pdb_extract - Workstation Version Manual

Extract information from each step of X-ray crystallographic and NMR software applications

(June, 18, 2004; last modified August 3, 2011) | (Latest version 3.11)

Table of Contents

What does pdb_extract do? (TABLE OF CONTENTS)

pdb_extract is used to extract statistical information from the output files produced by many software for protein structure determination using Xray Crystallography and NMR method. These statistical information will be written into a complete mmCIF file which is ready for PDB deposition.

In the case of Xray structure determination, **pdb_extract** merges all the information into two mmCIF (macromolecular Crystallographic Information File) files. One mmCIF file contains structure factors and the other contains atomic coordinates and statistics extracted from the steps of structure determination (data collection/integration/reduction, heavy atom phasing, molecular replacement, density modification, and final structure refinement) for various methods (MR, SAD, MAD, SIR, SIRAS, MIR, MIRAS). These two mmCIF files are ready for PDB deposition.

In the case of NMR structure determination, statistics from header section of PDB file and other LOG files produced by software is merged into one mmCIF file containing coordinates. This file along with other constrain files (if applicable) is ready for PDB deposition.

The current version supports 35 software packages and hundreds of different output files produced in various of steps. **Click here** to see the supported software lists.

The assembled mmCIF files by **pdb_extract** should be uploaded to the **ADIT** server. Enter any additional information into ADIT and submit your files directly from there.

The advantage of using pdb_extract:

- Faster to prepare your mmCIF file for deposition. Users only provide the output files produced from various software to get all the statistics. Some items (for example, Matthews coefficient and solvent constant, molecular entities ...) are pre-calculated for you.
- Complete and accurate to deposit your file. All the statistics (ranging from index to final refinement) can be automatically extracted. This reduces many typing errors.
- Great for multiple structural deposition. The data template file (called data_templete.text for non-electronically extracted information, like author name ...) can be re-used in each structure without re-entering the same information.
- Both Unix command options and Web interface are provided. It is flexible to use.
- Collectively, these software tools reduce the human effort required to assemble complete and validated protein structure entries ready for PDB deposition.

IMPORTANT NOTES:

1. The LOG or output files generated from any software should not be modified. Otherwise, information may not be extracted.
2. If you have several structures ready to be deposited to the PDB site, you need to apply the **pdb_extract** program to each individual structure, since each structure requires a single PDB ID for deposition.
3. You may have a lot of trials for each step (data processing, heavy atom phasing, or density modification, or final structure refinement), but information extracted from each step should be only from the best trial that leads to next step toward solving your structure.
4. You may use different programs for heavy atom phasing solution. For example, you used program A to locate heavy atom positions and you used program B to refine heavy atom parameters (like x, y, z, occupancy and B factors etc.). Phasing statistics information will be extracted from the output of program B; therefore, **pdb_extract** should be applied to the output of program B. However, if you want to give credit to program A, you can type '-p program-name' without giving LOG files.
5. You may also use different programs for final structure refinement, but **pdb_extract** should be only applied to the program which leads to your final structure deposition.

Program access TOP

The source and binary versions of **pdb_extract** can be downloaded from the address **http://deposit.pdb.org/software** . The source is available under an Open Source license. The binary distributions are available for Intel-Linux.

The web interface can be accessed at **http://pdb-extract.rutgers.edu**

pdb_extract has been integrated into **CCP4** and the CCP4i interface(Version 5.0 and above). Users can run **pdb_extract** under the CCP4 environment.

Installations TOP

System Requirements:

- platform Intel-Linux:
- C/C++ compilers

Installation of binary distribution TOP

It is recommended to install the binary distribution, since it is fast to install and it takes small space. The binary distributions are available for Intel-Linux.

Step 1. Uncompress and unbundle the distribution using the following command:

```
zcat pdb-extract-vX.XXX-XXX.tar.gz | tar -xf -
```

Step 2. Set up the environment variables.

```
* Define PDB_EXTRACT environment variable to point to the installation
directory. Assuming that the installation directory is
/home/username/pdb-extract-vX.XXX-XXX, execute in the shell:

    For C shell users:
      setenv PDB_EXTRACT /home/username/pdb-extract-vX.XXX-XXX

    For Bourne shell users:
      PDB_EXTRACT=/home/username/pdb-extract-vX.XXX-XXX; export PDB_EXTRACT

* Add "bin" subdirectory to the PATH environment variable.
Execute in the shell:

    For C shell users:
      setenv PATH "$PDB_EXTRACT/bin:"$PATH

    For Bourne shell users:
      PATH="$PDB_EXTRACT/bin:"$PATH; export PATH
```

Installation of source code distribution TOP

Step 1. Uncompress and unbundle the distribution using the following command:

```
zcat pdb-extract-vX.XXX-XXX.tar.gz | tar -xf -
```

Step 2. Set up the environment variables.

* Define PDB_EXTRACT environment variable to point to the installation directory. Assuming that the installation directory is /home/username/pdb-extract-vX.XXX-XXX, execute in the shell:

    For C shell users:
      setenv PDB_EXTRACT /home/username/pdb-extract-vX.XXX-XXX

    For Bourne shell users:
      PDB_EXTRACT=/home/username/pdb-extract-vX.XXX-XXX; export PDB_EXTRACT

* Add "bin" subdirectory to the PATH environment variable. Execute in the shell:

    For C shell users:
      setenv PATH "$PDB_EXTRACT/bin:"$PATH

    For Bourne shell users:
      PATH="$PDB_EXTRACT/bin:"$PATH; export PATH

Step 3. Building the Application (compile the program)

Position in the pdb-extract-vX.XXX-XXX directory and run "make" command:

```
cd pdb-extract-vX.XXX-XXX
make
```

The application executables will be placed in the "bin" subdirectory.

Run the program TOP

There is an example included in this distribution.

This example is located in the subdirectory of "pdb-extract-vX.X/examples/Example_1".

The directory contains the following:

- input_data - contains the input data for the example
- deposit - contains the resulting files (after running the program):

To execute the example, position in the appropriate directory and invoke test.sh and test_script.sh scripts.

cd pdb-extract-vX.XXX-XXX/pdb-extract-vX.X/examples/Example_1

A. Run the scripts test.sh

All the Unix commands were included in the script file test.sh.

./test.sh

B. Run the scripts test_script.sh

The script for *test_script.sh* is an alternative way to obtain the same result as above. It is also a combination of various programs. The difference is that it used the component extract instead of the **pdb_extract** and pdb_extract_sf. All the information is included in the file log_script.inp.

./test_script.sh

Please **click here** to see the script files and the explanations of arguments of input/output.

Tutorials TOP

There are four ways to extract crystallographic information and deposit complete data to the Protein Data Bank.

1. Use the **pdb_extract** **Web interface**
2. Use Unix Command Line Interface.
3. Use CNS-like Script Interface.
4. Use **CCP4i**

The four interfaces have different features. For example, The CCP4i or Web interface provide a simple graphic interface. Users only select the program name and output file names to do the job. The full Unix command line method provides the greatest flexibility. User need to read the command options to run the program. The script input method provides a simple local interface.

Here, we give a concrete example to show how to use **pdb_extract** for complete data extraction.

In this example, the experimental method for solving the protein structure was multiple anomalous diffraction (MAD). The information for the experiment is as the following:

- One crystal was used for data collection
- Three wavelengths (e.g. inflection, peak, remote edge) were tuned for diffraction.All three reflection data files were used for phasing.
- HKL2000 was used for indexing and data scaling. The program produced
  - four reflection data sets (data_for_refine.sca, scale1.sca, scale2.sca, scale3.sca).
  - four LOG files from scaling the four data sets (scale_refine.log, scale1.log, scale2.log, scale3.log).
  - one log file for index (index.log)
- SOLVE was used for heavy atom phase determination and phase refinement. The program produced
  - one log file (solve.prt).
- RESOLVE was used for density modification. The program produced
  - one log file (resolve.log).
- REFMAC5 was used for final structure refinement. The program produced
  - one data harvest file in mmcif format(native.refmac).
  - the final PDB file (refmac.pdb).

Use PDB-EXTRACT Web interface **TOP**

Follow on line tutorial

Use Unix Command Line Interface **TOP**

STEP 1. Obtain the template data file *data_template.text* using the command

extract -pdb refmac.pdb

After running the program, you will get a file called *data_template.text*. CATEGORY 1-2 contains the extracted unit cell parameters and the unique molecular chemical sequence group. Please modify the two CATEGORIES as necessary.

You may skip other categories until you submit your assembled mmCIF file into **ADIT** . However, if you have multiple structures to submit, you are commended to use the data_template file, since it can be re-used without re-entering the same information.

The content of the data template file *data_template.text* is given in **Appendix**
The command line options are given in the **Table**

STEP 2. Obtain coordinates and all the statistics

Run the **pdb_extract** program:

```
pdb_extract -e MAD  \            (MAD experiment)
-i HKL -iLOG index.log \             (from indexing)
-s HKL -iLOG scale_refine.log \        (from scaling for refinement)
-sp HKL scale1.log scale2.log scale3.log \       (from scaling for phasing)
-p SOLVE -iLOG solve.prt \           (from phasing)
-d RESOLVE -iLOG resolve.log \         (from density modification)
-r refmac5 -icif refmac -ipdb refmac.pdb \       (from final refinement)
-iENT date_template.text \         (structural & author information)
-o pdb_extract.cif              (output file in mmcif format)
```

Note: there must be a space before the sign \ and no space after, if you write the options into a script file.

STEP 3. Obtain structure factors

Run **pdb_extract_sf** to convert data into mmCIF format and merge all the files to one file.

```
pdb_extract_sf \
-rt F -rp MTZ -idat scale_refine.mtz  \       (data for refinement)
-dt I -dp HKL \                        (data for phasing)
-c 1 -w 1 -idat scale1.sca \       (crystal 1 & diffraction 1)
-c 1 -w 2 -idat scale2.sca \       (crystal 1 & diffraction 2)
-c 1 -w 3 -idat scale3.sca \        (crystal 1 & diffraction 3)
-o pdb_extract_sf.cif       (output file in mmcif format)
```

The output file (output_sf.cif) contains one reflection data block for refinement and one data block for protein phasing.

STEP 4. Validation and deposition

It is recommended to validate the two files (pdb_extract_sf.cif, pdb_extract.cif) from **ADIT** before submit your data.

Submit your data from **ADIT.**

Use the script interface **TOP**

STEP 1. obtain the plain text file log_script.inp

extract -pdb refmac.pdb

You will get one script file called *log_script.inp* and one data template file *data_template.text*.

- Edit the data template file according to the instruction in the file.
- Fill all the Log file names and the program names to the script file log_script.inp.

The content of the file log_script.inp is shown in the **Appendix**

STEP 2. run the program:

extract -ext log_script.inp

You will get the same results as using the Unix command line option.

STEP 3. Validation and deposition: (same as in the Unix command line option).

Use CCP4i interface **TOP**

Step 1. From the main window of CCP4i, select the *Data Harvesting Management Tool* option.

Step 2. From the option of *Run program* to select the *Extract additional information for deposition*

Step 3. Select the *Generate a data template file* from various steps

Type (or select using browse) in the yellow boxes either the PDB or mmCIF file name obtained from the final structure refinement and the output file name. In this case, the output coordinate file is refmac.pdb.

Run the **pdb_extract** program to obtain the data template file. Edit this file according to the instruction in the text file.

Step 4. Select the *Generate a complete mmCIF file for PDB deposition* from various steps

Select program names and log file names generated from the selected programs.

- Select the scaling program HKL and select the log file scale1.log to extract scaling statistics (data used for refinement).
- Select phasing method MAD and program SOLVE. Give the log file solve.prt to obtain phasing statistics.
- Select the density modification program RESOLVE and the log file resolve.log to obtain density modification statistics.
- Select the structure refinement program REFMAC5 and the PDB coordinate file refmac.pdb and the data harvest file native.refmac to obtain the PDB coordinates and refinement statistics

- Select the data template file generated from step 3 to obtain the chemical sequence and the non-electronically extracted information.

Run the **pdb_extract** program to obtain a complete data in mmCIF format. The final output file can be uploaded to **ADIT** for on line structure validation and submission.

NOTE: The characters of file name should always start from beginning of each yellow box. There should be no white space in each box, even no file name is typed in.

Use Unix Command Line Interface (NMR) TOP

STEP 1. Obtain the template data file *data_template.text* using the command

extract -pdb coordinate_PDB_file_name **-nmr** (if PDB format)

After running the program, you will get a data template file called *data_template.text*. This data template file contains 21 data fields for entering non-electronically extracted information. Please enter necessary information and carefully check CATEGORY 1 which contains the unique molecular chemical sequence. Please modify CATEGORY 1 as necessary. Additional structure information can be filled into CATEGORIES (2-21) for complete data deposition.

