# XBoard

Published by the Free Software Foundation
59 Temple Place - Suite 330
Boston, MA 02111-1307 USA

# Introduction

XBoard is a graphical chessboard that can serve as a user interface to chess engines (such as GNU Chess), the Internet Chess Servers, electronic mail correspondence chess, or your own collection of saved games.

This manual documents version 4.2.7 of XBoard.

# 1 Major modes

XBoard always runs in one of four major modes. You select the major mode from the command line when you start up XBoard.

xboard [options]
: As an interface to GNU Chess or another chess engine running on your machine, XBoard lets you play a game against the machine, set up arbitrary positions, force variations, watch a game between two chess engines, interactively analyze your stored games or set up and analyze arbitrary positions. (Note: Not all chess engines support analysis.)

xboard -ics -icshost hostname [options]
: As Internet Chess Server (ICS) interface, XBoard lets you play against other ICS users, observe games they are playing, or review games that have recently finished. Most of the ICS "wild" chess variants are supported, including bughouse.

xboard -ncp [options]
: XBoard can also be used simply as an electronic chessboard to play through games. It will read and write game files and allow you to play through variations manually. You can use it to browse games off the net or review games you have saved. These features are also available in the other modes.

|pxboard
: If you want to pipe games into XBoard, use the supplied shell script '`pxboard`'. For example, from the news reader '`xrn`', find a message with one or more games in it, click the Save button, and type '`|pxboard`' as the file name.

cmail [options]
: As an interface to electronic mail correspondence chess, XBoard works with the cmail program. See Chapter 10 [CMail], page 37 below for instructions.

# 2 Menus, buttons, and keys

To move a piece, you can drag it with the left mouse button, or you can click the left mouse button once on the piece, then once more on the destination square. To drop a new piece on a square (when applicable), press the middle or the right mouse button over the square and select from the popup menu. In cases where you can drop either a white or black piece, use the middle button (or shift+right) for white and the right button (or shift+middle) for black. When you are playing a bughouse game on an Internet Chess Server, a list of the offboard pieces that each player has available is shown in the window title after the player's name; in addition, the piece menus show the number of pieces available of each type.

All other XBoard commands are available from the menu bar. The most frequently used commands also have shortcut keys or on-screen buttons.

When XBoard is iconized, its graphical icon is a white knight if it is White's turn to move, a black knight if it is Black's turn. See Iconize in Section 2.7 [Keys], page 14 below if you have problems getting this feature to work.

## 2.1 File Menu

Reset
: Resets XBoard and the chess engine to the beginning of a new chess game. The *r* key is a keyboard equivalent. In Internet Chess Server mode, clears the current state of XBoard, then resynchronizes with the ICS by sending a refresh command. If you want to stop playing, observing, or examining an ICS game, use an appropriate command from the Action menu, not '`Reset`'. See Section 2.3 [Action Menu], page 8.

Load Game
: Plays a game from a record file. The *g* key is a keyboard equivalent. A popup dialog prompts you for the file name. If the file contains more than one game, a second popup dialog displays a list of games (with information drawn from their PGN tags, if any), and you can select the one you want. Alternatively, you can load the Nth game in the file directly, by typing the number *N* after the file name, separated by a space.

  The game file parser will accept PGN (portable game notation), or in fact almost any file that contains moves in algebraic notation. Notation of the form '`P@f7`' is accepted for piece-drops in bughouse games; this is a nonstandard extension to PGN. If the file includes a PGN position (FEN tag), or an old-style XBoard position diagram bracketed by '`[--`' and '`--]`' before the first move, the game starts from that position. Text enclosed in parentheses, square brackets, or curly braces is assumed to be commentary and is displayed in a pop-up window. Any other text in the file is ignored. PGN variations (enclosed in parentheses) are treated as comments; XBoard is not able to walk variation trees. The nonstandard PGN tag [Variant "varname"] functions similarly to the -variant command-line option (see below), allowing games in certain chess variants to be loaded. There is also a heuristic to recognize chess variants from the Event tag, by looking for the strings that the Internet Chess Servers put there when saving variant ("wild") games.

Load Next Game
: Loads the next game from the last game record file you loaded. The shifted *N* key is a keyboard equivalent.

Load Previous Game
: Loads the previous game from the last game record file you loaded. The shifted *P* key is a keyboard equivalent. Not available if the last game was loaded from a pipe.

Reload Same Game
: Reloads the last game you loaded. Not available if the last game was loaded from a pipe.

Save Game
: Appends a record of the current game to a file. A popup dialog prompts you for the file name. If the game did not begin with the standard starting position, the game file includes the starting position used. Games are saved in the PGN (portable game notation) format, unless the oldSaveStyle option is true, in which case they are saved in an older format that is specific to XBoard. Both formats are human-readable, and both can be read back by the '`Load Game`' command. Notation of the form '`P@f7`' is accepted for piece-drops in bughouse games; this is a nonstandard extension to PGN.

Copy Game
: Copies a record of the current game to an internal clipboard in PGN format and sets the X selection to the game text. The game can be pasted to another application (such as a text editor or another copy of XBoard) using that application's paste command. In many X applications, such as xterm and emacs, the middle mouse button can be used for pasting; in XBoard, you must use the Paste Game command.

Paste Game
: Interprets the current X selection as a game record and loads it, as with Load Game.

Load Position
: Sets up a position from a position file. A popup dialog prompts you for the file name. If the file contains more than one saved position, and you want to load the Nth one, type the number N after the file name, separated by a space. Position files must be in FEN (Forsythe-Edwards notation), or in the format that the Save Position command writes when oldSaveStyle is turned on.

Load Next Position
: Loads the next position from the last position file you loaded.

Load Previous Position
: Loads the previous position from the last position file you loaded. Not available if the last position was loaded from a pipe.

Reload Same Position
: Reloads the last position you loaded. Not available if the last position was loaded from a pipe.

Save Position
: Appends a diagram of the current position to a file. A popup dialog prompts you for the file name. Positions are saved in FEN (Forsythe-Edwards notation) format unless the `oldSaveStyle` option is true, in which case they are saved in an older, human-readable format that is specific to XBoard. Both formats can be read back by the '`Load Position`' command.

Copy Position
: Copies the current position to an internal clipboard in FEN format and sets the X selection to the position text. The position can be pasted to another application (such as a text editor or another copy of XBoard) using that application's paste command. In many X applications, such as xterm and emacs, the middle mouse button can be used for pasting; in XBoard, you must use the Paste Position command.

Paste Position
: Interprets the current X selection as a FEN position and loads it, as with Load Position.

Mail Move
Reload CMail Message
: See Chapter 10 [CMail], page 37.

Exit
: Exits from XBoard. The shifted *Q* key is a keyboard equivalent.

## 2.2 Mode Menu

Machine White
: Tells the chess engine to play White.

Machine Black
: Tells the chess engine to play Black.

Two Machines
: Plays a game between two chess engines.

Analysis Mode
: XBoard tells the chess engine to start analyzing the current game/position and shows you the analysis as you move pieces around. Note: Some chess engines do not support Analysis mode.

To set up a position to analyze, you do the following:

1. Select Edit Position from the Mode Menu

2. Set up the position. Use the middle and right buttons to bring up the white and black piece menus.

3. When you are finished, click on either the Black or White clock to tell XBoard which side moves first.

4. Select Analysis Mode from the Mode Menu to start the analysis.

Analyze File
: This option lets you load a game from a file (PGN, XBoard format, etc.) and analyze it. When you select this menu item, a popup window appears and

asks for a filename to load. If the file contains multiple games, another popup appears that lets you select which game you wish to analyze. After a game is loaded, use the XBoard arrow buttons to step forwards/backwards through the game and watch the analysis. Note: Some chess engines do not support Analysis mode.

ICS Client This is the normal mode when XBoard is connected to a chess server. If you have moved into Edit Game or Edit Position mode, you can select this option to get out.

To use xboard in ICS mode, run it in the foreground with the -ics option, and use the terminal you started it from to type commands and receive text responses from the chess server. See Chapter 4 [Chess Servers], page 29 below for more information.

XBoard activates some special position/game editing features when you use the *examine* or *bsetup* commands on ICS and you have '`ICS Client`' selected on the Mode menu. First, you can issue the ICS position-editing commands with the mouse. Move pieces by dragging with mouse button 1. To drop a new piece on a square, press mouse button 2 or 3 over the square. This brings up a menu of white pieces (button 2) or black pieces (button 3). Additional menu choices let you empty the square or clear the board. Click on the White or Black clock to set the side to play. You cannot set the side to play or drag pieces to arbitrary squares while examining on ICC, but you can do so in *bsetup* mode on FICS. In addition, the menu commands '`Forward`', '`Backward`', '`Pause`', and '`Stop Examining`' have special functions in this mode; see below.

Edit Game

Allows you to make moves for both Black and White, and to change moves after backing up with the '`Backward`' command. The clocks do not run.

In chess engine mode, the chess engine continues to check moves for legality but does not participate in the game. You can bring the chess engine into the game by selecting '`Machine White`', '`Machine Black`', or '`Two Machines`'.

In ICS mode, the moves are not sent to the ICS: '`Edit Game`' takes XBoard out of ICS Client mode and lets you edit games locally. If you want to edit games on ICS in a way that other ICS users can see, use the ICS *examine* command or start an ICS match against yourself.

Edit Position

Lets you set up an arbitrary board position. Use mouse button 1 to drag pieces to new squares, or to delete a piece by dragging it off the board or dragging an empty square on top of it. To drop a new piece on a square, press mouse button 2 or 3 over the square. This brings up a menu of white pieces (button 2) or black pieces (button 3). Additional menu choices let you empty the square or clear the board. You can set the side to play next by clicking on the word White or Black at the top of the screen. Selecting '`Edit Position`' causes XBoard to discard all remembered moves in the current game.

