wwPDB Processing Procedures and Policies Document

Section A: wwPDB processing procedures

Authored by the wwPDB annotation staff
March 2009 Version 2.3

Table of Contents

Preface 3
1 Sequences and sequence database reference assignment 4
2 Ligands 8
3 Coordinate section 12
4 Chain ID assignment 16
5 HEADER assignment 18
6 Compound information 19
7 Author information 22
8 Citation information 23
9 REMARKs 24
10 Miscellaneous records 29
11 Structural Genomics Entries 32
12 Information specific to X-ray structures 33
13 Information specific to NMR structures 35
14 Information specific to Electron Microscopy structures 36
15 Viral capsids and other complex assemblies 37
16 Work in progress 39
Appendices: 40
A. HEADER list 40
B. Format for Structure Factors 41

Preface

Since 1999, the wwPDB has been responsible for processing PDB data with deposition centers at RCSB PDB, PDBe, and PDBj. The processed entries follow the PDB format as described in the Protein Data Bank Contents Guide Version 3.2 (September, 2008), and the mmCIF format that complies with the PDB Exchange Dictionary (PDBx) http://mmcif.pdb.org/dictionaries/mmcif_pdbx.dic/Index/index.html[1].

There are some data items for which the processing procedures are ambiguous. Over the course of the last 12 months the annotation teams at the wwPDB deposition sites have worked to formalize many aspects of PDB annotation policies and procedures

This document presents the annotation processing rules that are a result of this review. These rules will be fully implemented by December 2008. The wwPDB staff will continue to update annotation practices in line with evolving structure determination and annotation methods.

The sections in this document are:
A: wwPDB processing procedures
B: wwPDB policies

December 2008: Initial release as version 2.2
March 17: minor revision 2.3, updates on citation, header list, SITE records and added cif templates for structure factors.

[1] PDBx Revision History: http://mmcif.rcsb.org/dictionaries/mmcif_pdbx.dic/Data/history.html

1 Sequences and sequence database reference assignment

(see PDB records DBREF and SEQRES)

What is the definition of sequence? (Maps to SEQRES and entity_poly_scheme)

Sequence is a list of the consecutive chemical components covalently linked in a linear fashion to form a polymer. The chemical components included in this listing may be standard or modified amino acid and nucleic acid residues. It may also include other residues that are linked to the standard backbone in the polymer. Chemical components or groups covalently linked to side-chains (in peptides) or sugars and/or bases (in nucleic acid polymers) will not be listed here.

Proteins containing 3 or more residues forming consecutive standard peptide bonds will be assigned sequence related records (SOURCE, COMPND, DBREF, SEQRES).

Nucleic acids containing two or more residues linked by standard nucleotide bonding will be assigned sequence related records (SOURCE, COMPND, DBREF, SEQRES).

The sequence records must represent all macromolecules used in the experiment, including HIS- or other expression tags, as well as residues missing from the coordinates due to disorder. Residues cleaved from the macromolecules prior to or during the experiment are not part of the sequence. The sequence can include neighboring cross-linked residues (such as chromophores) and modified amino acids.

What if the exact sequence of the sample is not known? If the exact sequence of the sample is not known, due to, for example, proteolysis, the sequence should match the coordinates and a REMARK 999 (_pdbx_entry_details.sequence_details) will be added. If the entry is a crystal structure, the Matthews coefficient and solvent content will list author-provided values instead of calculated values.

What is the sequence database reference (Maps to DBREF and struct_ref, struct_ref_seq)? The sequence database reference token DBREF and struct_ref, struct_ref_seq provides the mapping between a sequence in the sequence section against a valid sequence database reference.

Which polymer chains are assigned sequence database records? Each chain for which there is an appropriate sequence database reference will have DBREF, struct_ref, and struct_ref_seq records. The sequence will be self-referenced (i.e. the database reference will be the PDB entry itself) when no sequence database reference is available.

Which sequence databases will be used?

Proteins UniProt (UNP) is the current preferred protein sequence database. Where there are multiple UNP entries for the same protein from the same organism, strain and sequence identity, the UNP entry that has the most annotation will be used. Also, UNP entries that contain the complete protein sequence will be preferred over those that represent protein fragments.

Nucleic acids

Naturally obtained nucleic acid sequences can be referenced to GenBank, EMBL or DDBJ. For some of these database references, the residue numbers of the match in the sequence

database do not fit into the PDB file format, therefore the matching sequence database reference will be listed in the mmCIF file.

All synthetic DNA and RNA polymers will be self-referenced.

What if a UniProt reference is not available or is modified for a protein sequence? UniProt does not contain variable or hyper-variable regions of the immune system or unnatural sequences, so the PDB entries for such structures will be self-referenced. If the protein does not fall into these categories and does not have a UniProt reference, the author is encouraged to submit their sequence information via SPIN (the UniProtKB submission tool) at http://www.ebi.ac.uk/swissprot/Submissions/spin/index.jsp for directly sequenced proteins, or to EMBL/GenBank/DDBJ where the nucleotide sequence is available:
EMBL: http://www.ebi.ac.uk/embl/Submission/webin.html
GenBank: http://www.ncbi.nlm.nih.gov/Genbank/submit.html
DDBJ: http://sakura.ddbj.nig.ac.jp/
If at a point in the future the sequence does appear in UniProt, the updated version of the sequence database match will be available to users through the SIFTS file generated at MSD-EBI and distributed as part of the wwPDB ftp archive. The PDB file will not be modified to add the sequence cross-reference. The PDB file will always contain the sequence cross reference available at the time of processing of the entry. Information about SIFTS is available from http://www.ebi.ac.uk/msd-srv/docs/sifts/

If related entries with the same sequence do not have UNP references, the DBREF (struct_ref, struct_ref_seq_dif) will self-refer to the PDB entry and not refer back to the first PDB entry which contained the particular sequence. For example, if entries 1ABC, 1DEF and 1GHI all had the same antibody sequence, the DBREF (struct_ref, struct_ref_seq) in entries 1ABC, 1DEF and 1GHI will refer respectively to 1ABC, 1DEF and 1GEH.

How are chimeras handled? Chimeras should be deposited as a single chain with one chain ID because they were expressed as one chain. Chimeras that were refined with different chain IDs should be deposited with one chain ID for all parts of the chimera, including any linker regions.

The sections of the chimera which match the UNP entry or entries will each have database reference tokens. The sections of the chimera which do not match the UNP entry or entries, such as a linker region, will be self-referenced. Information about this can be added to REMARK 999.

What is the SEQADV (struct_ref_seq_dif) record? SEQADV, struct_ref_seq_dif describes any rational disagreement between a sequence database and the sequence in the PDB file.

The discrepancies listed in the SEQADV records are annotated as engineered mutation(s), cloning artifact(s), variants(s), his tags, insertions, deletions, etc.

What are the various types of conflicts and how are they listed in SEQADV?

- ***Engineered mutation*** Difference between the PDB sequence and the UNP entry that were engineered will be listed in SEQADV as “engineered mutation.”

- ***Cloning artifact*** The term cloning artifact is reserved for instances where a sequence difference is introduced, as in a PCR experiment during cloning or by random mutation. These instances are rare.
- ***Modified residue*** Only instances where the parent of a modified residue does not match the sequence database reference will be listed here. For example, if THR is listed in the UNP entry, and the PDB sequence is MSE (selenomethionine), then two records would be generated: one SEQADV for MSE/THR with the explanation of "engineered mutation" and one MODRES for MSE/MET.
- ***Microheterogeneity*** If a residue has more than one identity at a particular residue number, this is called microheterogeneity. The residue which does not match to the UNP reference or the one which has the higher occupancy (if both residues do not match with the UNP reference) will be listed in the sequence (SEQRES). All the residues which do not match to UNP reference will be listed in SEQADV records with the explanation of "microheterogeneity"
- ***Conflict*** Sequence conflicts which are listed in the UNP reference will be listed here with the explanation of "conflict" and also described in REMARK 999.
- ***Extra N- and/or C-terminal residues*** (including leader sequences, HIS-tags, other kinds of tags and/or initiating methionine(s)
- ***Insertions*** in the middle of the sequence such as linkers
- ***Deletions***

What is not listed in SEQADV? The following are not listed in the SEQADV records:

- Synthetic peptides
- Peptide inhibitors
- Modified residues where the parent residue matches the database reference. MODRES records (pdbx_struct_mod_residue) will be created for modified residues if the residue is derived from a parent residue.

How is microheterogeneity/polymorphism handled? Polymer chains which have the same sequence must have the same chemistry (homogeneous). If one chain has microheterogeneity at one position, but not in the other chain at the same position, then these two polymer chains will be treated as two different polymers with two sequences. For homogeneous chains, the residue which does not match the corresponding UNP residue will be listed in the sequence (SEQRES), regardless of its occupancy. The sum total of occupancies of the different identities of the residues which displays microheterogeneity should be less than or equal to 1.

If the microheterogeneity involves more than 2 identities for that residue number and neither residue identity matches the UNP residue at the corresponding UNP location, the residue with the higher occupancy will be listed in the sequence (SEQRES).

If a residue has two identities at a particular residue number, where one identity is a modified residue and one is the unmodified form, then the modified residue is listed in the sequence. A SEQADV record will be generated for the modified residue and the UNP entry, with the explanation of "microheterogeneity". This is the exception to our earlier decision that SEQADV records would not be created for modified residues.

Alternate position indicators will be used in the coordinates. The residue listed in the sequence will be listed first in the coordinates and be labeled as alternate position A. The other identity will be assigned alternate B. If there is a third or fourth residue identity, or if

one of the identities has its own alternate conformations, these will be assigned alternate IDs in alphabetical order. In the mmCIF file, the microheterogeneity flag is indicated in _entity_poly_seq.hetero and_poly_seq_scheme.hetero.

A full explanation of the microheterogeneity for all residues at a particular residue number will be elaborated in REMARK 999.

Neighboring intra-chain cross-linked groups This section describes cases of neighboring intra-chain cross-linked typically involving 2 or more sequential amino acids that react with each other to form one "residue". For example, residues 63, 64, and 65 in the PDB entry 1yjf underwent a reaction to form a circularized tri-peptide chromophore (Figure 1).

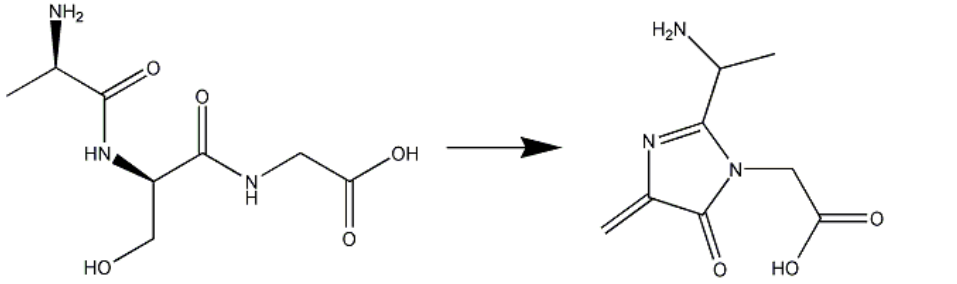

Figure 1: In entry 1yjf, residues 65, 66, and 67 (neighboring alanine, serine and glycine residues) reacted and generated a chromophore product called [2-(1-AMINOETHYL)-4-METHYLENE-5-OXO-4,5-DIHYDRO-1H-IMIDAZOL-1-YL]ACETIC ACID.

