GNU Backgammon

version 0.11
April, 2001

Gary Wong

1 Introduction

GNU Backgammon (`gnubg`) plays and analyses backgammon games and matches.

It is currently a work-in-progress. So far it is able to play individual games and tournament matches, evaluate and roll out positions, tune its own evaluation functions using either TD or supervised training, maintain databases of positions for training and other purposes, and more.

It is driven by a command-line interface, and displays an ASCII rendition of a board on text-only terminals, but also allows the user to play games and manipulate positions with a GTK+ board window where available. It is extensible on platforms which support Guile.

It currently plays at about the level of a championship flight tournament player (depending on its parameters and its luck in recent games, it rates from around 1900 to 2000 on FIBS, the First Internet Backgammon Server — at its strongest, it ranks in the top 5 of over 6000 rated players there) and is gradually improving; it should be somewhat stronger than this when released. Since almost all of the CPU time required during supervised training is spent performing rollouts, and rollouts can easily be performed in parallel, it is hoped that users will be able to pool rollout results and collectively train it to a level stronger than any individual could obtain.

If you can tolerate the rough edges, periodic pre-releases of the source will be made on the alpha.gnu.org FTP server (`ftp://alpha.gnu.org/gnu/gnubg/`).

Anonymous CVS access to the very latest code is available: if you have a CVS client, set `CVSROOT` to `:pserver:anoncvs@subversions.gnu.org:/cvs` and check out module '`gnubg`'. If not, daily snapshots of the CVS repository are made available by FTP (`ftp://alpha.gnu.org/gnu/gnubg/snapshots/`), or you can browse the repository through the WWW gateway (`http://subversions.gnu.org/cgi-bin/cvsweb/gnubg/`). Do not expect this code to be stable; it will generally include both more features and more bugs than the main pre-release version.

You can also compete against recent versions of `gnubg` on FIBS; it plays there under the names '`gnu`', '`mgnutest`', '`mpgnu`' and '`gnu_one_ply`'.

2 How to Play Backgammon

The rules presented in this chapter were written by Tom Keith for the Backgammon Galore web site (`http://www.bkgm.com/`), and are included here with his permission.

2.1 Rules of Backgammon

2.1.1 Setup

Backgammon is a game for two players, played on a board consisting of twenty-four narrow triangles called *points*. The triangles alternate in color and are grouped into four quadrants of six triangles each. The quadrants are referred to as a player's *home board* and *outer board*, and the opponent's home board and outer board. The home and outer boards are separated from each other by a ridge down the center of the board called the *bar*.

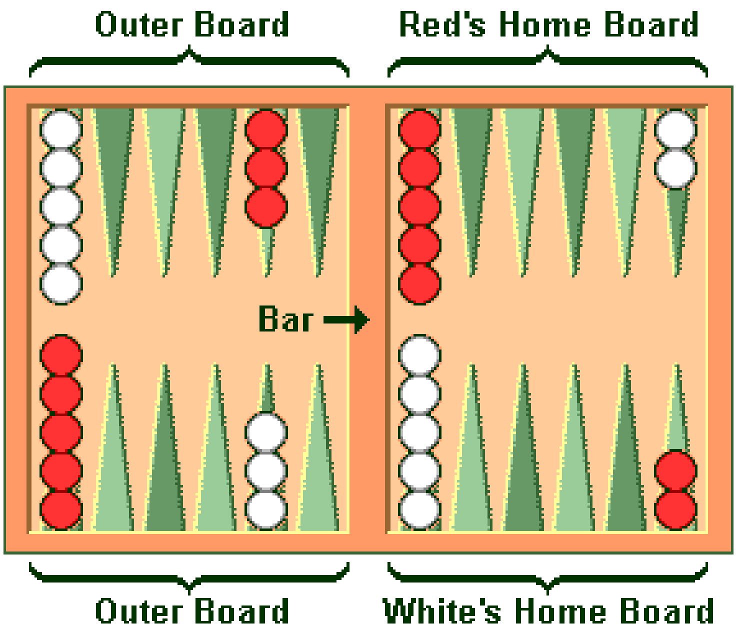


Figure 1. A board with the checkers in their initial position.

An alternate arrangement is the reverse of the one shown here, with the home board on the left and the outer board on the right.

The points are numbered for either player starting in that player's home board. The outermost point is the twenty-four point, which is also the opponent's one point. Each player has fifteen checkers of his own color. The initial arrangement of checkers is: two on each player's twenty-four point, five on each player's thirteen point, three on each player's eight point, and five on each player's six point.

Both players have their own pair of dice and a dice cup used for shaking. A *doubling cube*, with the numerals 2, 4, 8, 16, 32, and 64 on its faces, is used to keep track of the current stake of the game.

2.1.2 Object of the Game

The object of the game is for a player to move all of his checkers into his own home board and then bear them off. The first player to bear off all of his checkers wins the game.

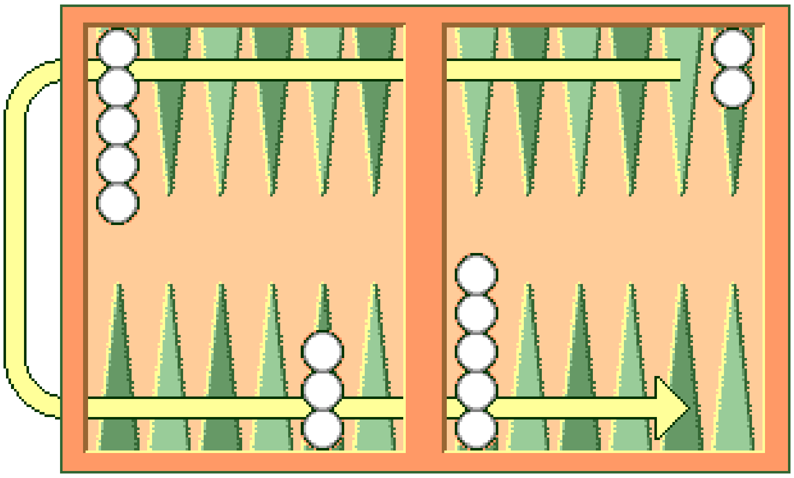


Figure 2. Direction of movement of White's checkers. Red's checkers move in the opposite direction.

