

Xnee Manual

Xnee Manual

Xnee is a suite of programs with,
recording, replaying and 'distribution'
capabilities for X Window System
version 1.08d 3 October 2003

Henrik Sandklef

Copyright (C) 2002-2003 Henrik Sandklef Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License"

Short Contents

1	Introduction	1
2	Getting started	2
3	General ideas	4
4	Installation	7
5	Uninstallation	9
6	Examples	9
7	Xnee Internals	16
8	Synchronisation	22
9	Xnee Requirements	25
10	FAQ	26
	Appendix A Copying This Manual	28
11	Index	35

1 Introduction

1.1 Short description

Xnee can record and replay user sessions in the X Windows System environment.

1.2 Summary

Xnee is a suite of programs that can record, replay and distribute user actions under the X11 environment. Think of it as a robot that can imitate the job you just did.

Xnee consists of a library and two applications

xnee - command line program

gnee - graphical user interface program

libxnee - library used by xnee and gnee

1.3 Xnee features

Test tool Instead of performing test cases for a GUI (or CLI program) over and over again, the test cases can be automated. Simply record a user session and replay it later.

Performance test tool If you want to simulate lots of simultaneous users in a network (or a local machine) you can use Xnee. Simply record a user session and start multiple instances of Xnee.

Demonstration tool You can use Xnee to demonstrate the features of your program. Simply record a user session and replay it later.

Distribution tool If you want to send over your mouse/keyboard actions to another display you can use the built-in distribution mechanism in Xnee.

Macro recorder/replayer By binding a key and modifier combination (e.g using xkeymouse) to replay a recorded session you will have a Window Manager and application independent macro.

File retyper Xnee can retype the contents of a file. This can be useful during tests or if you want xnee to answer some command session without having to record the session.

1.4 Description

The Xnee applications (gnee and xnee) receives X11 protocol data (e.g. events) from an X server (using libxnee) and print them to a file. Theses events are later read from the file and replayed.

Events directly generated by the user (e.g KeyPress) can be replayed or faked. Requests, replies, errors and events not directly generated by the user (e.g MapNotify) can be recorded as well. By using these data Xnee can replay with synchronisation. This is not only useful but essential.

1.5 Background

In order to verify that a program does the job it's supposed to do, certain tests have to be made. These tests are, IMHO, perhaps the most boring things a programmer can do. To release the programmer from this burden Xnee is made.

Xnee started out as a command line program. During the development phase the main functionality was broken out to a library, called libxnee. The command line program kept the name xnee. The thought behind making the library was to enable the writing of other clients than just the command line. Today there is a GUI program, gnee, that uses the library.

By using xnee your testcase(s) can be recorded and later on replayed. Xnee comes with other features. For more information about these, read the Introduction.

2 Getting started

2.1 Getting started

To get the first feel of Xnee some simple examples are presented.

2.1.1 Simple replay

Start a terminal emulator (e.g xterm) and then start Xnee,

```
xnee --replay --file example1.xnr &
```

....don't forget '&'. The file example1.xnr contains keyboard events recorded during development of this manual. When replayed you'll see what was typed and of course more important you'll get a first glimpse of Xnee and its capabilities. For information on where to find the example files, see below.

2.1.2 Simple recording of Key presses

We move on to a (very) simple recording session. Start a terminal emulator (e.g xterm) and your favorite editor. Move the pointer to one of the terminal windows and start Xnee.

```
xnee --record -o example2.xnr --device-event-range 2-3 \  
--time 5 --loops 20
```

Move the pointer to the editor and get focus (e.g click the window frame). After 5 seconds you can type whatever you want to record (20 press- and release events of the keyboard are recorded). We are done and you have recorded your first session! Leave the desktop as it is and go forward to the next example.

2.1.3 Simple replaying of your recorded file

Start one terminal emulator (e.g xterm). Let Xnee repeat the stuff you did in the example above. Undo all changes in the editor that was made in the previous example. Move the pointer to one of the terminal windows and start Xnee.

```
xnee --replay -f exmaple2.xnr --time 5
```

Move the pointer to the editor and get focus (e.g click the window frame). After 5 seconds you will see your typings in the example above being repeated.

2.1.4 Simple recording of mouse motions

We move on to another simple recording session. Start a terminal emulator (e.g xterm). Move the pointer to the terminal window and start Xnee.

```
xnee --record -o exmaple3.xnr --device-event-range 5-6 \  
--time 5 --loops 20
```

After 5 seconds you can move the pointer around (20 motion events are recorded).

2.1.5 Simple replaying of your recorded file

Let Xnee repeat the stuff you did in the example above.. Move the pointer to the terminal window and start Xnee.

```
xnee --replay -f exmaple3.xnr --time 5
```

After 5 seconds you will see your mouse motions in the example above being replayed.

2.1.6 Simple retyping of a text file

Let Xnee retype (type again) the text in a text file. Move the pointer to the terminal window and create a text file containing the command `ls -l`.

```
echo "ls -l" > ./mytext.txt
```

And after that you start Xnee.

```
xnee --retype-file ./mytext.txt --time 5
```

After 5 seconds you will see Xnee type `ls -l`, which probably will list the files in the current directory.

2.1.7 Example Xnee Session files

The example file above (example1.xnr) is a session file that has been delivered with the sources (although not installed), rpm and with the Xnee Documentation Package. The file(s) can be found:

Distribution	Location
RPM	/usr/lib/xnee/session
Source	./sessions/
Document Package	./sessions/

3 General ideas

This chapter will give information about key concepts in X11 and Xnee. It is vital that you read through this chapter.

3.1 Modes

Xnee has four modes:

- record (default mode)
- replay
- retype
- distribute

The distribution mechanism can be used together with the other three.

3.1.1 Record

When record mode is used Xnee receives a copy of the data sent to and from the X server. The copy is printed to a file. Xnee can record the whole X11 protocol, not just mouse and keyboard events.

3.1.2 Replay

When replay mode is used Xnee reads data from a file or stdin. These data is either sent to the server (if it is a keyboard or a mouse event) or used to synchronise with (if any of the other data).

3.1.3 Retype

Xnee can retype the contents of a text file. This is useful when combining replaying of different recorded session. You can change the text written in for example an editor (e.g emacs) without having to re-record the complete sessions.

3.1.4 Distribution

Xnee can fake mouse and keyboard events on multiple displays. This distribution mechanism can be used when recording, replaying or retyping.