The content of the data template file *data_template.text* is given in **Appendix**

STEP 2. Obtain coordinates and all the statistics

Run the **pdb_extract** program using the following command:

pdb_extract -r CNS -ipdb cns.pdb -ient data_template.text **-nmr**

Statistical information can be extracted from the header section of the PDB file.You will generate a complete mmCIF file containing atomic coordinates and other information about the structure.

STEP 3. Data validation and submmision

Please upload the extracted mmCIF file as well as other constraint files to the **ADIT** server for data validation and submmision.

Use PDB-EXTRACT Web interface TOP

Follow on line tutorial for NMR

helpful hints to get the LOG (or output) files from various programs TOP

Listed below are the programs used from data collection to structure determination.

Data collection/reduction TOP

This section is used to collect statistical information from the LOG files generated by the programs for Data Scaling/Merging/Averaging.

Important: The log files must be generated from the LAST (or BEST) trial which corresponds to the files used for phasing or molecular replacement.

The extracted information may be the following:

```
*    Intensities (or amplitude)  and standard deviations
*    Data completeness (overall, resolution shells)
*    Redundancy (overall, resolution shells), mosaicity
```

```
*     R-merge, R-sym (overall, resolution shells)
*     average(I/sigma), (overall, resolution shells)
*     Total  and unique reflections collected.
*     Resolution range
```

Some helpful hints for getting LOG files from the program of Data Scaling/Merging/Averaging

Using **HKL/HKL2000/scalepack**

HKL (or HKL2000 or Scalepack) is a package by Otwinowski for data collection/reduction/scaling. You can use the graphical interface or the scalepack script to scale your data. The LOG file (e.g. scale1.log) contains statistics for PDB deposition.
The generated LOG file type is 'LOG'.

Using **D*trek**

D*trek is a package by Jim Pflugrath at Rigaku/MSC for data collection/reduction/scaling. You can use the graphical interface to scale (or merge/average) your data. The LOG file (e.g. scale1.log) containing statistics is from the step of scaling data.
The generated LOG file type is 'LOG'.

Using **SAINT**

SAINT is a package by Bruker (Siemens Molecular Analytical Research Tool) for data collection/reduction/scaling. The LOG file (e.g. scale1.ls) containing statistics is from the step of scaling data.
The generated LOG file type is 'LOG'.

Using **SCALA**

SCALA is the CCP4 supported program. It scales together multiple observations of reflections. SCALA generates **mmCIF or LOG** file containing useful statistics. When you run the programs, you must ask the program to export the data harvest file (mmCIF type). The mmCIF file will be name.scala or name.truncate. Otherwise, it will generate LOG file.
The generated LOG file type is 'LOG or mmCIF'.

Molecular replacement **TOP**

This section is used to collect key statistical information from Molecular Replacement. You may first generate a LOG file from the rotation function, then generate a LOG file from the translation function. You can upload the two LOG files into this section for data extraction. You can also upload one LOG file which is generated from MR.

Important: The log files must be generated from the LAST (or BEST) trial which corresponds to the files used for density modification or refinement.

The extracted information may be the following:

```
*       Low and high resolution used in rotation and translation.
*       Rotation and translation methods
*       Reflection cut off criteria, reflection completeness.
*       Correlation coefficients for I or F between observed and calculated.
*       R_factor, packing information, and model details.
```

Some helpful hints for getting LOG files from the program molecular replacement

Using CNS/CNX/XPLOR

CNS can be used to do molecular replacement. After you finish the translation search, you can get a log file called translation.list which contains all the information of molecular replacement.

Using Amore (CCP4)

Amore is a program for molecular replacement. It is distributed in the CCP4 package. After rotation and translation search, you will generate two log files rotation.log and translation.log. You may extract information from both log files

If you run the program in one script, you may generate one LOG file. Upload this LOG file to the web interface.

Using Molrep(CCP4)

Molrep is a program for molecular replacement. It is distributed in the CCP4 package. When you run the script, you can specify a LOG file name (e.g. molrep.log). All the statistic information will be recorded in the log file.

Using EPMR

EPMR is a Unix command line program for molecular replacement. When you run the program, please give a log file name like the following Epmr [options] files > epmr.log All the statisticial information will be written in the log file.

Using Phaser

Phaser was developed by Randy Read's group at the University of Cambridge. It is a program for phasing macromolecular crystal structures with maximum likelihood methods. The program generates a LOG file which can be uploaded to the web interface for data extraction.

Heavy atom phasing TOP

Heavy atom phasing is performed at an earlier stage of structure determination. The log files generated from phasing contain important statistical information which should be deposited to the Protein Data Bank.

From heavy atom phasing, you may have LOG files and heavy atom coordinate file.

```
The phasing methods are the followings:
*      MR   molecular replacement.
*     SAD   single anomalous dispersion.
*     MAD   multiple anomalous dispersion.
*     SIR   single isomorphous replacement.
*   SIRAS   single isomorphous replacement with anomalous scattering.
*     MIR   multiple isomorphous replacement.
*   MIRAS   multiple isomorphous replacement with anomalous scattering.
```

Important: The log files must be generated from the LAST (or BEST) trial which corresponds to the files used for density modification or refinement.

```
The following items may be extracted:
*     Wavelength, f_prime, f_double_prime, resolution range
*     FOM (acentric, centric, overall, resolution shells)
*     R-Cullis (acentric, centric, overall, resolution shells)
*     R-Kraut (acentric, centric, overall, resolution shells)
*     Phasing power (acentric, centric, overall, resolution shells)
*     Number of heavy atom sites, heavy atom type.
*     Heavy atom location method.
*     Heavy atom B-factor, occupancies, and xyz coordinates.
```

Some helpful hints for getting the output files generated by various programs

Using SOLVE (version 2.00 and above):

SOLVE is a program for finding heavy atom location and refining heavy atom parameters. The statistical information is written to a file **solve.prt** (default name used by the program). The heavy atom coordinates are written to a file **ha.pdb**.

Note: You may upload the two file names **solve.prt** (file type: LOG) and **ha.pdb** (file type: PDB).

Using CNS/CNX/XPLOR

CNS is a complete software system for protein crystallography. The scripts for heavy atom location and phasing refinement are mad_phase.inp or ir_phase.inp. When you run these scripts, you will get output files like phase_final.summary, phase_final.sdb or mad_phase.fp.

The output file phase_final.summary has all the phasing statistics.
The output file phase_final.sdb has all the heavy atom coordinates, occupancies and B factors.
The output file mad_phase.fp has refined f_prime and f_double_prime.

(Note: The refined heavy atom coordinates, B factors and occupancies can be found in a file like phase_final.sdb. If you prefer to convert to the PDB format, you can run the script sdb_to_pdb.inp. You will get a file phase_final.pdb with PDB format.)

Note: You may input at most three files (as shown above) for extracting phase information.

Using MLPHARE (CCP4)

MLPHARE is a program in the CCP4 suite. It is used for refining heavy atom parameters.

If you use the CCP4i graphical interface or the script mode, you need to ask the program to write a harvesting file. Select the data havest button, when you use the CCP4i interface. Do not use the key word NOHARV, when you use script. After you finished running this program, you will get a file (e.g. name.mlphare) which is in mmCIF format. It contains all the information for heavy atom phasing refinement.

For extracting the wavelength information, you need to run program REVISE in the CCP4 (version 4.0-4.2.2). You may get a file (e.g. prephadata.log)

Note: You may input at most two files (as shown above) for extracting phase information.

Using SHARP (version 1.3.x and 2.0 and above):

SHARP is a program for finding heavy atom positions and refining heavy atom parameters. When you run SHARP or autoSHARP, the log files which have useful information are normally in the directory sharpfiles/logfiles_local/dirs, where dirs are all the subdirectories for your various structures. Please note that the location of generated log files may depend on how the program is installed!

SHARP produces many output files.

```
For version 1.3.x:
    Heavy.pdb  contains the heavy atom coordinates.
    FOMstats.html   contains figure of merit statistics.
    Otherstat.html  contains Rcullis, Rkraut, phasing power.

For version 2.0 and above:
    Heavy.pdb   contains the heavy atom coordinates.
    FOMstats.html   contains figure of merit statistics.
    RCullis_?.html  contains Rcullis.
    PhasingPower_?.html  contains phasing power
```

The easiest way to obtain these files is to run the program from the SUSHI interface. Review all the log files from the internet browser and save the files as plain text files.

Note: You may input at most four files (as shown above) for extracting phase information.

Using SnB (version 2.0 and above):

SnB has no heavy atom parameter refinement, and it has no corresponding statistics. SnB gives the heavy atom or substructure coordinates (e.g. *heavy.pdb*) in PDB format.

Note: You may input only one file (as shown above) for phasing extraction.

Using BnP (version 0.93 and above):

BnP is a combination of program SnB and Phases. The heavy atom positions are located by SnB and the heavy atom parameters will be refined by Phases.

The log file (e.g. auto.log) can be found from the directory ~/PHASES/*. Log file normally contains phasing power for each phasing set.

The file is in LOG format.

Note: You may input at most one file (as shown above) for extracting phase information.

Using SHELXD or SHELXS (version 97):

Heavy atom or substructure coordinates are produced in PDB format (e.g. *heavy.pdb*).

Note: You may input at most one file (as shown above) for extracting phase information.

Density modification TOP

Density modification is normally performed after obtaining phases. If you do density modification in your structure determination, statistics information is needed for PDB deposition.

If density modification is not done in a separate step, you may skip this step, since you do not have a log file specifically for density modification.

Important: The log files must be generated from the LAST (or BEST) trial which corresponds to the file used for refinement.

```
The following items may be extracted:
*     Density modification method.
*     FOM after density modification (overall, resolution shells)
*     Solvent mask determination method.
*     Structure solution software.
```

Some helpful hints for getting the output files from each program:

Using RESOLVE (version 2.00 and above):

RESOLVE is a density modification program in the SOLVE/RESOLVE package. Normally it runs together with SOLVE, but one can run it separately. When you run RESOLVE, you will get a log file like resolve.log.

Only one log file (resolve.log) is needed for extraction. File type is LOG.

Using CNS/CNX/XPLOR

The CNS user may need to run the input script like density_modify.inp. You will get a log file called density_modify.list.

Only one log file (density_modify.list) is needed for extraction. File type is LOG.

Using DM (CCP4)

DM is a density modification program in the CCP4 suit. When you run DM either by using the CCP4i graphic interface or the script, you will get a log file like dm.log.

Only one log file (dm.log) is needed for extraction. File type is LOG.

Using SOLOMON (CCP4)

SOLOMON is also a another density modification program in the CCP4 suite. When you run DM either by using the CCP4i graphic interface or the script, you will get a log file like Solomon.log.

Only one log file (Solomon.log) is needed for extraction. File type is LOG.

Final structure refinement TOP

Structure refinement is performed at the end of structure determination. The atom coordinates are generated in PDB or mmCIF format and the statistics are generated in log files. The **pdb_extract** program is applied to extract statistical information:

Since statistics can be carried at the header section of PDB file, you may not provide any LOG files for some programs like CNS, REFMAC5.

Important: The log file and the coordinate file must be generated from the LAST (or BEST) trial which corresponds to the file that is used for deposition to the PDB.

```
The following items may be extracted:
*    Resolution range (highest res. shell)
*    Number of reflections used in refinement, and in R-Free set.
*    R-factor (overall, resolution shells)
*    Number of atoms refined
*    Cell parameters and space group.
*    The xyz coordinates of all the atoms.
*    RMS Bond Distances, Bond Angles, Chiral Volume, Torsion Angles
*    Isotropic temperature factor restraints
*    Non-crystallographic symmetry restraints
*    Solvent model used
*    Overall Average Isotropic B Factor
*    Overall Anisotropic B Factor
*    Overall Isotropic B Factor
*    Topology/parameter data used to refine deposited model
*    Refinement software
```

Some helpful hints for getting the output files from each program:

Using REFMAC5 (CCP4):

REFMAC5 is a program for structure refinement used in the CCP4 suite. If you run this program using CCP4i or the script, you can get a PDB file with all the refinement information at the header section.

You may directly deposit this PDB file.

Using CNS/CNX/XPLOR

CNS/CNX/XPLOR is a program for final structure refinement. It exports coordinate file in both PDB and mmCIF format. You need the script deposit_mmcif.inp to generate the mmCIF format.

The mmCIF file carries more statistical information than the PDB file. Authors are encouraged to deposit the mmCIF file, otherwise authors may need to manually fill in more information.

You may not have to give any LOG file generated from CNS/CNX/XPLOR.

Using SHELXL (version 97):

SHELXL is a sub_program in the SHELX package. It is used for structure refinement. After you finish structure refinement, you need to run the shelxpro interactive program and use option B. After going through the shelxpro, you

will get a PDB file (e.g. name.pdb) with header information.

