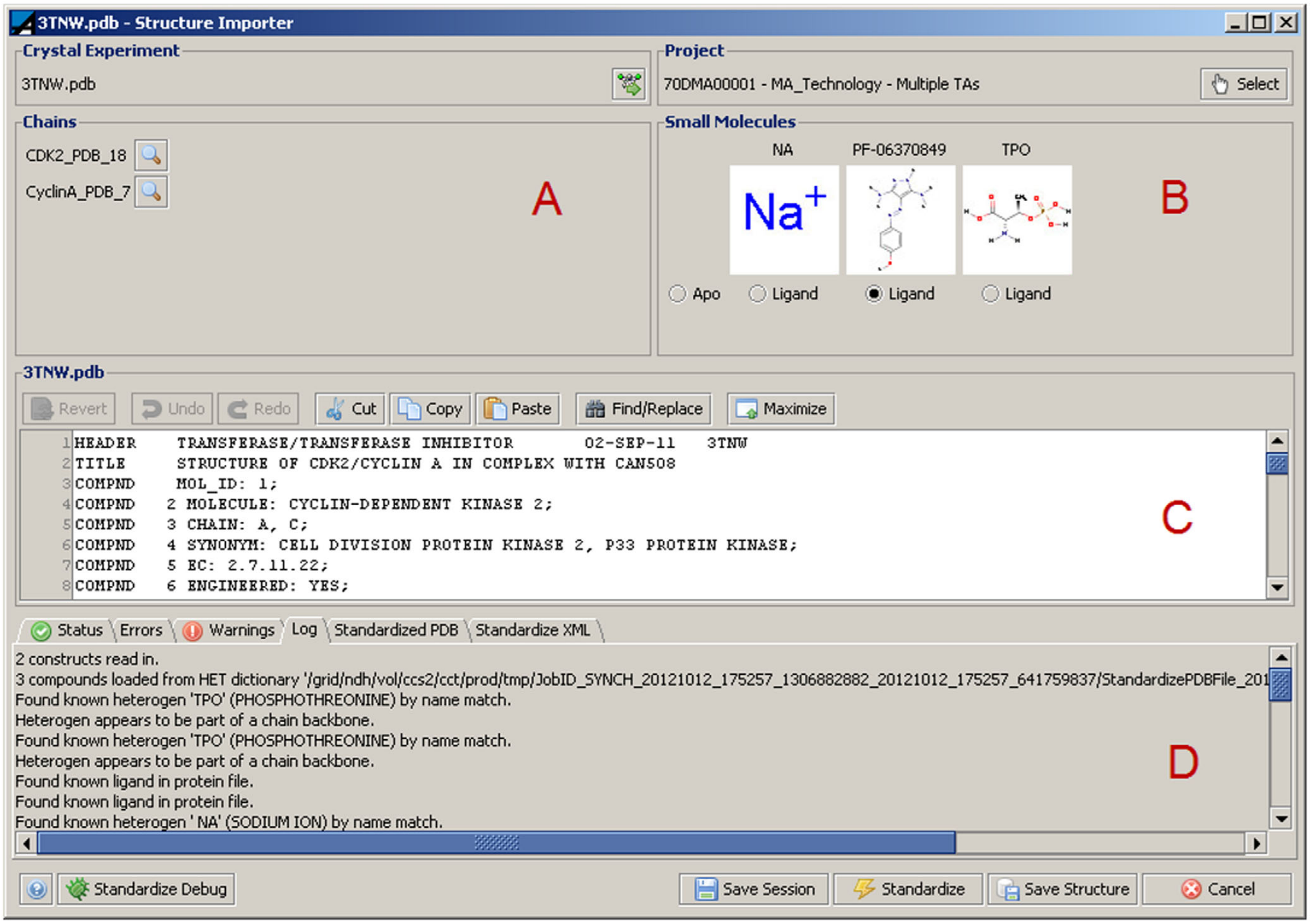


FIGURE 1 CSDB architecture overview. The CSDB Server exposes data from Oracle, calculated data on Pfizer high performance computing hardware, and external data such as from the PDB, to clients using a robust set REST-based APIs. CSDB, Pfizer's Crystal Structure Database; PDB, Protein Data Bank

2 | ARCHITECTURE

From the outset, the CSDB had several key goals, aside from addressing the issues previously mentioned. First and foremost, we realized that usability of the software must be emphasized from a very early stage in the development design process. The software must be easy and intuitive to use, both for structural biologists and computational chemists who deposit structures, and for the project chemists and designers who retrieve structures to enable and accelerate compound design. Second, we had to preserve as much of the original submitted data as possible. The final, prepared crystal structure must be fully compatible with the final refined electron density and have identical heavy atom positions as the submitted structure. The original asymmetric unit structure as submitted must be available for inspection, as should any other submitted data (refinement and scaling logs, reflection files, etc.). This is important not only because of the possibility of software errors—if there is an error in a parser, for example, the original data must be re-parsed once the issue is fixed—but also to maintain a clean intellectual property timeline. In short, the CSDB must add significant value, both in terms of efficiency and information



