# MINT V3.0

Dr. Andrew C.R. Martin, UCL



MINT is a graphical interface to the basic modelling functions of Andrej Šali's Modeller program. This interface will simply allow you to perform homology modelling; none of the other capabilities of Modeller is supported.

MINT may be used to perform 'full homology' modelling where you simply give Modeller the PDB files to use as templates and Modeller calculates a sequence alignment. Alternatively, you may provide an alignment and request Modeller to work from this. The latter protocol is recommended since an accurate alignment is crucial to obtaining a good model.

If MINT and/or Modeller have not been globally installed at your site, please read Appendix A.

## 1 Full Homology Modelling

In full homology modelling, you need only supply the sequence of the protein you wish to model and the PDB code of the template structures you wish to use.

Proceed as follows:

1. Create a file containing the sequence of the protein to be modelled in standard PIR format. This format is described in Appendix B
2. Start MINT by typing `mint` at the Unix prompt.
3. In the text box labelled **PIR Sequence File:**, enter the filename of the PIR sequence file you have created which contains the sequence of the protein to be modelled. If this box is grey and inactive (you can't type in it even after clicking in it with the mouse), click the **Specify Alignment** button and then click the **No Alignment** button in the window which appears.
4. In the text box labelled **Template PDB codes:**, enter the PDB codes of the structures you are using as templates for the homology model building. You should enter just the 4 character PDB code (e.g. 1crn), no preceeding characters (such as `pdb` or `p`) are required. Case is important; you should use the same case as used in your PDB directory where the PDB files are stored since these codes are used directly to access PDB files. This will generally be lower case.

Optionally, you may specify a 5th character in the PDB code. This is a chain label allowing you to model a single chain of a multi-chain protein.

If you are using more than one template structure, the PDB codes should be separated by spaces.

MINT will **first look in the current directory** for PDB files. **Note that any PDB files in your current directory must have the same prefix and extension as PDB files in the main PDB directory.** (See Appendix D.)

5. Select the number of models required and the refinement method you wish to use.

6. Click the **Run** button to start the Modeller program and to exit MINT. If you wish to exit without running Modeller, click the **Quit** button.

# 2 Alignment-Specified Modelling

In alignment-specified modelling, you must provide a sequence alignment containing the sequence of the protein to be modelled with the template structures you are going to use. This is the recommended protocol since a good alignment is critical in obtaining an accurate model.

Proceed as follows:

1. Create a file containing the sequence of the protein to be modelled aligned with the sequences of the template structures in Modeller's PIR-like alignment format. This format is described in Appendix C.

2. Start MINT by typing `mint` at the Unix prompt.

3. Click the **Specify Alignment** button. A new window will appear.

4. In the text box labelled **Alignment file:**, enter the name of the alignment file you have created. Ensure that the PIR radio button is active rather than the Quanta button.

5. Enter the identifier of the sequence to be modelled in the text box labelled **Target code:**.

6. Click on the button labelled **Exit**. The new window will disappear.

7. In the text box labelled **Template PDB codes:**, enter the identifiers used in the alignent file (Appendix C) for the structures you are using as templates for the homology model building.

If you are using more than one template structure, the codes should be separated by spaces.

Modeller will look for the PDB files in the current directory and the main PDB directory (see Appendix D) using filenames which are specified in the alignment file (see Appendix C). The codes entered here are not used directly to find the PDB files.

8. Select the number of models required and the refinement method you wish to use.

9. Click the **Run** button to start the Modeller program and to exit MINT. If you wish to exit without running Modeller, click the **Quit** button.

## 3 Advanced Options

Clicking the **Advanced options** button brings up a window in which three options may be set. These control the reading and writing of hetero (non-protein) atoms, waters and hydrogens. By default, all three of these are ignored.

In the advanced options window, clicking one of the three checkboxes will toggle the appropriate option on or off.

Click the **Exit** button at the bottom of the window to close it.

## 4 Other Information

Alignment files may also be specified in Quanta format. These may be generated from within the Quanta software by following the Quanta protocols for Homology Modelling (note that you do not need the MSI version of Modeller to do this). If you generate a Quanta alignment file, you simply specify this in the **Alignment File:** text box and click the **Quanta** radio button.

If you use a Quanta format alignment file, the codes used in the file and in the **Target code:** and **Template PDB codes:** text boxes must be valid PDB file stems (e.g. p1crn if your PDB files are stored as `/pdb/p1crn.pdb`).

# Appendices

## A Installation

Full installation instructions for Modeller are provided with the program and installation instructions for MINT are provided in the INSTALL file in the MINT directory.

If the software has been installed by another user, but not made globally available, you need to take the following steps:
1. Create the environment variables MODINSTALL and MINTDIR to point to where Modeller and MINT have been installed,
2. Create the KEY environment variable used by modeller, 3. Source the Modeller setup script,
4. Create an alias to run the MINT program.