Using TNT (version 5f):

TNT is a crystal structure refinement program. Data from this program can be extracted from the output PDB file and some LOG files. You can use the to_pdb command to convert coordinates in TNT format (name.cor) to the PDB format (name.pdb).

The command is: to_pdb name.cor

After finishing refinement, you must use command rfactor to generate a log file (e.g. rfactor.log) which contains the refinement statistics.

The command is: rfactor name.cor > rfactor.log

To extract the symmetry information, user must provide the symmetry file (e.g. p6122.dat). This information is in the control file name.tnt

Using ARP/wARP:

ARP/wARP is a automatic program for model building and refinement. REFMAC5 is used for the structure refinement step.

The new version (6.0 or above) can use CCP4i as graphic interface. You can run this program either by CCP4i or by using script. You will get a log file (for example warpNtrace_refine.log). You also get a PDB file like warpNtrace.pdb.

Note: If the coordinate file warpNtrace.pdb is directly used for deposition, you can use this option. Otherwise, use other program for final refinement.

Using PHENIX

PHENIX is a new software suite for the automated determination of macromolecular structures using X-ray crystallography and other methods.

The PDB file generated by phenix.refine has the non-standard 'REMARK' and the standard 'REMARK 3'. It is also OK to keep the non-standard REMARK for deposion.

Note: Sometimes, the MTZ file from PHENIX only contains 2Fo-Fc. Before deposition, you must make sure that the amplitude (Fo) or Intensity (I) is included in the MTZ file.

Program argument description and options TOP

There are three executable components (**pdb_extract, pdb_extract_sf, extract**) for the program. Argument description for the programs is given in details bellow.

Unix command options for pdb_extract TOP

PROGRAM DESCRIPTION:

pdb_extract is used to extract statistical information from the output files produced by the software for protein structural determination using Xray Crystallography and NMR method.

pdb_extract merges the information into two mmCIF (macromolecular Crystallographic Information File) files, one with structure factors and one with coordinate and statistic. These two files are ready for PDB deposition.

User can get help by typing '**pdb_extract** -h' or '**pdb_extract** -help' to get information how to do extractions and deposition to PDB

EXECUTABLE NAME: pdb_extract

SYNOPSIS: pdb_extract [OPTIONs]... [FILEs]...

ARGUMENT DESCRIPTION: (-o -e -i -s -sp -m -p -d -r -ipdb -ilog -icif -ient -idat)

1. **-o** Followed by a given output file name.

   For example: -o outfile.mmcif

   NOTE: if you do not give this description, the default output file name (pdb_extract.mmcif) will be used.

2. **-e** Followed by one of the following experimental methods:

```
The phasing methods are the followings:
*      MR   molecular replacement.
*     SAD   single anomalous dispersion.
*     MAD   multiple anomalous dispersion.
*     SIR   single isomorphous replacement.
*   SIRAS   single isomorphous replacement with anomalous scattering.
*     MIR   multiple isomorphous replacement.
*   MIRAS   multiple isomorphous replacement with anomalous scattering.
```

   example: -e MAD

   Note: If your structure was solved by combinations of above methods (e.g. MR with MAD), you may extract things from both methods (e.g. -e MR -m program_mr -ilog Log_file -e MAD -p program_mad -ilog file_name)

3. **-i** Followed by one of the following programs for data indexing:

   [HKL | DENZO | DTREK | MOSFLM]

   For example: -s HKL

4. **-s** Followed by one of the following programs for data scaling (for refinement):

   [SCALA | HKL | SCALEPACK | DTREK | SAINT | 3DSCALE | XSCALE | XENGEN | PROSCALE]

   For example: -s HKL

5. **-sp** Followed by one of the following programs for data scaling (for refinement):

   [SCALA | HKL | SCALEPACK | DTREK | SAINT | 3DSCALE | XSCALE | XENGEN | PROSCALE]

   For example: -sp HKL

   Note: The option is similar to **-s**, but it is used to extract statistics from multiple data reductions. The reflection data sets must be used to protein phasing solutions (SAD, MAD, SIR, MIR ,SIRAS, MIRAS). Normally, there are multiple data sets.

6. **-m** Followed by the one of following programs for molecular replacement:

   [AMORE | CNS | XPLOR | EPMR | MOLREP | BEAST | PHASER | COMO]

   For example: -m amore

7. **-p** Followed by the one of following program names for phasing:

   [CNS | XPLOR | MLPHARE | SOLVE | SHELX | SNB | BnP | BP3 | SHARP | PHASER | PHASES | WARP]

   For example: -p CNS

   Note: if the program that you used for phasing is not in the above list, you may still give the program name. Some information (like heavy atom coordinates) may still be extracted, if the produced file is in PDB or mmCIF format.

8. **-d** Followed by the one of following program names for density modification:

[CNS | XPLOR | DM | RESOLVE | SOLOMON | SHELXE | SHARP]

For example: -d CNS

9. **-r** Followed by one of the following program names for final structure refinement. [CNS | XPLOR | REFMAC5 | SHELX | TNT | BUSTER | PROLSQ | NUCLSQ | RESTRAIN | PHENIX | MAIN]

   For example: -r CNS

   Note: if the program that you used for final structure refinement is not in the above list, you may still give the program name. Some information (like atom coordinates) may still be extracted, if the produced file is in PDB or CIF format. (use -r program_name)

10. **-iPDB** Followed by a input file with PDB format.

    For example: -iPDB test1.pdb

    Note: The PDB files are usually generated from heavy atom phasing (heavy atom coordinates) or the final structure refinement.

11. **-iCIF** Followed by a input file with CIF format.

    For example: -iCIF deposit_cns.cif

    Note: This file can be produced during crystal structural determination. For instance: if you use MLPHARE for locating heavy atom position and do heavy atom phasing refinement, a file in mmCIF format will be generated. This file will contain statistics for heavy atom phasing. Another instance, if you use CNS for final structure refinement, running the deposit.inp macro will produce a CIF file containing the model coordinates and refinement statistics.

12. **-iLOG** Followed by one or more input LOG files

    For example: -iLOG mad_sdb.dat mad_summary.dat

    Note: Log files are usually generated during crystal structural determination. The format depends on the program used. They may contain phasing statistics or heavy atom coordinates. For instance, when people use CNS for heavy atom phasing, they will generate a file (e.g. mad_sdb.dat) which contains the heavy atom coordinates and a file (e.g. mad_summary.dat) which contains phase refinement statistics.

13. **-iENT** Followed by the either an mmCIF file or the data_template.text

    For example: -iENT data_template.text

    Note: The file data_template.text must be generated by the program **extract** using the command '**extract** -pdb coordinate_file'. It contains the full chemical sequence and related information to be filled for each macromolecule in the solved structure. The file is shown in **Appendix**

14. **-idat** Followed by reflection data used for refinement.

    For example: -idat reflection_data_file

    Note: This option is very special. It can be used ONLY with HKL/Scalepack output file. HKL/SCALEPACK does not export the average I/Simgal (overall and with resolution shells), but the items are required for PDB deposition. **pdb_extract** can calculate them for you when providing the data for refinement. The -s and -idat must be used together (for example: -s program_name_scaling -iLOG log_file -idat reflection_data_file)

Examples of pdb_extract using Unix command option TOP

You can extract statistics separately from each step of structure determination applications (index, data processing, heavy atom phasing, density modification, molecular replacement and final structure refinement), or you can put all the steps together, which is a complete deposition.

Note: option -iLOG may be followed by several LOG files for some program.

1. Extracting information from indexing:
   pdb_extract -i program_index -iLOG log_file -o output_file
2. Extracting information from data scaling LOG files (for refinement):
   pdb_extract -s program_name_scaling -iLOG log_file -o output_file_name

   Note: HKL/SCALEPACK does not export < I/Simgal >, but the item is required for the PDB deposition. **pdb_extract** can calculate this for you when providing the data for refinement. The command is

   pdb_extract -s HKL -iLOG log_file -idat reflection_data_file -o output_file_name
3. Extracting information from data scaling LOG files (for phasing):
   pdb_extract -sp program_name_scaling -iLOG log_file1 log_file2 -o output_file_name
4. Extracting information about heavy atom phasing: (The experimental_method must be given for this step)
   pdb_extract -e experimental_method -p program_name_phasing -iPDB pdb_files -iLOG log_files -iCIF mmCIF_files -o output_file_name
5. Extracting information about density modification (output from this program is normally the LOG file):
   pdb_extract -d program_name_for_dm -iLOG log_files -o output_file_name
6. Extracting information about molecular replacement (output from this program is normally the LOG file):
   pdb_extract -m program_name_for_mr -iLOG log_files -o output_file_name
7. Extracting information from final structure refinement:
   pdb_extract -r program_name_for_refinement -iPDB pdb_files -iLOG log_files -iCIF mmCIF_files -o output_file_name
8. **Extracting information for a complete structure:**

```
pdb_extract -e experimental_method \
-i program_name_for_index -iLOG log_files \
-s program_name_for_scaling -iLOG log_files \
-sp program_name_for_scaling -iLOG log_files \
-p program_name_for_phasing -iPDB pdb_files  -iLOG log_files -iCIF mmCIF_files \
-m program_name_for_MR  -iLOG log_files -iCIF mmCIF_files \
-d program_name_for_DM -iLOG log_files \
-r rogram_name_for_refinement -iPDB pdb_files -iLOG log_files -iCIF mmCIF_files \
-iENT data_template.text -o output_file_name \
-o output_file_name
```

Unix command options for pdb_extract_sf TOP

PROGRAM DESCRIPTION:

This program can be used to capture

- Reflection data used for final structure refinement.
- Multiple reflection data (eg. MAD, MIR ...) processed by the software at the data collection site.

EXECUTABLE NAME: pdb_extract_sf

SYNOPSIS: pdb_extract_sf [OPTIONs]... [FILEs]...

ARGUMENT DESCRIPTION: (-o -rt -rp -dt -dp -c -w -idat)

1. **-o** Followed by an output file name.

   Example: -o outfile.cif

NOTE: if you do not specify an output file, a default output file name (pdb_extract- _sf.mmcif) will be used.

2. **-dt** Followed by data type for initial data processing (normally intensity).

   It is followed by F (Amplitude) or I (Intensity)

   Example: -dt I

3. **-dp** Data format for initial data processing. It is followed by one of the following program names:

   HKL/SCALEPACK, DTREK, SAINT, XPREP, XSCALE,3DSCALE, SCALA, OTHER.

   For example: -dp HKL

4. **-c** crystal index. It is followed by crystal number (integers, like 1,2,3, ..)

   Example: -c 2

   (It means the reflection was from the second crystal).

5. **-w** wavelength index.

   It is followed by wavelength number (integers, like 1, 2, 3)

   Example: -w 2

   (This means the data was collected from the crystal using the second wavelength. This is MAD case).

6. **-idat** reflection data file It is followed by data file name

   Example: -idat scalepack.sca

   NOTE: You should always give the combination '-c i, -w j -idat file_name ' in the right order! Here i is the crystal index, j is wavelength index, and file_name is the file name containing the reflections.

7. **-rt** data type used for final structure refinement.

   It is followed by F (Amplitude) or I (Intensity)

   For example: -dt F

8. **-rp** data format in the final structure refinement.

   It is followed by one of the data format names: CNS/CNX/XPLOR, SHELX, TNT, HKL/SCALEPACK, DTREK, SAINT, XPREP, XSCALE,3DSCALE, SCALA,

Examples of pdb_extract_sf using Unix command options TOP

1. **Extracting reflection data used for final structure refinement:**
   pdb_extract_sf -rt data-type -rp data-format-for-refinement -idat data-file-name -o output-file-name

   NOTE: Normally, there is only one data set. If you have several data set used for final refinement, you need to merge all the data in one file.

2. **Extracting reflection data from initial data process (e.g. scaling ...):**
   pdb_extract_sf -dt data_type -dp program_name_for_scaling -c crystal_number_1 -w wavelength_number_1 -idat data_file_name_1 -c crystal_number_2 -w wavelength_number_2 -idat data_file_name_2 ... -o output_file_name

   NOTE: Normally, there are several data sets (e.g. in MAD, MIR ...). These reflections are used for protein phasing. The formats are from the initial data process.

3. **Converting all the reflection data in one mmCIF file (just combine the above two steps):**

```
 pdb_extract_sf \
-rt data-type_refine  -rp data-format-for_refine  -idat data-file-name_refine \
-dt data_type_scaling -dp program_name_for_scaling \
-c crystal_number_1 -w wavelength_number_1 -idat data_file_name_1 \
-c crystal_number_2 -w wavelength_number_2 -idat data_file_name_2 \
...  \
-o output_file_name
```

The output_file_name contains the reflections for refinement and the reflections for protein phasing.