Crystal Experiment: 3TNW.pdb

Project: 70DMA00001 - MA_Technology - Multiple TAs

Chains: CDK2_PDB_18, CyclinA_PDB_7

Small Molecules: NA, PF-06370849, TPO

○ Apo ○ Ligand ● Ligand ○ Ligand

```
HEADER    TRANSFERASE/TRANSFERASE INHIBITOR       02-SEP-11   3TNW
TITLE     STRUCTURE OF CDK2/CYCLIN A IN COMPLEX WITH CAN508
COMPND    MOL_ID: 1;
COMPND   2 MOLECULE: CYCLIN-DEPENDENT KINASE 2;
COMPND   3 CHAIN: A, C;
COMPND   4 SYNONYM: CELL DIVISION PROTEIN KINASE 2, P33 PROTEIN KINASE;
COMPND   5 EC: 2.7.11.22;
COMPND   6 ENGINEERED: YES;
```

```
2 constructs read in.
3 compounds loaded from HET dictionary '/grid/ndh/vol/ccs2/cct/prod/tmp/JobID_SYNCH_20121012_175257_1306882882_20121012_175257_641759837/StandardizePDBFile_201
Found known heterogen 'TPO' (PHOSPHOTHREONINE) by name match.
Heterogen appears to be part of a chain backbone.
Found known heterogen 'TPO' (PHOSPHOTHREONINE) by name match.
Heterogen appears to be part of a chain backbone.
Found known ligand in protein file.
Found known ligand in protein file.
Found known heterogen ' NA' (SODIUM ION) by name match.
```

FIGURE 2 Graphical interface for a representative run of STANDARDIZE, in this case for PDB structure 3TNW. (A) Two registered chains were automatically identified, in this case a CDK2 chain and a CyclinA chain. (B) STANDARDIZE automatically identifies small molecules, including looking up structures in the internal Pfizer database. In this case, the ligand of interest to modelers has been selected. (C) The original file is shown and is stored unmodified in the CSDB for future reference. (D) STANDARDIZE generates various human- and machine-readable outputs, including a summary of its processing, shown here. CSDB, Pfizer's Crystal Structure Database; PDB, Protein Data Bank

quality, in order for it to be widely accepted within the Pfizer scientific community.

The overall architecture of the CSDB is summarized in Figure 1. The CSDB in its current form can be neatly described in terms of two key parts. The first is a more-or-less traditional three-tier informatics platform. Users interact with the CSDB via a Java thick client that allows structure submission, various search and retrieval capabilities, and administrative functionality. The client connects to a server process, also implemented in Java, which manages transactions with an Oracle database, where the actual data (input structures, prepared structures, auxiliary files, and parsed data) are stored. Object-relational mapping is accomplished using Hibernate. Clients communicate with the server with a comprehensive set of REST-based APIs.

To a significant extent, the Java client and server are not “chemistry-aware,” in the sense that they, for example, contain little code to parse structure files or represent molecular structures. The bulk of the chemical/structural awareness resides in a separate set of code, written in C++, which constitutes the second key part of the CSDB system. This suite of applications is built on top of a Pfizer-proprietary C++ software library for representing and manipulating molecular structures, and performing force field-based methods. The C++ applications perform error checking and structure preparation roles, and will be described in detail below. They are invoked as services from the Java code on Pfizer's high performance computing cluster.

3 | STRUCTURE SUBMISSION

3.1 | Validation and annotation

Entering complex data, such as that associated with a protein-ligand crystallographic complex, into a database system inherently balances two often contradictory goals—ensuring data integrity and making the process as easy as possible for the scientists entering the data. Structures are entered into CSDB through the thick client GUI interface, starting with two essential pieces of data—the associated Pfizer project code, and either the asymmetric unit file (for in-house structures) or the PDB code (for imported structures; the asymmetric unit file is then downloaded). The structure is then run through a specialized "gatekeeper" application, STANDARDIZE, that is responsible for error checking and annotating the structures in a manner consistent with the PDB specification,[6] to generate a "standardized" form of the asymmetric unit that is stored and used for further processed (Figure 2). STANDARDIZE is invoked as a service and returns a detailed XML file that is parsed by the CSDB client and presented to the user.

For data consistency, there are a few key business rules that the CSDB enforces for each structure. For example, each identified sequence (e.g., SEQRES entry) must correspond to a construct that has been registered in CSDB. Similarly, each small molecule in the structure must be registered, either as a "known" small molecule or nonstandard residue ("heterogen") within the CSDB system, or as compound with an assigned Pfizer identifier. This is often done iteratively—STANDARDIZE is run on the structure and identifies unknown compounds or sequences; users identify the unknown sequences or small molecules; and then STANDARDIZE is re-run until all errors are resolved. Notably, if SEQRES annotations are absent in the input asymmetric unit file, STANDARDIZE derives the approximate sequence based on the information present in the file, and this is then used to look up possible sequence matches the CSDB database. Sequences are registered via a GUI that allows scientists to easily compare sequence numberings within a protein family using colors to distinguish matches, mismatches, and missing residues. This interface is particularly helpful when matching mutant sequences for reference structures, described below, as it highlights the superposition residues that will be used during structure preparation. CSDB allows deposition of both peptide and nucleotide (RNA and DNA) sequences.

The error checking performed by STANDARDIZE has evolved from a small set of basic rules to a complex and comprehensive set of checks, as driven by the errors or annotation issues found over the years in structures submitted to the system. Even in PDB structures, despite rigorous structure submission requirements, inconsistencies are common, such as noncovalent interactions annotated with CONECT records (especially with groups interacting with metals). In addition, by definition, bond orders are missing from the PDB-imported files (although they are available from online component dictionaries). For that reason, CONECT information is routinely stripped from structures retrieved from the PDB, and the connectivity information recalculated from the atomic coordinates. The set of error conditions and repair functions performed by STANDARDIZE include:

- Verifying that the expected ligands are present (missing atoms are allowed).
- The sequences present in the submitted structure matches the expected sequences (in particular, no added residues, and no unexpected connections between residues; the latter is a flaw commonly introduced by refinement programs).
- Repairing alanine substitutions (where ALA is added as a placeholder for a residue with undefined side chain density).
- Ensuring that there are not alignment errors in the atom name field for some metals or halogens, for example, differentiating calcium and alpha-carbon.
- Checking for missing backbone atoms or unconnected side chain density.
- Adding covalent polymerization bonds for specific heterogens; disulfide bridges; nonstandard residues and post-translational modifications (e.g., NAG); or those annotated via LINK records.
- Checking for both extreme close contacts and extremely long bonds which are often signs of either missing alternate position indicators or improper refinement.
- Ensure that small molecules have the same bond orders as the registered compounds.