In ICS mode, changes made to the position by '`Edit Position`' are not sent to the ICS: '`Edit Position`' takes XBoard out of '`ICS Client`' mode and lets you edit positions locally. If you want to edit positions on ICS in a way that other ICS users can see, use the ICS *examine* command, or start an ICS match against yourself. (See also the ICS Client topic above.)

Training
: Training mode lets you interactively guess the moves of a game for one of the players. You guess the next move of the game by playing the move on the board. If the move played matches the next move of the game, the move is accepted and the opponent's response is autoplayed. If the move played is incorrect, an error message is displayed. You can select this mode only while loading a game (that is, after selecting '`Load Game`' from the File menu). While XBoard is in '`Training`' mode, the navigation buttons are disabled.

Show Game List
: Shows or hides the list of games generated by the last '`Load Game`' command.

Edit Tags
: Lets you edit the PGN (portable game notation) tags for the current game. After editing, the tags must still conform to the PGN tag syntax:

```
<tag-section> ::= <tag-pair> <tag-section>
                        <empty>
<tag-pair> ::= [ <tag-name> <tag-value> ]
<tag-name> ::= <identifier>
<tag-value> ::= <string>
```

See the PGN Standard for full details. Here is an example:

```
[Event "Portoroz Interzonal"]
[Site "Portoroz, Yugoslavia"]
[Date "1958.08.16"]
[Round "8"]
[White "Robert J. Fischer"]
[Black "Bent Larsen"]
[Result "1-0"]
```

Any characters that do not match this syntax are silently ignored. Note that the PGN standard requires all games to have at least the seven tags shown above. Any that you omit will be filled in by XBoard with '`?`' (unknown value), or '`-`' (inapplicable value).

Edit Comment
: Adds or modifies a comment on the current position. Comments are saved by '`Save Game`' and are displayed by '`Load Game`', '`Forward`', and '`Backward`'.

ICS Input Box
: If this option is set in ICS mode, XBoard creates an extra window that you can use for typing in ICS commands. The input box is especially useful if you want to type in something long or do some editing on your input, because output from ICS doesn't get mixed in with your typing as it would in the main terminal window.

Pause
: Pauses updates to the board, and if you are playing against a chess engine, also pauses your clock. To continue, select '`Pause`' again, and the display will automatically update to the latest position. The '`P`' button and keyboard *p* key are equivalents.

If you select Pause when you are playing against a chess engine and it is not your move, the chess engine's clock will continue to run and it will eventually make a move, at which point both clocks will stop. Since board updates are paused, however, you will not see the move until you exit from Pause mode (or select Forward). This behavior is meant to simulate adjournment with a sealed move.

If you select Pause while you are observing or examining a game on a chess server, you can step backward and forward in the current history of the examined game without affecting the other observers and examiners, and without having your display jump forward to the latest position each time a move is made. Select Pause again to reconnect yourself to the current state of the game on ICS.

If you select '`Pause`' while you are loading a game, the game stops loading. You can load more moves manually by selecting '`Forward`', or resume automatic loading by selecting '`Pause`' again.

## 2.3 Action Menu

Accept
: Accepts a pending match offer. If there is more than one offer pending, you will have to type in a more specific command instead of using this menu choice.

Decline
: Declines a pending offer (match, draw, adjourn, etc.). If there is more than one offer pending, you will have to type in a more specific command instead of using this menu choice.

Call Flag
: Calls your opponent's flag, claiming a win on time, or claiming a draw if you are both out of time. You can also call your opponent's flag by clicking on his clock or by pressing the keyboard *t* key.

Draw
: Offers a draw to your opponent, accepts a pending draw offer from your opponent, or claims a draw by repetition or the 50-move rule, as appropriate. The *d* key is a keyboard equivalent.

Adjourn
: Asks your opponent to agree to adjourning the current game, or agrees to a pending adjournment offer from your opponent.

Abort
: Asks your opponent to agree to aborting the current game, or agrees to a pending abort offer from your opponent. An aborted game ends immediately without affecting either player's rating.

Resign
: Resigns the game to your opponent. The shifted *R* key is a keyboard equivalent.

Stop Observing
: Ends your participation in observing a game, by issuing the ICS observe command with no arguments. ICS mode only.

Stop Examining
: Ends your participation in examining a game, by issuing the ICS unexamine command. ICS mode only.

## 2.4 Step Menu

Backward

Steps backward through a series of remembered moves. The '`[<]`' button and the *b* key are equivalents. In addition, pressing the Control key steps back one move, and releasing it steps forward again.

In most modes, '`Backward`' only lets you look back at old positions; it does not retract moves. This is the case if you are playing against a chess engine, playing or observing a game on an ICS, or loading a game. If you select '`Backward`' in any of these situations, you will not be allowed to make a different move. Use '`Retract Move`' or '`Edit Game`' if you want to change past moves.

If you are examining an ICS game, the behavior of '`Backward`' depends on whether XBoard is in Pause mode. If Pause mode is off, '`Backward`' issues the ICS backward command, which backs up everyone's view of the game and allows you to make a different move. If Pause mode is on, '`Backward`' only backs up your local view.

Forward

Steps forward through a series of remembered moves (undoing the effect of '`Backward`') or forward through a game file. The '`[>]`' button and the *f* key are equivalents.

If you are examining an ICS game, the behavior of Forward depends on whether XBoard is in Pause mode. If Pause mode is off, '`Forward`' issues the ICS forward command, which moves everyone's view of the game forward along the current line. If Pause mode is on, '`Forward`' only moves your local view forward, and it will not go past the position that the game was in when you paused.

Back to Start

Jumps backward to the first remembered position in the game. The '`[<<]`' button and the shifted *B* key are equivalents.

In most modes, Back to Start only lets you look back at old positions; it does not retract moves. This is the case if you are playing against a local chess engine, playing or observing a game on a chess server, or loading a game. If you select '`Back to Start`' in any of these situations, you will not be allowed to make different moves. Use '`Retract Move`' or '`Edit Game`' if you want to change past moves; or use Reset to start a new game.

If you are examining an ICS game, the behavior of '`Back to Start`' depends on whether XBoard is in Pause mode. If Pause mode is off, '`Back to Start`' issues the ICS '`backward 999999`' command, which backs up everyone's view of the game to the start and allows you to make different moves. If Pause mode is on, '`Back to Start`' only backs up your local view.

Forward to End

Jumps forward to the last remembered position in the game. The '`[>>]`' button and the shifted *F* key are equivalents.

If you are examining an ICS game, the behavior of '`Forward to End`' depends on whether XBoard is in Pause mode. If Pause mode is off, '`Forward to End`' issues the ICS '`forward 999999`' command, which moves everyone's view of the game forward to the end of the current line. If Pause mode is on, '`Forward to`

`End`' only moves your local view forward, and it will not go past the position that the game was in when you paused.

Revert
: If you are examining an ICS game and Pause mode is off, issues the ICS command '`revert`'.

Truncate Game
: Discards all remembered moves of the game beyond the current position. Puts XBoard into '`Edit Game`' mode if it was not there already.

Move Now
: Forces the chess engine to move immediately. Chess engine mode only.

Retract Move
: Retracts your last move. In chess engine mode, you can do this only after the chess engine has replied to your move; if the chess engine is still thinking, use '`Move Now`' first. In ICS mode, '`Retract Move`' issues the command '`takeback 1`' or '`takeback 2`' depending on whether it is your opponent's move or yours.

## 2.5 Options Menu

Always Queen
: If this option is off, XBoard brings up a dialog box whenever you move a pawn to the last rank, asking what piece you want to promote it to. If the option is true, your pawns are always promoted to queens. Your opponent can still underpromote.

Animate Dragging
: If Animate Dragging is on, while you are dragging a piece with the mouse, an image of the piece follows the mouse cursor. If Animate Dragging is off, there is no visual feedback while you are dragging a piece, but if Animate Moving is on, the move will be animated when it is complete.

Animate Moving
: If Animate Moving is on, all piece moves are animated. An image of the piece is shown moving from the old square to the new square when the move is completed (unless the move was already animated by Animate Dragging). If Animate Moving is off, a moved piece instantly disappears from its old square and reappears on its new square when the move is complete.

Auto Comment
: If this option is on, any remarks made on ICS while you are observing or playing a game are recorded as a comment on the current move. This includes remarks made with the ICS commands *`say`*, *`tell`*, *`whisper`*, and *`kibitz`*. Limitation: remarks that you type yourself are not recognized; XBoard scans only the output from ICS, not the input you type to it.

Auto Flag
: If this option is on and one player runs out of time before the other, XBoard will automatically call his flag, claiming a win on time. In ICS mode, Auto Flag will only call your opponent's flag, not yours, and the ICS may award you a draw instead of a win if you have insufficient mating material. In local chess engine mode, XBoard may call either player's flag and will not take material into account.

Auto Flip View
: If the Auto Flip View option is on when you start a game, the board will be automatically oriented so that your pawns move from the bottom of the window towards the top.

Auto Observe
: If this option is on and you add a player to your `gnotify` list on ICS, XBoard will automatically observe all of that player's games, unless you are doing something else (such as observing or playing a game of your own) when one starts. The games are displayed from the point of view of the player on your gnotify list; that is, his pawns move from the bottom of the window towards the top. Exceptions: If both players in a game are on your gnotify list, if your ICS `highlight` variable is set to 0, or if the ICS you are using does not properly support observing from Black's point of view, you will see the game from White's point of view.

Auto Raise Board
: If this option is on, whenever a new game begins, the chessboard window is deiconized (if necessary) and raised to the top of the stack of windows.