To describe this situation in a PDB entry the following annotation applies:

- The neighboring intra-chain cross-linked group will be listed as a chemical group in the coordinates and in the sequence (SEQRES).
- The neighboring intra-chain cross-linked group will have 3 parent residues, if three amino acids were involved in the reaction to make the chromophore. Thus, there will be 3 MODRES records.
- Usually the chemical name of the chromophore is too long to fit into the description part of the MODRES section in the PDB file. A summary name can be used instead, such as CIRCULARIZED TRI-PEPTIDE CHROMOPHORE.
- There will be SEQADV record(s) with the explanation ”chromophore”. Any additional information can be added to REMARK 999.

2 Ligands

The wwPDB encourages depositors to provide a chemical name and/or chemical drawing including bond type, bond order and stereochemistry of the ligand in order to facilitate the correct annotation.

How is the identity of a ligand verified? Specialized software is used to get the bond types, the stereochemistry, and where possible the IUPAC-compliant name for each ligand in the structure.

How is the ligand identity assigned? Each ligand is assigned a unique ligand code (up to 3 alphanumeric characters). Annotators use multiple methods to determine if a ligand already exists in the chemical component dictionary. If the stereochemistry and bond types match an existing ligand, the code for that ligand is used in the entry. If there is no match, the new ligand is added to the dictionary with an arbitrarily assigned ID code.

How are new ligands added to the chemical component dictionary? Several software programs are used to generate a chemical component cif file based on the author's deposited coordinates. The ligand's IUPAC-compliant name is predicted based on the deposited coordinates. All atoms in the component cif file (including H-atoms) will have coordinates. Checks for chemical name, ligand code and connectivity are carried out to verify that if the new ligand exists in the chemical component dictionary.

What happens if a structure contains a ligand that already exists in the chemical component dictionary? If the ligand code used by the depositor does not match the ligand definition in the chemical component dictionary, the annotator will correct the author's ligand code. The atom labels and ligand name will be automatically updated according to the dictionary. It is noted that the atom names for most ligands are not IUPAC-compliant names (amino acids and nucleotides are the exception).

Are polypeptide inhibitors treated as ligands? If the inhibitor is polypeptide like or if it has sequence database reference, it is treated as a peptide chain and REMARK 400 (pdbx_entity_annotation) can be added for additional information. Otherwise the inhibitor is treated as a ligand and REMARK 630 (_entity.details and pdbx_struct_chem_comp_feature) will be used to describe this inhibitor.
For details, see current Format Guide.

Charge state Whenever possible, the overall charge for new ligand definitions should be neutral. This provides maximum compatibility with other chemical databases such as CAS and PubChem. The overall charge of the ligand is included in the chemical component dictionary and also listed in the FORMUL record of the PDB format file.
The individual atoms may have a charge in the atom records.

Exceptions:

- Tetravalent nitrogen atoms where all four bonding partners are heavy atoms must carry a positive charge to satisfy valence rules
- Nitro groups (R-NO2) are represented in a charge separated state (R-[N+](=O)[O-]), again, to better satisfy valence rules.

Ligands in NMR structures produce unique challenges, because all hydrogen atoms used in the structure must be present in the dictionary. The accurate chemical description of molecules which may exist in multiple protonation states is difficult to achieve in single chemical component definitions. To describe this complexity for the standard amino acids and nucleotides, a special component dictionary has been created. Within this dictionary are complete chemical descriptions of observed protonation states which include accurate formulae, formal charges and IUPAC atom nomenclature for each case.

How are ligand names and synonyms assigned?
HETNAM (chem_comp.name) Whenever possible, the name automatically produced by specialized chemical naming software is used. Exceptions to this rule are common (as judged by annotation staff) biological names, and brand names for drugs. Should ACDlabs, Chemdraw or Pubchem fail to predict a name, common names or names supplied by the depositor may be used.

HETSYN (chem_comp.pdbx_synonyms) Other names requested by depositors may be included as synonyms, at the discretion of the annotator. It should be noted any synonyms provided by the depositor should be meaningful and widely used names for the ligand.

Exceptions / Issues: Established common names can be used as HETNAM as long as the IUPAC-compliant name is stored in HETSYN. Examples would be “Fluconazole” as HETNAM and “2-(2,4-difluorophenyl)-1,3-bis(1,2,4-triazol-1-yl)propan-2-ol” as HETSYN. If a ligand name is changed or synonyms are added, every PDB entry containing that ligand will be updated.

Metals There are difficulties in handling dative bonds in coordination complexes and pi-bonding in organometallic complexes. These inorganic molecules need to be built on a case by case basis to reflect their chemical nature. An ambiguous flag (chem_comp.pdbx_ambiguous_flag) is included in the chemical component files for such molecules where current cheminformatics software is inadequate in describing the true chemical structure.

A number of ligands that were defined as “metal bound to multiple waters” have been flagged in the dictionary with chem_comp.status = “DEL” (deleted) meaning they are no longer valid ligands, and should not be used in the future. While water coordinated metals can be strongly bound, these groups are not consistently used among all structures, and user analysis would benefit from standardizing on the single ion representation. Software has been written to split existing ligands into metal and waters, while renumbering the waters.

Redundant ligands Ligands with different 3-letter codes but identical chemical structures have been identified and marked as redundant, and their chem_comp.status set to “OBS” (obsolete). These ligands will no longer be used, and the chem_comp.pdbx_replaced_by token has been set to the ligand code that is to be used. Any deposition that uses obsolete ligand codes will be automatically replaced with the existing code. For example SUL will be updated to SO4.

Leaving groups A new token introduced in the new dictionary is the concept of a “leaving atom.” It is denoted in a definition by the token chem_comp_atom.pdbx_leaving_atom_flag, in the chemical component file and indicates a heavy atom that leaves as part of a

polymerization reaction. Common examples are the O1 of sugars, the OXT atom of amino acids, and the O3P of nucleotides. Some caveats:

- By default, any hydrogen atom can be a leaving atom. So while both the carboxyl OH group and the amide H leave during a protein polymerization, only the OXT atom is specifically marked as a leaving atom.
- Only a single heavy atom can be a leaving atom, multiple leaving atoms cannot be connected to create a "leaving group".

Portions of ligands that are missing Ligands are defined as if all atoms were present in the experiment. If the ligand is only partially seen in the experiment (for example, a ring is missing in the density for a crystal structure), the ligand code that is used is for the fully defined ligand. If the ligand is new, the missing atoms are added to the definition of the ligand.

Modified amino acids and nucleotides If an amino acid or nucleotide is modified by a chemical group greater than 10 atoms, the residue will be split into two groups: the amino acid/nucleotide group and the modification. A link record will be generated between the amino acid/nucleotide group and the modification.

This choice reflects our experience that modifications greater than 10 atoms typically have an independent chemical identity in the archive and may exist in bound and unbound forms.

Incorporation of large modifying groups into standard monomers also leads to complex and non-intuitive chemical names for the modified group.

Nucleotide residue nomenclature Currently, regardless of whether a nucleotide sequence is DNA or RNA, the residues are labeled as A,G,C,T(U), creating confusion for the user. The only difference between DNA and RNA is the presence of the O2' group on the ribose ring. This will be rectified by using the following residues for DNA: DA, DG, DC, DT and the following residues for RNA: A, G, C, U. We realize this represents a major change from current practice with possible large implications both internally and externally, but feel it will ultimately serve the users of the PDB to greater effect.

Use of UNX/UNL/UNK There are times when an amino acid residue, nucleotide, atom, or ligand is unknown. These ligand codes should be used in the following cases:

UNX: unknown atom or ion
UNL: unknown ligand
UNK: unknown amino acid
N: unknown nucleotide

UNX UNX is the code for one atom or ion, by itself, when author does not know the identity of that atom or ion. NOTE: The ligand name is UNX, but the atom name is UNK. The atom type is "X".

UNL UNL is the code for unknown ligand. This is for when the author knows the atom types but not the connectivity of a cluster or group of atoms. (If the author knew the connectivity, they could determine the identity of the ligand). If the author's entry contains a cluster of 5 oxygens and 2 carbons, and one phosphate, the atom names are O1, O2, O3, O4, O5, C1, C2, C3, P1, and the atom types are O, C, and P respectively. The code is UNL, and the residue number for each atom within a cluster would be the same.

UNK UNK is the code for unknown amino acid only. For example, a poly-ALA or poly-GLY chain would be processed as poly-UNK, if the author does not know how the coordinates align with the sequence and the residue numbering is arbitrary. The sequence would be poly UNK and the residues in coordinates would be listed as UNK. The sequence, if it is known, would be listed in the REMARK 999 and its mmCIF tokens. (If the authors do know the alignment of sequence and coordinates, the poly-ALA or poly-GLY residues should be changed to match the sequence). The atom names of UNK are N,CA,CB,CG,O,C, and the atom types are N,C,C,C,O,C. It is noted that we need to decide how to handle cases where atoms past the CG are seen but the amino acid identity is not known. We also need to decide how to handle breaks in density between UNK residues.

N is the code for unknown nucleotide.

What if the ligand identity provided by the depositor conflicts with the software and annotator identification of a ligand? Conflicts between the author's identification of a ligand and the software and/or annotator identification of a ligand are brought to the attention of the author. In case of disagreement between the depositor and the wwPDB staff, the ligand name will be based on what is derived from the coordinates by specialized chemical naming software. If the stereochemistry cannot be determined by a program, the author's description of the stereochemistry will be used. Any discrepancy between the deposited coordinates and the dictionary definition will be described in REMARK 600 and a flag will be added to the mmCIF file.

Errors in stereochemistry and geometry of a chemical component Any errors in the stereochemistry or geometry of the chemical components in a file will be listed in an mmCIF token and a description of the discrepancy will be added to REMARK 600.

3 Coordinate section

(see ATOM and HETATM)

Alternate conformations

How are alternate conformations of individual atoms, side chains, or entire residues handled? Sometimes an atom, several atoms, or an entire side chain has more than one set of conformation. Each set of coordinates for the atom is assigned alternate position A and B, or if there are three alternate positions, A, B, and C, etc. The combined occupancies of the alternate positions should not exceed 1.00. Generally alternate conformation A should have the higher occupancy.

Alternate conformations for same atom, same residue, and same atom type with the same coordinates are not allowed.

How are alternate conformations of chemical groups handled? Chemical group alternate conformations are handled in the same way as amino acids, unless the alternate conformations involve two different chemical groups.

What if the same chemical group is in alternate conformations? For example, a zinc ion labeled with residue number 100 is in two alternate conformations. Generally, the higher occupancy will be labeled as alternate position A, the lower occupancy labeled as B. The occupancies of the two conformations should be less than or equal to 1.00.