2.1.3 Movement of the Checkers

To start the game, each player throws a single die. This determines both the player to go first and the numbers to be played. If equal numbers come up, then both players roll again until they roll different numbers. The player throwing the higher number now moves his checkers according to the numbers showing on both dice. After the first roll, the players throw two dice and alternate turns.

The roll of the dice indicates how many points, or *pips*, the player is to move his checkers. The checkers are always moved forward, to a lower-numbered point. The following rules apply:

1. A checker may be moved only to an *open point*, one that is not occupied by two or more opposing checkers.
2. The numbers on the two dice constitute separate moves. For example, if a player rolls 5 and 3, he may move one checker five spaces to an open point and another checker three spaces to an open point, or he may move the one checker a total of eight spaces to an open point, but only if the intermediate point (either three or five spaces from the starting point) is also open.

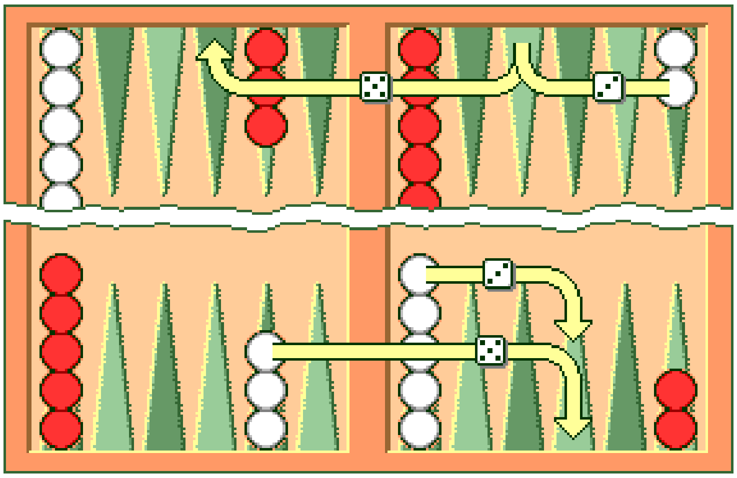


Figure 3. Two ways that White can play a roll of 53.

3. A player who rolls doubles plays the numbers shown on the dice twice. A roll of 6 and

6 means that the player has four sixes to use, and he may move any combination of checkers he feels appropriate to complete this requirement.

4. A player must use both numbers of a roll if this is legally possible (or all four numbers of a double). When only one number can be played, the player must play that number. Or if either number can be played but not both, the player must play the larger one. When neither number can be used, the player loses his turn. In the case of doubles, when all four numbers cannot be played, the player must play as many numbers as he can.

2.1.4 Hitting and Entering

A point occupied by a single checker of either color is called a *blot*. If an opposing checker lands on a blot, the blot is *hit* and placed on the bar.

Any time a player has one or more checkers on the bar, his first obligation is to *enter* those checker(s) into the opposing home board. A checker is entered by moving it to an open point corresponding to one of the numbers on the rolled dice.

For example, if a player rolls 4 and 6, he may enter a checker onto either the opponent's four point or six point, so long as the prospective point is not occupied by two or more of the opponent's checkers.

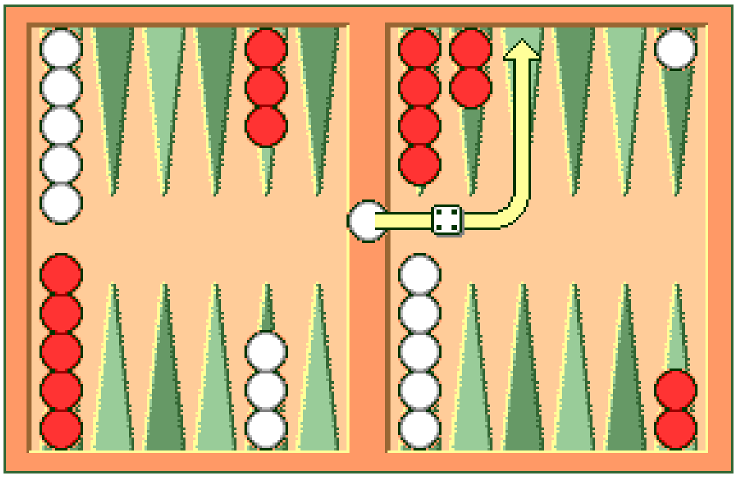


Figure 4. If White rolls 64 with a checker on the bar, he must enter the checker onto Red's four point since Red's six point is not open.

If neither of the points is open, the player loses his turn. If a player is able to enter some but not all of his checkers, he must enter as many as he can and then forfeit the remainder of his turn.

After the last of a player's checkers has been entered, any unused numbers on the dice must be played, by moving either the checker that was entered or a different checker.

2.1.5 Bearing Off

Once a player has moved all of his fifteen checkers into his home board, he may commence *bearing off*. A player bears off a checker by rolling a number that corresponds to the point on which the checker resides, and then removing that checker from the board. Thus, rolling a 6 permits the player to remove a checker from the six point.

If there is no checker on the point indicated by the roll, the player must make a legal move using a checker on a higher-numbered point. If there are no checkers on higher-numbered points, the player is permitted (and required) to remove a checker from the highest point on which one of his checkers resides. A player is under no obligation to bear off if he can make an otherwise legal move.

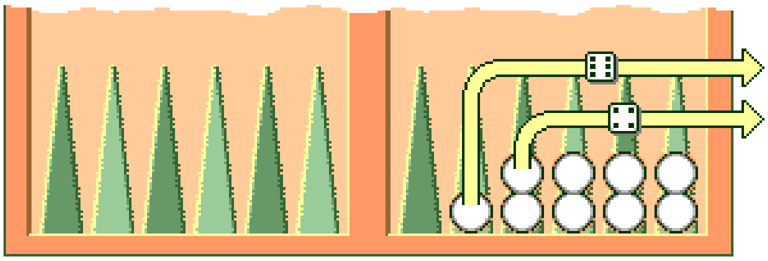


Figure 5. White rolls 64 and bears off two checkers.