As its name implies, a main purpose of STANDARDIZE is standardizing annotations. This process adds tremendous value to the submitted structures, ensuring that all structures in the CSDB conform to an absolute standard for basic annotations. The annotations generated by STANDARDIZE include:

- HET, HETNAM, and HETSYN records for all small molecules, especially the ligand of interest, which is assigned a HETNAM of "Ligand" and a HETSYN with the compound's Pfizer ID.
- Numbering residues according to that registered with the constructs in CSDB (usually the UniProt sequence numbering).
- Assigning chain IDs in a consistent fashion based on the mapped sequences.
- Ensuring that SITE annotations have corresponding REMARK 800 entries and vice versa.
- Annotate missing residues with REMARK 465 entries.
- Annotate missing side chain atoms with REMARK 470 entries.

3.2 | Nonstructural data

In addition to the basic structural data present in an asymmetric unit file (atom positions and connectivity plus the annotations added by STANDARDIZE), there are many other important pieces of data, such as the cell parameters, space group, resolution, crystallization conditions, and R-factors, that can be associated with a structure submission in CSDB. Most of these data are present in some form or another in either the submitted PDB-format file (e.g., refinement data in the "REMARK 3" records), or in auxiliary files such as scaling and refinement log files. As previously discussed, a key design point of CSDB was to make structure submission as straightforward as possible, so considerable effort was put into automatically parsing this nonstructural data from the input files and populating the corresponding data fields in the GUI. CSDB is capable of automatically parsing refinement, scaling, data collection, and crystallization condition information from most such applications in common use, including PHENIX,[7] SHELX,[8] TNT,[9] REFMAC,[10] Scala,[11] and many others. Additionally, CSDB is able to directly query the database used by the Rock Maker[12] application, which is utilized internally at Pfizer to track crystallization conditions and populates the crystallization conditions fields in the GUI. Together, these automated data extraction and formatting steps dramatically reduce the amount of work to fully submit a structure and has been key to the successful deployment of CSDB at Pfizer. Once these fields have been completely parsed, users can revise or edit any of the values, or enter data values absent in the submitted files.

3.3 | Reflection and structure factor data

No crystallographic data submission would be complete without a means to capture reflection intensity data, or the derived structure factors/phases. CSDB accepts reflection data in several common formats (MTZ, HKL, and others) and parses it into a common reduced format which are used to generate 2Fo-Fc electron density maps when combined with the submitted coordinates. Optionally, the MTZ file generated during refinement can be submitted as well, and the contained structure factor information is used directly to generate electron density. For structures imported from the PDB, structure factors are automatically imported in CIF format and parsed.

3.4 | Project reference structures and design units

The stoichiometry (i.e., number and type of protein chains) in the asymmetric unit is rarely the biological assembly that is meaningful for structure-based drug design. For example, HIV Protease is active in a homodimer, with the binding site being formed at the dimer interface; but many solved crystal structures contain four protein chains per asymmetric unit (two of the dimeric complexes). CSDB uses the concept of a “reference structure” to define how to split an asymmetric unit into subsets suitable for design; these subsets are referred to as “design units.” The reference structure also serves other purposes aside from defining how to split the asymmetric unit during structure preparation. Often, designers have a particular view of the protein or binding site that they are accustomed to, for example, certain residues “up” or “left.” The reference structure defines the standard view that all associated structures are superposed onto. The superposition is performed using a subset of residues selected by project team experts, generally with residues in the binding site, and are annotated in the reference structure via a PDB-format SITE entry.

To align an entire family of proteins onto a single reference frame, for example, all Kinases, project teams can choose a representative reference structure, and subsequently align the reference structures for related proteins onto the central reference. This process cannot be completely automated, as it does require some knowledge from a domain expert, including an alignment of all related protein sequences to use as a basis for correctly choosing superposition residues for each reference structure. However, the benefit of this wide-scale alignment of structures for use in derived studies (such as specificity analysis, hybrid design, or core hopping) cannot be overstated.

CSDB has specialized logic to account for two very common reference structure use cases. First, it is very common for a given protein to be crystallized with several different constructs, due to differences in protein preparation methodology, or addition of C- or N-terminal tags, especially between in-house and PDB-imported structures. In addition, point mutations are often introduced to investigate resistance mechanisms, crystal stability, solubility, and the like. Requiring a unique reference structure for each unique sequence would be unduly burdensome and time-consuming for users. Instead, CSDB can map related sequences onto a single reference structure, utilizing the common numbering defined when the sequences were registered into the system. Second, there are cases where a protein chain is found in some, but not all, structures for a given project. One example is CDK2 kinase, which is sometimes co-crystallized with, for example, Cyclin-A. A single reference structure can be used in this case, with the Cyclin-A chain annotated as being optional.