3.2 Ranges

What data to record is specified using ranges. Ranges has a start value and a stop value. The following data can be recorded:

Xnee name	X Protocol Name
core-requests	Request
device-event	Event

delivered-event	Event
error	Error
reply	Reply
ext-requests.ext-major	Extension Request
ext-requests.ext-minor	Extension Request
ext-replies.ext-major	Extension Reply
ext-replies.ext-minor	Extension Reply

When specifying the ranges when using xnee you can either type the integer value of the data or the name of the data. To find out what number belongs to what data name, you can use the `--print-data-name` option. For an explanation of the X protocol data, please read the "X Record Extension Library" or the "Record Extension Protocol Specification".

3.3 First and last motion event

Xnee has the ability to skip recording of successive motion events with no other data in between. This option is intended to reduce the number of data recorded by leaving out unnecessary data. This feature can be invoked with the `--first-last` flag.

3.4 Delay

Sometimes when Xnee starts recording data, the keyrelease (caused by pressing and releasing RETURN to execute the Xnee command line) is recorded. This single keyrelease (with no corresponding keypress) might confuse the X server. With the `--time <secs>` option Xnee can be paused for a number of seconds before recording/replaying/retyping starts.

3.5 Verbose

When enabling verbose mode (`--verbose`) Xnee prints a lot of information about it's state. This option is only intended for runtime debugging.

3.6 Human printouts

Sometimes it's hard to decide what data to use when synchronising. To do this you have to analyse what data is sent from the server when recording. Instead of reading the data number, a string representation of the data is printed out. To enable this option, use the `--human-printouts`.

3.7 Invoking Xnee

The mode(s) Xnee shall use and the ranges to use can be set by either:

- command line options

- project file (only applicable when in record mode)
- session file (only applicable when in replay mode)

3.7.1 Command line syntax

To get information about how to use Xnee's command line options see the man page, info page or use the `--help` option.

3.7.2 Project file

To use a project file use the `--project` option, e.g

```
xnee --project xnee/projects/netscape.xnp
```

3.7.3 Session file

To use a recorded session file use the `--file` option, e.g

```
xnee --file user1_session.xnee
```

3.8 Interrupting Xnee

Interrupting Xnee when recording or replaying can be done as follows

- user specified modifier and key
- limit the number of data to record
- sending a SIGTERM signal (e.g pressing Control-c in a terminal window)

The preferred way to interrupt xnee is to use the modifier+key.

3.8.1 modifier and key

It is possible to specify a modifier (e.g Control button) and a key (e.g 'a') that will stop the Xnee session. When using this option make sure that the modifier/key is not used in any way by the applications you are recording. You can specify a key+modifier to stop, pause and resume xnee. You can also insert a mark in the recorded session file.

3.8.2 limit the number of data to record

By specifying the number of data to record (`--loops`) xnee stops when this number of data is received from the server. When replaying the same amount of data is replayed.

3.8.3 sending a SIGTERM signal

The easiest way to send a signal to a process is by launching Xnee from a terminal window (e.g xterm) and then press Control-c which will send the SIGTERM signal to Xnee. When replaying it can sometimes be hard to move the pointer into the terminal window (e.g if a lot of motion events were recorded that will let you compete with Xnee on where the mouse pointer shall be located. Believe me, you'll end up losing that battle).

3.9 Using Xnee plugins

Xnee supports plugins since version 1.07. For information about how to write plugins, download the source code and look at the plugin example which shall be delivered with Xnee.

4 Installation

4.1 Installation from source with the configure script

To unpack, build and install Xnee from the sourcefiles do the following: Download the source files into a directory

Unzip the source file

```
gunzip xnee-1.0.tar.gz
```

Untar the source file

```
tar xvf xnee-1.0.tar
```

Enter the Xnee directory

```
cd xnee-1.0
```

Generate the makefiles

```
./configure
```

or if you want to specify which directory to install xnee to

```
./configure --prefix=<PATH TO INSTALLATION DIR>
```

Build Xnee

```
make clean all
```

Install (as root)

```
make install
```

4.2 Installation from source with default Makefile

To unpack, build and install Xnee from the sourcefiles do the following: Download the source files into a directory

Unzip the source file

```
gunzip xnee-1.0.tar.gz
```

Untar the source file

```
tar xvf xnee-1.0.tar
```

Enter the Xnee directory

```
cd xnee-1.0
```

Build Xnee

```
make -f Makefile.xnee clean all
```

Copy the Xnee binary (xnee/src/xnee) to a directory

```
cp xnee/src/xnee /usr/local/bin
```

4.3 Installation from SOURCE RPM

Use the rpm command, e.g

```
rpm --install xnee-0.9-1.i386.srpm
```

4.4 Installation from RPM

Use the rpm command, e.g

```
rpm --install xnee-0.9-1.i386.rpm
```

4.5 Installation from CVS

Download the xnee source code from the CVS repository at <http://savannah.gnu.org>. Instructions on how to do this can be found there as well.

Generate the configure script

```
make -f Makefile.cvs
```

Generate the Makefiles

```
./configure
```

Compile the binaries

```
make
```

Install the binaries

```
make install (as root)
```

5 Uninstallation

5.1 Uninstallation from SOURCE RPM

Use the rpm command, e.g

```
rpm --erase xnee
```

5.2 Uninstallation from RPM

Use the rpm command, e.g

```
rpm --erase xnee
```

5.3 Uninstallation from source installation

Remove the Xnee binaries

```
rm /usr/local/bin/xnee
rm /usr/local/bin/gnee
rm /usr/local/lib/libxnee*
```

6 Examples

6.1 Recorder

6.1.1 Record mouse motions

Record mouse motions only and save the session to mouse-rec.xnl.

```
xnee --record --mouse --out mouse-rec.xnl
```

After having typed this you can move your mouse round for a while. After Xnee has exited you will be able to replay your motions. Xnee will stop after having record 100 events (this is the default behaviour).

6.1.2 Record keyboard

Record keyboard events only and save log to kbd-rec.xnl.

```
xnee --record --mouse --out kbd-rec.xnl
```

After having typed this Xnee records all your keyboard actions. After Xnee has exited you will be able to replay your keyboard actions. Xnee will stop after having record 100 events (this is the default behaviour).