Assuming that Modeller has been installed in a directory which is called `/home/bsm/martin/sg/bin/modeller3` and MINT in `/home/bsm/martin/modeller`, the following script will set everything as required:

```
# Customise these 2 for your site...
# ----------------------------------
# Root directory for installed MODELLER:
setenv MODINSTALL /home/bsm/martin/sg/bin/modeller3
# Directory in which MINT is installed
setenv MINTDIR      '/home/bsm/martin/modeller'

if (-e $MODINSTALL) then
   # Root directory for the Protein DataBank (not essential, can be omitted):
   setenv PDB /pdb

   # MODELLER key. Change to the key provided by Andrej Sali:
   setenv KEY <modeller-key-here>

   # Set MODELLER environment variables and update the command path:
   if (-e $MODINSTALL/bin/setmodeller) source $MODINSTALL/bin/setmodeller

   # Read modeller HTML docs using Mosaic
   alias moddoc 'mosaic $MODINSTALL/doc/manual/manual.html'

   # Alias for mint
   alias  mint         $MINTDIR/mint.tcl
endif
```

**If you are at BSM Unit, UCL** simply add the following to your .cshrc file:

```
source /home/bsm/martin/modeller/setmodeller
```

# B PIR File Format

The PIR sequence file format consists of two header lines followed by the amino acid sequence using 1-letter code ending with an asterisk (*).

The first header line is of the form:

```
>P1;xxxxxx
```

where `xxxxxx` is an identifier for the sequence. Any sequence of up to 6 characters may be supplied.

The second header line is a title describing the sequence. Optionally this may consist of two fields separated by a dash (-). If so, the second field describes the source of the sequence (e.g. 'human').

The sequence follows using the standard 1-letter code. Spaces and line breaks are ignored and an asterisk (*) marks the end of the sequence. Note that chain breaks are indicated with a slash (/) not an asterisk as used in the standard PIR format.

# C Alignment File Format

The alignment file format is an extension of the PIR format described in Appendix B.

The alignment is simply created by using dash (-) characters to indicate deletions in the sequences.

The first header line is a standard PIR header line of the form:

```
>P1;xxxxxx
```

Note that the `xxxxxx` code (up to 6 characters) is used as an identifier by the **Target code:** and **Template PDB codes:** text boxes in MINT.

The second (comment) header line is modified and contains 10 fields separated by colons (:). These fields have the following meanings:

1. The type of structure associated with the sequence. This is specified as follows:

| | |
|---|---|
| sequence | No structure available, |
| structureX | An X-ray crystal structure, |
| structureN | An NMR structure, |
| structureM | A model structure. |

2. The filestem of the PDB file containing the associated structure. Any characters prepended onto the PDB code **are included**, but the extension and directory are not. Thus, if we are using crambin as a structure (PDB code 1crn) and we store the PDB files as `/pdb/pXXXX.pdb`, this would be specified as `p1crn`. This field is blank if there is no associated structure (i.e. this is the sequence to be modelled).

3. The residue number (in the PDB file) of the first residue of the sequence. Normally this will be the first residue number in the PDB file. This field is blank if there is no associated structure (i.e. this is the sequence to be modelled).

4. The chain name (in the PDB file) of the first residue in the sequence (or blank).

5. The residue number (in the PDB file) of the last residue of the sequence. Normally this will be the last residue number in the PDB file. This field is blank if there is no associated structure (i.e. this is the sequence to be modelled).

6. The chain name (in the PDB file) of the last residue in the sequence (or blank).

7. The name of the sequence. This is a text description of the protein and is normally the first COMPND record from a PDB file. This field may be left blank.

8. The source of the protein (e.g. 'HUMAN', 'MOUSE'). this field may be left blank.

9. The resolution of the crystal structure. This field is set to 0.00 if there is no associated structure (i.e. this is the sequence to be modelled) or this is an NMR structure.

10. The R-factor of the crystal structure. This field is set to 0.00 if there is no associated structure (i.e. this is the sequence to be modelled) or this is an NMR structure.

# D System Options

System options for MINT should be set at installation time. However, these may be overridden once MINT has been started by clicking on the **System** button.

These options have the following meanings:

**PDB Directory**

This is the location of the main store of PDB files. MINT and Modeller will always look in the current directory for PDB files first and will then look in the directory specified here. Note that a trailing slash (/) must be specified.

Note that the PDB filename will be created from the 4-letter PDB code you specify plus the prefix and extension (as described below). *Thus PDB files in your current directory **must** use the same naming convention as those in the main PDB directory* when doing full homology modelling.

**PDB Extension**
This is the filename extension used for your PDB files. Note that this must include a dot (.) if your PDB files have a dot before the extension.

**PDB Prefix**
This is any character(s) prepended onto the PDB code to create the filename. If your site has no prepended characters, place a - in this field.

## D.1 Examples

The follwing examples show how the PDB file with PDB code 2hfl are stored and how these values should be set:

| | |
|---|---|
| Name format | /pdb/p2hfl.pdb |
| PDB directory | /pdb/ |
| PDB extension | .pdb |
| PDB prefix | p |

| | |
|---|---|
| Name format | /data/pdb/pdb2hfl.ent |
| PDB directory | /data/pdb/ |
| PDB extension | .ent |
| PDB prefix | pdb |

## D.2 The current directory

When looking in the current directory for PDB files, the filename is built from the prefix, PDB code and extension. Therefore, **PDB files in your current directory must have the same prefix and extension as files in the main PDB directory**.

# E Frequently Asked Questions

**1. Q:** When doing full homology modelling using a PDB file in the current directory, why does MODELLER crash with a message of the form:

```
TOP______>    84  262 READ_ALIGNMENT FILE = SEGFILE, ALIGN_CODES = KNOWNS
rdseqpd_E> too many residues
recover__>  MODELLER_STATUS  >=  STOP_STATUS:   1  1
```

**A:** MODELLER is actually telling you that it cannot find the PDB file you specified for the template! The most probably reason for this is that the filenaming convention you have used for the file in the current directory doesn't match that in the PDB.