Scientists often want to annotate structures to define specific subsites, such as the hinge or activation loop regions of kinases. CSDB allows reference structures to contain an arbitrary number of SITE annotations, which are, along with their associated REMARK 800 annotations, transferred to the prepared structures. The residue specifications in the SITEs are automatically modified to account for differences in sequence numbering between the reference and submitted structures.

Scientists choose which reference structure they want to use for a given submitted asymmetric unit; in practice, most projects define a single reference structure. However, CSDB limits their choices to those which are compatible, based on which sequences are contained in the reference structure, and the stoichiometry of those sequences as compared to the submitted structure. Once a reference structure is assigned to a submitted asymmetric unit, the asymmetric unit is then processed via the CSDB structure preparation protocol.

4 | STRUCTURE PREPARATION

The CSDB structure preparation process takes the raw asymmetric unit files as submitted, and generates representations suitable for compound design and computational chemistry. Structure preparation consists of the following steps:

- Splitting asymmetric units into design units.
- Adding hydrogens, capping chain breaks and building missing side chains.
- Optimizing hydrogen positions.
- Superposing onto the reference structure.

Note that a given asymmetric unit file can result in any number of prepared structures, as the ratio of the number of output prepared structures to input structures depends on how many copies of the design unit can be found in the asymmetric unit (based on the stoichiometry of the chains in the assigned reference structure).

The fact that the exact same preparation process is applied to all structures in the database is of vital importance in terms of scientific rigor. If the preparation process was not consistent, it would not be possible to meaningfully compare numerical results from models derived from different crystal structures; there would always be debate as to whether the differences were due to the structures themselves or the differences in how they were processed. In addition, since this process is completely automated, if an error in the preparation process is found, it can easily be reapplied to all structures in the database, ensuring continued improvement in the quality of the stored structures.

4.1 | Splitting asymmetric units into design units

One of the largest sources of confusion in modeler's interpretations of crystal structures has to do with crystallographic symmetry. While crystallographers instinctively realize that each atom in the asymmetric unit may be repeated multiple times in the unit cell, this key point is often lost on modelers, who often simply take a given structure at face value, utilizing only the atoms as given in the asymmetric unit. This can lead to significant misinterpretation, as key interactions are overlooked. To complicate matters, the symmetry operations are often non-trivial, as they depend on nonorthogonal coordinate transformations and vary significantly between space groups.

To address this problem, the preparation process applies crystallographic symmetry operators to all water molecules and small molecules (ligands and noncovalent heterogens) to fully populate the hydration and small molecule packing environment around the submitted asymmetric unit. All such molecules within a parameterized cutoff (currently 4.0 Å) are kept. To ensure that all possible symmetrically related positions have been sampled, the process is extended to neighboring unit cells, sampling all possible $+/-1$ and $+/-2$ fractional coordinate shifts for all molecules, for each symmetry operator. A similar procedure is done for the protein chains in the structure; the symmetry-related chains that are close to the given asymmetric unit coordinates are kept and are written to a separate file. This “packing” file is essential for understanding the possible role of crystal packing forces on the observed protein structure and ligand binding mode.

Once the symmetry operations have been applied, a greedy algorithm is employed to locate the design units within the asymmetric unit. All combinations of chains in the asymmetric unit file are attempted to be mapped onto the reference structure chains, and for each such mapping, an RMS fit is attempted, using all matching alpha carbons. If the RMS deviation is less than a parameterized threshold (currently 7.5 Å), the match is declared to be a design unit and is removed from further consideration. While a greedy algorithm is inherently prone to finding non-optimal solutions, in practice, it has worked very well in this application.

Finally, each design unit is output in PDB format (maintaining all associated annotations from STANDARDIZE), keeping all waters and small molecules within a prescribed cutoff of the associated protein chains. Note that, in cases where a given small molecule or water is near two different design unit chains, that molecule is included in both design units (Figure 3).

4.2 | Adding hydrogens, capping chain breaks, and building missing side chains

The next step in the preparation process is perhaps the most controversial and speaks to the different requirements for crystallographers and computational chemists. While hydrogen atoms are not generally observable in macromolecular X-ray crystallographic experiments, they are essential for performing any force field-based molecular analysis or calculating interactions such as hydrogen bonds. Likewise, electron density for some protein atoms (such as flexible side chains or loops) can be undefined, and coordinates for those atoms are not determined as part of the crystallographic experiment. However, those atoms are indeed present, even if their exact locations are unknown. More significantly, arbitrary truncation at residue boundaries is very common in crystallographic experiments, resulting in strong monopoles and unbalanced dipoles (primary amines and aldehydes) that would negatively impact electrostatic calculations if used naively.

To address these problems, the preparation process adds hydrogen atoms, caps chain breaks, and builds in missing side chains, with all added atoms being given occupancy zero, which is consistent with the crystallographic experiment. Hydrogen atoms are added using a proprietary rule-based algorithm that takes into account element, hybridization state, and to a small extent, the local environment. Additionally, protonation state is adjusted for acidic and basic groups to be proper for pH 7.4, for example, acids are deprotonated and amines are protonated. Chain breaks—regions of undefined electron density where whole residues are not present—are capped with N-methyl (NME) groups on the carbonyl side and acetyl groups (ACE) on the amide side. This ensures that the electrostatics at these chain breaks correspond correctly with that actually present in the protein structure, and that terminal residues retain a biologically relevant protonation state.

Missing side chain atoms are built using templates, taking care to exactly use the atomic coordinates of any side chain atoms that are present in the submitted structure. Torsions for these added atoms are sampled based on the Penultimate Rotamer Library.[13] Rotamers with severe steric clashes are eliminated, and then the resulting subsets are iteratively screened based on energies calculated with an in-house implementation of the AMBER force field.[14] Once the best rotamer for each built side chain has been identified, all added atoms are minimized starting at the coordinates from the sampled rotamer. Again, all heavy atom coordinates from the submitted crystallographic experiment remain unchanged.