6.1.3 Record keyboard and mouse

Record keyboard and mouse and save log to km-rec.xml.

```
xnee --record --keyboard --mouse --out kbd-rec.xml
```

After having typed this Xnee records all your keyboard and mouse actions. So now move your pointer and write some stuff with your keyboard. After Xnee has exited you will be able to replay your keyboard and mouse actions. Xnee will stop after having record 100 events (this is the default behaviour).

6.1.4 Record a gnumeric session

Record a galeon session. Record 400 events. Save output in file galeon.xml Start a terminal emulator (e.g xterm)

```
xterm&
```

Start Xnee

```
xnee --record --keyboard --mouse --loops 400 --out gnumeric.xml&
```

Start galeon

```
gnumeric&
```

Start using gnumeric. Browse the menus above, reset the fonts etc.

6.1.5 Record a gnumeric session with synchronisation data

Record a gnumeric session. Record 400 events. Save output in file gnumeric2.xml

Start a terminal emulator (e.g xterm) `xterm&`

Start Xnee

```
xnee --record --keyboard --mouse --loops 400 --out gnumeric2.xml\  
--delivered-event-range Expose,MapRequest,LeaveNotify,EnterNotify &
```

Start galeon `gnumeric&` Start using gnumeric. Browse the menus above, reset the fonts etc.

6.2 Replayer

6.2.1 Replay mouse motions

Replay mouse motions as found in the file mouse-rec.xml.

```
xnee --replay --file mouse-rec.xml
```

Xnee will now imitate exactly what you did when you recorded this file.

6.2.2 Replay mouse motions using with half speed

Replay mouse motions as found in the file mouse-rec.xml but with the speed set to 50% of the recorded.

```
xnee --replay --file mouse-rec.xml --speed-percent 50
```

Xnee will now imitate exactly what you did when you recorded this file, although it will be done in 50% of the recorded time.

6.2.3 Replay mouse motions using with double speed

Replay mouse motions as found in the file `mouse-rec.xml` but with the speed set to 200% of the recorded.

```
xnee --replay --file mouse-rec.xml --speed-percent 200
```

Xnee will now imitate exactly what you did when you recorded this file, although it will be done twice as fast as when recorded.

6.2.4 Replay keyboard actions

Replay keyboard events from file `kbd-rec.xml`.

```
xnee --replay --file kbd-rec.xml
```

After having typed this Xnee replays all your keyboard actions. After Xnee has exited you will be able to replay your keyboard actions.

6.2.5 Replay keyboard and mouse

Replay keyboard and mouse from the file `km-rec.xml`.

```
xnee --replay --keyboard --mouse --file kbd-rec.xml
```

After having typed this Xnee replays all your keyboard and mouse actions. Xnee moves your pointer and writes the the same stuff as you did when recording.

6.2.6 Replay a gnumeric session

Replay the gnumeric session above

Start a terminal emulator (e.g `xterm`) `xterm`& Start a new fresh gnumeric spreadsheet
`gnumeric`&

Start Xnee

```
xnee --replay --file gnumeric.xml
```

Xnee will now do the same stuff you did when recording. It may happen that some user actions are replayed to early. This is so because Xnee has no way of knowing if it is in sync with the recorded session.

6.2.7 Replay a galeon session with synchronisation data

Replay the second gnumeric session above.

Start a terminal emulator (e.g `xterm`) `xterm`& Start a new fresh gnumeric spreadsheet
`gnumeric`&

Start Xnee

```
xnee --replay --file gnumeric2.xml
```

Xnee will now do the same stuff you did when recording. It may happen that the replaying slows down. This is because Xnee is currently out of sync. When being out of sync Xnee slows down a bit and checks the thresholds if it is allowed to continue. Xnee will most probably find itself in sync after a short while. All recorded user actions should have occurred the same way as when recording.

6.2.8 Replay a galeon session with synchronisation data setting threshold

Replay the second gnumeric session above.

Start a terminal emulator (e.g xterm) `xterm&` Start a new fresh gnumeric spreadsheet `gnumeric&`

6.3 Retyper

6.3.1 Retype the help printout

If you want Xnee to fake a user typing the help printout from xnee you can use the `--type-help` option.

Start a terminal emulator (e.g xterm) and an editor (e.g emacs).

```
xterm &
```

```
emacs &
```

Retype the help printout by starting xnee with a 10 seconds delay delay.

```
xnee --time 10 --type-help
```

Move your mouse to the editor and make the editor have focus. Wait a few seconds and xnee will type the help. You will now also have a copy of help text.

6.3.2 Retype a file

If you want Xnee to fake a user typing the letters as found in a text file you can use the retype mode. Note that it isn't possible to retype all characters yet. This will be implemented as soon as possible. We'll give an example on how to use this mode.

Start a terminal emulator (e.g xterm)

```
xterm &
```

Create a text file

```
echo "Hi Xnee" > testfile.txt
```

Retype the contents of this file to another file by starting xnee with a 10 seconds delay delay.

```
xnee --time 10 --retype-file testfile.txt
```

Start the fabulous editor `cat`

```
cat > copiedfile.txt
```

Wait a few seconds and xnee will retype the letters in the file `testfile.txt`. You will now also have a copy of that file. The copy is called `copiedfile.txt`. This is a really a stupid way to copy a file but this option opens up a few possibilities.

6.4 Distributor

With the distribution mode Xnee can send your device events to multiple displays.

6.4.1 Distribute your mouse motions

You can distribute your mouse motions to the displays frodo:0.0 and sam:0.0
Start a terminal emulator (e.g xterm)

```
xterm &
```

Start xnee

```
xnee --distribute frodo:0,sam:0.0  
--record --mouse
```

If you have setup authority correct on frodo and sam you will see all you mouse motions being done on thos displays as well.

6.4.2 Distribute the replaying of mouse motions

Replay and distribute mouse motions as found in the file `mouse-rec.xml`.

```
xnee --replay --file mouse-rec.xml  
--distribute frodo:0,sam:0.0
```

Xnee will now imitate exactly what you did when you recorded this file on your host as well on frodo and sam.

6.4.3 Distribute the retyping of a file

If you want Xnee to to distribute the fakeing of a user typing the letters as found in a text file you can use the retype mode together with the distribution mode.