Auto Save
: If this option is true, at the end of every game XBoard prompts you for a file name and appends a record of the game to the file you specify. Disabled if the `saveGameFile` command-line option is set, as in that case all games are saved to the specified file. See Section 3.3 [Load and Save options], page 23.

Blindfold
: If this option is on, XBoard displays the board as usual but does not display pieces or move highlights. You can still move in the usual way (with the mouse or by typing moves in ICS mode), even though the pieces are invisible.

Flash Moves
: If this option is on, whenever a move is completed, the moved piece flashes. The number of times to flash is set by the flashCount command-line option; it defaults to 3 if Flash Moves is first turned on from the menu.

Flip View
: Inverts your view of the chess board for the duration of the current game. Starting a new game returns the board to normal. The `v` key is a keyboard equivalent.

: If you are playing a game on an ICS, the board is always oriented at the start of the game so that your pawns move from the bottom of the window towards the top. Otherwise, the starting orientation is determined by the `flipView` command line option; if it is false (the default), White's pawns move from bottom to top at the start of each game; if it is true, Black's pawns move from bottom to top. See Section 3.4 [User interface options], page 24.

Get Move List
: If this option is on, whenever XBoard receives the first board of a new ICS game (or a different game from the one it is currently displaying), it retrieves the list of past moves from the ICS. You can then review the moves with the '`Forward`' and '`Backward`' commands or save them with '`Save Game`'. You might want to turn off this option if you are observing several blitz games at once, to keep from wasting time and network bandwidth fetching the move lists over

and over. When you turn this option on from the menu, XBoard immediately fetches the move list of the current game (if any).

Highlight Last Move
: If Highlight Last Move is on, after a move is made, the starting and ending squares remain highlighted. In addition, after you use Backward or Back to Start, the starting and ending squares of the last move to be unmade are highlighted.

Move Sound
: If this option is on, XBoard alerts you by playing a sound after each of your opponent's moves (or after every move if you are observing a game on the Internet Chess Server). The sound is not played after moves you make or moves read from a saved game file. By default, the sound is the terminal bell, but on some systems you can change it to a sound file using the soundMove option; see below.

  If you turn on this option when using XBoard with the Internet Chess Server, you will probably want to give the *`set bell 0`* command to the ICS, since otherwise the ICS will ring the terminal bell after every move (not just yours). (The '`.icsrc`' file is a good place for this; see Section 3.2 [ICS options], page 18.)

ICS Alarm
: When this option is on, an alarm sound is played when your clock counts down to the icsAlarmTime (by default, 5 seconds) in an ICS game. For games with time controls that include an increment, the alarm will sound each time the clock counts down to the icsAlarmTime. By default, the alarm sound is the terminal bell, but on some systems you can change it to a sound file using the soundIcsAlarm option; see below.

Old Save Style
: If this option is off, XBoard saves games in PGN (portable game notation) and positions in FEN (Forsythe-Edwards notation). If the option is on, a save style that is compatible with older versions of XBoard is used instead. The old position style is more human-readable than FEN; the old game style has no particular advantages.

Periodic Updates
: If this option is off (or if you are using a chess engine that does not support periodic updates), the analysis window will only be updated when the analysis changes. If this option is on, the Analysis Window will be updated every two seconds.

Ponder Next Move
: If this option is off, the chess engine will think only when it is on move. If the option is on, the engine will also think while waiting for you to make your move.

Popup Exit Message
: If this option is on, when XBoard wants to display a message just before exiting, it brings up a modal dialog box and waits for you to click OK before exiting.

If the option is off, XBoard prints the message to standard error (the terminal) and exits immediately.

Popup Move Errors
: If this option is off, when you make an error in moving (such as attempting an illegal move or moving the wrong color piece), the error message is displayed in the message area. If the option is on, move errors are displayed in small popup windows like other errors. You can dismiss an error popup either by clicking its OK button or by clicking anywhere on the board, including downclicking to start a move.

Premove
: If this option is on while playing a game on an ICS, you can register your next planned move before it is your turn. Move the piece with the mouse in the ordinary way, and the starting and ending squares will be highlighted with a special color (red by default). When it is your turn, if your registered move is legal, XBoard will send it to ICS immediately; if not, it will be ignored and you can make a different move. If you change your mind about your premove, either make a different move, or double-click on any piece to cancel the move entirely.

Quiet Play
: If this option is on, XBoard will automatically issue an ICS *`set shout 0`* command whenever you start a game and a *`set shout 1`* command whenever you finish one. Thus, you will not be distracted by shouts from other ICS users while playing.

Show Coords
: If this option is on, XBoard displays algebraic coordinates along the board's left and bottom edges.

Show Thinking
: If this option is set, the chess engine's notion of the score and best line of play from the current position is displayed as it is thinking. The score indicates how many pawns ahead (or if negative, behind) the chess engine thinks it is. In matches between two machines, the score is prefixed by '`W`' or '`B`' to indicate whether it is showing White's thinking or Black's, and only the thinking of the engine that is on move is shown.

Test Legality
: If this option is on, XBoard tests whether the moves you try to make with the mouse are legal and refuses to let you make an illegal move. Moves loaded from a file with '`Load Game`' are also checked. If the option is off, all moves are accepted, but if a local chess engine or the ICS is active, they will still reject illegal moves. Turning off this option is useful if you are playing a chess variant with rules that XBoard does not understand. (Bughouse, suicide, and wild variants where the king may castle after starting on the d file are generally supported with Test Legality on.)

## 2.6 Help Menu

Info XBoard
: Displays the XBoard documentation in info format. For this feature to work, you must have the GNU info program installed on your system, and the file '`xboard.info`' must either be present in the current working directory, or have been installed by the '`make install`' command when you built XBoard.

Man XBoard
: Displays the XBoard documentation in man page format. For this feature to work, the file '`xboard.6`' must have been installed by the '`make install`' command when you built XBoard, and the directory it was placed in must be on the search path for your system's '`man`' command.

Hint
: Displays a move hint from the chess engine.

Book
: Displays a list of possible moves from the chess engine's opening book. The exact format depends on what chess engine you are using. With GNU Chess 4, the first column gives moves, the second column gives one possible response for each move, and the third column shows the number of lines in the book that include the move from the first column. If you select this option and nothing happens, the chess engine is out of its book or does not support this feature.

About XBoard
: Shows the current XBoard version number.

## 2.7 Other Shortcut Keys

Iconize
: Pressing the *i* or *c* key iconizes XBoard. The graphical icon displays a white knight if it is White's move, or a black knight if it is Black's move. If your X window manager displays only text icons, not graphical ones, check its documentation; there is probably a way to enable graphical icons. If you get black and white reversed, we would like to hear about it; see Chapter 8 [Problems], page 35 below for instructions on how to report this problem.

You can add or remove shortcut keys using the X resources `form.translations`. Here is an example of what would go in your '`.Xdefaults`' file:

```
XBoard*form.translations: \
  Shift<Key>?: AboutGameProc() \n\
  <Key>y: AcceptProc() \n\
  <Key>n: DeclineProc() \n\
  <Key>i: NothingProc()
```

Binding a key to `NothingProc` makes it do nothing, thus removing it as a shortcut key. The XBoard commands that can be bound to keys are:

```
AbortProc, AboutGameProc, AboutProc, AcceptProc, AdjournProc,
AlwaysQueenProc, AnalysisModeProc, AnalyzeFileProc,
AnimateDraggingProc, AnimateMovingProc, AutobsProc, AutoflagProc,
AutoflipProc, AutoraiseProc, AutosaveProc, BackwardProc,
BlindfoldProc, BookProc, CallFlagProc, CopyGameProc, CopyPositionProc,
DebugProc, DeclineProc, DrawProc, EditCommentProc, EditGameProc,
EditPositionProc, EditTagsProc, EnterKeyProc, FlashMovesProc,
```

FlipViewProc, ForwardProc, GetMoveListProc, HighlightLastMoveProc, HintProc, Iconify, IcsAlarmProc, IcsClientProc, IcsInputBoxProc, InfoProc, LoadGameProc, LoadNextGameProc, LoadNextPositionProc, LoadPositionProc, LoadPrevGameProc, LoadPrevPositionProc, LoadSelectedProc, MachineBlackProc, MachineWhiteProc, MailMoveProc, ManProc, MoveNowProc, MoveSoundProc, NothingProc, OldSaveStyleProc, PasteGameProc, PastePositionProc, PauseProc, PeriodicUpdatesProc, PonderNextMoveProc, PopupExitMessageProc, PopupMoveErrorsProc, PremoveProc, QuietPlayProc, QuitProc, ReloadCmailMsgProc, ReloadGameProc, ReloadPositionProc, RematchProc, ResetProc, ResignProc, RetractMoveProc, RevertProc, SaveGameProc, SavePositionProc, ShowCoordsProc, ShowGameListProc, ShowThinkingProc, StopExaminingProc, StopObservingProc, TestLegalityProc, ToEndProc, ToStartProc, TrainingProc, TruncateGameProc, and TwoMachinesProc.

# 3 Options

This section documents the command-line options to XBoard. You can set these options in two ways: by typing them on the shell command line you use to start XBoard, or by setting them as X resources (typically in your '`.Xdefaults`' file). Many of the options cannot be changed while XBoard is running; others set the initial state of items that can be changed with the Chapter 3 [Options], page 16 menu.

Most of the options have both a long name and a short name. To turn a boolean option on or off from the command line, either give its long name followed by the value true or false ('`-longOptionName true`'), or give just the short name to turn the option on ('`-opt`'), or the short name preceded by '`x`' to turn the option off ('`-xopt`'). For options that take strings or numbers as values, you can use the long or short option names interchangeably.

Each option corresponds to an X resource with the same name, so if you like, you can set options in your '`.Xdefaults`' file or in a file named '`XBoard`' in your home directory. For options that have two names, the longer one is the name of the corresponding X resource; the short name is not recognized. To turn a boolean option on or off as an X resource, give its long name followed by the value true or false ('`XBoard*longOptionName: true`').