Examples of extract using Unix command options **TOP**

PROGRAM DESCRIPTION:

This program can be used to do the following:

- Generate data template file (data_template.text) which contains entries for author and structural information. It also generated the plain text file (log_script.inp) which contain entries for programs and LOG files.
- Add chain ID, if missing.
- Do structure and sequence alignment to figure out the unique molecular entity in the asymmetric unit.
- Calculate the Matthew coefficient and solvent constant.
- Assembly complete data using the script input file (log_script.inp).

EXECUTABLE NAME: extract

SYNOPSIS: extract [OPTIONs] [FILE]

ARGUMENT DESCRIPTION: (-nmr -pdb -cif -ext -sol -chain)

1. **-nmr** A switch between Xray and NMR system. It should not follow anything.

   NOTE: if you add **-nmr** , it will generate the data_template file for NMR system. if not, it will be for the Xray system (default).

2. **-pdb** Followed by the coordinate PDB file name

   example: -pdb pdb_file_name

3. **-cif** Followed by the coordinate mmCIF file name

   example: -cif mmCIF_file_name

   NOTE: it will generate two plain text files (data_template.text and log_script.inp) with the chemical sequences extracted from the coordinate mmCIF file.

4. **-ext** Followed by the generated file log_script.inp

   example: -ext log_script.inp

5. **-chain** Followed by the pdb file name to add chain ID to the file.

   example: -chain pdb_file_name

6. **-sol** Followed by the data template file to update the Matthew coefficient and solvent constant in the file, if sequence is modified.

   example: -sol data_template.text

Examples of extract using Unix command options TOP

1. Obtain the data template file and the LOG script file

   extract -pdb pdb_file_name (if PDB format)
   or
   extract -cif cif_file_name (if mmCIF format)

   NOTE: You will generate two plain text files. One is the data template file (data_template.text) which contains entries for author and structural information. Another is the script input file (log_script.inp) which contain entries for programs and LOG files.

   Sequences are extracted from SEQRES or coordinate. Unique molecular entity in the asymmetric unit are calculated by the structure and sequence alignment.

2. Obtain the data template file and the LOG script file for NMR system

   extract -pdb pdb_file_name -nmr (if PDB format)
   or
   extract -cif cif_file_name -nmr (if mmCIF format)

   NOTE: if you add -nmr , it will generate the data_template file for NMR system. if not, it will be for the Xray system (default).

3. Assembly the complete mmCIF file for deposition

   extract -ext log_script.inp

   NOTE: you need to fill the necessary LOG files and program names to the log_script.inp according to the instructions inside of the file.

4. Add chain ID to the PDB file

   extract -chain pdb_file_name

   NOTE: If the pdb file has multiple chains, each chain seperated by 'TER' or 'END'. The Chain ID will be given as A, B, C, ...

5. To update the Matthew coefficient and solvent constant

   extract -sol data_template.text

   NOTE: The values in the file data_template.text will be updated, if you modify the residue sequences in the entity_ploy field.

Tables TOP

Below are the two Tables. One is for all the Unix command options and the other is for the software supported by pdb_extract.

TOP Unix command options

Unix command line options consist of three executable components of pdb_extract.
pdb_extract is used to capture the details of molecular replacement, heavy atom phasing, density modification and structure refinement.
pdb_extract_sf is used to convert all other structure factor format to mmCIF format for PDB deposition,
extract is used to generate a data template file (data_template.text) and a script file (log_script.inp).

pdb_extract [OPTION]... [FILE]...

Option	Arguments followed by each option
-o	output file name (default name is pdb_extract.mmcif)
-e	one of experimental methods (MR| SAD| MAD| SIR| MIR| SIRAS| MIRAS)
-i	one of programs for indexing [HKL| DENZO| DTREK| MOSFLM]
-s	one of programs for reflection data scaling (used for refinement) [SCALA| HKL| SCALEPACK| DTREK| SAINT| 3DSCALE| XSCALE| XENGEN| PROSCALE]
-sp	one of programs for reflection data scaling (used for phasing) [SCALA| HKL| SCALEPACK| DTREK| SAINT| 3DSCALE| XSCALE| XENGEN| PROSCALE]
-m	one of programs for molecular replacement [AMORE| CNS| XPLOR| EPMR| MOLREP| BEAST| PHASER| COMO]
-p	one of programs for heavy atomphasing [CNS| XPLOR| MLPHARE| SOLVE| SHELX| SNB| BnP| BP3| SHARP| PHASER| PHASES| WARP]
-d	one of programs for density modification [CNS| XPLOR| DM| RESOLVE| SOLOMON| SHELXE| SHARP]
-r	one of programs for final structure refinement [CNS| XPLOR| REFMAC5| SHELX| TNT| BUSTER| PROLSQ| NUCLSQ| RESTRAIN| PHENIX| MAIN]
-ilog	the input file with format corresponding to the programused.
-ipdb	the input file with PDB format.
-icif	the input file with mmCIF format.
-ient	the input file data_template.text. (for complete sequence.) (It is generated by 'extract -pdb pdbfile')
-idat	the reflection data file to get <I/Sigmal> (optional)

pdb_extract_sf [OPTION]... [FILE]...

-o	output file name (default name is pdb_extract_sf.mmcif).
-rt	data type (I or F) in the reflection data file (used for final structure refinement!)
-rp	One of data formats (CNS| mmCIF| SHELX| TNT| HKL/Scalepack| DTrek| SAINT| OTHER) (used for final structure refinement!)
-dt	data type (I or F) after data reduction at beamline. (used for phase determination!)
-dp	One of programs for data reduction (HKL/Scalepack, DTrek, SAINT| OTHER). (used for phase determination!)
-c	crystal number (like 1, 2, 3...) used for diffraction.

-cif	input coordinate file name (mmCIF format)
-ext	input script file name log_script.inp (It must be generated by '**extract** -pdb pdb_file_name')
-chain	input coordinate file name (mmCIF format)
-sol	the data template file (data_template.text)
-NMR	(A switch between Xray & NMR, Nothing follows it)

Supported crystallographic software lists **TOP**

Software applications supported by **pdb_extract** are listed in the Table bellow.

Category	Software	Versions	References
Data collection integration reduction scaling averaging	HKL/HKL2000 SCALEPACK/DENZO	1.30 , 1.96 , 1.97.9, 1.98.7	Otwinowski & Minor (1997)
	D*trek	7.0SSI , 7.11 , 9.2 , 9.7 , 9.9.2	Pflugrath (1997)
	SCALA (CCP4)	CCP4(v4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02, 6.10)	Evans (1997)
	XDS/XSCALE	Nov. 2005 , June 2006, Dec. 2006 , Mar. 2007 , July 2007, January 2009	Kabsch
	MOSFLM	6.2.2 , 6.2.3, 6.2.5, 7.0.1	Leslie(1998)
	X-gen	5.5.5 , 5.8.3	Andrew J. Howard
	SAINT	V6.35A, V7.03A	Bruker (2002)
	3DSCALE/PROSCALE	N/A	Fu (2005)
Molecular replacement	CNS/CNX	0.9 , 1.0, 1.1 , 1.2	Brunger et al. (1998)
	XPLOR	3.1 , 3.851	Brunger et al (1998)
	AMORE (CCP4)	CCP4(V4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02)	Navaza (1994)
	MOLREP (CCP4)	CCP4(V4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02)	Vagin & Teplyakov (1997)
	EPMR	2.5	Kissinger et al. (1999)
	PHASER	1.2, 1.3 , 2.0 , 2.1	Read(2001)
	BEAST	1.1.1	Read(2001)
	COMO	1.2	Tong(1996)
Heavy atom phase determination	CNS/CNX	0.9 , 1.0 , 1.1 , 1.2	Brunger et al. (1998)
	XPLOR	3.1 , 3.851	Brunger et al (1998)
	SOLVE	2.0 , 2.01, 2.02, 2.06, 2.08 , 2.09, 2.10, 2.11, 2.13	Terwilliger & Berendzen. (1999)
	MLPHARE (CCP4)	CCP4(V4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02)	CCP4 (1994)
	SHARP/autoSHARP	1.3.x , 1.4.0 , 2.0 , 2.01 , 2.04 , 2.2.0	Fortelle & Bricogne (1997)
	SHELXD/SHELXS	97	Sheldrick (1997)
	PHASES	95	Furey (1997)
	PHASER	2.0 , 2.1	Read(2001)
	SnB	2.0 , 2.1 , 2.2	Weeks & Miller (1999)
	BnP	0.93 , 1.0 , 1.02 , 1.05	Weeks et al. (2002)
	BP3	1.0	Navraj S. Pannu(2003)

Density modification	CNS/CNX	0.9 , 1.0 , 1.1 , 1.2	Brunger et al. (1998)
	XPLOR	3.1 , 3.851	Brunger et al (1998)
	DM (CCP4)	CCP4(v4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02)	Cowtan (1994)
	SOLOMON (CCP4)	CCP4(V4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02)	Abrahams & Leslie (1996)
	RESOLVE	2.0 , 2.01, 2.02, 2.06, 2.08 , 2.09, 2.10, 2.11, 2.13	Terwilliger (2000)
	SHELXE	97	Sheldrick (1997)
Structure refinement	CNS/CNX	0.9 , 1.0 , 1.1 , 1.2	Brunger et al. (1998)
	XPLOR	3.1 , 3.851	Brunger et al (1998)
	REFMAC5 (CCP4)	CCP4(V4.0 , 5.0 , 5.01, 5.02 , 6.0 , 6.01 , 6.02, 6.13)	Murshudov (1997)
	PHENIX	1.0 , 1.1a , 1.22a , 1.3.1 , 1.3b, 1.3,1.4, 1.6	Adams et al (2002)
	SHELXL	97	Sheldrick (1997)
	TNT	5F	Tronrud (1997)
	BUSTER-TNT	1.0.2 -- 2.9	G.Bricogne (1993)
	ARP/wARP	5.0 , 6.1.1 , 7.0	Lamzin & Wilson, (1997)
	RESTRAIN (CCP4)	4.6	CCP4 (1994)
NMR structure determination	CNS/CNX	1.1 , 1.2	Brunger et al. (1998)
	XPLOR	3.1 , 3.851	Brunger et al (1998)
	CYANA	2.0	Güntert (1997)
	Xplor-NIH	2.13	G. Marius Clore(2003)

References TOP

1. Z. Otwinowski and W. Minor. (1997). Processing of X-ray Diffraction Data Collected in Oscillation Mode. *Methods in Enzymology*, Volume **276**: Macromolecular Crystallography, part A, p.307- 326
2. Pflugrath JW (1999). The finer things in X-ray diffraction data collection. *Acta Cryst.* **D55** 1718-25
3. Zheng-Qing Fu (2005), Three-dimensional model-free experimental error correction of protein crystal diffraction data with free-R test *Acta Cryst.* **D61** 1643-1648
4. SAINT V6.35A, Bruker Analytical X-Ray Systems, Madison, WI, (2002).
5. Evens, P. R. (1997). "the Scala" *Joint CCP4 and ESF-EACBM Newsletter.* **33**, 22-24
6. Kabsch, W. (1993). Automatic processing of rotation diffraction data from crystals of initially unknown symmetry and cell constants. *J. Appl. Cryst.* **26**, 795-800.
7. Leslie A. G. W. (1998), *J. Appl. Cryst.* **30**, 1036-1040.
8. Brunger, A.T., Adams, P.D., Clore, G.M., DeLano, W.L., Gros, P., Grosse-Kunstleve, R.W., Jiang, J.-S., Kuszewski, J., Nilges, N., Pannu, N.S., Read, R.J., Rice, L.M., Simonson, T., and Warren, G.L. (1998). Crystallography and NMR system (CNS): A new software system for macromolecular structure determination. *Acta Cryst.* **D54**, 905-921.
9. Navaza J. (1994) AMoRe: an Automated Package-- --for Molecular Replacement. *Acta Cryst.* **D50**, 157-163.