Example

```
HETATM 5255 CA  A CA A1677      24.997  -8.766   8.266  0.70  4.26          CA
HETATM 5256 CA  B CA A1677      25.089  -8.689   8.940  0.30  7.58          CA
```

What if two different chemical groups are in alternate conformations? The author may state that two chemical groups are in alternate conformations of each other, but have different identities. For example, the author may state that the chemical group at a particular location is both zinc and copper. Unlike polymorphic residues, the ligands can not be assigned the same residue number; different residue numbers must be assigned. In this example, zinc would be residue 100, and copper residue 101. Zinc would be assigned alternate conformation A, and copper would be assigned alternate conformation B. The ligand with the higher occupancy is generally, but not always, assigned alternate conformation A. The occupancies of each ligand should be less than 1.00, and combined should be less than or equal to 1.00.

Example

```
HETATM 2237  C1 AGLC A1145      24.054  15.397  10.953  0.50 23.40           C
HETATM 2238  C2 AGLC A1145      24.511  14.510  12.107  0.50 24.19           C
HETATM 2239  C3 AGLC A1145      24.469  13.035  11.755  0.50 23.62           C
HETATM 2240  C4 AGLC A1145      23.159  12.668  11.058  0.50 22.89           C
HETATM 2241  C5 AGLC A1145      22.775  13.655   9.964  0.50 22.60           C
HETATM 2242  C6 AGLC A1145      21.379  13.410   9.391  0.50 23.15           C
HETATM 2243  O1 AGLC A1145      24.965  15.362   9.884  0.50 25.48           O
HETATM 2244  O2 AGLC A1145      25.808  14.839  12.545  0.50 27.92           O
HETATM 2245  O3 AGLC A1145      24.599  12.347  12.979  0.50 23.35           O
HETATM 2246  O4 AGLC A1145      23.265  11.376  10.487  0.50 20.21           O
HETATM 2247  O5 AGLC A1145      22.790  14.942  10.534  0.50 22.57           O
HETATM 2248  O6 AGLC A1145      21.350  13.819   8.039  0.50 21.51           O
```

```
HETATM 2249  C1 BBGC A1146      22.835  15.552  11.694  0.50 28.27           C
HETATM 2250  C2 BBGC A1146      23.484  14.650  12.745  0.50 27.74           C
HETATM 2251  C3 BBGC A1146      23.616  13.199  12.308  0.50 26.77           C
HETATM 2252  C4 BBGC A1146      22.545  12.755  11.314  0.50 25.44           C
HETATM 2253  C5 BBGC A1146      22.324  13.751  10.180  0.50 25.87           C
HETATM 2254  C6 BBGC A1146      21.228  13.236   9.247  0.50 26.00           C
HETATM 2255  O1 BBGC A1146      23.216  16.914  11.644  0.50 29.46           O
HETATM 2256  O2 BBGC A1146      24.789  15.102  13.053  0.50 28.71           O
HETATM 2257  O3 BBGC A1146      23.586  12.419  13.484  0.50 27.44           O
HETATM 2258  O4 BBGC A1146      22.977  11.537  10.738  0.50 22.74           O
HETATM 2259  O5 BBGC A1146      21.961  15.012  10.707  0.50 27.65           O
HETATM 2260  O6 BBGC A1146      20.690  14.271   8.459  0.50 24.91           O
```

Handling OXTs Terminal oxygen, OXT, should be present on the final residue of a protein or peptide sequence. Some refinement programs use an OXT atom to denote the last atom of any polypeptide chain. Thus OXT atoms are added to

- the last residue before a gap in the chain (indicating a chain break) or
- the last observed residue in chain even though the residue is not the final residue of the sequence.

Inclusion of such OXT atoms in the middle of a polypeptide sequence does not describe the true contents of the crystal and is chemically incorrect.

The wwPDB corrects the OXT problem in one of the following 2 ways:

- Removing the OXT atom if it is not on the terminal residue of the sequence or
- Renaming the OXT to N of the next residue.

Authors are notified of how the annotation staff handled this case. If this atom is retained, the OXT will be changed to N of the following residue and the following remark is copied into REMARK 3, OTHER REFINEMENT REMARKS (refine.details):

“Due to a feature in the refinement program, the structure was refined with OXT on one or more residues that are not the terminal residues of the sequence. In all these instances the OXT was changed to N of the next residue.”

Any resulting C-N deviations will be removed from remark 500, because the atom was refined as oxygen, not nitrogen.

Zero occupancy residues (REMARK 475) **and atoms** (REMARK 480)
Missing residues (at the N- or C-termini or in flexible loops) and missing side chain atoms of the polymer component are listed in REMARK 465 (pdbx_unobs_or_zero_occ_residues with polymer_flag (Y) and occupancy_flag (0)) and REMARK 470 (pdbx_unobs_or_zero_occ_atoms with polymer_flag (Y) and occupancy_flag (0)), respectively, of the PDB format file and corresponding mmCIF tokens. Atoms which are leaving atoms such as polymer linkage (OXT in amino acids, OP3 in nucleic acids, O1 in saccharides) and hydrogens will not be listed as missing atoms in REMARK 470 and REMARK 480.

Some refinement programs allow inclusion of missing residues and side chain atoms in the coordinate files as atoms with occupancy 0.00. Since these atoms are usually ignored during refinement, their location and properties may not be reliable. Therefore, if such atoms are

included in the deposited coordinate file, these atoms will still be retained in the file but also listed in separate new remarks: REMARK 475 (for zero occupancy residues) and REMARK 480 (for zero occupancy atoms). These remarks will also be available in the corresponding mmCIF file. The text in these remarks will be as follows:

For zero occupancy residues (pdbx_unobs_or_zero_occ_residues with polymer_flag (Y) and occupancy_flag (1)):

```
REMARK 475
REMARK 475 ZERO OCCUPANCY RESIDUES
REMARK 475 THE FOLLOWING RESIDUES WERE MODELED WITH ZERO OCCUPANCY)
REMARK 475 THE LOCATION AND PROPERTIES OF THESE RESIDUES MAY NOT
REMARK 475 BE RELIABLE. (M=MODEL NUMBER; RES;
REMARK 475 C=CHAIN IDENTIFIER; SSSEQ=SEQUENCE NUMBER; I=INSERTION CODE)
REMARK 475
REMARK 475   M RES C SSSEQI
REMARK 475     MET A     1
REMARK 475     ALA A     2
```

For zero occupancy atoms: (pdbx_unobs_or_zero_occ_atoms with polymer_flag (Y) and occupancy_flag (1))

```
REMARK 480 ZERO OCCUPANCY ATOM
REMARK 480 THE FOLLOWING RESIDUES HAVE ATOMS MODELED WITH ZERO
REMARK 480 OCCUPANCY. THE LOCATION AND PROPERTIES OF THESE ATOMS
REMARK 480 MAY NOT BE RELIABLE. (M=MODEL NUMBER;
REMARK 480 RES=RESIDUE NAME; C=CHAIN IDENTIFIER; SSEQ=SEQUENCE NUMBER;
REMARK 480 I=INSERTION CODE):
REMARK 480   M RES C SSEQI  ATOMS
REMARK 480     GLU A   42   CG    CD    OE1   OE2
REMARK 480     GLN A   44   CG    CD    OE1   NE2
```

Ligands or hetgroups that are not part of any polymer (protein or nucleic acid) in the structure may also have missing atoms or atoms with zero occupancy. In such instances the name of the hetgroup or ligand, chain ID and model number (if applicable) will be listed in REMARK 610 (for missing atoms) or REMARK 615 (for atoms with 0.00 occupancy). Corresponding mmCIF categories listing this information will also be included in the mmCIF format file. As the list of specific atoms missing from a hetgroup may be really large, they will not listed in the remarks described above. The list of all missing atoms from the ligands may be easily derived by comparing the coordinates of the hetgroup to its definition in the ligand dictionary.

For non-polymer component with missing atoms (REMARK 610, pdbx_unobs_or_zero_occ_atoms with polymer_flag (N) and occupancy_flag (0))

```
REMARK 610
REMARK 610 MISSING HETEROATOM
REMARK 610 THE FOLLOWING RESIDUES HAVE MISSING ATOMS (M=MODEL NUMBER;
REMARK 610 ALT= ALT CODE; RES=RESIDUE NAME; C=CHAIN IDENTIFIER;
REMARK 610 SSEQ=SEQUENCE NUMBER; I=INSERTION CODE):
REMARK 610   M RES C SSSEQI
REMARK 610     GL5 A    42C
REMARK 610     GL7 A    44
```

For non-polymer component with zero occupancy atoms (REMARK 615, pdbx_unobs_or_zero_occ_atoms with polymer_flag (N) and occupancy_flag (1))

```
REMARK 615 ZERO OCCUPANCY ATOM
REMARK 615 THE FOLLOWING RESIDUES HAVE ATOMS MODELED WITH ZERO
```

```
REMARK 615 OCCUPANCY. THE LOCATION AND PROPERTIES OF THESE ATOMS
REMARK 615 MAY NOT BE RELIABLE. (M=MODEL NUMBER; ALT= ALT CODE
REMARK 615 RES=RESIDUE NAME; C=CHAIN IDENTIFIER; SSEQ=SEQUENCE NUMBER;
REMARK 615 I=INSERTION CODE):
REMARK 615   M RES C SSSEQI
REMARK 615     GL5 A    42
REMARK 615     GL7 A    44
```

4 Chain ID assignment

(see ATOM/HETATM record)

Definition of chain ID The chain ID is a unique identifier for each macromolecular polymer and all chemical groups (including waters) associated with it.

Which moieties are assigned chain IDs? All atoms in the coordinate section of the PDB file will be assigned a chain ID.

Why are chain IDs assigned in this way? The wwPDB has established this rule to improve the usability and interpretation of the structural data. Assigning one chain ID for all moieties associated with a polymer enables rapid and uniform identification of feature analysis.

How will chain IDs be assigned? Each polymer is assigned a unique chain ID. Chain IDs for all bound moieties and waters are assigned based on their proximity (number of contacts) to the nearest polymer. For example, all waters and chemical groups around a particular polymer are assigned the chain ID of the polymer they surround. Sugars are considered as ligands even if they have 2 or more covalently linked sugars. Therefore the mono and poly sugars are also assigned the chain ID of the polymer they surround.

How are chain IDs assigned to chemical groups and waters? All chemical groups and waters within 5 Angstroms of any atom of a polymeric chain will be considered to be associated with that chain and will be assigned the same chain ID as that polymer. If a particular chemical group/water is equidistant from more than 1 chain, then the chain ID is randomly assigned to be that of any one of these polymers.

Waters further than 5 Ångstroms away from the polymer that can be moved by symmetry to within 5 Ångstroms, will then be automatically moved and the author will be notified. If there are objections, the waters will be moved back to their original positions.