## 3.1 Chess Engine Options

-tc or -timeControl minutes[:seconds]
: Each player begins with his clock set to the `timeControl` period. Default: 5 minutes. The additional options `movesPerSession` and `timeIncrement` are mutually exclusive.

-mps or -movesPerSession moves
: When both players have made `movesPerSession` moves, a new `timeControl` period is added to both clocks. Default: 40 moves.

-inc or -timeIncrement seconds
: If this option is specified, `movesPerSession` is ignored. Instead, after each player's move, `timeIncrement` seconds are added to his clock. Use '`-inc 0`' if you want to require the entire game to be played in one `timeControl` period, with no increment. Default: -1, which specifies `movesPerSession` mode.

-clock/-xclock or -clockMode true/false
: Determines whether or not to display the chess clocks. If clockMode is false, the clocks are not shown, but the side that is to play next is still highlighted. Also, unless `searchTime` is set, the chess engine still keeps track of the clock time and uses it to determine how fast to make its moves.

-st or -searchTime minutes[:seconds]
: Tells the chess engine to spend at most the given amount of time searching for each of its moves. Without this option, the chess engine chooses its search time based on the number of moves and amount of time remaining until the next time control. Setting this option also sets clockMode to false.

-depth or -searchDepth number
: Tells the chess engine to look ahead at most the given number of moves when searching for a move to make. Without this option, the chess engine chooses

its search depth based on the number of moves and amount of time remaining until the next time control. With the option, the engine will cut off its search early if it reaches the specified depth.

-thinking/-xthinking or -showThinking true/false
: Sets the Show Thinking option. See Section 2.5 [Options Menu], page 10. Default: false.

-ponder/-xponder or -ponderNextMove true/false
: Sets the Ponder Next Move menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-mg or -matchGames n
: Automatically runs an n-game match between two chess engines, with alternating colors. If the `loadGameFile` or `loadPositionFile` option is set, XBoard starts each game with the given opening moves or the given position; otherwise, the games start with the standard initial chess position. If the `saveGameFile` option is set, a move record for the match is appended to the specified file. If the `savePositionFile` option is set, the final position reached in each game of the match is appended to the specified file. When the match is over, XBoard displays the match score and exits. Default: 0 (do not run a match).

-mm/-xmm or -matchMode true/false
: Setting `matchMode` to true is equivalent to setting `matchGames` to 1.

-fcp or -firstChessProgram program
: Name of first chess engine. Default: '`gnuchessx`'.

-scp or -secondChessProgram program
: Name of second chess engine, if needed. A second chess engine is started only in Two Machines (match) mode. Default: '`gnuchessx`'.

-fb/-xfb or -firstPlaysBlack true/false
: In games between two chess engines, firstChessProgram normally plays white. If this option is true, firstChessProgram plays black. In a multi-game match, this option affects the colors only for the first game; they still alternate in subsequent games.

-fh or -firstHost host
-sh or -secondHost host
: Hosts on which the chess engines are to run. The default for each is '`localhost`'. If you specify another host, XBoard uses '`rsh`' to run the chess engine there. (You can substitute a different remote shell program for rsh using the `remoteShell` option described below.)

-fd or -firstDirectory dir
-sd or -secondDirectory dir
: Working directories in which the chess engines are to be run. The default is `""`, which means to run the chess engine in the same working directory as XBoard itself. (See the CHESSDIR environment variable.) This option is effective only when the chess engine is being run on the local host; it does not work if the engine is run remotely using the -fh or -sh option.

-initString string
-secondInitString string

The string that is sent to initialize each chess engine for a new game. Default:

```
new
random
```

Setting this option from the command line is tricky, because you must type in real newline characters, including one at the very end. In most shells you can do this by entering a '\' character followed by a newline. It is easier to set the option from your '.Xdefaults' file; in that case you can include the character sequence '\n' in the string, and it will be converted to a newline.

If you change this option, don't remove the 'new' command; it is required by all chess engines to start a new game.

You can remove the 'random' command if you like; including it causes GNU Chess 4 to randomize its move selection slightly so that it doesn't play the same moves in every game. Even without 'random', GNU Chess 4 randomizes its choice of moves from its opening book. Many other chess engines ignore this command entirely and always (or never) randomize.

You can also try adding other commands to the initString; see the documentation of the chess engine you are using for details.

-firstComputerString string
-secondComputerString string

The string that is sent to the chess engine if its opponent is another computer chess engine. The default is 'computer\n'. Probably the only useful alternative is the empty string (''), which keeps the engine from knowing that it is playing another computer.

-reuse/-xreuse or -reuseFirst true/false
-reuse2/-xreuse2 or -reuseSecond true/false

If the option is false, XBoard kills off the chess engine after every game and starts it again for the next game. If the option is true (the default), XBoard starts the chess engine only once and uses it repeatedly to play multiple games. Some old chess engines may not work properly when reuse is turned on, but otherwise games will start faster if it is left on.

-firstProtocolVersion version-number
-secondProtocolVersion version-number

This option specifies which version of the chess engine communication protocol to use. By default, version-number is 2. In version 1, the "protover" command is not sent to the engine; since version 1 is a subset of version 2, nothing else changes. Other values for version-number are not supported.

## 3.2 Internet Chess Server Options

-ics/-xics or -internetChessServerMode true/false

Connect with an Internet Chess Server to play chess against its other users, observe games they are playing, or review games that have recently finished. Default: false.

-icshost or -internetChessServerHost host
: The Internet host name or address of the chess server to connect to when in ICS mode. Default: `chessclub.com`. Another popular chess server to try is `freechess.org`. If your site doesn't have a working Internet name server, try specifying the host address in numeric form. You may also need to specify the numeric address when using the icshelper option with timestamp or timeseal (see below).

-icsport or -internetChessServerPort port-number
: The port number to use when connecting to a chess server in ICS mode. Default: 5000.

-icshelper or -internetChessServerHelper prog-name
: An external helper program used to communicate with the chess server. You would set it to "timestamp" for ICC (chessclub.com) or "timeseal" for FICS (freechess.org), after obtaining the correct version of timestamp or timeseal for your computer. See "help timestamp" on ICC and "help timeseal" on FICS. This option is shorthand for `-useTelnet -telnetProgram program`.

-telnet/-xtelnet or -useTelnet true/false
: This option is poorly named; it should be called useHelper. If set to true, it instructs XBoard to run an external program to communicate with the Internet Chess Server. The program to use is given by the telnetProgram option. If the option is false (the default), XBoard opens a TCP socket and uses its own internal implementation of the telnet protocol to communicate with the ICS. See Chapter 5 [Firewalls], page 31.

-telnetProgram prog-name
: This option is poorly named; it should be called helperProgram. It gives the name of the telnet program to be used with the `gateway` and `useTelnet` options. The default is '`telnet`'. The telnet program is invoked with the value of `internetChessServerHost` as its first argument and the value of `internetChessServerPort` as its second argument. See Chapter 5 [Firewalls], page 31.

-gateway host-name
: If this option is set to a host name, XBoard communicates with the Internet Chess Server by using '`rsh`' to run the `telnetProgram` on the given host, instead of using its own internal implementation of the telnet protocol. You can substitute a different remote shell program for '`rsh`' using the `remoteShell` option described below. See Chapter 5 [Firewalls], page 31.

-internetChessServerCommPort or -icscomm dev-name
: If this option is set, XBoard communicates with the ICS through the given character I/O device instead of opening a TCP connection. Use this option if your system does not have any kind of Internet connection itself (not even a SLIP or PPP connection), but you do have dialup access (or a hardwired terminal line) to an Internet service provider from which you can telnet to the ICS.

  The support for this option in XBoard is minimal. You need to set all communication parameters and tty modes before you enter XBoard.

Use a script something like this:

```
stty raw -echo 9600 > /dev/tty00
xboard -ics -icscomm /dev/tty00
```

Here replace '/dev/tty00' with the name of the device that your modem is connected to. You might have to add several more options to these stty commands. See the man pages for 'stty' and tty if you run into problems. Also, on many systems stty works on its standard input instead of standard output, so you have to use '<' instead of '>'.

If you are using linux, try starting with the script below. Change it as necessary for your installation.

```
#!/bin/sh -f
# configure modem and fire up XBoard

# configure modem
(
  stty 2400 ; stty raw ; stty hupcl ; stty -clocal
  stty ignbrk ; stty ignpar ; stty ixon ; stty ixoff
  stty -iexten ; stty -echo
) < /dev/modem
xboard -ics -icscomm /dev/modem
```

After you start XBoard in this way, type whatever commands are necessary to dial out to your Internet provider and log in. Then telnet to ICS, using a command like *telnet chessclub.com 5000*. Important: See the paragraph below about extra echoes, in Chapter 7 [Limitations], page 34.

-icslogon or -internetChessServerLogonScript file-name

Whenever XBoard connects to the Internet Chess Server, if it finds a file with the name given in this option, it feeds the file's contents to the ICS as commands. The default file name is '.icsrc'. Usually the first two lines of the file should be your ICS user name and password. The file can be either in $CHESSDIR, in XBoard's working directory if CHESSDIR is not set, or in your home directory.

-msLoginDelay delay

If you experience trouble logging on to an ICS when using the -icslogon option, inserting some delay between characters of the logon script may help. This option adds delay milliseconds of delay between characters. Good values to try are 100 and 250.

-icsinput/-xicsinput or -internetChessServerInputBox true/false

Sets the ICS Input Box menu option. See Section 2.2 [Mode Menu], page 5. Default: false.