4.3 | Optimizing hydrogen positions

Positioning hydrogen atoms correctly is of vital importance to correctly evaluating molecular interactions such as hydrogen bonds, as well as for performing force field-based methods for understanding physical interactions. Additionally, macromolecular X-ray crystallographic experiments cannot readily distinguish between glutamine, asparagine, and histidine side chain flip states, or histidine tautomerization states. The CSDB preparation process employs a

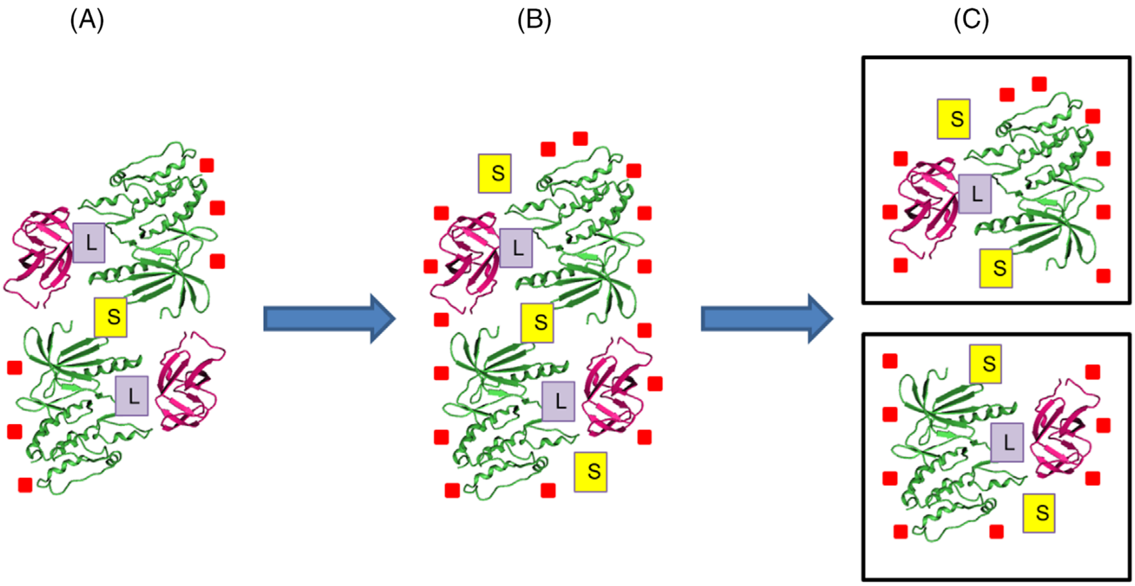


FIGURE 3 Overview of the splitting of an asymmetric unit into design units. Hypothetical protein chains are represented as ribbon structures. The ligands of interest to modelers are represented by an "L," while another small molecule, for example, a salt or metal ion, is represented by an "S." solvent molecules are represented by red squares. (A) The original input asymmetric unit. Note that only part of the structure has solvent present. (B) After application of crystallographic symmetry operators, solvents are fully populated. Also note that the small molecule has been reproduced as well. (C) The design units are extracted based on RMS calculations from a given reference structure. Solvents and small molecules that are near to the interface between design units will be included in both

proprietary algorithm that optimizes hydrogen positions as well as addresses the orientation of the aforementioned side chains. It samples a large number of states for each applicable group (water, hydroxyl, thiol, and primary amine groups, as well as the ASN, GLN, and HIS side chains), calculating energies with the CHARMm force field[15] and a TIP3P water model. It also samples flips, tautomers, and protonation states of histidines. Hydrogen bonding networks are identified, and each network is minimized using a self-consistent mean field algorithm.[16]

4.4 | Superposing onto the reference structure

The final step in the preparation process is the superposition of the design unit onto the reference structure coordinates. This is accomplished via a least-squares fit based on the alpha-carbon coordinates of the predefined superposition residues annotated in the reference structure. As previously mentioned, this step is vital in adding value, especially when structures exist for the same protein and/or family (Figure 4).

4.5 | Ancillary data

Once the preparation process has been completed, significant ancillary data is automatically generated for the convenience of users. A 2Fo-Fc electron density map file is generated in the proximity of the binding site, based on either the provided reflection intensities and asymmetric unit file coordinates, or the final structure factors from refinement, if provided. Three modeling sessions are automatically generated for each prepared structure for use in Pfizer's in-house molecular design tool, MoViT. The first session contains the ligand in the binding site with key interactions (e.g., hydrogen bonds) illustrated. The second illustrates the bound ligand in the presence of a