Start a terminal emulator (e.g xterm) on each of the hosts

```
xterm &
```

Create a text file.

```
echo "Hi again Xnee" > distfile.txt
```

Retype the contents of this file to another file by starting xnee with a 10 seconds delay delay.

```
xnee --time 10 --retype-file distfile.txt  
--distribute frodo:0,sam:0.0
```

Start the fabulous editor `cat` on the terminal emulators on each the terminals.

```
cat > copiedfile.txt
```

If you have setup authority correct on frodo and sam you will, after a few seconds, see xnee retype the letters in the file `distfile.txt`. You will now also have three copies of that file. On copy on each host. The copy is called `copiedfile.txt`. This might seem like a stupid way to copy a file to three locations but this is just an example.

6.5 Key and modifiers

6.5.1 Stop Xnee with key + modifier

You can stop xnee by specifying a key and modifier combination. Make sure that this key modifier isn't grabbed by another X client (e.g by the Window Manager). Let's say that you want Xnee to stop recording if you press Control and h.

```
xnee --record --mouse --loops -1 --stop-key Control,h
```

This will make xnee record mouse events until you press Control and h. All printouts are done to stdout so you can see that Xnee stops when you press the key and modifier.

Move your mouse for a while and you'll see xnee print out lots of lines.

Press Control and h.

Xnee will now have stopped recording.

6.5.2 Pausing and resuming Xnee with key + modifier

You can pause and resum xnee by specifying a key and modifier combination. Make sure that this key modifier isn't grabbed by another X client (e.g by the Window Manager). Let's say that you want Xnee to pause recording if you press Control and p and to resume when pressing Comntrol and r.

```
xnee --record --mouse --loops -1 --pause-key Control,p \
--resume-key Control,r
```

This will make xnee record mouse events until you press Control and p. All printouts are done to stdout so you can see that Xnee stops when you press the key and modifier.

Move your mouse for a while and you'll see xnee print out lots of lines.

Press Control and p.

Xnee will now have paused recording. Move your mouse for a while and note that nothing is printed.

Press Control and r.

Xnee will now have resumed recording. Move your mouse for a while and note that xnee begins its printouts.

6.6 Using macro

Macors can be used in various applications allthough many applicaions have a macro functionality built in (e.g emacs).

6.6.1 Define a simple macro

There are plenty of tools that bind a key + modifier combination to different actions. For various reasons the author of this manual is familliar with xkeymouse so we will use xkeymouse in this example.

The first thing to do is to decide which key + modifier combination to tie to the wanted action. Let's say we want to use one of the funtion keys, F1. We then have to find out which keycode belongs to that key. The action we will bind to this key + modifier combination will be the replaying of a recorded session from the previous examples.

We use Xnee to find the keycode for F1. Start xnee.

```
xnee --record --keyboard --loops 20
```

Press the F1 key and see what number was printed out. It will look something like this:

```
0,2,0,0,0,67,0,90300078
0,3,0,0,0,67,0,90300156
```

The interesting part here is the 6th column. In our example we find 67, which is the keycode for F1.

Now we move on to setup xkeymouse to grab F1 and bind that to replay the mouse motions from the file `mouse-rec.xml`. Open or create a new file in your home directory called `.xkmrc` and add the lines.

```
keycode=67, 0, Exec, xnee, --replay --file mouse-rec.xml, \
Fork, NoAutoRepeat
```

Let's try it. Start xkeymouse with verbose printouts.

```
xkeymouse --verbose
```

Press F1 and the recorded session from the previous example shall be replayed. You can also see in the verbose printouts that xkeymouse executes xnee.

6.6.2 Define another simple macro

Let's say we want to bind Control and e to execute the session as in the example above. This time setting up xkeymouse is a bit easier.

Setup xkeymouse to grab F1 and bind that to replay the mouse motions from the file `mouse-rec.xml` by opening or create a new file in your home directory called `.xkmrc` and add the lines.

```
e, Control, Exec, xnee, --replay --file mouse-rec.xml, \
Fork, NoAutoRepeat
```

Let's try it. Start xkeymouse with verbose printouts.

```
xkeymouse --verbose
```

Press Control and e and the recorded session from the previous example shall be replayed.

6.7 Various options

6.7.1 Using verbose mode

To enable verbose mode, start xnee like this

```
xkeymouse --verbose --record --mouse
```

Move the mouse for a while and you'll lots of verbose printouts that usually isn't there.

6.7.2 Using human readable printouts

To enable human printout mode, start xnee like this

```
xkeymouse --human-printouts --record --mouse
```

Move the mouse for a while and you'll see the data printed out in an almost human friendly format.

6.7.3 Using a different screen resolution

If a session was recorded on a screen with another resolution than on the one where we replay the session xnee will translate all coordinates automatically. However, you can force xnee to use a specific resolution when replaying. To do this, start xnee like this

```
xkeymouse --replay --file mouse-rec.xnl \  
--replay-resolution 800x600
```

Xnee will now replay the events recorded in the sessions file `mouse-rec.xnl` as if the screen has a resolution of 800x600.

6.7.4 Using no resolution translation

If a session was recorded on a screen with another resolution than the one where we replay the session xnee will translate all coordinates automatically. However, you can force xnee not to use translation. To do this, start xnee like this

```
xkeymouse --replay --file mouse-rec.xnl \  
--no-resolution-adjustment
```

Xnee will now replay the events recorded in the sessions file `mouse-rec.xnl` as if the screen had the same resolution the recorded one.

6.7.5 Record another display than the default

If you want to record another display than the default, as set in the `DISPLAY` variable, you use the `--display` option.

```
xkeymouse --record --mouse --display frodo:0.0
```

Xnee will now record the mouse events on the display `frodo:0.0`.

6.7.6 Replay to another display than the default

If you want to replay to another display than the default, as set in the `DISPLAY` variable, you use the `--display` option.

```
xkeymouse --replay --display frodo:0.0 --file mouse-rec.xnl
```

Xnee will now replay the mouse events on the display `frodo:0.0`.

7 Xnee Internals

This chapter is intended to explain the internal design of libxnee. Hopefully this will lead to a better understanding of how to use Xnee and why some features exist and why some don't.

7.1 Project file

Xnee can be set either using command line options (when using xnee) or by clicking the correct buttons etc in the GUI (using gnee). Instead of setting the same settings over and over again, you can use the Xnee Project File.

7.1.1 Create a project file

You can create a project by yourself. This can be done using the `write-settings` option in xnee or the “save settings to file” when using gnee. You can also write one by yourself in your favorite editor. For information about how to do this, read the chapter ‘Project file syntax’.