-autocomm/-xautocomm or -autoComment true/false

Sets the Auto Comment menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-autoflag/-xautoflag or -autoCallFlag true/false

Sets the Auto Flag menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-autobs/-xautobs or -autoObserve true/false
: Sets the Auto Observe menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-moves/-xmoves or -getMoveList true/false
: Sets the Get Move List menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-alarm/-xalarm or -icsAlarm true/false
: Sets the ICS Alarm menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-icsAlarmTime ms
: Sets the time in milliseconds for the ICS Alarm menu option. See Section 2.5 [Options Menu], page 10. Default: 5000.

-pre/-xpre \fRor\fB -premove true/false
: Sets the Premove menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-quiet/-xquiet or -quietPlay true/false
: Sets the Quiet Play menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-colorizeMessages or -colorize
: Setting colorizeMessages to true tells XBoard to colorize the messages received from the ICS. Colorization works only if your xterm supports ISO 6429 escape sequences for changing text colors.

-colorShout foreground,background,bold
-colorSShout foreground,background,bold
-colorChannel1 foreground,background,bold
-colorChannel foreground,background,bold
-colorKibitz foreground,background,bold
-colorTell foreground,background,bold
-colorChallege foreground,background,bold
-colorRequest foreground,background,bold
-colorSeek foreground,background,bold
-colorNormal foreground,background,bold
: These options set the colors used when colorizing ICS messages. All ICS messages are grouped into one of these categories: shout, sshout, channel 1, other channel, kibitz, tell, challenge, request (including abort, adjourn, draw, pause, and takeback), or normal (all other messages).

  Each foreground or background argument can be one of the following: black, red, green, yellow, blue, magenta, cyan, white, or default. Here "default" means the default foreground or background color of your xterm. Bold can be 1 or 0. If background is omitted, "default" is assumed; if bold is omitted, 0 is assumed.

  Here is an example of how to set the colors in your '`.Xdefaults`' file. The colors shown here are the default values; you will get them if you turn `-colorize` on without specifying your own colors.

```
xboard*colorizeMessages: true
xboard*colorShout: green
xboard*colorSShout: green, black, 1
xboard*colorChannel1: cyan
xboard*colorChannel: cyan, black, 1
xboard*colorKibitz: magenta, black, 1
xboard*colorTell: yellow, black, 1
xboard*colorChallenge: red, black, 1
xboard*colorRequest: red
xboard*colorSeek: blue
xboard*colorNormal: default
```

-soundProgram progname
: If this option is set to a sound-playing program that is installed and working on your system, XBoard can play sound files when certain events occur, listed below. The default program name is "play". If any of the sound options is set to "$", the event rings the terminal bell by sending a ^G character to standard output, instead of playing a sound file. If an option is set to the empty string "", no sound is played for that event.

-soundShout filename
-soundSShout filename
-soundChannel filename
-soundKibitz filename
-soundTell filename
-soundChallenge filename
-soundRequest filename
-soundSeek filename
: These sounds are triggered in the same way as the colorization events described above. They all default to "", no sound. They are played only if the colorizeMessages is on.

-soundMove filename
: This sound is used by the Move Sound menu option. Default: "$".

-soundIcsAlarm filename
: This sound is used by the ICS Alarm menu option. Default: "$".

-soundIcsWin filename
: This sound is played when you win an ICS game. Default: "" (no sound).

-soundIcsLoss filename
: This sound is played when you lose an ICS game. Default: "" (no sound).

-soundIcsDraw filename
: This sound is played when you draw an ICS game. Default: "" (no sound).

-soundIcsUnfinished filename
: This sound is played when an ICS game that you are participating in is aborted, adjourned, or otherwise ends inconclusively. Default: "" (no sound).

Here is an example of how to set the sounds in your .Xdefaults file:

```
xboard*soundShout: shout.wav
xboard*soundSShout: sshout.wav
xboard*soundChannel1: channel1.wav
xboard*soundChannel: channel.wav
xboard*soundKibitz: kibitz.wav
xboard*soundTell: tell.wav
xboard*soundChallenge: challenge.wav
xboard*soundRequest: request.wav
xboard*soundSeek: seek.wav
xboard*soundMove: move.wav
xboard*soundIcsWin: win.wav
xboard*soundIcsLoss: lose.wav
xboard*soundIcsDraw: draw.wav
xboard*soundIcsUnfinished: unfinished.wav
xboard*soundIcsAlarm: alarm.wav
```

## 3.3 Load and Save Options

-lgf or -loadGameFile file
-lgi or -loadGameIndex index
: If the `loadGameFile` option is set, XBoard loads the specified game file at startup. The file name '`-`' specifies the standard input. If there is more than one game in the file, XBoard pops up a menu of the available games, with entries based on their PGN (Portable Game Notation) tags. If the `loadGameIndex` option is set to '`N`', the menu is suppressed and the N th game found in the file is loaded immediately. The menu is also suppressed if `matchMode` is enabled or if the game file is a pipe; in these cases the first game in the file is loaded immediately. Use the '`pxboard`' shell script provided with XBoard if you want to pipe in files containing multiple games and still see the menu.

-td or -timeDelay seconds
: Time delay between moves during '`Load Game`'. Fractional seconds are allowed; try '`-td 0.4`'. A time delay value of -1 tells XBoard not to step through game files automatically. Default: 1 second.

-sgf or -saveGameFile file
: If this option is set, XBoard appends a record of every game played to the specified file. The file name '`-`' specifies the standard output.

-autosave/-xautosave or -autoSaveGames true/false
: Sets the Auto Save menu option. See Section 2.5 [Options Menu], page 10. Default: false. Ignored if `saveGameFile` is set.

-lpf or -loadPositionFile file
-lpi or -loadPositionIndex index
: If the `loadPositionFile` option is set, XBoard loads the specified position file at startup. The file name '`-`' specifies the standard input. If the `loadPositionIndex` option is set to N, the Nth position found in the file is loaded; otherwise the first position is loaded.

-spf or -savePositionFile file
: If this option is set, XBoard appends the final position reached in every game played to the specified file. The file name '-' specifies the standard output.

-oldsave/-xoldsave or -oldSaveStyle true/false
: Sets the Old Save Style menu option. See Section 2.5 [Options Menu], page 10. Default: false.

## 3.4 User Interface Options

-display
-geometry
-iconic
: These and most other standard Xt options are accepted.

-movesound/-xmovesound or -ringBellAfterMoves true/false
: Sets the Move Sound menu option. See Section 2.5 [Options Menu], page 10. Default: false. For compatibility with old XBoard versions, -bell/-xbell are also accepted as abbreviations for this option.

-exit/-xexit or -popupExitMessage true/false
: Sets the Popup Exit Message menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-popup/-xpopup or -popupMoveErrors true/false
: Sets the Popup Move Errors menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-queen/-xqueen or -alwaysPromoteToQueen true/false
: Sets the Always Queen menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-legal/-xlegal or -testLegality true/false
: Sets the Test Legality menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-size or -boardSize (sizeName | n1,n2,n3,n4,n5,n6,n7)
: Determines how large the board will be, by selecting the pixel size of the pieces and setting a few related parameters. The sizeName can be one of: Titanic, giving 129x129 pixel pieces, Colossal 116x116, Giant 108x108, Huge 95x95, Big 87x87, Large 80x80, Bulky 72x72, Medium 64x64, Moderate 58x58, Average 54x54, Middling 49x49, Mediocre 45x45, Small 40x40, Slim 37x37, Petite 33x33, Dinky 29x29, Teeny 25x25, or Tiny 21x21. Pieces of all these sizes are built into XBoard. Other sizes can be used if you have them; see the pixmapDirectory and bitmapDirectory options. The default depends on the size of your screen; it is approximately the largest size that will fit without clipping.

  You can select other sizes or vary other layout parameters by providing a list of comma-separated values (with no spaces) as the argument. You do not need to provide all the values; for any you omit from the end of the list, defaults are taken from the nearest built-in size. The value `n1` gives the piece size, `n2` the width of the black border between squares, `n3` the desired size for the clockFont, `n4` the desired size for the coordFont, `n5` the desired size for the default font, `n6` the smallLayout flag (0 or 1), and `n7` the tinyLayout flag (0 or 1). All dimensions are in pixels. If the border between squares is eliminated (0 width), the various highlight options will not work, as there is nowhere to draw the highlight. If smallLayout is 1 and `titleInWindow` is true, the window layout is rearranged to make more room for the title. If tinyLayout is 1, the labels on the menu bar are abbreviated to one character each and the buttons in the button bar are made narrower.

-coords/-xcoords or -showCoords true/false
: Sets the Show Coords menu option. See Section 2.5 [Options Menu], page 10. Default: false. The `coordFont` option specifies what font to use.

-autoraise/-xautoraise or -autoRaiseBoard true/false
: Sets the Auto Raise Board menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-autoflip/-xautoflip or -autoFlipView true/false
: Sets the Auto Flip View menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-flip/-xflip or -flipView true/false
: If Auto Flip View is not set, or if you are observing but not participating in a game, then the positioning of the board at the start of each game depends on the flipView option. If flipView is false (the default), the board is positioned so that the white pawns move from the bottom to the top; if true, the black pawns move from the bottom to the top. In any case, the Flip menu option (see Section 2.5 [Options Menu], page 10) can be used to flip the board after the game starts.

-title/-xtitle or -titleInWindow true/false
: If this option is true, XBoard displays player names (for ICS games) and game file names (for '`Load Game`') inside its main window. If the option is false (the default), this information is displayed only in the window banner. You probably won't want to set this option unless the information is not showing up in the banner, as happens with a few X window managers.

-buttons/-xbuttons or -showButtonBar True/False
: If this option is False, xboard omits the [`<<`] [`<`] [P] [`>`] [`>>`] button bar from the window, allowing the message line to be wider. You can still get the functions of these buttons using the menus or their keyboard shortcuts. Default: true.