A player must have all of his active checkers in his home board in order to bear off. If a checker is hit during the bear-off process, the player must bring that checker back to his home board before continuing to bear off. The first player to bear off all fifteen checkers wins the game.

2.1.6 Doubling

Backgammon is played for an agreed stake per point. Each game starts at one point. During the course of the game, a player who feels he has a sufficient advantage may propose doubling the stakes. He may do this only at the start of his own turn and before he has rolled the dice.

A player who is offered a double may *refuse*, in which case he concedes the game and pays one point. Otherwise, he must *accept* the double and play on for the new higher stakes. A player who accepts a double becomes the *owner of the cube* and only he may make the next double.

Subsequent doubles in the same game are called *redoubles*. If a player refuses a redouble, he must pay the number of points that were at stake prior to the redouble. Otherwise, he becomes the new owner of the cube and the game continues at twice the previous stakes. There is no limit to the number of redoubles in a game.

2.1.7 Gammons and Backgammons

At the end of the game, if the losing player has borne off at least one checker, he loses only the value showing on the doubling cube (one point, if there have been no doubles). However, if the loser has *not* borne off any of his checkers, he is *gammoned* and loses *twice* the value of the doubling cube. Or, worse, if the loser has not borne off any of his checkers and still has a checker on the bar or in the winner's home board, he is *backgammoned* and loses *three times* the value of the doubling cube.

2.1.8 Optional Rules

The following optional rules are in widespread use.

Automatic doubles
If identical numbers are thrown on the first roll, the stakes are doubled. The doubling cube is turned to 2 and remains in the middle. Players usually agree to limit the number of automatic doubles to one per game.

Beavers When a player is doubled, he may immediately redouble (beaver) while retaining possession of the cube. The original doubler has the option of accepting or refusing as with a normal double.

The Jacoby Rule
Gammons and backgammons count only as a single game if neither player has offered a double during the course of the game. This rule speeds up play by eliminating situations where a player avoids doubling so he can play on for a gammon.

2.1.9 Irregularities

1. The dice must be rolled together and land flat on the surface of the right-hand section of the board. The player must reroll both dice if a die lands outside the right-hand board, or lands on a checker, or does not land flat.
2. A turn is completed when the player picks up his dice. If the play is incomplete or otherwise illegal, the opponent has the option of accepting the play as made or of requiring the player to make a legal play. A play is deemed to have been accepted as made when the opponent rolls his dice or offers a double to start his own turn.
3. If a player rolls before his opponent has completed his turn by picking up the dice, the player's roll is voided. This rule is generally waived any time a play is forced or when there is no further contact between the opposing forces.

2.2 Rules for Match Play

When backgammon tournaments are held to determine an overall winner, the usual style of competition is *match play*. Competitors are paired off, and each pair plays a series of games to decide which player progresses to the next round of the tournament. This series of games is called a *match*.

Matches are played to a specified number of points. The first player to accumulate the required points wins the match. Points are awarded in the usual manner: one for a single game, two for a gammon, and three for a backgammon. The doubling cube is used, so the winner receives the value of the game multiplied by the final value of the doubling cube.

Matches are normally played using the *Crawford rule*. The Crawford rule states that if one player reaches a score one point short of the match, neither player may offer a double in the immediately following game. This one game without doubling is called the Crawford game. Once the Crawford game has been played, if the match has not yet been decided, the doubling cube is active again.

Match to 5		**White**	**Black**	
Game 1:	White wins 2 points	2	0	Doubling Allowed
Game 2:	Black wins 1 point	2	1	
Game 3:	White wins 2 points	4	1	
Game 4:	Black wins 1 point	4	2	Crawford Game
Game 5:	Black wins 2 points	4	4	Doubling Allowed
Game 6:	White wins 2 points	6	4	

In this example, White and Black are playing a 5-point match. After three games White has 4 points, which is just one point short of what he needs. That triggers the Crawford rule which says there can be no doubling in next game, Game 4.

There is no bonus for winning more than the required number of points in match play. The sole goal is to win the match, and the size of the victory doesn't matter.

Automatic doubles, beavers, and the Jacoby rule are not used in match play.

3 Sample Game

Although GNU Backgammon has many commands and options, only a few of them are used during routine play. This chapter demonstrates some of those most commonly used.

If your gnubg installation is complete, you should be able to start it and see output similar to the following:

```
GNU Backgammon 0.02  Copyright 1999, 2000, 2001 Gary Wong.
GNU Backgammon is free software, covered by the GNU General Public
License, and you are welcome to change it and/or distribute copies
of it under certain conditions.  Type "show copying" to see the
conditions.
There is absolutely no warranty for GNU Backgammon.  Type "show
warranty" for details.
(No game)
```

When you start a game, gnubg will (by default) show which player's move it is; it is displaying '(No game)' at the moment to indicate that it isn't anybody's turn.

GNU Backgammon uses a pseudo-random number generator to produce dice rolls. It will normally initialise the state of the generator to some value based on the current time (and perhaps other system information, depending what features are available on your platform). But for purposes of this example, it will be better to start the generator from a known state, so that the rolls it generates will be repeatable. Enter the following command to change the generator state ("seed"):

```
(No game) set seed 15
Seed set to 15.
(No game)
```

The default settings allow you to play X (the black pieces, if you have a graphical board window) and the computer will play O (red). It will play with no lookahead by default; it is very fast at choosing moves in this mode (instantaneous, on most hardware) though it does not play at its full strength. Let's start a game.

```
(No game) new game
gnubg rolls 1, user rolls 3.
 GNU Backgammon  Position ID: 4HPwATDgc/ABMA
 +13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
 | X           O    |   | O              X |
 | X           O    |   | O              X |
 | X           O    |   | O                |
 | X                |   | O                |
 | X                |   | O                |
v|                  |BAR|                  |     (Cube: 1)
 | O                |   | X                |
 | O                |   | X                |
 | O           X    |   | X                |
 | O           X    |   | X              O |
 | O           X    |   | X              O |     Rolled 13
 +12-11-10--9--8--7-------6--5--4--3--2--1-+     user

(user)
```

This is the output you will see if a graphical board is unavailable or disabled; if your environment supports a graphical board, you will see a better representation of a backgammon board than this. Exactly the same information is available, either way.