10. Vagin A. , Teplyakov A. (1997) , MOLREP: an automated program for molecular replacement. *J. Appl. Cryst.* **30**, 1022-1025.
11. Charles R. Kissinger, Daniel K. Gehlhaar & David B. Fogel, (1999) Rapid automated molecular replacement by evolutionary search. *Acta Cryst.* , **D55**, 484-491
12. R. J. Read (2001) Pushing the boundaries of molecular replacement with maximum likelihood. *Acta Cryst.* **D57**, 1373-1382
13. Terwilliger, T.C. and J. Berendzen. (1999) Automated MAD and MIR structure solution. *Acta Cryst.* **D55**, 849-861.
14. COLLABORATIVE COMPUTATIONAL PROJECT, NUMBER 4. 1994. The CCP4 Suite: Programs for Protein Crystallography. *Acta Cryst.* **D50**, 760-763
15. E. de La Fortelle & G. Bricogne (1997) Maximum-Likelihood Heavy-Atom Parameter Refinement for the Multiple Isomorphous Replacement and Multiwavelength Anomalous Diffraction Methods. *Methods in Enzymology* **276** 472-494
16. Furey, W. & Swaminathan, S. (1997), PHASES-95: A Program Package for the Processing and Analysis of Diffraction Data from Macromolecules. *Methods in Enzymology,* **277**, 590-620
17. Weeks, C.M. & Miller, R. (1999). The design and implementation of SnB v2.0, *J. Appl. Cryst.***32**, 120-124.
18. Weeks, C.M., Blessing, R.H., Miller, R., Mungee, S., Potter, Rappleye, A., Simith, G.D. Xu, H., Furey, W. (2002), Towards automated protein structure determination: BnP, the SnB-PHASES Interface. *Z. Kristallogr.* **217**, 686-693
19. Navraj S. Pannu,Airlie J. McCoy, Randy J. Read(2003), Application of the-- --complex multivariate normal distribution to crystallographic methods with insights into multiple isomorphous replacement phasing *ACTA CRYSTALLOGR.,SECT.D.* **59**, 1801-1808
20. Sheldrick G. (1997) The SHELX-97 homepage http://shelx.uni-ac.gwdg.de/SHELX/
21. K. Cowtan (1994), *Joint CCP4 and ESF-EACBM Newsletter on Protein Crystallography.* **31**, p34-38.
22. Abrahams J. P. and Leslie A. G. W.(1996). *Acta Cryst.* **D52**, 30-42
23. Terwilliger, T. C. (2000) Maximum likelihood-- --density modification. *Acta Cryst.* **D56**, 965-972.
24. G. Bricogne (1993), Direct Phase-- --Determination by Entropy Maximisation and Likelihood Ranking: Status Report and Perspectives. *ACTA CRYSTALLOGR.,SECT.D* **49**, 37-60
25. Tronrud, D, E., (1997). The TNT Refinement Package. in Macromolecular Crystallography, Part B, *Methods Enzymol.* **277**, 306-318
26. Lamzin, V.S. & Wilson, K.S. (1997). Automated refinement for protein crystallography. *Methods Enzymol.* (Carter, C. & Sweet, B. eds.) **277**, 269-305
27. G.N. Murshudov, A.A.Vagin and E.J.Dodson, (1997) Refinement of Macromolecular Structures by the Maximum-Likelihood Method. *Acta Cryst.* **D53**, 240-255.
28. P.D. Adams, R.W. Grosse-Kunstleve,-- --L.-W. Hung, T.R. Ioerger, A.J. McCoy, N.W. Moriarty, R.J. Read, J.C. Sacchettini, N.K. Sauter and T.C. Terwilliger.(2002) PHENIX: building new software for automated crystallographic structure determination. *Acta Cryst.* **D58**, 1948-1954
29. Güntert, P., Mumenthaler, C. & Wüthrich, K. (1997). Torsion angle dynamics for NMR structure calculation with the new program DYANA. *J. Mol. Biol.* **273**, 283-298.
30. C.D. Schwieters, J.J. Kuszewski, N. Tjandra and G.M. Clore (2003), "The Xplor-NIH NMR Molecular Structure Determination Package," *J. Magn. Res.* **160**, 66-74.

Frequently asked questions TOP

1. Question: What should I do, if the program that I used for solving a structure is not supported by pdb_extract?

   Answer: If the program exports log files in mmCIF format or the PDB format for atomic coordinates, you just give the program name, information is still extracted. However, if the unknown program only generates LOG file which is neither mmCIF no PDB format, please send us deposit@deposit.rcsb.org the log file and the program name. We will add the program to our list.

2. Question: If I used high throughput mode to determine the structure, which may involve several programs and several steps (for example, phase determination & density modification), how can I use the LOG file to **pdb_extract**?

   Answer: If each program generates its own output file, please follow the normal extraction procedure, which means to apply each program name and LOG file to the **pdb_extract.**

   For example, if the high throughput structure determination involves SOLVE (phase determination) and RESOLVE (density modification) and each program exports its own log file (solve.prt from SOLVE, and

resolve.log from RESOLVE), you can use **pdb_extract** in the following way
pdb_extract -e MAD -p SOLVE -ilog solve.prt -d RESOLVE -ilog resolve.log

If there is only one large LOG file (e.g. phase.log) generated in the high throughput mode, you may only apply this log file to **pdb_extract.** For example,
pdb_extract -e MAD -p prog_A -ilog phase.log -p prog_B -ilog phase.log -d prog_C -ilog phase.log.

3. Question: If I used several programs (for example CNS, PHENIX, and REFMAC5) to do final refinement, which log file should I use for **pdb_extract**?

   Answer: you can use the LOG file and the program which exports the final PDB coordinate file. For example, if REFMAC5 is the last program to produce the PDB file, your extraction can be
   pdb_extract -r REFMAC5 -ipdb pdb_file -icif native.refmac

4. Question: If I used several programs (for example SOLVE, BP3, MLPHARE) to determine phase, which log file should I use for **pdb_extract**?

   Answer: you can use the LOG file and the program which produced the phase. For example, if SOLVE is the last program to get the final phase, your extraction can be
   pdb_extract -e MAD -p SOLVE -ilog solve.prt.

   However, if other programs were also important for your phase determination and you want to add other program's name to the data base, you can do the following (no LOG files for other programs) :
   pdb_extract -e MAD -p SOLVE -ilog solve.prt -p BP3 -p MLPHARE

5. Question: If it takes really long time between each crystallographic step (like from phasing to refinement), I may not keep the old log files.

   Answer: I suggest you apply the **pdb_extract** program as soon as you finished this step. Then, you will generate one mmCIF file for this step. You may only keep this mmCIF file somewhere in your disk. Finally, you just use the same program to merge all the steps together. (Your options should all be -icif cif_file_name ...).

6. Question: How do I know that I obtained the correct mmCIF file?

   Answer: Normally the program gives a warning message. But it is a good idea to check if the mmCIF file has the right PDB coordinates (_atom_site. ?). If you encounter an error when running the program, please take a look if you used the correct options. Otherwise, send a message to deposit@deposit.rcsb.org

7. Question: I have installed the CCP4 suit. do I have to install the **pdb_extract** again.

   Answer: You do not have to install the standalone version of pdb_extract, if you prefer to do validation by the ADIT server. In addition to using the CCP4i interface, you can also do all the Unix command line option under the CCP4 environment.

Explanations of arguments and input/output files **TOP**

The script file test.sh:

```
#!/bin/sh

############### testing command line ####################
# use pdb_extract to extract the required statistics and get a mmcif file.
pdb_extract  -e  MAD \
-s HKL -ilog input_data/sclepack1.log  \
-p CNS -iLOG  input_data/mad_sdb.dat input_data/mad_summary.dat input_data/mad_fp.dat \
-d CNS -iLOG input_data/density_modify.dat  \
-r CNS -iCIF input_data/deposit_cns.mmcif \
-iENT input_data/data_template.text \
-o Example_1.cif
```

```
# use pdb_extract_sf to convert the structure factor to mmcif format.
pdb_extract_sf  -rt F -rp CNS -idat input_data/gere-nat.cv  \
-dt I -dp HKL -c 1 -w 1 -idat input_data/w1.sca  \
-c 1 -w 2 -idat input_data/w2.sca  \
-c 1 -w 3 -idat input_data/w3.sca -o Example_1.sf.cif


# move the files to some directory and delete some log files.
mv Example_1.cif deposit
mv Example_1.sf.cif deposit
```

The alternative script file test_script.sh:

```
#!/bin/sh

############### testing the script inp ####################

# use extract to run everything in example_1.inp and get a mmcif file.
extract -ext input_data/example_1.inp


# move the files to some directory and delete some log files.
mv script_example_1.cif deposit/
mv script_example_1_sf.cif deposit/
#rm -f *log *err procheck* SEQUENCE.DAT *ERR validation.alignment
```

The output files:

After you run the above commands (for example ./test.sh), you will get the following files in the directory pdb-extract-vX.X/examples/Example_1/deposit/

- Example_1.cif is the merged mmCIF file created by "pdb_extract"
- Example_1.sf.cif is the structure factor created by "pdb_extract_sf"

You can deposit the two files Example_1.sf.cif and either Example_1.cif to ADIT

The input files:

```
MAD experiment
    Phasing calculation by program CNS (version 1.1).
    Density modification by program CNS (version 1.1).
    Final structure refinement by program CNS (version 1.1).
Data files:
    pdb-extract-vX.X /examples/Example_1/input_data/mad_sdb.dat
           o File format: CNS log format.
           o File source: run CNS (mad_phase.inp)
           o Data to be extracted: heavy atom coordinates, B factors, etc.
    pdb-extract-vX.X /examples/Example_1/input_data/mad_summary.dat
           o File format: CNS log format.
           o File source: run CNS (mad_phase.inp)
           o Data to be extracted: all the phasing statistics
    pdb-extract-vX.X /examples/Example_1/input_data/mad_fp.dat
           o File format: CNS log format.
           o File source: run CNS (mad_phase.inp)
           o Data to be extracted: wavelengths, f_prime, f_double_prime.
    pdb-extract-vX.X /examples/Example_1/input_data/density_modify.dat
           o File format: CNS log format.
           o File source: run CNS (fourier_map_dm.inp)
```

```
        o Data to be extracted: FOM after density modification, dm method
  pdb-extract-vX.X /examples/Example_1/input_data/deposit_cns.mmcif
        o File format: mmCIF
        o File source: run CNS (deposit_mmcif.inp)
        o Data to be extracted: the atom coordinates and B factors and
          structure refinement statistics.
  pdb-extract-vX.X /examples/Example_1/input_data/data_template.text
        o File format: mmCIF
        o File source: Generated by ' extract -pdb pdb_file_name'.
        o Data to be extracted: a complete chemical sequence.
```

Appendix TOP

Data template file: (data_template.text) TOP

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                    THE DATA_TEMPLATE.TEXT FILE FOR X-RAY
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                           NOTES AND REMINDER
The data template file contains data entries for unique chemical sequences
present in the structure and other non-electronically captured information.

PLEASE CHECK CATEGORIES 1 & 2: Before proceeding any further, make necessary
corrections here so that all information in these categories are complete
and correct.

You may choose to fill in CATEGORIES (3-19) either here or later in ADIT.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                        GUIDELINES FOR USING THIS FILE
  1. Only strings included between the 'lesser than' and 'greater than'
     signs (<.....>) will be parsed for evaluation by the program. Therefore,
     DO NOT write either on the left or right of the 'less than' and 'greater
     than' signs respectively.