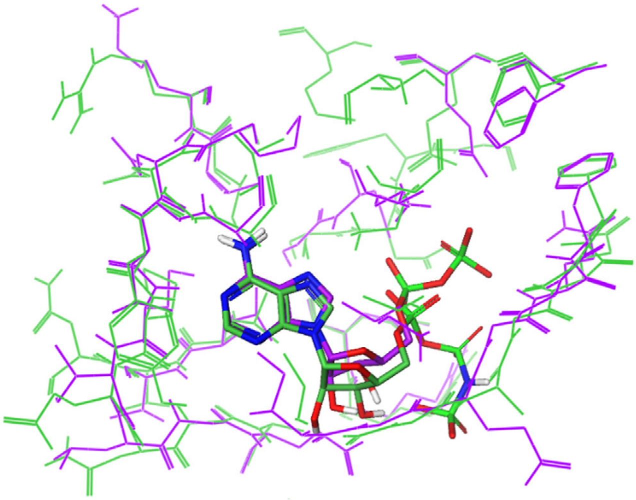
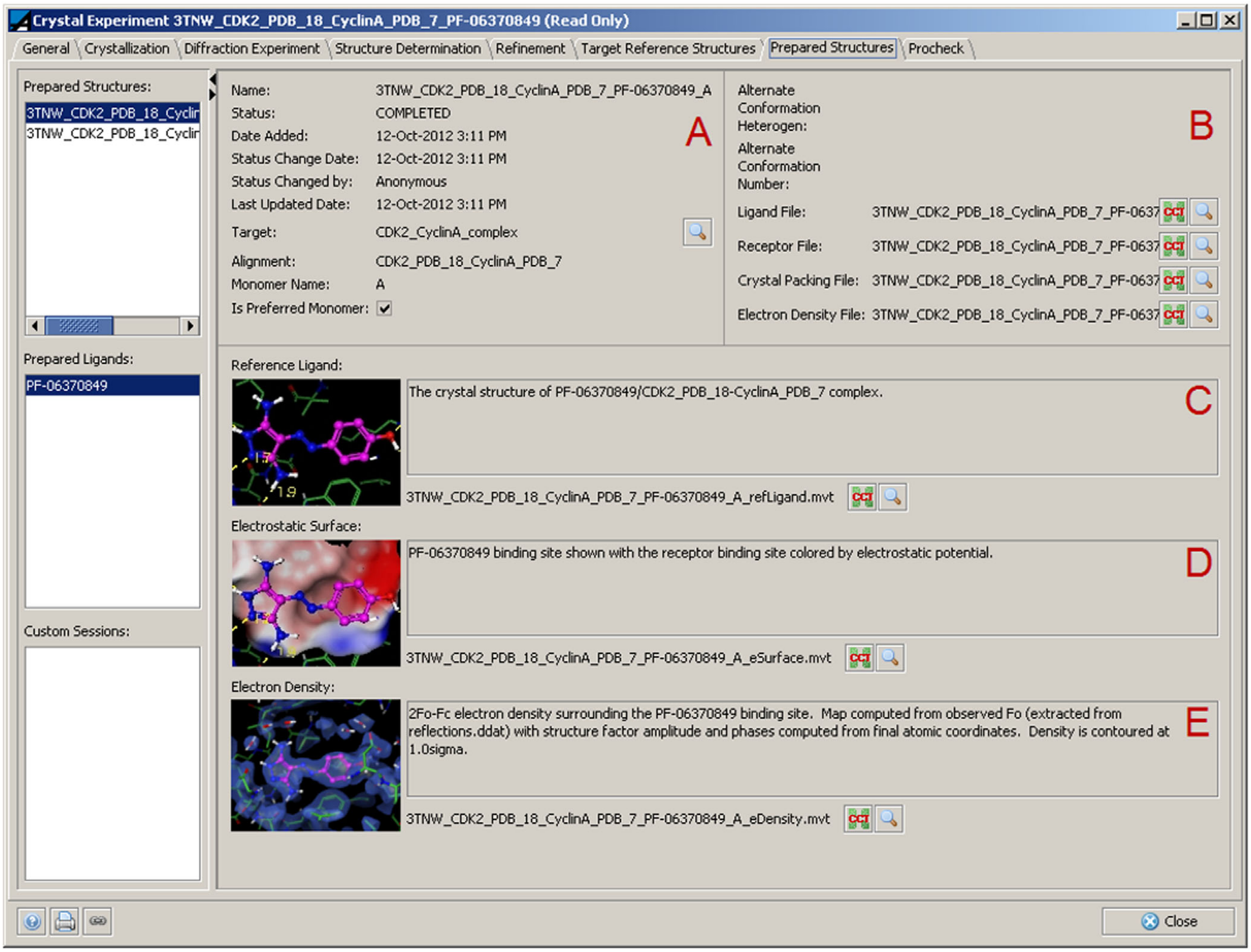


FIGURE 4 Superposition of two prepared kinase structures: 1MP8 (FAK, purple) and 1MQB (EphA2, green) bound to their respective nucleotide analogues. Although there are clear similarities, for example, in the hinge region, this superposed view shows clear areas of distinction that can be exploited to created targeted compounds



Crystal Experiment 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-06370849 (Read Only)

General | Crystallization | Diffraction Experiment | Structure Determination | Refinement | Target Reference Structures | Prepared Structures | Procheck

Prepared Structures:
3TNW_CDK2_PDB_18_Cyclir
3TNW_CDK2_PDB_18_Cyclir

Prepared Ligands:
PF-06370849

Custom Sessions:

Name: 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-06370849_A
Status: COMPLETED
Date Added: 12-Oct-2012 3:11 PM
Status Change Date: 12-Oct-2012 3:11 PM
Status Changed by: Anonymous
Last Updated Date: 12-Oct-2012 3:11 PM
Target: CDK2_CyclinA_complex
Alignment: CDK2_PDB_18_CyclinA_PDB_7
Monomer Name: A
Is Preferred Monomer: ☑

Alternate Conformation Heterogen:
Alternate Conformation Number:
Ligand File: 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-0637
Receptor File: 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-0637
Crystal Packing File: 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-0637
Electron Density File: 3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-0637

Reference Ligand:
The crystal structure of PF-06370849/CDK2_PDB_18-CyclinA_PDB_7 complex.
3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-06370849_A_refLigand.mvt