You have just won the opening roll (you have a 3, and `gnubg` has a 1) — the dice roll is in the bottom right hand corner for the TTY board, or shown as graphical dice on the right hand side of the board window. The position ID (the '`4HPwATDgc/ABMA`' in this example) is an encoded representation of the current position. It comes in useful when using some of `gnubg`'s advanced commands, but you can safely ignore it for now. Notice that the prompt has changed to '`(user)`', to indicate that it is your turn; on systems where a login name is available, `gnubg` will use that instead.

If you are familiar with backgammon, you will know that the best play for this roll is to make your 5 point. The command to do that is:

```
(user) 8 5 6 5
```

in the terminal window (i.e. move one chequer from your 8 point to your 5 point, and move another from your 6 to your 5 point); or by dragging each chequer to the destination point in the board window. Click mouse button 1 on the dice to indicate you have finished your move (just as you would pick up your dice to end your move if you were playing on a real board).

Once you have done that, `gnubg` will take its turn, rolling 63 and playing 24/15. It is now your move again:

```
 GNU Backgammon  Position ID: 4HPwCSCwZ/ABMA
 +13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
 | X           O    |   | O              X |
 | X           O    |   | O              X |
 | X           O    |   | O                |
 | X                |   | O                |
 | X                |   | O                |
v|                  |BAR|                  |     (Cube: 1)
 | O                |   |                  |
 | O                |   | X                |
 | O                |   | X                |
 | O           X    |   | X  X             |
 | O     O     X    |   | X  X           O |     On roll
 +12-11-10--9--8--7-------6--5--4--3--2--1-+     user
(user)
```

Since it is no longer the opening roll, you have the option of doubling before rolling your dice. To roll without doubling, just enter the command ***roll*** (or ***r*** for short):

```
(gnubg) roll
```

```
 GNU Backgammon  Position ID: 4HPwCSCwZ/ABMA
 +13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
 | X           O    |   | O              X |
 | X           O    |   | O              X |
 | X           O    |   | O                |
 | X                |   | O                |
 | X                |   | O                |
v|                  |BAR|                  |     (Cube: 1)
 | O                |   |                  |
 | O                |   | X                |
 | O                |   | X                |
 | O           X    |   | X  X             |
 | O     O     X    |   | X  X           O |     Rolled 35
 +12-11-10--9--8--7-------6--5--4--3--2--1-+     user
(user)
```

(You can also roll by clicking the dice just underneath the board window.)

Let's play this roll by hitting twice: 13/10*, 6/1*. You can enter this move with *13 10 6 1* — note that you don't need to specify anything special to hit; `gnubg` will automatically move blots that are hit to the bar. You can also make this move by dragging chequers in the board window, of course.

```
(user) 13 10 6 1
 GNU Backgammon  Position ID: 4HPwAWBhZ+IBMA
 +13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
 | X           O    | O | O              X |
 | X           O    | O | O              X |
 | X           O    |   | O                |
 | X                |   | O                |
 |                  |   | O                |
v|                  |BAR|                  |     (Cube: 1)
 | O                |   |                  |
 | O                |   |                  |
 | O                |   | X                |
 | O           X    |   | X  X             |
 | O     X     X    |   | X  X           X |     On roll
 +12-11-10--9--8--7-------6--5--4--3--2--1-+     user
(user)
```

GNU Backgammon has rolled 65 and been unable to move. Let's take another roll:

```
(user) roll
```

```
GNU Backgammon  Position ID: 4HPwAWBhZ+IBMA
+13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
| X           O    | O | O              X |
| X           O    | O | O              X |
| X           O    |   | O                |
| X                |   | O                |
|                  |   | O                |
v|                 |BAR|                  |     (Cube: 1)
| O                |   |                  |
| O                |   |                  |
| O                |   | X                |
| O           X    |   | X  X             |
| O     X     X    |   | X  X           X |     Rolled 25
+12-11-10--9--8--7-------6--5--4--3--2--1-+     user

(user)
```

Play this roll as *13 11 6 1*. `gnubg` will now roll 16 — and still can't move!

```
GNU Backgammon  Position ID: 4HPwAWDDZsoBMA
+13-14-15-16-17-18------19-20-21-22-23-24-+     gnubg
| X           O    | O | O              X |
| X           O    | O | O              X |
| X           O    |   | O                |
|                  |   | O                |
|                  |   | O                |
v|                 |BAR|                  |     (Cube: 1)
| O                |   |                  |
| O                |   |                  |
| O                |   |                  |
| O           X    |   | X  X           X |
| O  X  X     X    |   | X  X           X |     On roll
+12-11-10--9--8--7-------6--5--4--3--2--1-+     user

(user)
```

Our position is now strong. Let's try doubling — this can be done by entering the command *double* at the prompt, or by clicking the doubling cube in the board window. When you are doubled, you should use the command *take* or *drop* to accept or refuse the cube. GNU Backgammon will drop this double:

```
(user) double
gnubg refuses the cube and gives up 1 point.
user wins a single game and 1 points.
The score (after 1 game) is: gnubg 0, user 1 (money session,
without Jacoby rule).
```

That concludes the example game. You have seen most of the commands normally used while playing; others you will need include *bar* when re-entering a hit chequer (e.g. *bar 20 24 20*) and *off* to remove a chequer from the board when bearing off (e.g. *6 off 5 off*).

Type *help* from within `gnubg` for a summary of all commands, or browse the rest of this manual to learn more. Have fun playing GNU Backgammon!

4 Starting and Leaving gnubg

This chapter describes how to run `gnubg`, and how to end it when you are finished.

4.1 Invoking gnubg

Start GNU Backgammon by running the program *gnubg*. Once started, it will process commands from the terminal.

You can also instruct `gnubg` to read an existing game or match from a file, to play or analyse. To read from *filename*, specify *gnubg filename*.

`gnubg` will give you a short summary of how to invoke it if given the option '`--help`' ('`-h`' for short), and report its version with '`--version`' ('`-v`').