Waters further than 5 Ångstroms away from any polymer, which cannot be brought closer to a polymer chain by application of symmetry, will be brought to the attention of the depositor. The waters will be listed in REMARK 525 with the distance to the nearest macromolecule listed. Authors will be given the opportunity to remove the waters or update the coordinates. Authors may choose to retain the distant waters in the entry.

```
REMARK 525
REMARK 525 SOLVENT
REMARK 525
REMARK 525 THE SOLVENT MOLECULES HAVE CHAIN IDENTIFIERS THAT
REMARK 525 INDICATE THE POLYMER CHAIN WITH WHICH THEY ARE MOST
REMARK 525 CLOSELY ASSOCIATED.  THE REMARK LISTS ALL THE  SOLVENT
REMARK 525 MOLECULES WHICH ARE MORE THAN 5A AWAY FROM THE
REMARK 525 NEAREST POLYMER CHAIN ( M=MODEL NUMBER;
REMARK 525 RES=RESIDUE NAME; C=CHAIN IDENTIFIER; SSEQ=SEQUENCE
REMARK 525 NUMBER; I=INSERTION CODE):
REMARK 525
REMARK 525  M RES CSSEQI
REMARK 525  0 HOH A 561      DISTANCE =  5.07 ANGSTROMS
REMARK 525  0 HOH A 791      DISTANCE =  5.08 ANGSTROMS
```

How are chain IDs related to residue numbering? All residues and chemical groups in a file should be uniquely identified. Once the polymers and chemical groups associated with them are assigned chain IDs, the numbering of all residues, chemical groups and waters for each chain ID must be unique. Numbering of residues that were not modelled due to limited experimental data should also be considered here. The wwPDB encourages deposition of polymer chains with sequential residue numbering. For protein chains, the authors are encouraged to follow the UniProt residue numbering, wherever possible. The use of non-sequential residue numbering and insertion codes should be avoided as far as possible in order to make structures easily interpretable by the larger scientific community. If the coordinate residue numbers, as provided by the author, are unique and sequential within a particular chain ID, the residues will not be renumbered. If the author has already sorted ligands to polymer chains with which they are associated, these are not reassigned.

What is the maximum number of chain IDs in a file? Up to 62 chains can be included in the PDB entry. Upper case letters and numbers should be used first for chain IDs. Lower case letters should be used only after all upper letters and numbers 0-9 have been used. Symbols should never be used for chain IDs.

Consistent use of chain IDs in PDB structures and manuscripts During the annotation of structures the numbering and chain IDs of the chemical groups and waters associated with the polymeric chains may be changed in accordance with the wwPDB chain ID rules. The wwPDB strongly encourages depositors to use the wwPDB-assigned chain ID and residue numbers in any publication material. Deposition and processing of structures prior to preparation of the manuscript will ensure consistent usage of chain IDs and residue numbers in the manuscript and the PDB file. Validation and structure analysis reports generated during annotation may also be helpful in the manuscript preparation.

5 HEADER assignment

How is the HEADER record (function) assigned? The HEADER (struct_keywds.pdbx_keywords) is used to briefly describe the broad function of the macromolecules present in the PDB entry. A list of classifications is available.

See Appendices for the header list.

Two or more HEADERs can be combined using the following rules:

- For macromolecular complexes, the headers are separated with a forward slash (HEADER1/HEADER2). e.g. TRANSCRIPTION/DNA (Note that protein is listed first in protein/nucleic acid complexes).
- The word "complex" will not be used in the header record but will appear in the keywords record whenever there is a "/" in the header.
- A multifunctional macromolecule will have commas between the different headers: HEADER1,HEADER2
- Functions such as inhibitor, activator, receptor of a macromolecule can be added to existing headers (HEADER INHIBITOR) e.g. HYDROLASE INHIBITOR, HYDROLASE ACTIVATOR, HYDROLASE RECEPTOR, HYDROLASE REGULATOR.

Header functions are assigned by the annotator The annotator typically assigns the function after reviewing the UniProt keywords and keywords provided by the author. The keywords should include the header and if the header is a complex (/ separated), the word complex will be added in the keywords. If the protein is an enzyme, the general class of enzyme is used. For example, the header is assigned as oxidoreductase if the E.C. number starts with 1.

- If protein has no known function, "UNKNOWN FUNCTION" is used.
- If the structure is involved in structural genomics and the function is not known, the header "STRUCTURAL GENOMICS, UNKNOWN FUNCTION" is used.
- If the function is putative, such as a "putative hydrolase" the header will be assigned based on the putative assignment, i.e. "HYDROLASE".
- If the annotator is unsure of the function, the annotator asks the author to choose the appropriate header from within the standard header list.
- If the function is new to the PDB, a generalized header describing the function will be added to the standard header list based on UniProt Keywords if available and the list will be exchanged among the 3 wwPDB sites.

6 Compound information

(see COMPND, SOURCE and KEYWDS records)

How are the molecule name and synonyms assigned?

Protein names Protein molecule names and synonyms are copied from the corresponding UniProt entry and inserted in the MOLECULE and SYNONYM field of the COMPND section (entity.pdbx_description and entity_name_com.name). Exceptions are as follows:

- If the UNP name refers to a precursor and the entry contains the mature protein, the word precursor is removed for the PDB entry. If it includes putative, then the wording will be kept.
- In the case of zymogens, if the UNP name is “trypsinogen” but the activated protein is present in the structure, then trypsin is used as the name.
- UniProt contains the complete gene sequence. If the final product is made of more than one chain, the corresponding polypeptide name will be given as the name in the PDB entry. For example, if the UNP name is insulin, the protein names correspond to the chain names: such as Insulin alpha 1 and Insulin alpha 2
- For viral proteins where a polyprotein is synthesized, if the PDB entry contains all the components, then the name polyprotein can be used. Otherwise the name of the fragment from UNP will be used. For example if a “genome polyprotein” is composed of capsid, envelope protein, and major envelope proteins but the deposited structure is only of the envelope protein, the name used will be “envelope protein”.
- If the full UNP sequence for the protein is not present in the sample, the fragment section will be filled in. However, the fragment section will not be filled in where complete mature protein is represented, for example for a complete trypsin molecule or envelope protein etc.
- If the UNP name is not assigned (i.e. it is “hypothetical”) and the author has a name for the protein, then the author’s protein name will be used.
- If the author’s name for the molecule differs from the UNP molecule name, UNP primary name is the molecule name. The author’s molecule name is listed first in the synonyms list, followed by the synonyms provided in UNP.
- If there is no corresponding UNP entry and a UNP entry can not be created for it, the author’s protein name is used.
- Antibodies will be named using as much information as the author provides.
  Examples:
  (i) Fab fragments will be named as Fab heavy chain and Fab light chain, respectively.
  (ii) For immunoglobulin G chains (which include fab and fc regions), the protein names will be antibody heavy chain and antibody light chain.
  (iii) If the author provided specific names for the antibody, then those names will be used e.g. Fr62 monoclonal antibody light chain and Fr62 monoclonal heavy chain.
- For chimeric proteins, the protein name is comma separated and may refer to the presence of a linker (protein_1, linker, protein_2). Other details about the chimera can be mapped to OTHER_DETAILS (entity.details) and REMARK 999.

Nucleic acid molecule names For all nucleic acid sequences, a biological name should be used when available. The biological name is either provided by the author or it is obtained from a sequence database, such as "16S RIBOSOMAL RNA".

If the sequence is shorter than 24 nucleotides and no biological name is available, the molecule name is given as the short sequence:

5'-D(*CP*GP*CP*GP*(8OG)P*AP*TP*TP*CP*GP*CP*G)-3'

For sequences greater than or equal to 24 nucleotides, and no biological name, then the name may be something such as: 50-mer.

Typically, “fragments” are not used for nucleic acids. For example, the name "Loop E from 5S RNA" is represented as molecule: “5S RNA” but the fragment is NOT represented as “fragment: Loop E”.

How is fragment information indicated? (COMPND, entity_keywords.pdbx_fragment) A protein fragment is a sequence where one or more residues are missing compared to the corresponding sequence in the Uniprot entry.

If the PDB entry contains a fragment of the protein in the sequence records when compared to the UniProt entry, the fragment field will also record the residue numbers listed as per the numbering in the corresponding Uniprot entry. If additional annotation is available for the complete fragment, for example annotation about ligand binding domain, this will be added to the fragment field e.g. “ligand binding domain, UNP residues 100-200” where the residue numbers are the UniProt numbers. The numbering from Uniprot is universal and sequential and is preferred.

Any further information available on the fragment or provided by the authors will be listed in OTHER_DETAILS field in the COMPND section (entity.details).

How is the EC number assigned (COMPND, E.C., entity_keywords.pdbx_ec)? The EC number is automatically extracted from the UNP entry. If the author disagrees with the UNP assignment of EC number, the wwPDB staff will contact the Uniprot staff and try to clarify the matters. In the case where the matter can not be resolved the EC assignment will be removed and added to the OTHER_DETAILS field in the COMPND section and will be indicated as “Author provided EC number is xxxx”.

How are mutations indicated (COMPND, MUTATION, entity_keywords.pdbx_mutation)? Mutations are indicated in the COMPND section as “COMPND MUTATION: YES”. The exact mutation will be described in the mmCIF file using numbering from the Uniprot entry, i.e. Y20K. The SEQADV records will contain the annotation “ENGINEERED MUTATION” which indicates the mutation(s), furthermore extra information will be added to the REMARK 999.

How is the source indicated? The NCBI_TAXID is added to the source section of the PDB file and mmCIF tokens are created for the tax ID.
The source organism is indicated by the taxonomy id as listed in the NCBI Taxonomy database. The official scientific name, ORGANISM_SCIENTIFIC, _entity_src_gen.pdbx_gene_src_scientific_name fields of the PDB entry, is obtained from UNP database which is based on the NCBI Taxonomy id. If a common name is listed in the NCBI taxonomy database, then the common name is mapped to the ORGANISM COMMON, _entity_src_gen.gene_src_common_name field, otherwise this will be left blank. An exception to this rule is listed below. If an author wishes to provide a synonym for the

scientific name, the name is mapped to SOURCE, OTHER_DETAILS, _entity_src_gen.pdbx_description. If the NCBI Taxonomy database name is “Escherichia coli (strain K12)”, then “Escherichia coli” would map to scientific name and “K12” would map to strain. Note that the scientific name of the source and host organisms will be included in mixed case to match with standard scientific literature. Plasmid and gene names will also be represented similarly.

The scientific names of chimeric proteins are listed as a comma separated list.
If an expression system was used, the expression system scientific name will be taken from the NCBI Taxonomy database. Other information about the source and expression system is not mandatory and will be included in the file only if the author has provided it.
Tax ID for synthetic is 32630
Tax ID for undefined is 32644
Tax ID for hybrid is 37965

Phage display Information about phage display can be mapped to source, "OTHER_DETAILS".

Cell-free synthesis, in vitro transcription and in vitro translation Cell-free synthesis, in vitro transcription and in vitro translation will all be described as "cell-free synthesis". All will have genetically manipulated sources and will be listed as expression system: cell-free synthesis in the PDB file. Other information about the synthesis, such as “wheat germ” can be added to the OTHER_DETAILS.

Baculovirus If a baculovirus was used, it will be listed under vector type. If cell line is provided, it will be added to the entry.

Synthetic The term “synthetic” was clarified to mean "synthesized using non biological methods".