  2. All alphanumeric values or strings that you include in the different
     categories should be within double-quotes. Blank spaces or carriage
     returns within a pair of double quotes are ignored by the program.
     DO NOT use double quotes (") within strings that you enter.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

~~~~~~~~~~~~~~~~~~~~~~~~~~~START INPUT DATA BELOW~~~~~~~~~~~~~~~~~~~~~~

================CATEGORY 1:   Crystallographic Data=======================
Enter crystallographic data

<space_group = "P 1 21 1"> (use International Table conventions)
<space_group_number = "? ">

<unit_cell_a     = "  56.800 " >
<unit_cell_b     = "  69.950 " >
```

```
<unit_cell_c     = "  60.530 " >
<unit_cell_alpha = " 90.00 " >
<unit_cell_beta  = "114.50 " >
<unit_cell_gamma = " 90.00 " >


================CATEGORY 2:   Sequence Information ======================
Enter one letter sequence for each polymeric entity in asymmetric unit

-------------------------------------------------------------------------
                          SOME DEFINITIONS
     An ENTITY is defined as any unique molecule present in the asymmetric
     unit. Each unique biological polymer (protein or nucleic acids) in the
     structure is considered an entity. Thus, if there are five copies of
     a single protein in the asymmetric unit, the molecular entity is still
     only one. Water and non-polymers like ions, ligands and sugars are
     also entities.

     Here we only consider the sequences of polymeric entities (protein or
     nucleic acid).

               GUIDELINES FOR COMPLETING THIS CATEGORY
      * In a PDB or mmCIF format file, all residues of a single polymeric
      entity should have one chain ID. Multiple copies of the same entity
      should each be assigned a unique chain ID. The multiple chain IDs
      should be separated by commas as 'A,B,C,...'. If incorrect chain IDs
      are used the entity groups extracted by this program will not be
      correct. To avoid this, make necessary corrections in the PDB or mmCIF
      file used to generate the data_template file and regenerate the
      data_template.text file. Alternatively, edit the extracted sequence
      in this file to correctly represent the sequence and chain IDs of each
      polymeric entity.

      * In addition to chain IDs, this program uses distance geometry to
      asses if there are any breaks in the polymer sequence. These breaks
      may occur due to missing residues (not included in the model due to
      missing electron density) or due to poor geometry. Four question marks
      '????' are used to denote these chain breaks. Replace these question
      marks with the sequence of residues missing from the coordinates. Also
      add any residues missing from the N- and/or C-termini here.

      * If there are non-standard residues in the coordinates, this program
      lists them according to the three letter code used in the coordinate
      file as (ABC). If all the residues in your sequence are nonstandard,
      check and edit the sequence manually to represent it correctly in this
      file.

      * If any residue was modeled as Ala or Gly due to lack of the side-chain
      density, the sequence extracted here will represent them as A or G
      respectively. Correct this to the original sequence that was present in
      the crystal.
-------------------------------------------------------------------------

        Below is the one letter chemical sequence extracted from your PDB
        coordinate file. The molecular entities are grouped and listed
        together.

PLEASE CHECK THE SEQUENCE of each entity carefully and modify it, as necessary.
Make sure that you REVIEW THE FOLLOWING:
```

```
   * chain breaks due to missing residues,
   * missing residues in the N- and/or C-termini,
   * non-standard residues and
   * cases of residues modeled as Ala or Gly due to missing side-chain density.

<molecule_entity_id="1" >
<molecule_entity_type="polypeptide(L)" >
<molecule_one_letter_sequence="
MENFQKVEKIGEGTYGVVYKARNKLTGEVVALKKIRLDTETEGVPSTAIREISLLKELNHPNIVKLLDVI
HTENKLYLVFEFLHQDLKKFMDASALTGIPLPLIKSYLFQLLQGLAFCHSHRVLHRDLKPQNLLINTEGA
IKLADFGLARAFGVPVRTYTHEVVTLWYRAPEILLGCKYYSTAVDIWSLGCIFAEMVTRRALFPGDSEID
QLFRIFRTLGTPDEVVWPGVTSMPDYKPSFPKWARQDFSKVVPPLDEDGRSLLSQMLHYDPNKRISAKAA
LAHPFFQDVTKPVPHLRL" >
< molecule_chain_id="A" >
< target_DB_id=" " > (if known)

<molecule_entity_id="2" >
<molecule_entity_type="polypeptide(L)" >
<molecule_one_letter_sequence="
MSHKQIYYSDKYDDEEFEYRHVMLPKDIAKLVPKTHLMSESEWRNLGVQQSQGWVHYMIHEPEPHILLFR
RPLPKKPKK" >
< molecule_chain_id="B" >
< target_DB_id=" " > (if known)

<molecule_entity_id=" " >
<molecule_entity_type=" " >
<molecule_one_letter_sequence="  " >
<molecule_chain_id=" " >

<target_DB_id=" " >  (if known)


================CATEGORY 3:   Contact Authors=============================
Enter information about the contact authors.
    Note: items marked by (e.g. ) are manditory.
          PI information should be always given.

1.  Information about the Principal investigator (PI) should be given.

<contact_author_PI_id = "1 ">           (must be given 1)
<contact_author_PI_salutation = " ">     ( Dr./Prof./Mr./Mrs./Ms.)
<contact_author_PI_first_name = " ">      (e.g. John)
<contact_author_PI_last_name = " ">        (e.g. Rodgers)
<contact_author_PI_middle_name = " ">
<contact_author_PI_role = " ">   (e.g. investigator/responsible scientist)
<contact_author_PI_organization_type = " ">  (e.g. academica/commercial/goverment/other)
<contact_author_PI_email = " ">        (e.g.   name@host.domain.country)
<contact_author_PI_address = " ">             (e.g. 610 Taylor road)
<contact_author_PI_city = " ">                (e.g. Piscataway)
<contact_author_PI_State_or_Province = " ">   (e.g.  New Jersey)
<contact_author_PI_Zip_Code = " ">            (e.g.  08864)
<contact_author_PI_Country = " ">          (e.g.  UNITED STATES)
<contact_author_PI_fax_number = " ">
<contact_author_PI_phone_numer = " ">

2. Information about other contact authors

<contact_author_id = "2 ">       (e.g. 2,3,4..)
<contact_author_salutation = " ">
```

```
<contact_author_first_name = " ">
<contact_author_last_name = " ">
<contact_author_middle_name = " ">
<contact_author_role = " ">
<contact_author_organization_type = " ">
<contact_author_email = " ">
<contact_author_address = " ">
<contact_author_city = " ">
<contact_author_State_or_Province = " ">
<contact_author_Zip_Code = " ">
<contact_author_Country = " ">
<contact_author_fax_number = " ">
<contact_author_phone_numer = " ">


...(add more if needed)...

================CATEGORY 4:   Structure Genomics=========================
If it is the structure genomics project, give the information

<SG_project_id = " 1">
<SG_project_name = " ">        (e.g. NPPSFA/PSI, Protein Structure Initiative)
<full_name_of_SG_center = " ">   (e.g. Berkeley Structural Genomics Center)


================CATEGORY 5:   Release Status==============================
Enter release status for the coordinates,structure_factor, and sequence

   Status for sequence should be chosen from one of the following:
   (release now, hold for release)

   Status for others should be chosen from one of the following:
  (release now, hold for publication,  hold for 4 weeks, hold for 6 weeks,
   hold for 6 months, hold for 1 year)

<Release_status_for_coordinates = " ">      (e.g. release now)
<Release_status_for_structure_factor = " ">
<Release_status_for_sequence = " ">

================CATEGORY 6:   Title=======================================
Enter the title for the structure

<structure_title = " ">     (e.g. Crystal Structure Analysis of the B-DNA)
<structure_details = " ">


================CATEGORY 7: Authors of Structure==========================
Enter authors of the deposited structures (e.g. Surname, F.M.)

<structure_author_name = " ">
<structure_author_name = " ">
<structure_author_name = " ">
<structure_author_name = " ">
...add more if needed...


================CATEGORY 8:   Citation Authors============================
Enter author names for the publications associated with this deposition.
```

```
      The primary citation is the article in which the deposited coordinates
      were first reported. Other related citations may also be provided.

1. For the primary citation
<primary_citation_author_name = " ">    (e.g. Surname, F.M.)
<primary_citation_author_name = " ">
<primary_citation_author_name = " ">
<primary_citation_author_name = " ">
...add more if needed...

2. For other related citations  (if applicable)
<citation_author_id = " ">    (e.g. 1, 2 ..)
<citation_author_name = " ">
<citation_author_name = " ">
<citation_author_name = " ">
<citation_author_name = " ">
...add more if needed...


...(add more other citations if needed)...

================CATEGORY 9:   Citation Article============================
Enter citation article (journal, title, year, volume, page)

      If the citation has not yet been published, use 'To be published'
      for the category 'journal_abbrev' and leave pages and volume blank.

1. For primary citation
<primary_citation_id = "primary">
<primary_citation_journal_abbrev = " ">     (e.g. to be published)
<primary_citation_title = " ">
<primary_citation_year = " ">
<primary_citation_journal_volume = " ">
<primary_citation_page_first = " ">
<primary_citation_page_last = " ">

2. For other related citation (if applicable)
<citation_id = "1 ">               (e.g. 1, 2, 3 ...)
<citation_journal_abbrev = " ">
<citation_title = " ">
<citation_year = " ">
<citation_journal_volume = " ">
<citation_page_first = " ">
<citation_page_last = " ">


...(add more citations if needed)...


================CATEGORY 10:   Molecule Names=============================
Enter the names of the molecules (entities) that are in the asymmetric unit

NOTE: The number of molecular names should be the same as CATEGORY 2 !
      The name of molecule should be obtained from the appropriate
      sequence database reference, if available. Otherwise the gene name or
      other common name of the entity may be used.
      e.g. HIV-1 integrase for protein
           RNA Hammerhead Ribozyme for RNA
```

```
<molecule_name = " ">    (entity 1)
<molecule_name = " ">    (entity 2)

...(add more if needed)...


================CATEGORY 11:   Molecule Details============================
Enter additional information about each entity, if known. (optional)

      Additional information would include details such as fragment name
      (if applicable), mutation, and E.C.number.

1. For entity 1
<Molecular_entity_id = "1 ">       (e.g. 1, 2, ...)
<Fragment_name = " ">             (e.g. ligand binding domain, hairpin)
<Specific_mutation = " ">         (e.g. C280S)
<Enzyme_Comission_number = " ">   (if known: e.g. 2.7.7.7)

2. For entity 2
<Molecular_entity_id = "2 ">
<Fragment_name = " ">
<Specific_mutation = " ">
<Enzyme_Comission_number = " ">

...(add more if needed)...

================CATEGORY 12:   Genetically Manipulated Source=============
Enter data in the genetically manipulated source category

      If the biomolecule has been genetically manipulated, describe its
      source and expression system here.

1. For entity 1
<Manipulated_entity_id = "1 ">               (e.g. 1, 2, ...)
<Source_organism_scientific_name = " ">      (e.g. Homo sapiens)
<Source_organism_gene = " ">                 (e.g. RPOD, ALKA...)
<Source_organism_strain = " ">               (e.g. BH10 ISOLATE, K-12...)
<Expression_system_scientific_name = " ">    (e.g. Escherichia coli)
<Expression_system_strain = " ">             (e.g. BL21(DE3))
<Expression_system_vector_type = " ">        (e.g. plasmid)
<Expression_system_plasmid_name = " ">       (e.g. pET26)
<Manipulated_source_details = " ">           (any other relevant information)

2. For entity 2
<Manipulated_entity_id = "2 ">
<Source_organism_scientific_name = " ">
<Source_organism_gene = " ">
<Source_organism_strain = " ">
<Expression_system_scientific_name = " ">
<Expression_system_strain = " ">
<Expression_system_vector_type = " ">
<Expression_system_plasmid_name = " ">
<Manipulated_source_details = " ">


...(add more if needed)...

================CATEGORY 13:   Natural Source============================
Enter data in the natural source category  (if applicable)
```

```
    If the biomolecule was derived from a natural source, describe it here.


1. For entity 1
<natural_source_entity_id = " ">          (e.g. 1, 2, ...)
<natural_source_scientific_name = " ">    (e.g. Homo sapiens)
<natural_source_organism_strain = " ">    (e.g. DH5a , BMH 71-18)
<natural_source_details = " ">            (e.g. organ, tissue, cell ..)


2. For entity 2
<natural_source_entity_id = " ">
<natural_source_scientific_name = " ">
<natural_source_organism_strain = " ">
<natural_source_details = " ">


...(add more if needed)...

================CATEGORY 14:  Synthetic Source==============================
If the biomolecule has not been genetically manipulated or synthesized,
describe its source here.

1. For entity 1
<synthetic_source_entity_id = " ">          (e.g. 1, 2, ...)
<synthetic_source_description = " ">      (if known)

2. For entity 2
<synthetic_source_entity_id = " ">
<synthetic_source_description = " ">

...(add more if needed)...


================CATEGORY 15:   Keywords=====================================
Enter a list of keywords that describe important features of the deposited
structure.

      For example, beta barrel, protein-DNA complex, double helix,
      hydrolase, structural genomics etc.

<structure_keywords = " ">

================CATEGORY 16:   Biological Assembly=========================
Enter data in the biological assembly category (if applicable)

      Biological assembly describes the functional unit(s) present in the
      structure. There may be part of a biological assembly, one or more
      than one biological assemblies in the asymmetric unit.
      Case 1
      * If the asymmetric unit is the same as the biological assembly
        nothing special needs to be noted here.
      Case 2
      * If the asymmetric unit does not contain a complete biological unit.
        Please provide symmetry operations including translations required
        to build the biological unit.
        (example:
        The biological assembly is a hexamer generated from the dimer
```

```
        in the asymmetric unit by the operations:  -y, x-y-1, z-1 and
        -x+y, -x-1, z-1.)
      Case 3
      * If the asymmetric unit has multiple biological units
        Please specify how to group the contents of the asymmetric unit into
        biological units.
        (example:
        The biological unit is a dimer. There are 2 biological units in the
        asymmetric unit (chains A & B and chains C & D).

<biological_assembly = " ">     (biological unit 1)
<biological_assembly = " ">     (biological unit 1)

....(add more if needed)....

================CATEGORY 17:   Methods and Conditions=====================
Enter the crystallization conditions for each crystal

1. For crystal 1:
<crystal_number = "1 ">            (e.g. 1, 2, ...)
<crystallization_method = " ">      (e.g. vapor diffusion, hanging drop)
<crystallization_pH = " ">          (e.g. 7.5 ...)
<crystallization_temperature = " "> (e.g. 298) (in Kelvin)
<crystallization_details = " ">  (e.g. PEG 4000, NaCl etc.)

2. For crystal 2:
<crystal_number = " ">
<crystallization_method = " ">
<crystallization_pH = " ">
<crystallization_temperature = " ">
<crystallization_details = " ">

...(add more if needed)...

================CATEGORY 18:   Crystal Property============================
Enter solvent content, Matthews coefficient
      These values were calculated based on the sequence as shown in
      CATEGORY 2. If there are missing residues, you need to add the
      missing residues and re-run the program to get accurate values.
      (The command to re-run is 'extract -sol data_template.text')

1. For crystal 1:
<crystals_number = " 1 ">                  (e.g. 1, 2, ...)
<crystals_solvent_content = "50.6 ">
<crystals_matthews_coefficient = "2.5 ">
<crystals_mosaicity = " ">    (e.g. 0.5 ...)


2. For crystal 2:
<crystals_number = "  ">
<crystals_solvent_content = "50.6 ">
<crystals_matthews_coefficient = "2.5 ">
<crystals_mosaicity = " ">



...(add more if needed)...
```

```
================CATEGORY 19:   Radiation Source (experiment)============
Enter the details of the source of radiation, the X-ray generator,
and the wavelength for each diffraction.

1. For experiment 1:
<radiation_experiment = "1 ">      (e.g. 1, 2, ...)
<radiation_source = " ">           (e.g. SYNCHROTRON, ROTATING ANODE ...)
<radiation_source_type = " ">      (e.g. NSLS BEAMLINE X8C ...)
<radiation_wavelengths= " ">       (e.g. 1.502 ...)
<radiation_detector = " ">         (e.g. CCD/AREA DETECTOR/IMAGE PLATE ...)
<radiation_detector_type= " ">     (e.g. SIEMENS-NICOLET/RIGAKU RAXIS ...)
<radiation_detector_details = " ">    (e.g. mirrors...)
<data_collection_date = " ">             (e.g. 2004-11-27)
<data_collection_temperature = " ">      (e.g. 100 for crystal  1:)
<data_collection_protocol= " ">          (e.g. SINGLE WAVELENGTH, MAD, ...)
<data_collection_monochromator= " ">     (e.g. GRAPHITE, Ni FILTER ...)

2. For experiment 2:

<radiation_experiment = "2 ">
<radiation_source = " ">
<radiation_source_type = " ">
<radiation_wavelengths= " ">
<radiation_detector = " ">
<radiation_detector_type= " ">
<radiation_detector_details = " ">
<data_collection_data = " ">
<data_collection_temperature = " ">
<data_collection_protocol= " ">
<data_collection_monochromator= " ">


....(add more if needed)....

====================================END=================================
```