7.1.2 Project file syntax

`‘display displayname’`
This sets the display to record or send events to when replaying

`‘file <file-name>’`
Use the file <file-name> when replaying

`‘out <file-name>’`
Redirect printouts to <file-name> (stdout is default)

`‘err <file-name>’`
Redirect error printouts to <file-name> (stderr is default)

`‘resource <file-name>’`
Use resource named <resource-name>

`‘plugin <file-name>’`
Use the lib pointed out by file name as plugin.

`‘first-last’`
Sets the first-last option to true

`‘everything’`
Tells Xnee to record everything.

`‘mouse’` Tells Xnee to record mouse events.

`‘keyboard’`
Tells Xnee to record keyboard eventes.

`‘loops <n>’`
How many data to record. -1 means for ever.

`‘k-log’` Record 1000 data.

`‘10k-log’` Record 100 000 data.

`‘100k-log’`
Record 100 000 data.

`‘m-log’` Record 1000 000 data.

`‘verbose’` Turns on verbose mode

`‘buffer-verbose’`
Turns on verbose mode showing replay buffers

`‘time <secs>’`
Pauses Xnee for secs seconds

- ‘stop-key modifier key’
Recording/replaying stops when pressing modifier + key.
See chapter "Key and modifier syntax" for syntax
- ‘pause-key modifier key’
Recording/replaying pauses when pressing modifier + key.
See chapter "Key and modifier syntax" for syntax
- ‘resume-key modifier key’
Recording/replaying resumes when pressing modifier + key.
See chapter "Key and modifier syntax" for syntax
- ‘insert-key modifier key’
A mark is inserted into the session file. See chapter "Key and modifier syntax" for syntax
- ‘future-clients’
TO BE DOCUMNETED
- ‘all-clients’
TO BE DOCUMNETED
- ‘human-printouts’
Print recorded data in a human friendlier format
- ‘record’ Use record mode
- ‘replay’ Use replay mode
- ‘distribute <LIST>’
Sets the list of displays to distribute events to
- ‘device-event-range <X-LIST>’
Sets the list of device events to record to <X-LIST>
- ‘delivered-event-range <X-LIST>’
Sets the list of delivered events to record to <X-LIST>
- ‘error-range <X-LIST>’
Sets the list of errors to record to <X-LIST>
- ‘request-range <X-LIST>’
Sets the list of requests to record to <X-LIST>
- ‘reply-range <X-LIST>’
Sets the list of replies to record to <X-LIST>
- ‘extension-request-major-range <X-LIST>’
Sets the list of extension request major to record to <X-LIST>
- ‘extension-request-minor-range <X-LIST>’
Sets the list of extension request minor to record to <X-LIST>
- ‘extension-reply-major-range <X-LIST>’
Sets the list of extension reply major to record to <X-LIST>
- ‘extension-reply-mainor-range <X-LIST>’
Sets the list of extension reply minor to record to <X-LIST>

LIST is a comma separated list if displays, e.g

```
192.168.1.2:9,10.0.0.2:1
```

RANGE is either a single data name or number or a range between two such (seperated with a "-")

X-LIST is a comma separated list of RANGE, e.g

```
12-15,17,22-29,KeyPress-MotionNotify,ButtonPress
```

7.2 Session file

The Xnee session files are the printouts from a recorded session. The lines in the session file can be either of the following:

‘Replayable data’

‘Synchronisation data’

‘Xnee setup data’

‘Insert marks’

7.2.1 Replayable data

A text representation of the recorded device event. Example: 0,3,0,0,0,28,0,3182009792)■

7.2.2 Synchronisation data

A text representation of the recorded non device event

7.2.3 Xnee setup data

Information about how Xnee was setup when recording this session. Examples: #
device_event: 2-3

7.2.4 Xnee meta data

Information about the evnironment used when recording the seesion Examples: #
Nodename: laphroaig

7.2.5 Insert marks

This is intended to be used in the future when Xnee will be scriptable. Insert marks are inserted using a key+modifier during recording.

7.3 Key and modifier

When wanting to interrupt Xnee from its current action it may sometimes be hard to send a signal. Because of this Xnee has been added support to grab key and modifiers that can be bound to various xnee actions. You can specify a key + modifier combination and bind that to either one of the following actions:

```
'stop'  
'pause'  
'resume'  
'insert mark'
```

7.3.1 Stop Xnee with key+modifier combination

Xnee stops its current action when the user presses the key+modifier combination as specified during setup. Xnee will be shut gracefully.

7.3.2 Pause Xnee with key+modifier combination

Xnee pause its current action when the user presses the key+modifier combination as specified during setup. Xnee will be in paused mode until the user stops or resumes Xnee.

7.3.3 Resume Xnee with key+modifier combination

Xnee resumes its current paused action when the user presses the key+modifier combination as specified during setup. Xnee will continue where it was paused.

7.3.4 Insert marks Xnee with key+modifier combination

When the user presses the key+modifier combination as specified during setup Xnee will print a mark in the session file containing a time stamp. This feature is intended be used when you want to mark an interesting time/event during recording. After recording has finished you can add Xnee scripting calls to Xnee which will be interpreted and executed as if they were recorded.

7.3.5 Stop key and modifier syntax

Setting stop key and modifier is done using the `--stop-key` option.

```
xnee --record --mouse --stop-key Control,e
```

This will make Xnee stop its current action when the user presses Control and e.

7.3.6 Pause key and modifier syntax

Setting pause key and modifier is done using the `--pause-key` option.

```
xnee --record --mouse --pause-key Control,p
```

This will make Xnee pause its current action when the user presses Control and p.

7.3.7 Resume key and modifier syntax

Setting stop resume and modifier is done using the `--resume-key` option.

```
xnee --record --mouse --resume-key Control,r
```

This will make Xnee resume its paused action when the user presses Control and r.

7.3.8 Insert key and modifier syntax

Setting insert key and modifier is done using the `--insert-key` option.

```
xnee --record --mouse --insert-key Control,i
```

This will make Xnee insert a mark in the session file when the user presses Control and i.

7.3.9 Key syntax

You can set the key by entering one character or the keycode for the character you wish to use.

7.3.10 Modifier syntax

Modifiers can be set to a '+' separated list of any of the following.