GNU Backgammon will normally use a graphical board window if a window system is available. To prevent this behaviour and use standard terminal output instead, you can specify the '`--tty`' option (which can be abbreviated to '`-t`').

By default, `gnubg` will attempt to load pre-trained neural net weights; if you have no weights or wish to train a net from scratch, you can supply the '`--no-weights`' ('`-n`') option, and new weights will be randomly generated.

FIXME the previous command is out of date; –no-weights is now –new-weights.

FIXME document –no-bearoff, –no-rc, –datadir and –window-system-only options.

4.2 Leaving gnubg

`exit`
`quit`
: To exit `gnubg`, use the *exit* or *quit* commands (they are synonymous). If you are currently playing a game, the match in progress will be lost, so `gnubg` will ask you if you are sure you want to exit in this case.

`gnubg` will also exit upon end-of-file; you can therefore exit by typing an end-of-file character (usually *C-d*, i.e. hold the ⟨CTRL⟩ key and press D).

If you send an interrupt sequence (often by pressing *C-c*), the current `gnubg` command (if any) will be terminated. This can be useful if you do not want to wait for a slow command to complete.

4.3 Shell Commands

You can execute shell commands from `gnubg` without leaving or suspending it.

`![`*command*`]`
: Invoke a subshell to execute *command*. If you do not give a command, a shell will be invoked (the shell is specified by the environment variable `SHELL` if it is set, or defaults to '`/bin/sh`' if not).

5 Basic Commands

There are a few simple commands you can use within GNU Backgammon to display information you might need to know:

`help [`*command*`]`
`? [`*command*`]`
: Describe the commands that `gnubg` understands. `help` by itself lists the main commands with a one-line description of each; when *command* is specified, that command is explained in more detail, and if it has any subcommands then they are listed in brief.

  `?` is a synonym for `help`.

`show copying`
`show warranty`
: These commands are used to show the conditions GNU Backgammon is distributed under (the GNU General Public License). Essentially, there is no warranty for `gnubg`, and everybody is free to use, inspect, copy, modify and distribute it. You are permitted to do just about anything with `gnubg`, except deny these freedoms to others. For a precise description of your rights and obligations, please use the `show copying` and `show warranty` commands, or see the file '`COPYING`' in the `gnubg` distribution (the contents are identical).

6 Playing Backgammon with gnubg

This chapter introduces several more commands used while playing games and matches in `gnubg`.

6.1 Commands Used During the Game

`new game`
: This command is used to start a game within a match or session. (Note that all games are played within matches or sessions; see Section 6.4 [Matches and Sessions], page 19. To play a single game, use either a 1 point match or a session lasting for one game, depending whether you want gammons and the doubling cube to be active.) `new game` will set up the board in the starting position, and roll one die for each player for the opening move.

  If you are in the middle of a game, `new game` will ask you if you want to abort the game in progress. If you do, a new game will replace the current one (i.e. the partially completed game will have no effect on the score). If you want the current game to be scored for either player, you should use the `resign` command instead.

`move` *from to* ...
`move =`*number*
from to ...
`=`*number*
: The `move` command allows you to make chequer plays when it is your turn. In its normal form, you should specify pairs of point numbers indicating the points you want to move a chequer from and to. Specify one pair for each chequer you want to move. (For instance, on an opening roll of 31, you might use `move 8 5 6 5` to move two chequers — one from your 8 point to your 5 point, and the other from your 6 point to your 5 point.) For several example moves, see Chapter 3 [Sample Game], page 8.

  You should use the words `bar` and `off` when moving a chequer from the bar or bearing it off the board, e.g. `move bar 20` or `move 3 off`. These words can be abbreviated to `b` and `o` respectively.

  If there is only one legal move you can make, then the command `move` by itself will make it for you without requiring you to specify it in full. Similarly, if there is no play available at all, then `move` will end your turn without moving.

  As long as you specify at least one pair of points, then the word `move` is optional — the command `bar 20 24 20` means exactly the same thing as `move bar 20 24 20`, for instance.

  FIXME document '=n' notation.

  If you are using a window system, you can also move chequers using the board window. One way to do this is to use any mouse button to drag a chequer (that is, press the button when the pointer is over the chequer you wish to move; move the pointer to the point you wish to play it to, and then release the button).

An alternative is to click a mouse button on the chequer; button 1 will move it by the number of pips showing on the left die, and any other button will move it according to the right die. If you don't like the order the dice are displayed in, pressing either button 2 or 3 on the dice will swap their positions.

Whichever method you use to move the chequers, once you have made a legal move you can end your turn by clicking mouse button 1 on the dice.

FIXME reference set auto move and set auto bearoff.

`resign` [*type*]
`agree`
`accept`
`decline`
`reject`

The `resign` command is used to give up a game without playing it to completion. It is often useful during endgame play when the game reaches a position where it is impossible for one player to win. If you do not specify a value *type*, then the player whose turn it is offers to give up one game (at the current cube value) to the opponent; you can also specify `1`, `2` or `3` to resign a single, double or triple game. Specifying `normal`, `gammon` or `backgammon` is also legal, and is identical to expressing the number of points as a digit. `single` is yet another synonym for one game.

The opponent may accept the resignation with either the `agree` or `accept` commands, but is not obliged to. To ignore the resignation and continue play normally, use either the `decline` or `reject` commands. (`accept` and `reject` are also legal commands in response to a double; see Section 6.5 [Cube], page 20.)

`set board` *id*|=*number*

If you wish to directly set the chequers to a different position, you can use the `set board` command. You need to know the *position ID* of the chequer arrangement you want; position IDs are always displayed when the board is shown. (If you are playing on a text terminal, the position ID is in the upper right hand corner of the board diagram; when using the board window, the ID is shown below the board on the right hand side.) For instance, to set the chequers to the starting position, use the command `set board 4HPwATDgc/ABMA`.

FIXME reference =n notation and describe the GTK edit mode

`show board` [*id*|=*number*]

The `show board` command is used to display a particular position — normally the board of the current game, but it is also possible to view unrelated layouts. When specified without the optional *id* parameter, the current position is displayed. (When using a text terminal, this is useful if subsequent output has caused the board diagram to scroll off the screen. In the board window, the command can be used to undo erroneous chequer plays by resetting the window to the position at the start of the turn.)