How are keywords (KEYWDS, _struct_keywords.text) assigned? Keywords are provided by the author. Keywords are also added from the corresponding UNP entry. Some of the UNP terms such as signalling or 3-D structure are not included, and redundant terms between the author’s keywords and UNP are removed. If the author does not agree with the UNP keywords and/or if there is a valid scientific argument for removing the UNP keywords from the PDB entry, then the UNP keywords will be removed. The HEADER should be included in the keywords list. In the case of a complex (as indicated by a / in the HEADER), the word “complex” will be added to the keywords.

7 Author information

Who should be indicated as an author for the PDB entry (AUTHOR, audit_author.name)? The authors for a PDB entry can be the same as the author list for the primary citation, or a subset of citation authors. Alternatively, there may be more authors listed for the entry compared to the citation author. Generally, at least one of the authors for the entry should be included in the author list for the primary citation.

The authorship of the entry is at the discretion of the principal investigator (PI). If more than one PI is responsible for the entry, they will need to come to a mutual decision on the authorship.

See section 11 for author information for structural genomics structures

8 Citation information

(JRNL records)

Authors are encouraged to deposit their structures in advance of publication. The primary citation is the paper that describes the structure in the PDB entry.

Thesis Conference Proceedings and Thesis can only be included as primary citation, but not in reference citation.

PubMed Ids PubMed IDs are available for the primary citations of entries in the PDB, mmCIF and XML files. DOI numbers are also included.

Unpublished If the author indicates that the entry will never be published, the journal section is not included in the PDB entry. An mmCIF token will be added to the mmCIF file to flag situations when the author has indicated the entry will never be published.

Jr. Journal author names that have the suffix “junior” will be represented as “Jr.” and not as “junior”.

Title case All titles, names etc. are included in mixed case in the mmCIF format file to match with the literature. Although the legacy PDB format files will have titles in upper case, in future the PDB format files will also have titles in mixed case.

9 REMARKs

Numbered remarks, including validation & biological unit

REMARKs 40 and 42- other programs used for validation Use of REMARKs 40 and 42 is discontinued.

There will be new validation package developed by the software community through Validation Task Force. Therefore any other programs used for validation will not be listed in the PDB or mmCIF files.

REMARK 100 Processing sites
NDB id has been removed from this remark. This remark now indicates one of the four process sites: RCSB, PDBe, PDBj or BNL with uniform date and site id code with exception of BNL entries which will not have process date and site id code.

REMARK 300 and REMARK 350 Biological unit and quaternary assembly These REMARKs describe quaternary structure which may include software calculated quaternary assembly and/or author determined biological unit (also called the biological assembly), biologically relevant form of the molecule for which there is experimental evidence. These remarks do not apply to solution NMR entries.

Quaternary structure indicates the way in which many protein chains associate with one another. For example, hemoglobin consists of four protein chains of two slightly different types, all attached to an iron atom. In general two or more polypeptide chains that behave in many ways as a single structural and functional entity are said to exhibit quaternary structure. The separate chains are not linked through covalent chemical bonds but by weak forces of association. The quaternary structure is the shape or structure that results from the interactions of more than one protein molecule, usually called protein subunits in this context, which function as part of the larger assembly or protein complex. In biochemistry, quaternary structure is the arrangement of multiple folded protein molecules in a multi-subunit complex.

Many proteins are actually assemblies of more than one polypeptide chain, which in the context of the larger assemblage are known as protein subunits. In addition to the tertiary structure of the subunits, multiple-subunit proteins possess a quaternary structure, which is the arrangement into which the subunits assemble. Enzymes composed of subunits with diverse functions are sometimes called holoenzymes, in which some parts may be known as regulatory subunits and the functional core is known as the catalytic subunit. Examples of proteins with quaternary structure include hemoglobin, DNA polymerase, and ion channels. Other assemblies referred to instead as multiprotein complexes also possess quaternary structure. Examples include nucleosomes and microtubules. Changes in quaternary structure can occur through conformational changes within individual subunits or through reorientation of the subunits relative to each other. It is through such changes, which underlie cooperativity and allostery in "multimeric" enzymes, that many proteins undergo regulation and perform their physiological function.

The above definition follows a classical approach to biochemistry, established at times when the distinction between a protein and a functional, proteinaceous unit was difficult to elucidate. More recently, people refer to protein-protein interaction when discussing

quaternary structure of proteins and consider all assemblies of proteins as protein complexes.

We derive a likely oligomeric state of the structure based on the surface area of interactions and the association of macromolecules. Frequently, this derived structure and the biological unit are the same. However, due to crystal packing forces, experimental evidence, or author opinion, the biological unit and the PDB derived oligomeric structure may be different. For example, a macromolecule may be assigned a hexameric quarterary structure in the crystal but the biological unit may be monomeric. It is recognized that quaternary structure programs such as PISA[2] and PQS[3] do not work with every case, such as antibodies, or any case where a biological process has a low association/dissociation property. The quaternary assembly is calculated and evaluated by the annotation staff, while the biological unit is provided by the author.

Each chain must be used once at the identity in building the biological unit and/or quaternary structure. If necessary, the wwPDB staff will move the coordinates of the chain(s) so that every chain is used at identity. There is the potential for a PDB entry to have more than one oligomeric state.

The wwPDB is adding information about quaternary structure to make it more convenient for the larger scientific community to understand the quaternary assembly of the protein in any given experiment which could be different from the assumed biological unit.

Quaternary structure will be calculated automatically, based on what is in the coordinate file and known about the macromolecule. The matrices forming the quaternary structure will be reported as BIOMT in REMARK 350 and will be assigned by the wwPDB annotators. In instances where the author's description of the biological unit disagrees with what the crystal structure appears to present, the biological unit can be chosen by the depositor, and reported in the file. However, inclusion of this in the file is NOT MANDATORY and in most cases it is likely to be identical to the quaternary structure. The total surface area, buried surface area and free energy gain will be listed if two polymers have an interface. If more than two proteins have an interface, the remark will list the "average buried surface area".

Nomenclature: The number of subunits in an oligomeric complex are described using names that end in -mer (Greek for "part, subunit"). Formal Greco-Latinate names are generally used for the first ten types and can be used for up to twenty subunits, whereas higher order complexes are usually described by the number of subunits, followed by -meric.

In PDB REMARK 300 and REMARK 350, any polypeptide of length of 3 or more amino acids or 2 or more nucleotides is considered in the naming of a quaternary structure as monomeric or dimeric etc.:

1 = monomeric	8 = octameric	15 = pentadecameric
2 = dimeric	9 = nonameric	16 = hexadecameric
3 = trimeric	10 = decameric	17 = heptadecameric
4 = tetrameric	11 = undecameric	18 = octadecameric
5 = pentameric	12 = dodecameric	19 = nonadecameric
6 = hexameric	13 = tridecameric	20 = eicosameric

[2] E. Krissinel and K. Henrick (2005). Detection of Protein Assemblies in Crystals. In: M.R. Berthold et.al. (Eds.): CompLife 2005, LNBI 3695, pp. 163--174. Springer-Verlag Berlin Heidelberg.
[3] Henrick, K., and J. M. Thornton. 1998. PQS: a protein quaternary structure file server. Trends. Biochem. Sci. 23:358-361.

7 = heptameric 14 = tetradecameric 21-meric etc.

Please note that the multi-mer described in PDB remark 350 represents either homo or hetero multi-mer for that entry.

For examples, see Format Guide.

REMARK 400 (COMPOUND, **_pdbx_entry_details.compound_details)** REMARK 400 may be used to annotate additional functional details of the compounds. This information is not mandatory and will not be verified by the wwPDB. For peptide inhibitor which is treated as polymer, _pdbx_entity_annotation will be used to map to REMARK 400.

REMARK 500, VALIDATION, Calculation of bond, angle, torsion deviations, etc. The calculation of bond and angle deviations for protein entries will be based on the updated Engh & Huber amino acid target values[4]. For nucleic acids, the Parkinson et al., statistics are used for these calculations[5]. All bonds and angles that deviate more than 6 times from their standard target values will be flagged as a deviation. The PHI/PSI values are based on Kleywegt's calculations[6].

REMARK 500
CLOSE CONTACTS IN SAME ASYMMETRIC UNIT (pdbx_validate_close_contact)
SYMMETRY RELATED CLOSE CONTACTS (pdbx_validate_symm_contact)
BOND LENGTHS (pdbx_validate_rmsd_bond)
BOND ANGLES (pdbx_validate_rmsd_angle)
TORSION ANGLES (pdbx_validate_torsion)
NON-CIS, NON-TRANS (pdbx_validate_peptide_omega)
SIDE CHAIN PLANAR GROUPS (pdbx_validate_planes)
MAIN CHAIN PLANARITY (protein only) (pdbx_validate_main_chain_plane)
CHIRAL CENTERS (protein C-alpha only) (pdbx_validate_chiral)

REMARK 620 HETEROGEN REMARK 620 contains any additional annotation specific to the heterogens present in the structure. By default, software adds coordination angles for any metal coordination and surrounding residues (if present) in REMARK 620. Any other information about the ligand may be entered in REMARK 600 (_pdbx_entry_details.nonpolymer_details).

For examples, see Format Guide.

REMARK 630 INHIBITORS Details of inhibitors/peptide inhibitors which are presented as a single molecule (het group) are provided in REMARK 630. By default, molecule type and inhibitor's name will be provided in this REMARK.

For details, see Format Guide.

REMARK 650, HELIX We encourage authors to use the calculated helix records and not provide their own remarks. HELIX records which have been provided by the author will have

[4] Structure quality and target parameters. R. A. Engh and R. Huber. International Tables for Crystallography (2006). Vol. F, ch. 18.3, pp. 382-392
[5] "New Parameters for the Refinement of Nucleic Acid Containing Structures." Gary Parkinson, Jaroslav Vojtechovsky, Lester Clowney, Axel Brunger*, and Helen M. Berman. (1996) Acta Cryst. D 52, 57-64
[6] "PHI/PSI- Chology: Ramachandran revisited. " GJ Kleywegt and TA Jones (1996) Structure 4, 1395-1400.

a remark added to the header REMARK 650 to indicate the helix records have been author provided.

```
HELIX
DETERMINATION METHOD:  AUTHOR PROVIDED.
```

REMARK 700, SHEET We encourage authors to use the calculated sheet records and not provide their own remarks. SHEET records which have been provided by the author will have a remark added to the header REMARK 700 to indicate the sheet records have been author provided.

```
SHEET
DETERMINATION METHOD:  AUTHOR PROVIDED.
```

REMARK 800, SITE record commentary REMARK 800 is used to annotate the binding environment of any non-polymeric heterogen. The site identifiers provided in these records are listed in the corresponding SITE records. This information may be provided by the depositors.

REMARK 900, related entries Authors may provide REMARK 900 (pdbx_database_related), related entries, to relate other entries to the current entry. The related entries will not be automatically assigned.