Electrostatic Surface:
PF-06370849 binding site shown with the receptor binding site colored by electrostatic potential.
3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-06370849_A_eSurface.mvt

Electron Density:
2Fo-Fc electron density surrounding the PF-06370849 binding site. Map computed from observed Fo (extracted from reflections.ddat) with structure factor amplitude and phases computed from final atomic coordinates. Density is contoured at 1.0sigma.
3TNW_CDK2_PDB_18_CyclinA_PDB_7_PF-06370849_A_eDensity.mvt

FIGURE 5 Graphical interface showing details for a representative prepared structure, in this case, PDB structure 3TNW. (A) Basic information about the structure, including the submission date and the reference structure used to generate the design units. (B) The various files (ligand, receptor, packing, and electron density) are available for viewing and/or export. (C) An auto-generated MoViT session showing significant interactions. (D) An auto-generated MoViT session showing an electrostatic surface of the binding site. (E) An auto-generated MoViT session showing a 2Fo-fc electron density map in the neighborhood of the ligand. All sessions are available for viewing or download. PDB, Protein Data Bank

molecular surface of the binding site, colored by the protein's electrostatic potential. The final modeling session contains a view of the binding site with the calculated electron density. These modeling sessions are stored in the database and are directly accessible through the Java thick client and MoViT (Figure 5).

5 | QUALITY OF LIGAND FIT

One possible significant source of error in the interpretation of X-ray crystallographic structures is with respect to the quality of the fit of ligands to electron density, especially with respect to the strain energy of the ligand in the putative bound orientation.[17] Assessing the quality of the ligand fit is currently not included in the CSDB, although it has been discussed and may well be included in a future release. The primary issue is that while automated measures of ligand fit to electron density do exist,[18] manual inspection is generally preferred,[17] which is impossible at the scale of CSDB. In addition, while an estimate of strain energy can be calculated using molecular mechanics, it is difficult to reach scientific consensus on issues such as global versus local strain, sampling methods, choice of force field, and so forth. As a whole, errors with ligand fit in CSDB, especially with respect to structures solved in-house, have not been identified as a major source of problems for structure-based design teams at Pfizer.

6 | SEARCHING AND STRUCTURE RETRIEVAL

A database is only useful if it is easy to find and retrieve the stored information that users are interested in. CSDB offers a robust set of search and retrieval functions and exposes these through a set of

REST APIs that are used by the Java thick client, as well as other applications (e.g., an in-house molecular visualization tool, MoViT). Additionally, CSDB provides REST APIs to easily retrieve specific information or structures without the need for direct database access or UI interaction.

The most common search method by users is simple browsing of list of available structures. Structures are organized in two parallel trees, one by the protein sequences involved in the structures (a two-level hierarchy by protein family and protein; e.g., "Kinase," "CDK2"), and the other by the associated therapeutic area and project, as defined by the Pfizer project code assigned to the structure at the time of submission. Users can choose to view a list of prepared structures (the most common case), the asymmetric units, or the original submission in the case of homology models.

CSDB supports robust searching by keywords, which includes searching of the name, author, protein, project, and comment fields. The search supports full use of binary operators (NOT, AND, OR) and parentheses, which allows very complex queries to be submitted. Additionally, there are two key pieces of data that are often very important for finding structures of interest—the resolution of the structure and the submission date. The search term parsing algorithm identifies the keywords "RESOLUTION" and "DATE" and allows common relational operators (<, >, >=, etc.) to allow users to quickly narrow their search to the specific structures of interest.

It is very common for users to want to find if the deposited structure whose bound ligand is most similar to a query. Likewise, it is often desired to find all structures in the database whose ligands contain a particular substructure, such as a given core structure. Similarity and substructure searching is enabled through an in-house REST API to internally hosted service providing this functionality.

Finally, CSDB allows searches by protein sequence, using the BLAST[19] algorithm to identify structures containing sequences that match the query sequence at a given expectation value. An internal BLAST database is maintained for this purpose and is updated whenever a new sequence is registered or modified in CSDB.

The search functionalities are exposed on the CSDB server via REST APIs implemented with the Jersey framework. The queries are submitted in a proprietary XML format that encodes all of the allowed query components, as described above. These APIs return the number of structures of each type, the associated protein and project trees, and details for the structures themselves, including date, author, structure name, crystallographer comments, and hyperlinks to download receptor and ligand structures, electron density, and packing files. These APIs are currently used by several clients (including the CSDB thick client, and the MoViT molecular design tool) to directly query and access the CSDB data.

7 | MAINTAINING DATA INTEGRITY

The CSDB contains a large amount of interdependent data. For example, sequences are registered with a specific residue numbering scheme, which is then propagated (via STANDARDIZE) to the asymmetric unit files, and subsequently (via the structure preparation process) to the prepared files. If a piece of information changes—a sequence numbering is changed, bond orders in registered compounds are modified, or a reference structure is updated, for instance—it is important that all affected structures are correctly updated. To account for this, the CSDB was designed to be "self-healing." Several times per hour, processes, referred to as "agents," run that check for changes to the data, resolve which structures are affected, and automatically process the required changes. This includes, for example, preparing newly deposited structures, but also includes, for example, re-running STANDARDIZE (and subsequently, the preparation process) on any structures associated with a modified ligand, heterogen, or sequence. Again, this process follows the core CSDB design themes of ease of use and maintenance of data integrity. Notably, if an agent process fails, it simply generates an error message, and whatever failed will be reattempted at the next invocation. This allows for a very fault-tolerant architecture, as dependent servers and the like will inevitably have occasional failures. This architecture greatly limits the chances for significant data corruption.