When a position ID *id* is given, the chequers are arranged into the position specified and that board is displayed. Note that this command affects the display only; the current game is unchanged. Use the `show board` command with no parameter if you want to see the current game again.

`hint` At any time during the game when the dice have been rolled, you can use the `hint` command to see the moves `gnubg` recommends. The output is of the following form:

```
Win    W(g)   W(bg)  L(g)   L(bg)  Equity        Move
0.542  0.142  0.008  0.113  0.008  (+0.114)      6/5 8/5
0.505  0.120  0.008  0.122  0.007  (+0.009)      24/23 23/20
0.498  0.126  0.008  0.123  0.007  (+0.000)      24/23 13/10
0.499  0.113  0.008  0.121  0.007  (-0.011)      24/23 24/21
0.486  0.125  0.008  0.120  0.009  (-0.024)      13/10 10/9
0.481  0.116  0.008  0.129  0.008  (-0.051)      6/5 24/21
0.472  0.122  0.008  0.129  0.009  (-0.064)      6/5 13/10
```

The moves are listed in descending order of preference, so in this case, `gnubg` recommends the move '`6/5 8/5`'. The first five columns are its estimates of the probability of the player on roll winning ('`Win`'), winning a gammon ('`W(g)`'), winning a backgammon ('`W(bg)`'), losing a gammon ('`L(g)`'), and losing a backgammon ('`L(bg)`') if the game is played to completion without use of the doubling cube, after the candidate move in that row is made[1]. The sixth column, '`Equity`', is the estimated cubeless equity following the move — this is the expected number of points per game won by the player on roll.

FIXME describe =n notation.

`gnubg` will 'look ahead' a certain number of moves when evaluating the probabilities, according to the search depth set by the `set plies` command (see Chapter 8 [Analysis], page 22).

`show pipcount [`*id*`|=`*number*`]`
Use `show pipcount` to automatically count the number of 'pips' each player needs to bear off. Depending on the position, the output will look something like:

```
The pip counts are: X 103, O 112.
```

`show player`

`set player` *name* `human`

`set player` *old-name* `name` *new-name*
Both players have a certain amount of configuration information; use the command `show player` to summarise these settings. By default, `gnubg` will play for player 0, whose name is initially '`gnubg`'. Player 1 defaults to a human (i.e. `gnubg` will prompt the user for a move when it is player 1's turn) whose name is the user's login name, on systems where this information is available; on single-user systems, the default name '`user`' applies.

Either player can be set to a human with the command `set player` *name* `human`, where *name* is either the number of the player (0 or 1) or that player's name (initially '`gnubg`' and '`user`' or the user's login name). You can also specify `both` which will set both players simultaneously. There are also options for computer players (see Section 6.3 [Computer Player], page 19).

[1] Note that the probability of winning includes the probability of winning a gammon or backgammon, and likewise the gammon probabilities include the backgammon probabilities.

You can change the names of the players with the `set player` *old-name* `name` *new-name* command. Again, either the player numbers or names are valid for the *old-name* parameter. Names may not contain whitespace characters, and may not be longer than 31 characters. The names '`0`', '`1`' and '`both`' are not permitted, to avoid ambiguities, and the players may not both share the same name. Names are not case sensitive.

`set turn` *player*

`show turn` The command `set turn` *player* (where *player* can be the player's name or number, as above) is used to control which player is on roll. It will cancel the current dice roll and cube action (if any), and set the named player on roll.

`set automatic bearoff`

`set automatic move`

FIXME

`set display` *value*

By default, `gnubg` will update the display before and after every move, whether made by a human or automatically. (When using the graphical board, the board window is redrawn to the current position; when using the ASCII board, a new diagram is displayed on the terminal.) Information about cube actions and resignations offered will also appear as appropriate.

The command `set display off` will suppress this output, which can speed up the display and reduce clutter (this might be useful when completing a game where both sides are played by the computer, for instance). `set display on` will restore the default behaviour. The standard toggle synonyms may be substituted for `on` and `off`.

The board will always be updated when it is a human player's turn to move, regardless of the `display` setting.

6.2 Controlling Dice Rolls

`roll` This is the basic command used to roll the dice before each turn. It will use the current random number generator to produce a dice roll, and then allow the player to move (if there is at least one legal play for the dice rolled). By rolling the dice, the player gives up any opportunity to double this turn.

`set automatic roll` *value*

When the cube is not being used, or is owned by the opponent, or at certain scores in a match, it is either illegal or pointless for a player to double. Under these circumstances, the player might as well roll the dice immediately — the `set automatic roll` command instructs `gnubg` to go ahead and roll the dice without waiting for the player to issue `roll`, whenever no doubling decision is necessary. The standard toggle commands may be used to turn this option on and off.

`set dice` *pips pips*

To set the dice to a particular roll (whether they have already been rolled or not), use the `set dice` command. The player can then play a move according to the dice specified. Like the `roll` command, this will also forego any opportunity to double; to disregard a dice roll and allow the player to roll again (or double, if permitted), use `set turn` *player*.

`set rng` *generator* `[`*seed*`]`

`show rng` gnubg allows a variety of methods for generating dice rolls, including several built-in pseudo-random number generators and a facility for external libraries to be used. It is also possible to enter a roll manually whenever required.

The `set rng` command is used to select which generator will be used. The *generator* parameter must be one of the following:

`ansi` The ANSI C standard `rand()` random number generator. The behaviour of this generator will depend on the C library linked with `gnubg`, but is typically a linear congruential pseudo-random number generator. Such generators have fairly weak distribution properties, but are generally adequate for producing backgammon dice. However, the ANSI generator is not recommended for performing rollouts, because any small biases in the dice could accumulate over hundreds or thousands of trials and distort the results. Using a better generator would be safer for rollouts.

`bsd` The 4.3BSD `random()` non-linear additive feedback random number generator. This is a good quality generator, but is not available on all systems. `gnubg` will report an error if you attempt to use this generator if the C library used in your installation does not include the BSD `random` code.