REMARK 0, re-refinements of another author's data The following text is for a dedicated remark for cases where an author re-refines another author's data (REMARK 0).The remark would always appear in entries where the author refined someone else's data. The entry would be treated as an experimental structure.

```
REMARK   0
REMARK   0 THIS ENTRY yyyy REFLECTS AN ALTERNATIVE MODELING OF THE
REMARK   0 ORIGINAL STRUCTURAL DATA (RxxxxSF or xxxx.MR) DETERMINED BY
REMARK   0 AUTHORS OF THE PDB ENTRY xxxx:
REMARK   0 AUTHOR INITIALS, AUTHOR LAST NAME
```

Note: In entries where REMARK 0 is included as described above, remarks REMARK 1, REMARK 200 and REMARK 900 will also be annotated as show in the example below:

For the entry 1ZET,

- PDB entry title: "X-Ray Data Do not Support Hoogsteen Base-Pairing During Replication by Human Polymerase Iota":
- Reference 1 includes the original author's paper

```
REMARK   1 REFERENCE 1
REMARK   1  AUTH   D.T.NAIR,R.E.JOHNSON,S.PRAKASH,L.PRAKASH,
REMARK   1  AUTH 2 A.K.AGGARWAL
REMARK   1  TITL   REPLICATION BY HUMAN DNA POLYMERASE-I OCCURS BY
REMARK   1  TITL 2 HOOGSTEEN BASE-PAIRING
REMARK   1  REF    NATURE                        V. 430   377 2004
REMARK   1  REFN   ASTM NATUAS  UK ISSN 0028-0836
```

- REMARK 200

```
REMARK 200 REMARK: AUTHOR USED THE SF DATA FROM ENTRY XXXX.
```

(NOTE: rest of REMARK 200 is blank, since the re-refinement author did not collect the original data)

- Related entries

```
REMARK 900 RELATED ENTRIES
REMARK 900 RELATED ID: R1T3NSF   RELATED DB: PDB
```

```
REMARK 900 THIS ENTRY 1ZET REFLECTS AN ALTERNATIVE MODELING OF X-RAY
REMARK 900 DATA R1T3NSF
```

- The SF file has the following, with additional information on what the author added:

_audit.revision_id 1_0
_audit.creation_date 2005-07-19
_audit.update_record
;Initial release, author used sf file from pdb entry 1t3n, and added columns Fcalc, phases and FOM
;

File Format Guide

The version of the PDB file and its correspondence to the file format guide will be included in all files processed and released by the wwPDB.

10 Miscellaneous records

(SITE, LINK, SSBOND, HELIX, SHEET)

REVDAT records REVDAT records contain a history of the modifications made to an entry since its release.

Record Format

COLUMNS	DATA TYPE	FIELD	DEFINITION
1 - 6	Record name	"REVDAT"	
8 - 10	Integer	modNum	Modification number.
11 - 12	Continuation	continuation	Allows concatenation of multiple records
14 - 22	Date	modDate	Date of modification (or release for new entries). This is not repeated on continuation lines.
24 - 28	String(5)	modId	Identifies this particular modification. It links to the archive used internally by PDB. This is not repeated on continuation lines
32	Integer	modType	An integer identifying the type of modification. In case of revisions with more than one possible modType, the highest value applicable will be assigned
40 - 45	LString(6)	record	Modification detail.
47 - 52	LString(6)	record	Modification detail.
54 - 59	LString(6)	record	Modification detail.
61 - 66	LString(6)	record	Modification detail.

Each time revisions are made to the entry, a modification number is assigned in increasing (by 1) numerical order. REVDAT records appear in descending order (most recent modification appears first). New entries have a REVDAT record with modNum equal to 1 and modType equal to 0. Allowed modTypes are:

0 Initial released entry.
1 Other modification.

Each revision may have more than one REVDAT record, and each revision has a separate continuation field.

Modification details are typically PDB record names such as SOURCE, TITLE, or COMPND. A special modification detail VERSN indicates that the file has undergone a change in version. The current version will be specified in REMARK 4.

Verification/Validation/Value Authority Control

The modType must be one of the defined types, and the given record type must be valid. If modType is 0, the modId must match the entry's ID code in the HEADER record.

Relationships to Other Record Types

In the case of a version revision, the current will be specified in REMARK 4.

Template

```
         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234567890
REVDAT   2   15-OCT-99 1ABC    1       REMARK
REVDAT   1   09-JAN-89 1ABC    0


         1         2         3         4         5         6         7
12345678901234567890123456789012345678901234567890123456789012345678901234567890
REVDAT   2   11-MAR-08 2ABC    1       JRNL   VERSN
REVDAT   1   09-DEC-03 2ABC    0
```

Annotation of SITE records The SITE records supply the identification of groups comprising important sites in the macromolecule. Historically SITE records have been only used to annotate the catalytic residues in an enzyme. In new PDB entries, SITE records will define any interacting residues, based on distance. An evidence code (_struct_site.pdbx_evidence_code) has been added to identify whether the SITE records is software calculated or author provided.

The SITE records are created to annotate the residues in the environment of all free floating ligands and hetgroups (not around modified residues). These records can be created automatically using a script that selects all protein, water and hetgroup atoms within a specified distance of any atom of a ligand. Site identifiers are 3-letter codes in a character range of AC1-ZZ9 if it is software determined.
as shown in PDB entry 1W3M.

Software generated SITE records
In Jan 2009 the wwPDB annotation software is using software 'getsite.f'. getsite is a Fortran program derived from the CCP4 program contact.f (http://www.ccp4.ac.uk/html/contact.html) where the original keyword input is automated to find all atom-atom contacts between every hetgroup in a PDB entry that is not in a polymer chain designated by SEQRES, and then list in the SITE record the unique set of residues in PDB format irrespective of symmetry while in mmcif format the unique set of residues plus symmetry are listed.

The distance restraints are limited to non H-atom contacts less than 3.7 and greater than 0.8 Ang. The 3.7 is rounded 3.66 Ang which is the limit for O--H....O van der Waals with

O--H bonded	1.08
H van der Waals	1.09
O van der Waals	1.52
	3.69 Ang

Only Atoms with either blank or the same atom_altcode are used.

No atoms with zero occupancy are used.

Only model 1 atoms are used.

The unique chain-chain contacts are listed to a chain for the case where a complete chain is the 'ligand'.

Additional SITE records may also be included upon author request to highlight biologically important residues in the protein (like catalytic residues and metal binding site).

Note that the metal coordination interactions will appear both in LINK and SITE records. The difference between these 2 instances is that the LINK records may be built between 2 symmetry related molecules, while site records are usually listed within the same asymmetric unit.

REMARK 800 is mandatory if SITE records exist.

LINK records LINK records will be automatically generated using the standard software. There are various cutoff distances specified in this software for various kinds of LINK records. LINK records can include covalent bonding, metal coordination etc. LINK records may also be added by the author.

Standard software will automatically add REMARK 620 to the header of the file listing the angles of the link records between metal ions and surrounding residues following standard coordination geometry. The distances between the atoms and the symmetry operations will be added to the end of the LINK record in the PDB file. An example of SITE and LINK records are shown in PDB entry 1W3M.

SSBOND records SSBOND records are created for cysteine residues involved in disulfide bonds and do not include disulfide bonds between other residues or ligands. The distance between the atoms will be added to the end of the SSBOND line in the PDB file. The symmetry operation for the atom, including the identity, will also be indicated.

HELIX and SHEET records Helix and sheet records are automatically generated by Promotif software. Authors who wish to provide their own helix and sheet records may do so; a remark will be added to REMARK 650 and REMARK 700 of the PDB entry to indicate that the helix and/or sheet records were author provided.

REMARK 3 wwPDB will not accept REMARK 3 template which uses seq ids as selection range in tls/ncs groups.

11 Structural Genomics Entries

Structural genomics (SG) entries are usually either X-ray or NMR structures deposited by the various structural genomics groups around the world. In the USA, these structures are deposited primarily by the several Protein Structure Initiative (PSI) groups. Europe, UK, Canada and Japan also have several structural genomics centers. The SG structures are processed in the same way as any other depositions to the PDB. There are just a few additional rules for annotation of these entries. These are listed below:

- Usually the SG entries are deposited with a “release immediately” status. In special instances (like for the CASP competition) the depositions may be processed and held for a pre-determined period before its release.
- If the function of the protein or complex in the deposition is not known, the HEADER (struct_keywords.) is listed as 'STRUCTURAL GENOMICS, UNKNOWN FUNCTION' as opposed to ‘UNKNOWN FUNCTION’. If at a later date the function of the protein is determined, the author may request the header to be updated.
- For entries deposited by an SG group, the author list also includes the name of the SG center (like JCSG, MCSG, BSGC, etc.).
- The following words and phrases are also included in the keywords: SG center name (in full and also the abbreviation), Structural Genomics. If the entry is from a PSI center, the initials “PSI” and the words “Protein Structure Initiative” are added to the keywords. Entries that are part of the second phase of the PSI project are labeled as PSI-2.
- For SG entries deposited by centers which also deposit targets to TargetDB, the TargetDB ID for each sequence in the entry is included in the file and it appears in REMARK 900 (_pdbx_database_related).
- The project name, center name and center abbreviation are included in the _pdbx_SG_project mmCIF category.

12 Information specific to X-ray structures

Deposition of X-ray structures. All structures determined by single crystal X-ray diffraction where the structure is that of a non-virus capsid should contain the atomic coordinates for the whole crystallographic asymmetric unit (ASU). The ASU is defined as the smallest unit that can be rotated and translated to generate one unit cell using only the symmetry operators allowed by the crystallographic symmetry. The asymmetric unit may be one molecule or one subunit of a multimeric protein, but it can also be more than one.

Information contained in structure factor files Structure factor (sf) files should include information such as cell, space group, symmetry, wavelength, number of reflections and title of the entry. The date included in the header of the sf file will be the date of release, not the deposition date. If authors include multiple structure factor files (such as a set for refinement, multiple sets for phasing, etc.), the data will be archived as one data_block with multiple categories. For example, the reflections for refinement will be listed in the category '_refln' and the phasing data sets will be listed in the category '_phasing_set_refln.' Use of multiple data blocks within one sf file will be discontinued. The different data sets can be distinguished by crystal and/or wavelength ID as appropriate. If different cell dimensions are present, this information will also be included in the file.

REMARK 3 & REMARK 200 significant figures The values provided by the author will be retained in the mmCIF and PDB files.

Hydrogens in crystal structures Hydrogens in crystal structures will be retained regardless of resolution with the occupancy provided by the depositor (even if the structure is a low resolution structure and the occupancy is 1.00).

Matthews coefficient and solvent content For crystal structures, the Matthews coefficient and solvent content will be automatically calculated using the following equations:

Matthews coefficient[7] = volume of unit cell/(the molecular weight of macromolecule*Y*X)
Where Y is the number of asymmetric units in the unit cell (i.e. the number of symmetry operators in the space group). The unknown variable, X, is the number of molecules in the asymmetric unit.

Solvent content = 1 - 1.23/ (Matthews coefficient)

The molecular weight includes protein and nucleic acids based on sequences, no water and ligands. In cases of viral capsids and proteolytic fragments, the Matthews coefficient and solvent content should be author provided and will not be automatically calculated.