-mono/-xmono or -monoMode true/false
: Determines whether XBoard displays its pieces and squares with two colors (true) or four (false). You shouldn't have to specify `monoMode`; XBoard will determine if it is necessary.

-flashCount count
-flashRate rate
-flash/-xflash
: These options enable flashing of pieces when they land on their destination square. `flashCount` tells XBoard how many times to flash a piece after it lands

on its destination square. `flashRate` controls the rate of flashing (flashes/sec). Abbreviations: `flash` sets flashCount to 3. `xflash` sets flashCount to 0. Defaults: flashCount=0 (no flashing), flashRate=5.

-highlight/-xhighlight or -highlightLastMove true/false
: Sets the Highlight Last Move menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-blind/-xblind or -blindfold true/false
: Sets the Blindfold menu option. See Section 2.5 [Options Menu], page 10. Default: false.

-clockFont font
: The font used for the clocks. If the option value is a pattern that does not specify the font size, XBoard tries to choose an appropriate font for the board size being used. Default: -*-helvetica-bold-r-normal–*-*-*-*-*-*-*-*.

-coordFont font
: The font used for rank and file coordinate labels if `showCoords` is true. If the option value is a pattern that does not specify the font size, XBoard tries to choose an appropriate font for the board size being used. Default: -*-helvetica-bold-r-normal–*-*-*-*-*-*-*-*.

-font font
: The font used for popup dialogs, menus, comments, etc. If the option value is a pattern that does not specify the font size, XBoard tries to choose an appropriate font for the board size being used. Default: -*-helvetica-medium-r-normal–*-*-*-*-*-*-*-*.

-fontSizeTolerance tol
: In the font selection algorithm, a nonscalable font will be preferred over a scalable font if the nonscalable font's size differs by `tol` pixels or less from the desired size. A value of -1 will force a scalable font to always be used if available; a value of 0 will use a nonscalable font only if it is exactly the right size; a large value (say 1000) will force a nonscalable font to always be used if available. Default: 4.

-bm or -bitmapDirectory dir
-pixmap or -pixmapDirectory dir
: These options control what piece images xboard uses. The XBoard distribution includes one set of pixmap pieces in xpm format, in the directory '`pixmaps`', and one set of bitmap pieces in xbm format, in the directory '`bitmaps`'. Pixmap pieces give a better appearance on the screen: the white pieces have dark borders, and the black pieces have opaque internal details. With bitmaps, neither piece color has a border, and the internal details are transparent; you see the square color or other background color through them.

  If XBoard is configured and compiled on a system that includes libXpm, the X pixmap library, the xpm pixmap pieces are compiled in as the default. A different xpm piece set can be selected at runtime with the `pixmapDirectory` option, or a bitmap piece set can be selected with the `bitmapDirectory` option.

  If XBoard is configured and compiled on a system that does not include libXpm (or the `--disable-xpm` option is given to the configure program), the bitmap pieces are compiled in as the default. It is not possible to use xpm pieces in this case, but pixmap pieces in another format called "xim" can be used by giving the `pixmapDirectory` option. Or again, a different bitmap piece set can be selected with the `bitmapDirectory` option.

Files in the `bitmapDirectory` must be named as follows: The first character of a piece bitmap name gives the piece it represents ('`p`', '`n`', '`b`', '`r`', '`q`', or '`k`'), the next characters give the size in pixels, the following character indicates whether the piece is solid or outline ('`s`' or '`o`'), and the extension is '`.bm`'. For example, a solid 80x80 knight would be named '`n80s.bm`'. The outline bitmaps are used only in monochrome mode. If bitmap pieces are compiled in and the bitmapDirectory is missing some files, the compiled in pieces are used instead.

If the bitmapDirectory option is given, it is also possible to replace xboard's icons and menu checkmark, by supplying files named '`icon_white.bm`', '`icon_black.bm`', and '`checkmark.bm`'.

For more information about pixmap pieces and how to get additional sets, see Section 11.3 [zic2xpm], page 41 below.

-whitePieceColor color
-blackPieceColor color
-lightSquareColor color
-darkSquareColor color
-highlightSquareColor color

Colors to use for the pieces, squares, and square highlights. Defaults:

```
-whitePieceColor       #FFFFCC
-blackPieceColor       #202020
-lightSquareColor      #C8C365
-darkSquareColor       #77A26D
-highlightSquareColor  #FFFF00
-premoveHighlightColor #FF0000
```

On a grayscale monitor you might prefer:

```
-whitePieceColor       gray100
-blackPieceColor       gray0
-lightSquareColor      gray80
-darkSquareColor       gray60
-highlightSquareColor  gray100
-premoveHighlightColor gray70
```

-drag/-xdrag or -animateDragging true/false

Sets the Animate Dragging menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-animate/-xanimate or -animateMoving true/false

Sets the Animate Moving menu option. See Section 2.5 [Options Menu], page 10. Default: true.

-animateSpeed n

Number of milliseconds delay between each animation frame when Animate Moves is on.

## 3.5 Other Options

-ncp/-xncp or -noChessProgram true/false

If this option is true, XBoard acts as a passive chessboard; it does not start a chess engine at all. Turning on this option also turns off clockMode. Default: false.

-mode or -initialMode modename

If this option is given, XBoard selects the given modename from the Mode menu after starting and (if applicable) processing the loadGameFile or loadPositionFile option. Default: `""` (no selection). Other supported values are MachineWhite, MachineBlack, TwoMachines, Analysis, AnalyzeFile, EditGame, EditPosition, and Training.

-variant varname

Activates preliminary, partial support for playing chess variants against a local engine or editing variant games. This flag is not needed in ICS mode. Recognized variant names are:

```
normal        Normal chess
wildcastle    Shuffle chess, king can castle from d file
nocastle      Shuffle chess, no castling allowed
fischerandom  Fischer Random shuffle chess
bughouse      Bughouse, ICC/FICS rules
crazyhouse    Crazyhouse, ICC/FICS rules
losers        Lose all pieces or get mated (ICC wild 17)
suicide       Lose all pieces including king (FICS)
giveaway      Try to have no legal moves (ICC wild 26)
twokings      Weird ICC wild 9
kriegspiel    Opponent's pieces are invisible
atomic        Capturing piece explodes (ICC wild 27)
3check        Win by giving check 3 times (ICC wild 25)
shatranj      An ancient precursor of chess (ICC wild 28)
unknown       Catchall for other unknown variants
```

In the shuffle variants, xboard does not shuffle the pieces, but you can do it by hand using Edit Position. Some variants are supported only in ICS mode, including fischerandom, bughouse, and kriegspiel. The winning/drawing conditions in crazyhouse (offboard interposition on mate), losers, suicide, giveaway, atomic, and 3check are not fully understood. In crazyhouse, xboard does not yet keep track of offboard pieces. Shatranj is unsupported, but it may be usable if you turn off Test Legality.

-debug/-xdebug or -debugMode true/false

Turns on debugging printout.

-rsh or -remoteShell shell-name

Name of the command used to run programs remotely. The default is '`rsh`' or '`remsh`', determined when XBoard is configured and compiled.

-ruser or -remoteUser user-name

User name on the remote system when running programs with the `remoteShell`. The default is your local user name.

# 4 Chess Servers

An *Internet Chess Server*, or *ICS*, is a place on the Internet where people can get together to play chess, watch other people's games, or just chat. You can use either `telnet` or a client program like XBoard to connect to the server. There are thousands of registered users on the different ICS hosts, and it is not unusual to meet 200 on both chessclub.com and freechess.org.

Most people can just type *`xboard -ics`* to start XBoard as an ICS client. Invoking XBoard in this way connects you to the Internet Chess Club (ICC), a commercial ICS. You can log in there as a guest even if you do not have a paid account. To connect to the largest Free ICS (FICS), use the command *`xboard -ics -icshost freechess.org`* instead, or substitute a different host name to connect to your favorite ICS. For a full description of command-line options that control the connection to ICS and change the default values of ICS options, see Section 3.2 [ICS options], page 18.

While you are running XBoard as an ICS client, you use the terminal window that you started XBoard from as a place to type in commands and read information that is not available on the chessboard.

The first time you need to use the terminal is to enter your login name and password, if you are a registered player. (You don't need to do this manually; the `icsLogon` option can do it for you. see Section 3.2 [ICS options], page 18.) If you are not registered, enter *`g`* as your name, and the server will pick a unique guest name for you.

Some useful ICS commands include

*`help <topic>`*
: to get help on the given <topic>. To get a list of possible topics type *help* without topic. Try the help command before you ask other people on the server for help.

  For example *`help register`* tells you how to become a registered ICS player.

*`who <flags>`*
: to see a list of people who are logged on. Administrators (people you should talk to if you have a problem) are marked with the character '`*`', an asterisk. The <flags> allow you to display only selected players: For example, *`who of`* shows a list of players who are interested in playing but do not have an opponent.

*`games`*
: to see what games are being played

*`match <player> [<mins>] [<inc>]`*
: to challenge another player to a game. Both opponents get <mins> minutes for the game, and <inc> seconds will be added after each move. If another player challenges you, the server asks if you want to accept the challenge; use the *`accept`* or *`decline`* commands to answer.

*`accept`*
*`decline`*
: to accept or decline another player's offer. The offer may be to start a new game, or to agree to a *`draw`*, *`adjourn`* or *`abort`* the current game. See Section 2.3 [Action Menu], page 8.

  If you have more than one pending offer (for example, if more than one player is challenging you, or if your opponent offers both a draw and to adjourn the game), you have to supply additional information, by typing something like *accept <player>*, *accept draw*, or *draw*.