`isaac` Bob Jenkins' ISAAC random number generator. This is believed to be a high quality generator.

`manual` By setting the generator to `manual`, `gnubg` will not generate the dice itself; rather, it will prompt for a roll to be entered whenever one is required.

`md5` MD5 is a public domain message digest algorithm invented by Ron Rivest and documented in RFC 1321. `gnubg` can use MD5 as a procedure for generating dice rolls — this generator has the property that the seed increments by one each roll, and so the sequence generated by seed '`n+1`' will be identical to that generated by seed '`n`', with the first roll omitted. The command `show seed` is available when using the *md5* generator, for obtaining the current seed value.

`mersenne` Choosing this option will enable the Mersenne Twister generator designed by Matsumoto and Nishimura. This should be an excellent pseudo-random number generator.

`user` Use the `user` generator to dynamically load a user library which will be used to produce the dice rolls. See the file '`userrng.c`' in the `gnubg` distribution for an example user generator, and instructions on how to write your own.

FIXME explain optional seed parameter

`set seed` *seed*
: You can change the random number generator seed at any time with the `set seed` command. FIXME

`show seed` FIXME

6.3 Having `gnubg` Make Moves

`set player` *name* `gnubg`
: This command will instruct `gnubg` to choose moves (and make cube decisions) for the specified player, using its evaluation engine.

`set player` *name* `pubeval`
: An alternative computer player can be chosen by setting a player to `pubeval`. This will cause moves for that player to be made by Gerry Tesauro's benchmark player. `pubeval` is much weaker than `gnubg`'s own evaluator, but provides a constant level of play which is useful for measuring different players against.

  `pubeval` is not capable of making cube or resignation decisions based on the position. It will never accept resignations at less than triple (backgammon) stakes, and will take all cubes. It nevers offers resignations or doubles of its own.

`set player` *name* `external`
: FIXME

`set player` *name* `evaluation`
: FIXME

`play` FIXME

`external` FIXME

6.4 Matches and Sessions

This section describes how to use GNU Backgammon to play series of games, whether those games are part of a match (as in tournament backgammon) or a session of independent games (conventionally called "money" play, regardless of whether any payment is involved).

`new match` *length*
`new session`
`set automatic game`
`set crawford`
`set postcrawford`
`set automatic crawford`
`set jacoby`
`set automatic doubles` *limit*
`set score` *points points*
`show crawford`
`show postcrawford`
`show jacoby`
`show score`

FIXME

6.5 The Doubling Cube

`double`
`redouble`
`beaver`
`take`
`accept`
`drop`
`pass`
`reject`
`set cube centre`
`set cube owner` *player*
`set cube use`
`set cube value` *points*

FIXME

6.6 Saving Games and Matches

`save game` *file*
`save match` *file*
`load game` *file*
`load match` *file*

FIXME

7 The Graphical Interface to gnubg

`set colours`
`set delay`
`show delay`
FIXME

8 Using gnubg to Analyse Positions

`eval [`*id*`|=`*number*`]`
`rollout [`*id*`|=`*number* `...]`
`set evaluation`
`set rollout`
`set cache` *size*
`show cache`
`show gammonprice`
`show marketwindow`
`show matchequitytable`
`analysis` FIXME

9 Position Databases

`database dump`
`database export`
`export database`
`database generate`
`database import`
`import database`
`database rollout`

FIXME

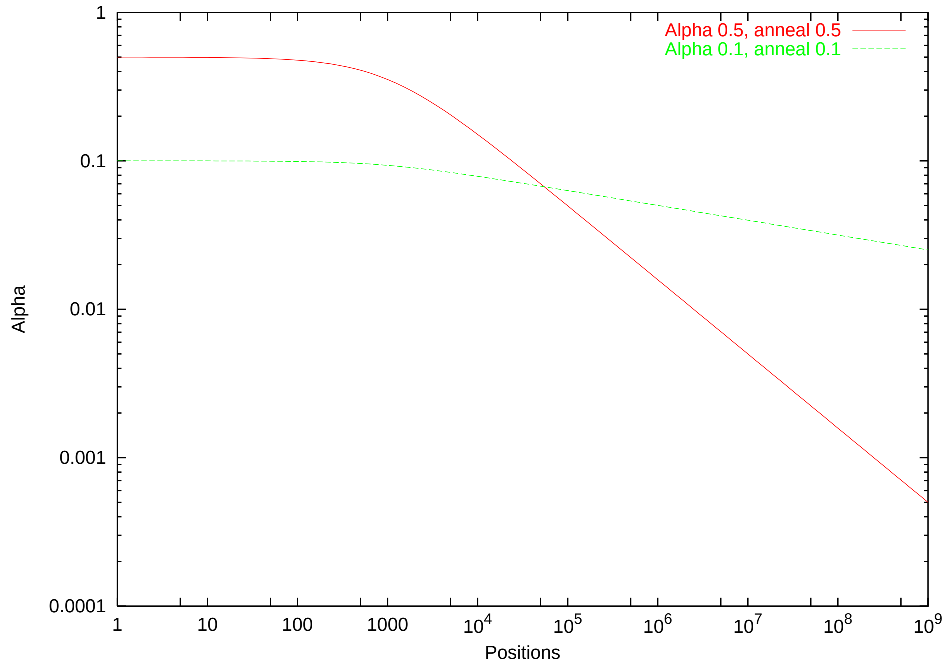
`database train`
`train database`

FIXME pointer only

10 Modifying gnubg's Neural Nets

`new weights`

`save weights` *file*
`load weights` *file*
`database train`
`train database`
`train td` FIXME



11 Extending gnubg

: FIXME

12 Frequently Asked Questions

12.1 Where can I get GNU Backgammon?

Pre-release snapshots of GNU Backgammon are periodically made available for FTP at `ftp://alpha.gnu.org/gnu/gnubg/`.

If you want to experiment with the very latest code, the development sources are kept in a CVS repository at `http://subversions.gnu.org/cgi-bin/cvsweb/gnubg/`; see the `gnubg` web page at `http://www.gnu.org/software/gnubg/gnubg.html` for instructions on checking out sources from the repository. Daily snapshots of the main branch of the repository are automatically placed in the '`snapshots`' directory of the FTP area above, if you wish to retrieve this experimental code via FTP.