script file: (log_script.inp) **TOP**

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                        THE LOG_SCRIPT.INP FILE
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                         NOTES AND REMINDER
This script file is used to enter the names of the crystallographic
software used for structure determination and the log, PDB, mmCIF or
text files generated by them.

PLEASE COMPLETE the ENTRY FIELDS according to the type of your experiment
and use the command 'extract -ext log_script.inp' to obtain the completed
structure data ready for validation and deposition.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                        GUIDELINES FOR USING THIS FILE
```

```
  1. Only strings included between the 'lesser than' and 'greater than'
     signs (<.....>) will be parsed for evaluation by the program. Therefore,
     DO NOT write either on the left or right of the 'less than' and 'greater
     than' signs respectively.

  2. All alphanumeric values or strings that you include in the different
     categories should be within double-quotes. Blank spaces or carriage
     returns within a pair of double quotes are ignored by the program.
     DO NOT use double quotes (") within strings that you enter.

  3. Log files used for generating the deposition should be generated from
     the best (usually the last) trial for each crystallographic software.
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

~~~~~~~~~~~~~~~~~~~~~~~~~~START INPUT DATA BELOW~~~~~~~~~~~~~~~~~~~~~~

===============PART 1: Structure Factor for Final Refinement==============
Enter reflection data file used for final structure refinement

     NOTE:
     *  Usually the highest resolution or best data set is used for the
        refinement. Use that structure factor file here.

     *  In some cases, it may not be possible to collect a complete dataset
        from a single crystal. Thus, multiple data sets have to be scaled
        and merged together for refinement. Use the merged reflection file
        here.

     *  If the reflection data format is not one of those listed below,
        please use OTHER for the data format, and provide an ASCII file
        that has at least five values [H, K, L, I (or F), sigmaI (or sigmaF)]
        for each reflection and seperate each item by one or more spaces.
        Include the test flags as the sixth column in the file (if available).

     *  If the reflection file is in mtz format (e.g. using REFMAC5), convert
        it to mmCIF format using the mtz2various application provided by CCP4.

        Reflection data format:
        CNS|SHELX|TNT|REFMAC5|HKL|SCALEPACK|DTREK|SAINT|SCALA|3DSCALE

<reflection_data_type = "F" >      [enter I (intensity) or F (amplitude)]
<reflection_data_format = "CNS" >
<reflection_data_file_name = " " >

==============PART 2: Structure Factors for Protein Phasing===============
Enter reflection data files used for heavy atom or MAD phasing

     NOTE:
     *  Enter this category if you have more than one complete reflection
        file (e.g. in the case of MAD,SIRAS, MIR). The LOG files generated
        from data scaling software for all these data sets are also needed.

     *  If the scaling program is not one of those listed below
        (HKL|SCALEPACK|DTREK|SAINT|3DSCALE), enter OTHER for the program
        name and provide an ASCII file with five values
        [H, K, L, I (or F), sigmaI (or sigmaF)] for each reflection and
        seperate each item by a space

     *  If the same crystal was used for collecting multiple data sets, the
```

```
        crystal number will remain '1' as the wavelength numbers change.
        However, if multiple crystals were used, for the data collections,
        the corresponding crystal numbers should be used for each data set.

     *  IT IS IMPORTANT THAT THE LOG FILE AND DATA FILE COME FROM THE
        SAME PROGRAM.

<scale_data_type = "I" >          [enter I (intensity) or F (amplitude)]
<scale_program_name = "HKL" >

For data set 1:
<crystal_number  = "1" >
<diffract_number = "1" >
<scale_data_file_name  = " " >
<scale_log_file_name   = " " >

For data set 2:
<crystal_number  = "1" >
<diffract_number = "2" >
<scale_data_file_name  = " " >
<scale_log_file_name   = " " >

For data set 3:
<crystal_number  = "1" >
<diffract_number = "3" >
<scale_data_file_name  = " " >
<scale_log_file_name   = " " >

==================PART 3: Statistics for Indexing====================
Enter log file and software name for data indexing

     NOTE:
     *  This is only for the data of final structure refinment.

        Software for indexing is one of the following:
        (HKL|DENZO|DTREK|MOSFLM)

<data_indexing_software = "HKL" >
<data_indexing_LOG_file_name = " " >
<data_indexing_CIF_file_name = " " >  (if mmCIF format)

==================PART 4: Statistics for Data Scaling====================
Enter log file and software name for data scaling

     NOTE:
     *  The log file included here should have scaling statistics of
        the file used for the final structure refinement. If multiple data
        sets were scaled and merged for refinement (as described in Part 1
        above) use the log file generated during merging of the data sets.

        Software for scaling is one of the following:
        (HKL|SCALEPACK|DTREK|SAINT|3DSCALE|SCALA)

<data_scaling_software = "HKL" >
<data_scaling_LOG_file_name = " " >
<data_scaling_CIF_file_name = " " >  (if  mmCIF format)

==============PART 5: Statistics for Molecular Replacement===============
Enter log files and software name for molecular replacement
```

```
     NOTE:
        Software is one of the following:
        (CNS|AMORE|MOLREP|EPMR|PHASER)
        The log file should be from the best trial of MR.

<mr_software = " " >
<mr_log_file_LOG_1 = " " >
<mr_log_file_LOG_2 = " " >

=================PART 6: Statistics for Protein Phasing==================
Enter log files and software name for heavy atom phasing

     NOTE:
        The phasing method should be one of (SAD|MAD|SIR|SIRAS|MIR|MIRAS).
        Software is one of the following:
        (CNS|MLPHARE|SOLVE|SHELXS|SHELXD|SNB|BNP|SHARP|PHASES)
        The log file should be from the best trial of phasing.

<phasing_method = "MAD" >
<phasing_software = "SOLVE" >

<phasing_log_file_LOG_1 = " " >
<phasing_log_file_PDB_1 = " " >    (if PDB format (heavy atom coordinates))
<phasing_log_file_CIF_1 = " " >    (if mmCIF format)

<phasing_log_file_LOG_2 = " " >
<phasing_log_file_PDB_2 = " " >
<phasing_log_file_CIF_2 = " " >

... add more if needed ...

===============PART 7: Statistics for Density Modification===============
Enter log files and software name for density modification

     NOTE:
        Software is one of the following:
        (CNS|DM|RESOLVE|SOLOMON|SHELXE)
        The log file should be from the best trial of density modification.

<dm_software = "RESOLVE " >
<dm_log_file_LOG_1 = " " >
<dm_log_file_CIF_1 = " " >         (if mmCIF format)

===============PART 8: Statistics for Structure Refinement===============
Enter log files and software name used for final structure refinement

     NOTE:

        Software is one of the following:
        (CNS|REFMAC5|SHELXL|TNT|PROLSQ|NUCLSQ|RESTRAIN)
        The log file should be from the final trial of structure refinement.

<refine_software = "REFMAC5" >

<refine_log_file_PDB_1 = " " >  (coordinate file in PDB format)
<refine_log_file_CIF_1 = " " >  (mmCIF file containing refinement statistics)
<refine_log_file_LOG_1 = " " >
```

```
=======================PART 9: Data Template File=========================
Enter file name of the data template file

     NOTE:
        This file 'data_template.text' was generated by using the
        command 'extract -pdb pdb_file' or 'extract -cif cif_file'. It
        contains the sequences of all unique polymers (protein or nucleic
        acid) present in the structure. It also contains other
        non-electronically captured information. Please complete the
        data template file before running pdb_extract.

<data_template_file = "data_template.text" >


==========================PART 10: Output Files============================
Enter the output file names

     NOTE:
        If you do not give the output file names, the default names
        pdb_extract_sf.mmcif containing structure factors and
        pdb_extract.mmcif containing coordinates will be assigned
        by the program


<sf_output= " " >            (for structure factors)
<statistics_output= " " >    (for coordinates and statistics)

====================================END=====================================
```

Data template file for NMR: (data_template.text) **TOP**

```
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
                    THE DATA_TEMPLATE.TEXT FILE FOR NMR
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                           NOTES AND REMINDER
The data template file contains data entries for unique chemical sequences
present in the structure and other non-electronically captured information.

PLEASE CHECK CATEGORIES 1. Before proceeding any further, make necessary
corrections here so that all information in these categories are complete
and correct.

You may choose to fill in CATEGORIES (2-21) either here or later in ADIT.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

                         GUIDELINES FOR USING THIS FILE
  1. Only strings included between the 'lesser than' and 'greater than'
     signs (<.....>) will be parsed for evaluation by the program. Therefore,
```

```
     DO NOT write either on the left or right of the 'less than' and 'greater
     than' signs respectively.

  2. All alphanumeric values or strings that you include in the different
     categories should be within double-quotes. Blank spaces or carriage
     returns within a pair of double quotes are ignored by the program.
     DO NOT use double quotes (") within strings that you enter.


~~~~~~~~~~~~~~~~~~~~~~~~~~~START INPUT DATA BELLOW~~~~~~~~~~~~~~~~~~~~~~~

================CATEGORY 1:   Molecular Entity Sequence==================
Enter one letter code sequence for each molecular entity

A Molecular entity is defined as a unique monomer in each model.The
molecular entities are calculated and grouped together.
Please carefully check the entity and modify it, if necessary.

If a chain is broken, four question marks ???? are given at the broken
point. Please REPLACE the ? by the missing sequences including N and C
terminals. If residue name is not the standard one letter code (due to
modification), the full residue (three letter name) name should be given
and parenthesized.

NOTE: If all the residues are modified, sequence may not be extracted.
      Please manually add the sequence.