Reflections The number of reflections for refinement is the number of crystallographically unique measured reflections that satisfy both the resolution and the observation limits. The number of reflections for data collection is the total number of crystallographically unique measured reflections that are labeled as observed by the criterion on sigma(I) or on

[7] See Matthews, B.W. 1968. Solvent content of protein crystals. J. Mol. Biol. 33: 491–497 and http://www.doe-mbi.ucla.edu/~sawaya/tutorials/Characterize/characterize.html

sigma(F). The number of reflections reported for refinement should be less than or equal to the number reported for data collection, even if Freidel pairs were used.

Twinned structures Structures based on twinned crystal diffraction data can be identified through use of the recent addition of twinning tokens to the mmCIF public exchange dictionary, pdbx_twin. The tokens can be used for identification of twinning operator, type, and fraction. The information contained in these tokens should be automatically mapped to REMARK 3, OTHER REFINEMENT REMARKS, in the PDB file header except PHENIX and REFMAC refinement.

The structure factor file for a twinned structure should include the detwinned data used for refinement first. If the authors have the twinned data, it may also be included in the file. If authors include both the twinned and detwinned data, the pdbx_reflns_twin tokens should also be included in the sf file.

MAD data If MAD data was collected, authors are encouraged to provide all data sets used in structure solution and refinement. The data set used for refinement should be listed first in the structure factor file. Authors are encouraged to provide the other (phasing) datasets. Wavelengths, source and other data collection information for all data sets should also be provided.

For examples see Format Guide.

BioSync and information about synchrotron data collection Information about synchrotron sources and beamlines will be made consistent with the standard names used in the BioSync database (http://biosync.rcsb.org/).

13 Information specific to NMR structures

Pseudoatoms (Q atoms) Pseudoatoms (also known as Q atoms) submitted for NMR entries will be removed from the entry.

Superimposed models At least one domain of the NMR entry should be superimposed across all models. It is recognized that for multi-domain NMR structures, domain movements prevent the whole structure from being aligned through the length of the molecule. However, in order to highlight the relative movement of the domains, it makes sense to superpose at least one part of the structure across all the models deposited under a PDB accession code. This does not detract from the scientific value of the coordinate set or the experiment, but on the contrary, serves to highlight domain motion with respect to a fixed point. The superposition need not be arbitrary but may be done at the choosing of the depositor. This will allow the larger scientific community easy identification of the protein folds. It also facilitates identification of model variations across different parts of the structure.
There are two types of NMR experimental methods (EXPDTA, _exptl.method):
SOLID-STATE NMR
SOLUTION NMR

All models in a deposition should be superimposed in an appropriate author determined manner and only one superposition method should be used. Structures from different experiments, or different domains of a structure should not be superimposed and deposited as models of a deposition.

All models in an NMR ensemble must be homogeneous – each model must have the exact same atoms (hydrogen and heavy atoms), sequence and chemistry.

Deposition of minimized average structure must be accompanied with ensemble and must be homogeneous with ensemble.
MDLTYP record contains additional annotation pertinent to the coordinates presented in the entry. This record will indicate minimized average structure with model number of the minized average structure. The corresponding cif is _struct.pdbx_model_type_details.

MDLTYP MINIMIZED AVERAGE, MODEL X

REMARK 465
For homogeneous NMR ensemble, the missing residues will be listed in model range.
Example,

```
REMARK 465 MISSING RESIDUES
REMARK 465 THE FOLLOWING RESIDUES WERE NOT LOCATED IN THE
REMARK 465 EXPERIMENT. (RES=RESIDUE NAME; C=CHAIN IDENTIFIER;
REMARK 465 SSSEQ=SEQUENCE NUMBER; I=INSERTION CODE.)
REMARK 465   MODELS 1-20
REMARK 465     RES C SSSEQI
REMARK 465     MET A     1
REMARK 465     GLY A     2
```

REMARK 470
For homogeneous NMR ensemble, the missing atoms will be listed in model range.

Example,

```
REMARK 470 MISSING ATOM
REMARK 470 THE FOLLOWING RESIDUES HAVE MISSING ATOMS(RES=RESIDUE NAME;
REMARK 470 C=CHAIN IDENTIFIER; SSEQ=SEQUENCE NUMBER; I=INSERTION CODE):
REMARK 470   MODELS 1-25
REMARK 470     RES CSSEQI  ATOMS
REMARK 470     ILE A  20    CD1
REMARK 470     THR A  59    CG2
```

14 Information specific to Electron Microscopy structures

There are 2 types of data files associated with cryo-EM structures – EM volumes (maps) and atomic coordinates. The atomic coordinate data used in fitting to EM maps are submitted to the PDB and processed by all centers of the wwPDB. The EM (and Electron Tomography) maps (volume data) are deposited, processed and archived only in the Electron Microscopy Database (EMDB) at EBI. The atomic coordinates and their corresponding EM maps will be cross-referenced in both the EMDB and PDB depositions where appropriate. The EMDB ID will be listed in REMARK 900, related entries. Further information should also be included in REMARK 999.

New templates and mmCIF tokens were created in consultation with the EM scientific community to archive relevant details of the EM experiment, data collection, processing and structure solution.

There are two types of experimental methods for EM:
ELECTRON CRYSTALLOGRAPHY
ELECTRON MICROSCOPY

REMARK 240 This remark is mandatory for ELECTRON CRYSTALLOGRAPHY. Three new PDB records have been added: RECONSTRUCTION METHOD, SAMPLE TYPE and SPECIMEN TYPE. The DATE OF DATA COLLECTION will have uniform date, dd-mmm-yy.

REMARK 245 This remark is mandatory for ELECTRON MICROSCOPY. Four new PDB records have been added: RECONSTRUCTION METHOD, SAMPLE TYPE, SPECIMEN TYPE and PARTICLE TYPE if SAMPLE TYPE is PARTICLE. The DATE OF EXPERIMENT will have uniform date, dd-mmm-yy.

For examples see Format Guide.

15 Viral capsids and other complex assemblies

For the purposes of annotation, a complex assembly is defined as a structure for which the full biological assembly and/or crystallographic asymmetric unit is built by applying a set of non-crystallographic rotation/translation transformations to a set of deposited coordinates.

Icosahedral Viruses The icosahedral virus is the most common complex assembly deposited to the PDB. The author generally deposits the coordinates of the icosahedral asymmetric unit and supplies a set of 60 transformation matrices to be applied to the coordinates to produce the full biological assembly. We will continue to request these matrices from the authors. From the author-provided matrices and coordinates we will calculate a standard set of 60 ordered matrices as well as the transformation that moves the complex to the standard icosahedral frame (same frame used by ViperDB). The calculated matrices, the frame transformation, and the description of how they are to be applied to the coordinates to build the assembly will be stored in _pdbx_struct records.

For crystal structures we will also request a complete description of how to build the crystal asymmetric unit, and the description will be archived in _pdbx_struct records. If the coordinates are provided in the crystal frame, the non-crystallographic symmetry transformations will also be placed in struct_ncs_oper records and will appear in MTRIX records, enabling validation against the structure factor data.

Regular Symmetries Icosahedral point symmetry is just one type of symmetry that can be adopted by a complex assembly. Other point symmetries (see table below) or helical symmetries are possible. For all structures deposited as complex assemblies, we will archive symmetry information as appropriate in _pdbx_point_symmetry or _pdbx_helical_symmetry records.

point symmetry	Schoenflies symbol	# equivalent positions
circular	Cn	n
dihedral	Dn	2n
tetrahedral	T	12
octahedral	O	24
icosahedral	I	60

From: International Tables for Crystallography, Volume A, 4th edition, Table 10.4.2, p. 782-783

REMARK 300 The point symmetry will reflect in the last line of REMARK 300

Example,

```
REMARK 300 THE ASSEMBLY REPRESENTED IN THIS ENTRY HAS REGULAR
REMARK 300 CYCLIC POINT SYMMETRY (SCHOENFLIES SYMBOL = C38).
```

C = CYCLIC
T = TETRAHEDRAL
D = DIHEDRAL
O = OCTAHEDRAL

I = ICOSAHEDRAL

The mapping cif is _pdbx_point_symmetry. For example, _pdbx_point_symmetry.entry_id 2BK1
_pdbx_point_symmetry.Schoenflies_symbol C
_pdbx_point_symmetry.circular_symmetry 38

16 Work in progress

Sequence variants The wwPDB staff recognize that there are different ways to handle variants in the DBREF section of the PDB format file. The best way to handle these instances is currently being discussed.

Patents Some authors wish to include information regarding patents as the primary citation for depositions. The wwPDB staff is discussing this matter.

TLS tensors The wwPDB recognizes there is a problem regarding the way B factors are represented in CCP4 refined depositions where TLS refinement was used. The wwPDB staff are in contact with CCP4 regarding the resolution of this.

Source information for electron microscopy depositions The wwPDB is working on developing a better data model for electron microscopy, especially to represent cases where the sequence of the model is different from the sequence used in the experiment.

Large size files The wwPDB recognizes that there are cases where the coordinates do not fit into the PDB file format because the limit on the maximum number of chain IDs and/or atoms is exceeded. The best way to address this issue is being discussed.

Multiple models for structure with alternate conformations The wwPDB recognizes that there are different ways to represent entries containing alternate conformations. The best possible and scientifically accurate representations of these structures are being considered.

Appendices:

A. HEADER list

Below is the list of headers to be used when processing PDB entries:

ALLERGEN
ANTIBIOTIC (peptidic, saccharide containing)
ANTIFREEZE PROTEIN
ANTIFUNGAL PROTEIN
ANTIMICROBIAL PROTEIN
ANTITOXIN
ANTITUMOR PROTEIN
ANTIVIRAL PROTEIN
APOPTOSIS
ATTRACTANT
BIOSYNTHETIC PROTEIN
BLOOD CLOTTING
CARBOHYDRATE
CELL ADHESION
CELL CYCLE
CELL INVASION
CHAPERONE
CIRCADIAN CLOCK PROTEIN
CONTRACTILE PROTEIN
CYTOKINE (includes interleukins, interferons)
DE NOVO PROTEIN (ARTIFICIALLY DESIGNED, OFTEN SYNTHETIC)
DNA
DNA-RNA HYBRID (used when biological unit contains mixed DNA and RNA residues or strands)
ELECTRON TRANSPORT
ENDOCYTOSIS
EXOCYTOSIS
FLAVOPROTEIN
FLUORESCENT PROTEIN
GENE REGULATION (use only when TRANSCRIPTION, REPLICATION, TRANSLATION are not applicable)
HORMONE
HYDROLASE (E.C.3.-.-.-)
IMMUNE SYSTEM (includes antibodies, antigens)
ISOMERASE (E.C.5.-.-.-)
LIGASE (E.C.6.-.-.-)
LIPID TRANSPORT
LUMINESCENT PROTEIN
LYASE (E.C.4.-.-.-)
MEMBRANE PROTEIN (no other function known)
METAL TRANSPORT
MOTOR PROTEIN
NEUROPEPTIDE
ONCOPROTEIN
OXIDOREDUCTASE (E.C.1.-.-.-)
OXYGEN BINDING
OXYGEN STORAGE
OXYGEN TRANSPORT
PHOTOSYNTHESIS