12.2 How do I play a game?

Once you are running `gnubg`, enter *`new game`* at the '`(gnubg)`' prompt to start a game against the computer opponent. You should now see a board (if a window system is available, `gnubg` will use a graphical board window; otherwise, it will display an ASCII board on your terminal). If `gnubg` won the opening roll, it will have moved; you can now type *`roll`* (or click on the dice below the board window) to roll the dice yourself. In either case, it will now be your move; you should enter the moves for each chequer as pairs of numbers. For instance, if you have rolled 3 and 1, you could type *`8 5 6 5`* to move one chequer from the 8 point to the 5 point and another from the 6 point to the 5 point. Use *`bar`* to move from the bar, and *`off`* to bear off. If you are using the board window, you can also drag chequers around the board. Click on the dice when you have finished.

12.3 I only see '`gnubg.bd: No such file or directory`'. What's wrong?

The file '`gnubg.bd`' is the bearoff database that `gnubg` uses to evaluate endgame positions. `gnubg` will look for it first in the current directory and then in the installed directory ('`/usr/local/share/gnubg/`' by default).

The `gnubg` distribution is set up to create '`gnubg.bd`' by itself during compilation, but it can be a slow process (taking half an hour or more, depending on the speed of your computer). If you would rather not wait to generate the database yourself, you can obtain a copy via FTP from `ftp://alpha.gnu.org/gnu/gnubg/gnubg.bd.gz`.

FIXME this is out of date; gnubg can cope without gnubg.bd if it has to.

12.4 Now I get '`gnubg.weights: No such file or directory`'. What's that?

The file '`gnubg.weights`' contains the trained neural net weights for most of `gnubg`'s position evaluators. Like '`gnubg.bd`', it should be kept in either the current directory or the installed directory.

A binary variant of the weights file named '`gnubg.wd`' can also be used (this version can be loaded more quickly, but is not portable between different computer architectures). It will automatically be built during the compilation process, and used in preference to '`gnubg.weights`' if found.

You should be able to obtain a copy of the weights from wherever you found the `gnubg` distribution. If you wish to start `gnubg` without any weights (and train your own), you can use the '`--no-weights`' option (see see Section 4.1 [Invoking gnubg], page 12).

12.5 The board window is shown in black and white and looks awful! What's wrong?

If you see this problem, your X server is probably using a *PseudoColor* visual, and is dynamically allocating colours to clients from a limited colourmap. On these types of displays, `gnubg` tries to be a well-behaved client by using colours from the default standard colourmap. This will allow it to share colours with other clients that use the same scheme, which will help them all to use as many colours as possible without exhausting the colourmap.

Unfortunately, not all X clients use standard colourmaps. If other clients have been run before `gnubg` and allocated most of the colours, then there may not be enough left for `gnubg` to allocate a standard colourmap. If this happens, it will just take what it can get (which in the worst case might be black and white only). You can try to avoid this problem by running `gnubg` before other colour-hungry clients, or by using the '`xstdcmap`' utility to install the default standard colourmap early in your session while colours are still available. Depending on your hardware, you may be able to configure your display to allow more colours, or use a *DirectColor* or *TrueColor* visual which should resolve the problem.

Of course, if your X server can only display black and white (not even shades of grey), then there's not much you can do!

GNU Free Documentation License

Preamble

The purpose of this License is to make a manual, textbook, or other written document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you".

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (For example, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License.

The “Cover Texts” are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, whose contents can be viewed and edited directly and straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup has been designed to thwart or discourage subsequent modification by readers is not Transparent. A copy that is not “Transparent" is called "Opaque”.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML designed for human modification. Opaque formats include PostScript, PDF, proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML produced by some word processors for output purposes only.

The “Title Page” means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, “Title Page” means the text near the most prominent appearance of the work’s title, preceding the beginning of the body of the text.

VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

COPYING IN QUANTITY

If you publish printed copies of the Document numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a publicly-accessible computer-network location containing a complete Transparent copy of the Document, free of added material, which the general network-using public has access to download anonymously at no charge using public-standard network protocols. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has less than five).

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.
I. Preserve the section entitled “History”, and its title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled “History” in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
K. In any section entitled “Acknowledgements" or "Dedications”, preserve the section's title, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
M. Delete any section entitled “Endorsements”. Such a section may not be included in the Modified Version.
N. Do not retitle any existing section as “Endorsements” or to conflict in title with any Invariant Section.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties–for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled “History” in the various original documents, forming one section entitled “History"; likewise combine any sections entitled "Acknowledgements”, and any sections entitled “Dedications”. You must delete all sections entitled “Endorsements.”

COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, does not as a whole count as a Modified Version of the Document, provided no compilation copyright is claimed for the compilation. Such a compilation is called an “aggregate”, and this License does not apply to the other self-contained works thus compiled with the Document, on account of their being thus compiled, if they are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one quarter of the entire aggregate, the Document’s Cover Texts may be placed on covers that surround only the Document within the aggregate. Otherwise they must appear on covers around the whole aggregate.

TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License provided that you also include the original English version of this License. In case of a disagreement between the translation and the original English version of this License, the original English version will prevail.

TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided for under this License. Any other attempt to copy, modify, sublicense or distribute the Document 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.

FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation 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. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright (c) YEAR YOUR NAME.
> Permission is granted to copy, distribute and/or modify this document
> under the terms of the GNU Free Documentation License, Version 1.1
> or any later version published by the Free Software Foundation;
> with the Invariant Sections being LIST THEIR TITLES, with the
> Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.
> A copy of the license is included in the section entitled "GNU
> Free Documentation License".

If you have no Invariant Sections, write "with no Invariant Sections" instead of saying which ones are invariant. If you have no Front-Cover Texts, write "no Front-Cover Texts" instead of "Front-Cover Texts being LIST"; likewise for Back-Cover Texts.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Command Index

T

Index

P

Q

R

S

T

Table of Contents