Any data from previous submissions can be edited, either by the scientist(s) who submitted them, or by privileged curators. The background agents then automatically determine if dependent data needs to be recomputed. For example, a scientist could enter an updated reflection file, causing recalculation of electron density for that structure.

Other agent processes are run with less frequency—from daily to weekly, depending on their function—to perform actions such as dumping structure data to flat files, updating metadata from external data sources such as the PDB, and other ancillary functions. The modular nature of the agent processes allows new ones to be added easily, to expand CSDB functionality without significantly changing the core data structures and design.

8 | PDB SEARCH

The PDB repository is constantly being updated, with new structures being added on a constant basis. Many of these new structures are of direct interest to project teams, either because they are of the intended drug design target, or are related in some way, such as a related target with specificity implications. CSDB provides a GUI to search the PDB for deposited structures that match a given sequence at some predefined similarity. The search is performed using BLAST and can be restricted by the release date and the resolution of the structure. The resulting list indicates whether the given structure has already been imported into the CSDB, and provides methods to download or view the structures, import the structures, or add the structures to a list of import requests. The searches can be set up to email results on a weekly basis, which, when combined with the date restrictions, can provide scientists with a very clean, minimal list of key PDB structures of interest to their project teams.

ACKNOWLEDGMENTS

Aside from the authors, there is a long list of people who have directly or indirectly contributed to the CSDB, either with respect to development, or simply testing or lending ideas. These include David Airmet, Anthony Colson, Dr. Djamal Bouzida, Dr. David G. Brown, Dr. Barry Finzel, Dr. Stephan T. Freer, Dr. Ketan Gajiwala, Kenny Leung, Dr. Joseph Moon, Veda Larson Palmer, Dr. Jayvardhan Pandit, Dr. Paul Rejto, David Schumann, Shao-Tien Sng, and Eric Trull.

DATA AVAILABILITY STATEMENT

Corresponding author will share details of all algorithms described in the manuscript by request. All data detailed in the manuscript is available from public sources such as the PDB. Proprietary crystal structures solved at Pfizer cannot be made available.

ORCID

Daniel K. Gehlhaar https://orcid.org/0000-0002-7462-5519
Peter W. Rose https://orcid.org/0000-0001-9981-9750

REFERENCES

[1] H. Berman, K. Henrick, H. Nakamura, *Nat. Struct. Biol.* **2003**, *10*, 980.

[2] S. E. O'Brien, D. G. Brown, J. E. Mills, C. Phillips, G. Morris, *J. Mol. Graph. Modell.* **2005**, *24*, 186.

[3] PSILO, *Chemical Computing Group*, PSILO, Montreal, Canada **2021**.

[4] PLDB, *Protein-Ligand Database (PLDB)*, Schrodinger Inc., New York **2021**.

[5] SPRUCE, *OpenEye Scientific Software*, SPRUCE, Santa Fe **2021**.

[6] PDB, File format documentation, http://www.wwpdb.org/docs.html#format (accessed June 6, 2021).

[7] P. V. Afonine, R. W. Grosse-Kunstleve, N. Echols, J. J. Headd, N. W. Moriarty, M. Mustyakimov, T. C. Terwilliger, A. Urzhumtsev, P. H. Zwart, P. D. Adams, *Acta Cryst. D: Biol. Cryst* **2012**, *68*, 352.

[8] G. Sheldrick, *Acta Cryst. C.* **2015**, *71*, 3.

[9] D. E. Tronrud, L. F. Ten Eyck, International Tables for Crystallography. in *International Union of Crystallography*, Vol. F (Eds: E. Arnold, D. M. Himmel, M. G. Rossmann) **2012**, p. 520, Ch.18.7. https://it.iucr.org/Fb/https://doi.org/10.1107/97809553602060000111

[10] G. N. Murshudov, P. Skubák, A. A. Lebedev, N. S. Pannu, R. A. Steiner, R. A. Nicholls, M. D. Winn, F. Long, A. A. Vagin, *Acta Cryst. D* **2011**, *67*, 355.

[11] R. H. Blessing, *Acta Cryst. A* **1995**, *51*, 33.

[12] Rock Maker, *Formulatrix*, Rock Maker, Massachusetts **2021**.

[13] S. C. Lovell, J. M. Word, J. S. Richardson, D. C. Richardson, *Proteins* **2000**, *40*, 389.

[14] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, P. A. Kollman, *J. Am. Chem. Soc.* **1995**, *117*, 5179.

[15] A. D. MacKerell, D. Bashford, M. Bellott, R. L. Dunbrack, J. D. Evaseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wiorkiewicz-Kuczera, D. Yin, M. Karplus, *J. Phys. Chem. B* **1998**, *102*, 3586.

[16] P. Koehl, M. Delarue, *Nat. Struct. Biol.* **1995**, *2*, 163.

[17] J. W. Liebeschuetz, *J. Med. Chem.* **2021**, *64*, 7533.

[18] A. Meyder, E. Nittinger, G. Lange, R. Klein, M. Rarey, *J. Chem. Inf. Model.* **2017**, *57*, 2437.

[19] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, D. J. Lipman, *Nucleic Acids Res.* **1997**, *25*, 3389.

How to cite this article: D. K. Gehlhaar, B. A. Luty, P. P. Cheung, A. H. Litman, R. M. Owen, P. W. Rose, *J. Comput. Chem.* **2022**, *43*(15), 1053. https://doi.org/10.1002/jcc.26862