'Shift or Shift'

Modifier is set to the Shift key

'LockMask or Lock'

Modifier is set to the Lock key

'Control or ctrl'

Modifier is set to the Control key

'Mod1Mask or m1m'

Modifier is set to

'Mod2Mask or m2m'

Modifier is set to

'Mod3Mask or m3m'

Modifier is set to

'Mod4Mask or m4m'

Modifier is set to

'Mod5Mask or m5m'

Modifier is set to

'Alt_L' Modifier is set to Alt_L

'Alt_R' Modifier is set to Alt_R

'Alt' Modifier is set to be any of Alt_L and Alt_R

'Shift_R' Modifier is set to Shift_R

‘Shift_L’ Modifier is set to Shift_L
 ‘Shift’ Modifier is set to be any of Shift_L and Shift_R
 ‘Control_R’
 Modifier is set to Control_R
 ‘Control_L’
 Modifier is set to Control_L
 ‘Control’ Modifier is set to be any of Control_L and Control_R
 ‘Scroll’ Modifier is set Scroll
 ‘Caps_Lock’
 Modifier is set Caps Lock
 ‘AnyModifier or any’
 Modifier is set to any of the modifier
 ‘none or 0’
 No modifier is used

You can also specify the keycode for the modifier you wish to use

7.4 Modifier+key examples

If you want to set the modifier to be a combination of Control, Alt and Shift you separate them with a + sign.

`Control+Alt+Shift`

If you want to make Xnee stop when pressing the above modifier combination and the key y, you type

`--stop-key Control+Alt+Shift,y`

8 Synchronisation

We will try to go through the basics of how Xnee implements synchronisation and try to tell you, by using examples, why synchronisation is important.

8.1 Why synchronise

To understand why synchronisation during replay is needed an example is given.

In this example only mouse and keyboard events are recorded. Think of a session with a web browser.

During record the following is done:

- Start galeon (or another web browser) via the GNOME panel
- Press Ctrl-O which pops up a window
- Press the left button in the textfield of the popup window
- Enter the URL you want to enter (e.g <http://www.gnu.org>)

- Click on the OK button
- Then click on another URL (e.g GNU Documentation)
- Then click on another URL (e.g On-Line Documentation)

When replaying this session it is often to synchronise the recorded session with what's happening "right now" on the display since sometimes (or rather always) there can be different response times from the same URL.

During replay the following is done:

- Galeon is started
- Ctrl-O is typed which pops up a window
- Press the left button in the textfield of the popup window
- Enter the URL you want to enter (e.g `http://www.gnu.org`)
- Click on the OK button
- ... due to an enormous amount of visitors the GNU web server can't respond as quick as it did when recording. So when the next thing happens
- Then click on another URL (e.g GNU Documentation)
- ... the page hasn't been loaded and when the next event is replayed
- Then click on another URL (e.g On-Line Documentation)
- ... the link isn't there and we're really out of sync with the recorded session

8.2 How to synchronise

Instead we could record some more data than just the mouse and keyboard events.

During record the following is done:

- Start galeon (or another web browser) via the GNOME panel
- Record some X data that tells us that a window have been created
- Press Ctrl-O which pops up a window
- Record some X data that tells us that a window have been created
- Press the left button in the textfield of the popup window
- Enter the URL you want to enter (e.g `http://www.gnu.org`)
- Click on the OK button
- Record some X data that tells us that a window have been destroyed
- Then click on another URL (e.g GNU Documentation)
- Record some X data that tells us that a some text has been displayed in a window
- Then click on another URL (e.g On-Line Documentation)
- Record some X data that tells us that a some text has been displayed in a window

The non-mouse-or-keyboard events recorded (window created & text displayed) are record for synchronisation purposes.

During replay the following is done:

- Start galeon (or another web browser) via the GNOME panel
- wait for: the recorded X data to be sent again

- Press Ctrl-O which pops up a window
- wait for: the recorded X data to be sent again
- Press the left button in the textfield of the popup window
- Enter the URL you want to enter (e.g `http://www.gnu.org`)
- Click on the OK button
- wait for: the recorded X data to be sent again
- Then click on another URL (e.g GNU Documentation)
- wait for: the recorded X data to be sent again
- Then click on another URL (e.g On-Line Documentation)
- wait for: the recorded X data to be sent again

8.3 Synchronisation is needed

So by recording more data than just the events to be replayed we can synchronise what was recorded with what is going on when replaying. But the data has to be chosen with respect to that the data:

- differs from different sessions (Gimp and Xterm are really different)
- slows down the replay session if there are too many
- is hard to choose since the X protocol is rich
- differs (comparing record and replaying)
- can have different ordering (comparing record and replaying)

8.3.1 Different data for different kind of sessions

If we record an xterm session with all data being recorded and compare that to a recorded GIMP session with all data being recorded we can see that the data to use as synchronisation data differs. As an example there aren't so many windows created/destroyed during an xterm session.

The solve to the the problem of finding out what data to use as synchronisation data one can:

- use the project files delivered with Xnee
- analyse the application (using Xnee's `--human-prints` option) and do some "trial and error"

8.3.2 Slow replay session due to too many synchronise data

The synchronisation itself doesn't take much time but there are timeouts that makes Xnee pauses for a short while (see above). If there are many such timeouts it will lead to a slow or shaky replaying session.

8.3.3 X protocol is rich

For an end user (with no X expertise) it is hard to read the X protocol specification and make assumptions on what data to use.

8.3.4 Different data sent

Even if one starts up a machine from scratch (reboot) when recording and from scratch when replaying there is no guarantee that the data is sent in the same order or that exactly the same amount of data is sent.

8.4 Buffers and timeouts

To enable synchronisation Xnee buffers data:

- that was read in the session file but hasn't been sent during replay
- that was sent during replay but hasn't been seen in the session file being replayed

For every data read from session file (during replay) that isn't replayable (i.e. device event) Xnee stores the data in a buffer. Xnee also stores the data sent from the X server during playback. The data received from the server make the buffer entry for that specific data be decremented. If, on the other hand, the same data was read from file the buffer entry for that data is incremented. Before replaying any replayable event Xnee makes sure it is in sync. If Xnee is in sync the replaying continues. If Xnee is out of sync it will look for its thresholds and see if it is inside a user specified range. There are three thresholds:

- **positive maximum** nr data read from session file
- **negative minimum** nr of data sent from X server
- **absolute total maximum** sum of the absolute values above

If Xnee read one data from file (e.g. the event MapNotify) Xnee checks if the buffer entry for the specific data is bigger than the positive maximum value (after having incremented the buffer value).

If Xnee receives one data from the X server (e.g. the event MapNotify) it checks if the buffer entry for the specific data is bigger than the negative minimum value (after having decremented the buffer value).