*draw*
*adjourn*
*abort* asks your opponent to terminate a game by mutual agreement. Adjourned games can be continued later. Your opponent can either *decline* your offer or accept it (by typing the same command or typing *accept*). In some cases these commands work immediately, without asking your opponent to agree. For example, you can abort the game unilaterally if your opponent is out of time, and you can claim a draw by repetition or the 50-move rule if available simply by typing *draw*.

*finger <player>*
to get information about the given <player>. (Default: yourself.)

*vars* to get a list of personal settings

*set <var> <value>*
to modify these settings

*observe <player>*
to observe an ongoing game of the given <player>.

*examine*
*oldmoves* to review a recently completed game

Some special XBoard features are activated when you are in examine mode on ICS. See the descriptions of the menu commands '`Forward`', '`Backward`', '`Pause`', '`ICS Client`', and '`Stop Examining`' on the Section 2.4 [Step Menu], page 9, Section 2.2 [Mode Menu], page 5, and Section 2.5 [Options Menu], page 10.

# 5 Firewalls

By default, XBoard communicates with an Internet Chess Server by opening a TCP socket directly from the machine it is running on to the ICS. If there is a firewall between your machine and the ICS, this won't work. Here are some recipes for getting around common kinds of firewalls using special options to XBoard. Important: See the paragraph in the below about extra echoes, in Chapter 7 [Limitations], page 34.

Suppose that you can't telnet directly to ICS, but you can telnet to a firewall host, log in, and then telnet from there to ICS. Let's say the firewall is called '`firewall.example.com`'. Set command-line options as follows:

```
xboard -ics -icshost firewall.example.com -icsport 23
```

Or in your '`.Xdefaults`' file:

```
XBoard*internetChessServerHost: firewall.example.com
XBoard*internetChessServerPort: 23
```

Then when you run XBoard in ICS mode, you will be prompted to log in to the firewall host. This works because port 23 is the standard telnet login service. Do so, then telnet to ICS, using a command like '`telnet chessclub.com 5000`', or whatever command the firewall provides for telnetting to port 5000.

If your firewall lets you telnet (or rlogin) to remote hosts but doesn't let you telnet to port 5000, you may be able to connect to the chess server on port 23 instead, which is the port the telnet program uses by default. Some chess servers support this (including chessclub.com and freechess.org), while some do not.

If your chess server does not allow connections on port 23 and your firewall does not allow you to connect to other ports, you may be able to connect by hopping through another host outside the firewall that you have an account on. For instance, suppose you have a shell account at '`foo.edu`'. Follow the recipe above, but instead of typing '`telnet chessclub.com 5000`' to the firewall, type '`telnet foo.edu`' (or '`rlogin foo.edu`'), log in there, and then type '`telnet chessclub.com 5000`'.

Suppose that you can't telnet directly to ICS, but you can use rsh to run programs on a firewall host, and that host can telnet to ICS. Let's say the firewall is called '`rsh.example.com`'. Set command-line options as follows:

```
xboard -ics -gateway rsh.example.com -icshost chessclub.com
```

Or in your '`.Xdefaults`' file:

```
XBoard*gateway: rsh.example.com
XBoard*internetChessServerHost: chessclub.com
```

Then when you run XBoard in ICS mode, it will connect to the ICS by using '`rsh`' to run the command '`telnet chessclub.com 5000`' on host '`rsh.example.com`'.

Suppose that you can telnet anywhere you want, but you have to run a special program called '`ptelnet`' to do so.

First, we'll consider the easy case, in which '`ptelnet chessclub.com 5000`' gets you to the chess server. In this case set command line options as follows:

```
xboard -ics -telnet -telnetProgram ptelnet
```

Or in your '`.Xdefaults`' file:

```
XBoard*useTelnet: true
XBoard*telnetProgram: ptelnet
```

Then when you run XBoard in ICS mode, it will issue the command '`ptelnet chessclub.com 5000`' to connect to the ICS.

Next, suppose that '`ptelnet chessclub.com 5000`' doesn't work; that is, your '`ptelnet`' program doesn't let you connect to alternative ports. As noted above, your chess server may allow you to connect on port 23 instead. In that case, just add the option '`-icsport ""`' to the above command, or add '`XBoard*internetChessServerPort:`' to your '`.Xdefaults`' file. But if your chess server doesn't let you connect on port 23, you will have to find some other host outside the firewall and hop through it. For instance, suppose you have a shell account at '`foo.edu`'. Set command line options as follows:

```
xboard -ics -telnet -telnetProgram ptelnet -icshost foo.edu -icsport ""
```

Or in your '`.Xdefaults`' file:

```
XBoard*useTelnet: true
XBoard*telnetProgram: ptelnet
XBoard*internetChessServerHost: foo.edu
XBoard*internetChessServerPort:
```

Then when you run XBoard in ICS mode, it will issue the command '`ptelnet foo.edu`' to connect to your account at '`foo.edu`'. Log in there, then type '`telnet chessclub.com 5000`'.

ICC timestamp and FICS timeseal do not work through some firewalls. You can use them only if your firewall gives a clean TCP connection with a full 8-bit wide path. If your firewall allows you to get out only by running a special telnet program, you can't use timestamp or timeseal across it. But if you have access to a computer just outside your firewall, and you have much lower netlag when talking to that computer than to the ICS, it might be worthwhile running timestamp there. Follow the instructions above for hopping through a host outside the firewall (foo.edu in the example), but run timestamp or timeseal on that host instead of telnet.

Suppose that you have a SOCKS firewall that will give you a clean 8-bit wide TCP connection to the chess server, but only after you authenticate yourself via the SOCKS protocol. In that case, you could make a socksified version of XBoard and run that. If you are using timestamp or timeseal, you will to socksify it, not XBoard; this may be difficult seeing that ICC and FICS do not provide source code for these programs. Socksification is beyond the scope of this document, but see the SOCKS Web site at http://www.socks.permeo.com/. If you are missing SOCKS, try http://www.funbureau.com/.

# 6 Environment variables

Game and position files are found in a directory named by the CHESSDIR environment variable. If this variable is not set, the current working directory is used. If CHESSDIR is set, XBoard actually changes its working directory to $CHESSDIR, so any files written by the chess engine will be placed there too.

# 7 Limitations and known bugs

There is no way for two people running copies of XBoard to play each other without going through an Internet Chess Server.

Under some circumstances, your ICS password may be echoed when you log on.

If you are connecting to the ICS by running telnet on an Internet provider or firewall host, you may find that each line you type is echoed back an extra time after you hit ⟨Enter⟩. If your Internet provider is a Unix system, you can probably turn its echo off by typing *stty -echo* after you log in, and/or typing ⟨^E⟩⟨Enter⟩ (Ctrl+E followed by the Enter key) to the telnet program after you have logged into ICS. It is a good idea to do this if you can, because the extra echo can occasionally confuse XBoard's parsing routines.

The game parser recognizes only algebraic notation.

The internal move legality tester does not look at the game history, so in some cases it misses illegal castling or en passant captures. It permits castling with the king on the d file because this is possible in some "wild 1" games on ICS. It does not check piece drops in bughouse and crazyhouse to see if you actually hold the piece you are trying to drop. However, if you attempt an illegal move when using a chess engine or chess server, XBoard will accept the error message that comes back, undo the move, and let you try another.

Fischer Random castling is not understood. You can probably play Fischer Random successfully on ICS by typing castling moves into the ICS Interaction window, but they will not be animated correctly, and saved games will not be loaded correctly if castling occurs.

FEN positions saved by XBoard never include correct information about whether castling is legal or how many half-moves have been made since the last irreversible move, and sometimes may not correctly indicate when en passant capture is available.

The mate detector does not understand that non-contact mate is not really mate in bughouse and crazyhouse. The only problem this causes while playing is minor: a '#' (mate indicator) character will show up after a non-contact mating move in the move list; XBoard will not assume the game is over at that point. However, if you are editing a game, Edit Game mode will be terminated by a non-contact mate.

The menus may not work if your keyboard is in Caps Lock or Num Lock mode. This seems to be a problem with the Athena menu widget, not an XBoard bug.

Also see the ToDo file included with the distribution for many other possible bugs, limitations, and ideas for improvement that have been suggested.

# 8 Reporting problems

Report bugs and problems with XBoard to <bug-xboard@gnu.org>.

Please use the 'script' program to start a typescript, run XBoard with the '-debug' option, and include the typescript output in your message. Also tell us what kind of machine and what operating system version you are using. The command 'uname -a' will often tell you this. Here is a sample of approximately what you should type:

```
script
uname -a
./configure
make
./xboard -debug
exit
mail bug-xboard@gnu.org
Subject: Your short description of the problem
Your detailed description of the problem
~r typescript
.
```

If you improve XBoard, please send a message about your changes, and we will get in touch with you about merging them in to the main line of development. Also see our Web site at http://savannah.gnu.org/projects/xboard/.

# 9 Authors and contributors

Tim Mann has been responsible for XBoard versions 1.3 and beyond, and for WinBoard, a port of XBoard to Microsoft Win32 (Windows NT and Windows 95).

Mark Williams contributed the initial (WinBoard-only) implementation of many new features added to both XBoard and WinBoard in version 4.1.0, including copy/paste, premove, icsAlarm, autoFlipView, training mode, auto raise, and blindfold. Ben Nye contributed X copy/paste code for XBoard.

Hugh Fisher added animated piece movement to XBoard, and Henrik Gram (henrikg@funcom.com) added it to WinBoard. Frank McIngvale added click/click moving, the Analysis modes, piece flashing, ZIICS import, and ICS text colorization to XBoard. Jochen Wiedmann ported XBoard to the Amiga, creating AmyBoard, and converted the documentation to texinfo. Elmar Bartel contributed the new piece bitmaps introduced in version 3.2. John Chanak contributed the initial implementation of ICS mode. The color scheme and the old 80x80 piece bitmaps were taken from Wayne Christopher's `XChess` program.