PLANT PROTEIN (no other function known)
PROTON TRANSPORT
PROTEIN TRANSPORT (a protein involved in transporting other protein)
RECOMBINATION
REPLICATION
RIBOSOME (use only when TRANSLATION is not correct; do not specify /RNA even when present!)
RIBOSOMAL PROTEIN
RNA
SIGNALING PROTEIN (includes G-proteins)
SPLICING
STRUCTURAL GENOMICS (product of a probable gene)
STRUCTURAL PROTEIN
TOXIN (not antibiotic, can use e.g. HYDROLASE INHIBITOR, TOXIN)
TRANSFERASE (E.C.2.-.-.-)
TRANSCRIPTION (DNA to RNA)
TRANSLATION (protein synthesis; prefer over RIBOSOME)
TRANSPORT PROTEIN (a protein that transports anything)
NUCLEAR PROTEIN (whether involved in binding RNA/DNA or some sort of nuclear processing is unclear)
VIRUS (for entire viral capsid)
VIRAL PROTEIN (viral protein not involved in the viral capsid)
VIRUS LIKE PARTICLE ((for cases where virus like particles are assembled , but are not the standard virus)

When no other function is known use the following:
CHOLINE-BINDING PROTEIN
CYTOSOLIC PROTEIN (a protein whose function is not known well but is known to be found in the cytosol of a cell.)
DNA BINDING PROTEIN
RNA BINDING PROTEIN
LIPID BINDING PROTEIN
METAL BINDING PROTEIN (such as ZN, FE)
PEPTIDE BINDING PROTEIN
PROTEIN BINDING (implies binding of protein by protein)
SUGAR BINDING PROTEIN
xxx-BINDING PROTEIN (for any xxx ligand if none of above applies, such as HEME, AVIDIN, BIOTIN)
PROTEIN FIBRIL
UNKNOWN FUNCTION

B. Format for Structure Factors

Example 1

```
data_rxxxxsf
#
_audit.revision_id      1_0
_audit.creation_date  ?
_audit.update_record    'Initial release'
#
#
_entry.id   rxxxxsf
```

```
#
#
_cell.entry_id     rxxxxsf
_cell.length_a             118.8600
_cell.length_b             155.0300
_cell.length_c             155.5400
_cell.angle_alpha           90.0000
_cell.angle_beta            90.0000
_cell.angle_gamma           90.0000
#
_symmetry.entry_id     rxxxxsf
_symmetry.Int_Tables_number        20
_symmetry.space_group_name_H-M     'C 2 2 21'
#
loop_
_symmetry_equiv.id
_symmetry_equiv.pos_as_xyz
1 'X,  Y,  Z'
2 '-X,  -Y,  Z+1/2'
3 'X,  -Y,  -Z'
4 '-X,  Y,  -Z+1/2'
5 'X+1/2,  Y+1/2,  Z'
6 '-X+1/2,  -Y+1/2,  Z+1/2'
7 'X+1/2,  -Y+1/2,  -Z'
8 '-X+1/2,  Y+1/2,  -Z+1/2'
#
_reflns.entry_id     rxxxxsf
_reflns.d_resolution_high          2.598
_reflns.d_resolution_low         47.140
_reflns.limit_h_max         45
_reflns.limit_h_min          0
_reflns.limit_k_max         59
_reflns.limit_k_min          0
_reflns.limit_l_max         59
_reflns.limit_l_min          0
_reflns.number_all        44453
_reflns.number_obs        44453
#
_diffrn_radiation_wavelength.id     1
#
_exptl_crystal.id     1
#
_reflns_scale.group_code     1
#
loop_
_refln.wavelength_id
_refln.crystal_id
_refln.scale_group_code
_refln.index_h
_refln.index_k
_refln.index_l
_refln.status
_refln.F_meas_au
_refln.F_meas_sigma_au
_refln.F_calc
_refln.phase_calc
_refln.fom
```

```
_refln.pdbx_HL_A_iso
_refln.pdbx_HL_B_iso
_refln.pdbx_HL_C_iso
_refln.pdbx_HL_D_iso
1 1 1    0    0    6 o   299.0    6.4   1306.2   0.0    0.32  0.33
0    0.00    0.00
1 1 1    0    0   10 o   726.8   15.0   1756.7   180.0  0.99  2.86
0    0.00    0.00
...
#END OF REFLECTIONS
```

Example 2

```
data_rxxxxsf
#
_audit.revision_id      1_0
_audit.creation_date  ?
_audit.update_record    'Initial release'
#
#This file contains two data sets. The first data set is used for
refinement.
#The second data set is used for phasing.
#
_entry.id   rxxxxsf
#
#
_cell.entry_id     rxxxxsf
_cell.length_a             108.7420
_cell.length_b              61.6790
_cell.length_c              71.6520
_cell.angle_alpha           90.0000
_cell.angle_beta            97.1510
_cell.angle_gamma           90.0000
#
_symmetry.entry_id     rxxxxsf
_symmetry.Int_Tables_number         5
_symmetry.space_group_name_H-M     'C 1 2 1'
#
loop_
_symmetry_equiv.id
_symmetry_equiv.pos_as_xyz
1 'X,  Y,  Z'
2 '-X,  Y,  -Z'
3 'X+1/2,  Y+1/2,  Z'
4 '-X+1/2,  Y+1/2,  -Z'
#
_reflns.entry_id     rxxxxsf
_reflns.d_resolution_high          2.148
_reflns.d_resolution_low         70.711
_reflns.limit_h_max         50
_reflns.limit_h_min        -50
_reflns.limit_k_max         28
_reflns.limit_k_min          0
```

```
_reflns.limit_l_max         33
_reflns.limit_l_min          0
_reflns.number_all        25739
_reflns.number_obs        25471
#
_diffrn_radiation_wavelength.id     1
#
_exptl_crystal.id     1
#
_reflns_scale.group_code     1
#
loop_
_refln.wavelength_id
_refln.crystal_id
_refln.scale_group_code
_refln.index_h
_refln.index_k
_refln.index_l
_refln.status
_refln.F_meas_au
_refln.F_meas_sigma_au
1 1 1     -50    0    1 x         ?      ?
1 1 1      49    5    1 o      37.7    9.4
1 1 1      50    0    0 x         ?      ?
...
#END

data_rxxxxAsf
#
#This is second data set for phasing.
#
#
#
loop_
_cell.entry_id
_cell.CCP4_wavelength_id
_cell.CCP4_crystal_id
_cell.length_a
_cell.length_b
_cell.length_c
_cell.angle_alpha
_cell.angle_beta
_cell.angle_gamma
PHASE  1  1  108.7420  61.6790  71.6520  90.0000  97.1510  90.0000
PHASE  2  1  108.7420  61.6790  71.6520  90.0000  97.1510  90.0000
PHASE  3  1  108.7420  61.6790  71.6520  90.0000  97.1510  90.0000
PHASE  4  1  108.7420  61.6790  71.6520  90.0000  97.1510  90.0000
#
_symmetry.entry_id     PHASE
_symmetry.Int_Tables_number        5
_symmetry.space_group_name_H-M    'C 1 2 1'
#
loop_
_symmetry_equiv.id
_symmetry_equiv.pos_as_xyz
1 'X,  Y,  Z'
2 '-X,  Y,  -Z'
```

```
3 'X+1/2,  Y+1/2,  Z'
4 '-X+1/2,  Y+1/2,  -Z'
#
loop_
_reflns.entry_id
_reflns.CCP4_wavelength_id
_reflns.CCP4_crystal_id
_reflns.d_resolution_high
_reflns.d_resolution_low
_reflns.limit_h_max
_reflns.limit_h_min
_reflns.limit_k_max
_reflns.limit_k_min
_reflns.limit_l_max
_reflns.limit_l_min
_reflns.number_all
_reflns.number_obs
PHASE  1  1    2.148   71.095    50   -50    28  0   33  0  25739  11565
PHASE  2  1    2.148   71.095    50   -50    28  0   33  0  25739  11895
PHASE  3  1    2.148   71.095    50   -50    28  0   33  0  25739  12000
PHASE  4  1    2.148   71.095    50   -50    28  0   33  0  25739  13346
#

loop_
_diffrn_radiation_wavelength.id
_diffrn_radiation_wavelength.CCP4_crystal_id
_diffrn_radiation_wavelength.wavelength
1  1      0.00000
2  1      0.00000
3  1      0.00000
4  1      0.00000
#
loop_
_exptl_crystal.id
1
#
_reflns_scale.group_code     1
#
loop_
_refln.wavelength_id
_refln.crystal_id
_refln.scale_group_code
_refln.index_h
_refln.index_k
_refln.index_l
_refln.status
_refln.F_meas_au
_refln.F_meas_sigma_au
_refln.pdbx_anom_difference
_refln.pdbx_anom_difference_sigma
1 1 1    -50    0    1 x        ?        ?        ?        ?
1 1 1    -50    0    2 x        ?        ?        ?        ?
...
#END OF REFLECTIONS
```

Example 3

```
data_rxxxxsf
_entry.id   XXXX
_database_2.database_code   PDB
_database_2.database_id     XXXX
_audit.creation_date    'YYYY-MM-DD'

_cell.entry_id      XXXX
_cell.length_a      117.259
_cell.length_b      127.319
_cell.length_c      191.227
_cell.angle_alpha   90.00
_cell.angle_beta    90.29
_cell.angle_gamma   90.00
_cell.formula_units_Z   44

_symmetry.entry_id    XXXX
_symmetry.space_group_name_H-M  'P 1 21 1'

loop_
_symmetry_equiv.id
_symmetry_equiv.pos_as_xyz
       1  'X,Y,Z'
       2  '-X,Y+1/2,-Z'

_atom_sites.entry_id     XXXX
_atom_sites.fract_transf_matrix[1][1]     0.008528
_atom_sites.fract_transf_matrix[1][2]     0.000000
_atom_sites.fract_transf_matrix[1][3]     0.000043
_atom_sites.fract_transf_matrix[2][1]     0.000000
_atom_sites.fract_transf_matrix[2][2]     0.007854
_atom_sites.fract_transf_matrix[2][3]     0.000000
_atom_sites.fract_transf_matrix[3][1]     0.000000
_atom_sites.fract_transf_matrix[3][2]     0.000000
_atom_sites.fract_transf_matrix[3][3]     0.005229

_atom_sites.fract_transf_vector[1]         0.00000
_atom_sites.fract_transf_vector[2]         0.00000
_atom_sites.fract_transf_vector[3]         0.00000


_reflns.entry_id     XXXX
_reflns_scale.group_code 1
_exptl_crystal.id  1

_diffrn_radiation_wavelength.id 1
_diffrn_radiation_wavelength.wavelength 1.5418
_reflns.d_resolution_high          2.700
_reflns.d_resolution_low        182.574
_reflns.number_all       149457

loop_
_refln.wavelength_id
_refln.crystal_id
_refln.scale_group_code
```

```
_refln.index_h
_refln.index_k
_refln.index_l
_refln.status
_refln.F_meas_au
_refln.F_meas_sigma_au
…
#END OF REFLECTIONS
```