Xnee also checks if the absolute sum of the differences for every entry in the buffer is higher than a total threshold.

If Xnee is getting out of sync it slows down the speed a bit and tries to continue. However after a while it may happen that Xnee considers that it is no use to continue since we are too much out of sync.

Xnee compensates for the delay during replay that is caused when being out of sync.

It is possible to tweak the thresholds using the `--maximum-threshold`, `--negative-threshold` and `--total-diff-threshold` options. It is also possible to turn off synchronisation completely using the `--no-sync` option.

9 Xnee Requirements

9.1 Runtime requirements

Xnee requirements:

- RECORD extension
- XTest extension

You can use Xnee in replaying mode without RECORD extension if synchronisation is turned off.

9.2 Development requirements

For development requirements, please look at the DEVELOPMENT file that is distributed with all packages and with CVS.

10 FAQ

‘Where do I send questions?’
 info-xnee@gnu.org

‘Where and how do I report bugs?’
 Turn on verbose mode `xnee --verbose` and save the printouts. Include all printouts and log files in the mail which is sent to `xnee-bug@gnu.org`

‘When setting ranges (integers), how do I know what numbers belong’
 to X11 data? You can either type the data name directly (e.g MotionNotify)
 or you can use the print-xxx-name options.

- `--print-data-name`
- `--print-event-name`
- `--print-error-name`
- `--print-reply-name`
- `--print-request-name`

‘Can Xnee record more than one display?’
 No, but we are considering adding the functionality (see the TODO file)

‘Where is the CVS repository?’
 You can find a CVS tree at <http://savannah.gnu.org>. You are more than welcome to join

‘Is there a GUI frontend for Xnee ?’
 Yes! Gnee.

‘What license is Xnee released under ?’
 GPL. Which can be found at <http://www.gnu.org/copyleft/gpl.html>

‘Why the name Xnee?’

We wanted to use a recursive acronym, as GNU (“GNU’s Not Unix”). After having read the Wine documentation, we thought that Xnee is not an Event Emulator would work fine since Xnee is Not an Event Emulator but rather a event recorder and faker.

‘What does the name Gnee mean?’

Gnee’s not an Emulator Either

‘What doesn’t the name Gnee mean?’

Gnee’s not an Event Emulator

‘Can you add xxx functionality to Xnee’

Send an email to xnee-devel@gnu.org and we’ll consider it.

‘When starting Xnee I get noticed that I am missing RECORD extension’

Your X server doesn’t have the RECORD extension either loaded or compiled. To solve this you have to, depending on what Xserver you have do the following:

- XFree86 4.0 and higher Make sure that the RECORD extension is loaded. This is done by letting the Module section in the X config file (e.g /etc/X11/XF86Config-4) look something like:

Section "Module"

Load "dbe" # Double-buffering

Load "GLcore" # OpenGL support

Load "dri" # Direct rendering infrastructure

Load "glx" # OpenGL X protocol interface

Load "extmod" # Misc. required extensions

Load "v4l" # Video4Linux

Load "pex5" # PHIGS for X 3D environment (obsolete)

Load "record" # X event recorder

Load "xie" # X Image Extension (obsolete)

You only need the following two modules if you do not use xfs.

Load "freetype" # TrueType font handler

Load "type1" # Adobe Type 1 font handler

EndSection

The important load directive (in this case) is the following line

Load "record" # X event recorder

- X.org, XFree86 (3.3 and lower) or any other Xserver Recompile and reinstall the Xserver and make sure that RECORD extension is built into the Xserver. Please look at the documentation from your Xserver "vendor" on how to do that.

‘How do I build VNC so that I can use Xnee together with it?’

Download vnc source from:

<http://www.uk.research.att.com/vnc/xvnchistory.html>

In the file Xvnc/config/cf/vnc.def change NO on the following items to YES

as below:

```
#define BuildRECORD YES
#define BuildRECORDlib YES
```

Download the Xfree86 distribution from <http://www.xfree86.org>. Specifically, the following dir, (currently in the file X430src-3.tgz file):

```
tar zxvf \
X430src-3.tgz
xc/programs/Xserver/record/
xc/programs/Xserver/record/Imakefile
xc/programs/Xserver/record/record.c
xc/programs/Xserver/record/recordmod.c
xc/programs/Xserver/record/set.c
xc/programs/Xserver/record/set.h
xc/programs/Xserver/record/module/
xc/programs/Xserver/record/module/Imakefile
cp -rf \
xc/programs/Xserver/record \
vnc_unixsrc/Xvnc/programs/Xserver
cd libvncauth/ xmkmf make all
cd Xvnc make World |& tee LOG
```

Verify the build by running `xdpyinfo` in an xterm in the vncserver and verify that RECORD and XTEST extensions are loaded.

Appendix A Copying This Manual

A.1 GNU Free Documentation License

Version 1.2, November 2002

Copyright © 2000,2001,2002 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful 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 non-commercially. Secondly, 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.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

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. (Thus, 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. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

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 Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A “Transparent” copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document 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, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. 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, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include 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, PostScript or PDF 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.

A section “Entitled XYZ” means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as “Acknowledgements”, “Dedications”, “Endorsements”, or “History”.) To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. 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.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) 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 computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. 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.

4. 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 fewer than five), unless they release you from this requirement.
- 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”, Preserve 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. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, 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 to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

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.

5. 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, and that you preserve all their Warranty Disclaimers.

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.”

6. 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.

7. 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, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included an aggregate, this License does not apply to the other works in the aggregate which 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 half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. 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, and all the

license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. 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.

10. 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.