Chris Sears and Dan Sears wrote the original XBoard. They were responsible for versions 1.0 through 1.2.

Evan Welsh wrote `CMail`. Patrick Surry helped in designing, testing, and documenting CMail.

# 10 CMail

The '`cmail`' program can help you play chess by email with opponents of your choice using XBoard as an interface.

You will usually run '`cmail`' without giving any options.

## 10.1 CMail options

-h
: Displays '`cmail`' usage information.

-c
: Shows the conditions of the GNU General Public License. See [Copying], page 43.

-w
: Shows the warranty notice of the GNU General Public License. See [Copying], page 43.

-v
-xv
: Provides or inhibits verbose output from '`cmail`' and XBoard, useful for debugging. The **`-xv`** form also inhibits the cmail introduction message.

-mail
-xmail
: Invokes or inhibits the sending of a mail message containing the move.

-xboard
-xxboard
: Invokes or inhibits the running of XBoard on the game file.

-reuse
-xreuse
: Invokes or inhibits the reuse of an existing XBoard to display the current game.

-remail
: Resends the last mail message for that game. This inhibits running XBoard.

-game <name>
: The name of the game to be processed.

-wgames <number>
-bgames <number>
-games <number>
: Number of games to start as White, as Black or in total. Default is 1 as white and none as black. If only one color is specified then none of the other color is assumed. If no color is specified then equal numbers of White and Black games are started, with the extra game being as White if an odd number of total games is specified.

-me <short name>
-opp <short name>
: A one-word alias for yourself or your opponent.

-wname <full name>
-bname <full name>
-name <full name>
-oppname <full name>
: The full name of White, Black, yourself or your opponent.

-wna <net address>
-bna <net address>
-na <net address>
-oppna <net address>
: The email address of White, Black, yourself or your opponent.

-dir <directory>
: The directory in which 'cmail' keeps its files. This defaults to the environment variable $CMAIL_DIR or failing that, $CHESSDIR, '$HOME/Chess' or '~/Chess'. It will be created if it does not exist.

-arcdir <directory>
: The directory in which 'cmail' archives completed games. Defaults to the environment variable $CMAIL_ARCDIR or, in its absence, the same directory as cmail keeps its working files (above).

-mailprog <mail program>
: The program used by cmail to send email messages. This defaults to the environment variable $CMAIL_MAILPROG or failing that '/usr/ucb/Mail', '/usr/ucb/mail' or 'Mail'. You will need to set this variable if none of the above paths fit your system.

-gamesFile <file>
: A file containing a list of games with email addresses. This defaults to the environment variable $CMAIL_GAMES or failing that '.cmailgames'.

-aliasesFile <file>
: A file containing one or more aliases for a set of email addresses. This defaults to the environment variable $CMAIL_ALIASES or failing that '.cmailaliases'.

-logFile <file>
: A file in which to dump verbose debugging messages that are invoked with the '-v' option.

-event <event>
: The PGN Event tag (default 'Email correspondence game').

-site <site>
: The PGN Site tag (default 'NET').

-round <round>
: The PGN Round tag (default '-', not applicable).

-mode <mode>
: The PGN Mode tag (default 'EM', Electronic Mail).

Other options
: Any option flags not listed above are passed through to XBoard. Invoking XBoard through CMail changes the default values of two XBoard options: The default value for '-noChessProgram' is changed to true; that is, by default no chess engine is started. The default value for '-timeDelay' is changed to 0; that is, by default XBoard immediately goes to the end of the game as played so far, rather than stepping through the moves one by one. You can still set these

options to whatever values you prefer by supplying them on CMail's command line. See Chapter 3 [Options], page 16.

## 10.2 Starting a CMail Game

Type 'cmail' from a shell to start a game as white. After an opening message, you will be prompted for a game name, which is optional—if you simply press ⟨Enter⟩, the game name will take the form 'you-VS-opponent'. You will next be prompted for the short name of your opponent. If you haven't played this person before, you will also be prompted for his/her email address. 'cmail' will then invoke XBoard in the background. Make your first move and select 'Mail Move' from the 'File' menu. See Section 2.1 [File Menu], page 3. If all is well, 'cmail' will mail a copy of the move to your opponent. If you select 'Exit' without having selected 'Mail Move' then no move will be made.

## 10.3 Answering a Move

When you receive a message from an opponent containing a move in one of your games, simply pipe the message through 'cmail'. In some mailers this is as simple as typing `| cmail` when viewing the message, while in others you may have to save the message to a file and do *cmail* < *file* at the command line. In either case 'cmail' will display the game using XBoard. If you didn't exit XBoard when you made your first move then 'cmail' will do its best to use the existing XBoard instead of starting a new one. As before, simply make a move and select 'Mail Move' from the 'File' menu. See Section 2.1 [File Menu], page 3. 'cmail' will try to use the XBoard that was most recently used to display the current game. This means that many games can be in progress simultaneously, each with its own active XBoard.

If you want to look at the history or explore a variation, go ahead, but you must return to the current position before XBoard will allow you to mail a move. If you edit the game's history you must select 'Reload Same Game' from the 'File' menu to get back to the original position, then make the move you want and select 'Mail Move'. As before, if you decide you aren't ready to make a move just yet you can either select 'Exit' without sending a move or just leave XBoard running until you are ready.

## 10.4 Multi-Game Messages

It is possible to have a 'cmail' message carry more than one game. This feature was implemented to handle IECG (International Email Chess Group) matches, where a match consists of one game as white and one as black, with moves transmitted simultaneously. In case there are more general uses, 'cmail' itself places no limit on the number of black/white games contained in a message; however, XBoard does.

## 10.5 Completing a Game

Because XBoard can detect checkmate and stalemate, 'cmail' handles game termination sensibly. As well as resignation, the 'Action' menu allows draws to be offered and accepted for 'cmail' games.

For multi-game messages, only unfinished and just-finished games will be included in email messages. When all the games are finished, they are archived in the user's archive directory, and similarly in the opponent's when he or she pipes the final message through '`cmail`'. The archive file name includes the date the game was started.

## 10.6 Known CMail Problems

It's possible that a strange conjunction of conditions may occasionally mean that '`cmail`' has trouble reactivating an existing XBoard. If this should happen, simply trying it again should work. If not, remove the file that stores the XBoard's PID ('`game.pid`') or use the '`-xreuse`' option to force '`cmail`' to start a new XBoard.

Versions of '`cmail`' after 2.16 no longer understand the old file format that XBoard used to use and so cannot be used to correspond with anyone using an older version.

Versions of '`cmail`' older than 2.11 do not handle multi-game messages, so multi-game correspondence is not possible with opponents using an older version.

# 11 Other programs you can use with XBoard

Here are some other programs you can use with XBoard

## 11.1 GNU Chess

The GNU Chess engine is available from:

ftp://ftp.gnu.org/gnu/gnuchess/

You can use XBoard to play a game against GNU Chess, or to interface GNU Chess to an ICS.

## 11.2 Crafty

Crafty is a chess engine written by Bob Hyatt. You can use XBoard to play a game against Crafty, hook Crafty up to an ICS, or use Crafty to interactively analyze games and positions for you.

Crafty is a strong, rapidly evolving chess program. This rapid pace of development is good, because it means Crafty is always getting better. This can sometimes cause problems with backwards compatibility, but usually the latest version of Crafty will work well with the latest version of XBoard. Crafty can be obtained from its author's FTP site: ftp://ftp.cis.uab.edu/hyatt/.

To use Crafty with XBoard, give the -fcp and -fd options as follows, where <crafty's directory> is the directory in which you installed Crafty and placed its book and other support files.

## 11.3 zic2xpm

The "zic2xpm" program is used to import chess sets from the ZIICS(*) program into XBoard. "zic2xpm" is part of the XBoard distribution. ZIICS is available from:

ftp://ftp.freechess.org/pub/chess/DOS/ziics131.exe

To import ZIICS pieces, do this:

1. Unzip ziics131.exe into a directory:
   ```
   unzip -L ziics131.exe -d ~/ziics
   ```
2. Use zic2xpm to convert a set of pieces to XBoard format.
   For example, let's say you want to use the FRITZ4 set. These files are named "fritz4.*" in the ZIICS distribution.
   ```
   mkdir ~/fritz4
   cd ~/fritz4
   zic2xpm ~/ziics/fritz4.*
   ```
3. Give XBoard the "-pixmap" option when starting up, e.g.:
   ```
   xboard -pixmap ~/fritz4
   ```
   Alternatively, you can add this line to your .Xdefaults file:
   ```
   xboard*pixmapDirectory: ~/fritz4
   ```

(*) ZIICS is a separate copyrighted work of Andy McFarland. The "ZIICS pieces" are copyrighted works of their respective creators. Files produced by "zic2xpm" are for PERSONAL USE ONLY and may NOT be redistributed without explicit permission from the original creator(s) of the pieces.

# GNU GENERAL PUBLIC LICENSE

Version 2, June 1991



## Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

## TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

   Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

   a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

   b. You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

   c. If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

   These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:
   a. Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   b. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
   c. Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

   The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

   If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you

indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

   If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

   It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

   This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

   Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

## NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## END OF TERMS AND CONDITIONS

## How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
one line to give the program's name and an idea of what it does.
Copyright (C) 19yy  name of author

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111 USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type 'show w'.  This is free software, and you are welcome
to redistribute it under certain conditions; type 'show c'
for details.
```

The hypothetical commands '`show w`' and '`show c`' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than '`show w`' and '`show c`'; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright
interest in the program 'Gnomovision'
(which makes passes at compilers) written
by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

# Index

## D



## E



## F



## G



## H



## I



## K

## L



## M



## N



## O



## P



## Q

## R



## S



## T



## U



## V



## W



## X

# Table of Contents