<molecule_entity_id="1" >
<molecule_entity_type="polypeptide(L)" >
<molecule_one_letter_sequence="
MENFQKVEKIGEGTYGVVYKARNKLTGEVVALKKIRLDT????TAIREISLLKELNHPNIVKLLDVIHTENKLY
LVFEFLHQDLKKFMDASALTGIPLPLIKSYLFQLLQGLAFCHSHRVLHRDLKPQNLLINTEGAIKLADFG
LARAFGVPVRTYTHEVVTLWYRAPEILLGCKYYSTAVDIWSLGCIFAEMVTRRALFPGDSEIDQLFRIFR
TLGTPDEVVWPGVTSMPDYKPSFPKWARQDFSKVVPPLDEDGRSLLSQMLHYDPNKRISAKAALAHPFFQ
DVTKPVP" >
< molecule_chain_id="A" >
< target_DB_id=" " > (if known)

<molecule_entity_id="2" >
<molecule_entity_type="polypeptide(L)" >
<molecule_one_letter_sequence="
QIYYSDKYDDEEFEYRHVMLPKDIAKLVPKTHLMSESEWRNLGVQQSQGWVHYMIHEPEPHILLFRRPLP
" >
< molecule_chain_id="B" >
< target_DB_id=" " > (if known)

<molecule_entity_id=" " >
<molecule_entity_type=" " >
<molecule_one_letter_sequence="  " >
<molecule_chain_id=" " >

<target_DB_id=" " >  (if known)


================CATEGORY 2:   Contact Authors============================
Enter information about the contact authors.
    Note: items marked by (e.g. ) are manditory.
          PI information should be always given.
```

```
1.  Information about the Principal investigator (PI) should be given.

<contact_author_PI_id = "1 ">           (must be given 1)
<contact_author_PI_salutation = " ">     ( Dr./Prof./Mr./Mrs./Ms.)
<contact_author_PI_first_name = " ">      (e.g. John)
<contact_author_PI_last_name = " ">        (e.g. Rodgers)
<contact_author_PI_middle_name = " ">
<contact_author_PI_role = " ">   (e.g. investigator/responsible scientist)
<contact_author_PI_organization_type = " ">  (e.g. academica/commercial/goverment/other)
<contact_author_PI_email = " ">        (e.g.   name@host.domain.country)
<contact_author_PI_address = " ">            (e.g. 610 Taylor road)
<contact_author_PI_city = " ">               (e.g. Piscataway)
<contact_author_PI_State_or_Province = " ">   (e.g.  New Jersey)
<contact_author_PI_Zip_Code = " ">           (e.g.  08864)
<contact_author_PI_Country = " ">          (e.g.  UNITED STATES)
<contact_author_PI_fax_number = " ">
<contact_author_PI_phone_numer = " ">

2. Information about other contact authors

<contact_author_id = "2 ">       (e.g. 2,3,4..)
<contact_author_salutation = " ">
<contact_author_first_name = " ">
<contact_author_last_name = " ">
<contact_author_middle_name = " ">
<contact_author_role = " ">
<contact_author_organization_type = " ">
<contact_author_email = " ">
<contact_author_address = " ">
<contact_author_city = " ">
<contact_author_State_or_Province = " ">
<contact_author_Zip_Code = " ">
<contact_author_Country = " ">
<contact_author_fax_number = " ">
<contact_author_phone_numer = " ">


...(add more if needed)...

================CATEGORY 3:   Structure Genomics=========================
If it is the structure genomics project, give the information

<SG_project_id = " 1">
<SG_project_name = " ">        (e.g. NPPSFA/PSI, Protein Structure Initiative)
<full_name_of_SG_center = " ">   (e.g. Berkeley Structural Genomics Center)


================CATEGORY 4:   Release Status==============================
Enter Release Status for Coordinates, Constraints, Sequence

   Status for sequence should be chosen from one of the following:
   (release now, hold for release)

   Status for others should be chosen from one of the following:
  (release now, hold for publication,  hold for 4 weeks, hold for 6 weeks,
   hold for 6 months, hold for 1 year)

<Release_status_for_coordinates = " ">
<Release_status_for_NMR_constraints = " ">
```

```
<Release_status_for_sequence = " ">

================CATEGORY 5:   Title=======================================
Enter a title for the structure

<structure_title = " ">     (e.g. Crystal Structure Analysis of the B-DNA)
<structure_details = " ">

================CATEGORY 6: Authors of Structure============================
Enter authors of the deposited structures (e.g. Surname, F.M.)

<structure_author_name = " ">
<structure_author_name = " ">
<structure_author_name = " ">
<structure_author_name = " ">
...add more if needed...


================CATEGORY 7:   Citation Authors=============================
Enter author names for the publications associated with this deposition.

      The primary citation is the article in which the deposited coordinates
      were first reported. Other related citations may also be provided.

1. For the primary citation
<primary_citation_author_name = " ">    (e.g. Surname, F.M.)
<primary_citation_author_name = " ">
<primary_citation_author_name = " ">
<primary_citation_author_name = " ">
...add more if needed...

2. For other related citations  (if applicable)
<citation_author_id = " ">    (e.g. 1, 2 ..)
<citation_author_name = " ">
<citation_author_name = " ">
<citation_author_name = " ">
<citation_author_name = " ">
...add more if needed...


...(add more other citations if needed)...
================CATEGORY 8:   Citation Article=============================
Enter citation article (journal, title, year, volume, page)

      If the citation has not yet been published, use 'To be published'
      for the category 'journal_abbrev' and leave pages and volume blank.

1. For primary citation
<primary_citation_id = "primary">
<primary_citation_journal_abbrev = " ">     (e.g. to be published)
<primary_citation_title = " ">
<primary_citation_year = " ">
<primary_citation_journal_volume = " ">
<primary_citation_page_first = " ">
<primary_citation_page_last = " ">

2. For other related citation (if applicable)
<citation_id = "1 ">               (e.g. 1, 2, 3 ...)
<citation_journal_abbrev = " ">
```

```
<citation_title = " ">
<citation_year = " ">
<citation_journal_volume = " ">
<citation_page_first = " ">
<citation_page_last = " ">


...(add more citations if needed)...
================CATEGORY 9:   Molecule Names==============================
Enter the name of the molecule for each entity

      The name of molecule should be obtained from the appropriate
      sequence database reference, if available. Otherwise the gene name or
      other common name of the entity may be used.
      e.g. HIV-1 integrase for protein
           RNA Hammerhead Ribozyme for RNA
      The number of entities should be the same as in CATEGORY 1.

<molecule_name = " ">    (entity 1)
<molecule_name = " ">    (entity 2)

...(add more if needed)...

================CATEGORY 10:  Molecule Details============================
Enter additional information about each entity, if known. (optional)

      Additional information would include details such as fragment name
      (if applicable), mutation, and E.C.number.

1. For entity 1
<Molecular_entity_id = "1 ">       (e.g. 1, 2, ...)
<Fragment_name = " ">              (e.g. ligand binding domain, hairpin)
<Specific_mutation = " ">          (e.g. C280S)
<Enzyme_Comission_number = " ">    (if known: e.g. 2.7.7.7)

2. For entity 2
<Molecular_entity_id = "2 ">
<Fragment_name = " ">
<Specific_mutation = " ">
<Enzyme_Comission_number = " ">

...(add more if needed)...

================CATEGORY 11:   Genetically Manipulated Source=============
Enter data in the genetically manipulated source category

      If the biomolecule has been genetically manipulated, describe its
      source and expression system here.

1. For entity 1
<Manipulated_entity_id = "1 ">               (e.g. 1, 2, ...)
<Source_organism_scientific_name = " ">      (e.g. Homo sapiens)
<Source_organism_gene = " ">                 (e.g. RPOD, ALKA...)
<Expression_system_scientific_name = " ">    (e.g. Escherichia coli)
<Expression_system_strain = " ">             (e.g. BL21(DE3))
<Expression_system_vector_type = " ">        (e.g. plasmid)
<Expression_system_plasmid_name = " ">       (e.g. pET26)
<Manipulated_source_details = " ">           (any other relevant information)
```

```
2. For entity 2
<Manipulated_entity_id = "2 ">
<Source_organism_scientific_name = " ">
<Source_organism_gene = " ">
<Expression_system_scientific_name = " ">
<Expression_system_strain = " ">
<Expression_system_vector_type = " ">
<Expression_system_plasmid_name = " ">
<Manipulated_source_details = " ">


...(add more if needed)...

================CATEGORY 12:   Natural Source=============================
Enter data in the natural source category  (if applicable)

    If the biomolecule was derived from a natural source, describe it here.


1. For entity 1
<natural_source_entity_id = " ">          (e.g. 1, 2, ...)
<natural_source_scientific_name = " ">    (e.g. Homo sapiens)
<natural_source_organism_strain = " ">    (e.g. DH5a , BMH 71-18)
<natural_source_details = " ">            (e.g. organ, tissue, cell ..)


2. For entity 2
<natural_source_entity_id = " ">
<natural_source_scientific_name = " ">
<natural_source_organism_strain = " ">
<natural_source_details = " ">


...(add more if needed)...

================CATEGORY 13:  Synthetic Source=============================
If the biomolecule has not been genetically manipulated or synthesized,
describe its source here.

1. For entity 1
<synthetic_source_entity_id = " ">          (e.g. 1, 2, ...)
<synthetic_source_description = " ">      (if known)

2. For entity 2
<synthetic_source_entity_id = " ">
<synthetic_source_description = " ">

...(add more if needed)...

================CATEGORY 14:   Keywords===================================
Enter a list of keywords that describe important features of the deposited
structure.

      For example, beta barrel, protein-DNA complex, double helix,
      hydrolase, structural genomics etc.

<structure_keywords = " ">

================CATEGORY 15:   Ensemble===================================
```

```
Enter data in category ensemble

  Skip this section, if only one average structure has been deposited.

<conformers_calculated_total_number = " ">   (e.g. 200)
<conformers_submitted_total_number = " ">    (e.g. 20)
<conformers_selection_criteria = " ">  (e.g. 20 structures for lowest energy)

================CATEGORY 16:   Representative Conformers==================
Enter data in category representative conformers

  Normally, only one of the ensemble is selected as a representative
  structure.

<conformer_id = " ">      (e.g. 1,2..)
<conformer_selection_criteria = " ">  (e.g.lowest energy, fewest violations)

================CATEGORY 17:   Sample Details=============================
Enter a description of each NMR sample, including the solvent system used.

1. for sample 1.
<solution_id_1= "1 ">       (e.g. 1, 2.. )
<solution_content_1= " ">  (e.g. 50mM phosphate buffer NA; 90% H2O, 10% D2O)
<solvent_system_1= " ">    (e.g. 90% H2O, 10% D2O )

2. for sample 2.
<solution_id_2= " ">
<solution_content_2= " ">
<solvent_system_2= " ">

....add more if needed....

================CATEGORY 18:   Sample Conditions==========================
Enter experimental conditions used for each sample.

  Each set of conditions is identified by a numerical code.

1. for sample 1.
<Conditions_id_1 = "1 ">    (e.g. 1, 2..)
<Temperature_1 = " ">      (e.g. 298)  (in Kelvin)
<Pressure_1 = " ">         (e.g. ambient, 1atm)
<pH_value_1 = " ">         (e.g. 7.2)
<Ionic_strength_1 = " ">   (e.g.  100MM KCL)

2. for sample 2.
<Conditions_id_2 = " ">
<Temperature_2 = " ">
<Pressure_2 = " ">
<pH_value_2 = " ">
<Ionic_strength_2 = " ">

....add more if needed....

================CATEGORY 19:   Spectrometer================================
Enter the details about each spectrometer used to collect data.

1. for experiment 1:
<spectrometer_id_1 = "1 ">              (e.g. 1, 2..)
<spectrometer_manufacturer_1 = " ">    (e.g. Bruker ..)
```

```
<spectrometer_model_1 = " ">            (e.g. DRX)
<spectrometer_field_strength_1 = " ">  (e.g. 500, 700)

2. for experiment 2:
<spectrometer_id_2 = " ">
<spectrometer_manufacturer_2 = " ">
<spectrometer_model_2 = " ">
<spectrometer_field_strength_2 = " ">

....add more if needed....

================CATEGORY 20:   Experiment Type============================
Enter information for those experiments that were used to generate
constraint data. For each NMR experiment, indicate which sample and
which sample conditions were used for the experiment.

1. for experiment type 1:
<experiment_type_id_1 = "1 ">    (e.g. 1, 2..)
<solution_type_id_1= " 1">       (same ID as solution_id_1 in CATEGORY 17)
<conditions_type_id_1 = "1 ">    (same ID as conditions_id_1 in CATEGORY 18)
<Experiment_type_1= " ">        (e.g. 3D_15N-separated_NOESY)

2. for experiment type 2:
<experiment_type_id_2 = " ">    (e.g. 1, 2..)
<solution_type_id_2= " ">       (same ID as solution_id_1 in CATEGORY 17)
<conditions_type_id_2 = " ">    (same ID as conditions_id_1 in CATEGORY 18)
<Experiment_type_2= " ">

....add more if needed....

================CATEGORY 21:   Method and Details=========================
Enter the method and details of the refinement for the deposited structure.

<NMR_method = " ">   (e.g. simulated annealing)
<NMR_details = " ">  (enter details about the NMR refinement)


====================================END==================================
```