11 Index

B

- background..... 2
- Build VNC to be used with Xnee 27

C

- can Xnee record more than one display 26
- command line syntax..... 6
- Create a project file..... 17

D

- Define a simple macro..... 14
- Define another simple macro..... 15
- description 1
- development requirements 26
- Distribute the replaying of mouse motions 13
- Distribute the retyping of a file 13
- Distribute your mouse motions..... 13
- distributing 4

E

- Example Xnee Session files 3

F

- FDL, GNU Free Documentation License 28
- features 1

G

- General ideas..... 4
- Getting started 2
- GUI frontend for Xnee 26

H

- how to synchronise..... 23

I

- Insert key and modifier syntax..... 21
- Insert marks Xnee with key+modifier combination
..... 20
- Interrupting xnee 6

K

- key 6
- Key and modifier 20
- Key syntax..... 21

L

- license 26

M

- missing RECORD extension 27
- modes 4
- modifier..... 6
- Modifier syntax..... 21
- Modifier+key examples 22

P

- Pause key and modifier syntax..... 20
- Pause Xnee with key+modifier combination 20
- Pausing and resuming Xnee with key + modifier
..... 14
- plugins..... 7
- Project file 16
- Project file syntax 17

Q

- questions is sent where? 26

R

- ranges 4
- Record a gnumeric session 10
- Record a gnumeric session with synchronisation
data..... 10
- Record another display than the default 16
- Record keyboard..... 9
- Record keyboard and mouse 10
- Record mouse motions 9
- recording..... 4
- Replay a galeon session with synchronisation data
..... 11
- Replay a galeon session with synchronisation data
setting threshold 12
- Replay a gnumeric session 11
- Replay keyboard actions 11

Replay keyboard and mouse	11
replay mouse motions	10
Replay to another display than the default.....	16
replay using double speed	11
replay using half speed	10
replaying	4
Report bugs?	26
requirements	25
Resume key and modifier syntax	21
retype	4
Retype a file	12
Retype the help printout	12
runtime requirements	26

S

Session file	19
SIGTERM	6
Stop key and modifier syntax	20
Stop Xnee with key + modifier	14

Stop Xnee with key+modifier combination	20
Synchronisation	22

U

Using a different screen resolution	16
Using human readable printouts	15
Using no resolution translation	16
Using verbose mode	15

W

What does the name Gnee mean?	27
what numbers belong to X11 data?	26
why name it Xnee	27
why synchronise	22

X

Xnee Internals	16
----------------------	----

Table of Contents

1	Introduction	1
1.1	Short description	1
1.2	Summary	1
1.3	Xnee features	1
1.4	Description	1
1.5	Background	2
2	Getting started	2
2.1	Getting started	2
2.1.1	Simple replay	2
2.1.2	Simple recording of Key presses	2
2.1.3	Simple replaying of your recorded file	3
2.1.4	Simple recording of mouse motions	3
2.1.5	Simple replaying of your recorded file	3
2.1.6	Simple retyping of a text file	3
2.1.7	Example Xnee Session files	3
3	General ideas	4
3.1	Modes	4
3.1.1	Record	4
3.1.2	Replay	4
3.1.3	Retype	4
3.1.4	Distribution	4
3.2	Ranges	4
3.3	First and last motion event	5
3.4	Delay	5
3.5	Verbose	5
3.6	Human printouts	5
3.7	Invoking Xnee	5
3.7.1	Command line syntax	6
3.7.2	Project file	6
3.7.3	Session file	6
3.8	Interrupting Xnee	6
3.8.1	modifier and key	6
3.8.2	limit the number of data to record	6
3.8.3	sending a SIGTERM signal	6
3.9	Using Xnee plugins	7

4	Installation	7
4.1	Installation from source with the configure script	7
4.2	Installation from source with default Makefile	7
4.3	Installation from SOURCE RPM	8
4.4	Installation from RPM	8
4.5	Installation from CVS	8
5	Uninstallation	9
5.1	Uninstallation from SOURCE RPM	9
5.2	Uninstallation from RPM	9
5.3	Uninstallation from source installation	9
6	Examples	9
6.1	Recorder	9
6.1.1	Record mouse motions	9
6.1.2	Record keyboard	9
6.1.3	Record keyboard and mouse	10
6.1.4	Record a gnumeric session	10
6.1.5	Record a gnumeric session with synchronisation data	10
6.2	Replayer	10
6.2.1	Replay mouse motions	10
6.2.2	Replay mouse motions using with half speed	10
6.2.3	Replay mouse motions using with double speed	11
6.2.4	Replay keyboard actions	11
6.2.5	Replay keyboard and mouse	11
6.2.6	Replay a gnumeric session	11
6.2.7	Replay a galeon session with synchronisation data	11
6.2.8	Replay a galeon session with synchronisation data setting threshold	12
6.3	Retyper	12
6.3.1	Retype the help printout	12
6.3.2	Retype a file	12
6.4	Distributor	12
6.4.1	Distribute your mouse motions	13
6.4.2	Distribute the replaying of mouse motions	13
6.4.3	Distribute the retyping of a file	13
6.5	Key and modifiers	13
6.5.1	Stop Xnee with key + modifier	14
6.5.2	Pausing and resuming Xnee with key + modifier	14
6.6	Using macro	14
6.6.1	Define a simple macro	14
6.6.2	Define another simple macro	15
6.7	Various options	15
6.7.1	Using verbose mode	15
6.7.2	Using human readable printouts	15

6.7.3	Using a differet screen resolution.....	16
6.7.4	Using no resolution translation	16
6.7.5	Record another display than the default	16
6.7.6	Replay to another display than the default.....	16
7	Xnee Internals.....	16
7.1	Project file	16
7.1.1	Create a project file.....	17
7.1.2	Project file syntax	17
7.2	Session file.....	19
7.2.1	Replayable data	19
7.2.2	Synchronisation data.....	19
7.2.3	Xnee setup data	19
7.2.4	Xnee meta data	19
7.2.5	Insert marks	19
7.3	Key and modifier.....	20
7.3.1	Stop Xnee with key+modifier combination	20
7.3.2	Pause Xnee with key+modifier combination	20
7.3.3	Pause Xnee with key+modifier combination	20
7.3.4	Insert marks Xnee with key+modifier combination	20
7.3.5	Stop key and modifier syntax.....	20
7.3.6	Pause key and modifier syntax.....	20
7.3.7	Resume key and modifier syntax.....	21
7.3.8	Insert key and modifier syntax.....	21
7.3.9	Key syntax.....	21
7.3.10	Modifier syntax	21
7.4	Modifier+key examples.....	22
8	Synchronisation	22
8.1	Why synchronise	22
8.2	How to synchronise.....	23
8.3	Synchronisation is needed.....	24
8.3.1	Different data for different kind of sessions.....	24
8.3.2	Slow replay session due to too many synchronise data	24
8.3.3	X protocol is rich	24
8.3.4	Different data sent.....	25
8.4	Buffers and timeouts	25
9	Xnee Requirements	25
9.1	Runtime requirements	26
9.2	Development requirements	26
10	FAQ	26

Appendix A	Copying This Manual.....	28
A.1	GNU Free Documentation License	28
11	Index	35