Linux Sockets – the select() function

/tmp/tmp1

accept()

connect()

connect()
connect()
connect()
connect()

master_fd

Server

fd **Client1**

fd **Client2**

fd **Client3**

fd **Client4**

fd **Client5**

The server binds its master socket to the file /tmp/tmp1.

The clients connect to the server through the file /tmp/tmp1.

The server accepts the connections from the clients through the file /tmp/tmp1.

The server adds each connection to its active_fd_set using the macro FD_SET.

/tmp/tmp1

accept()

read()/write()

master_fd

cl[0]

fd

Server

Client1

Client2

Client3

Client4

Client5

cl[1]

cl[2]

cl[3]

cl[4]

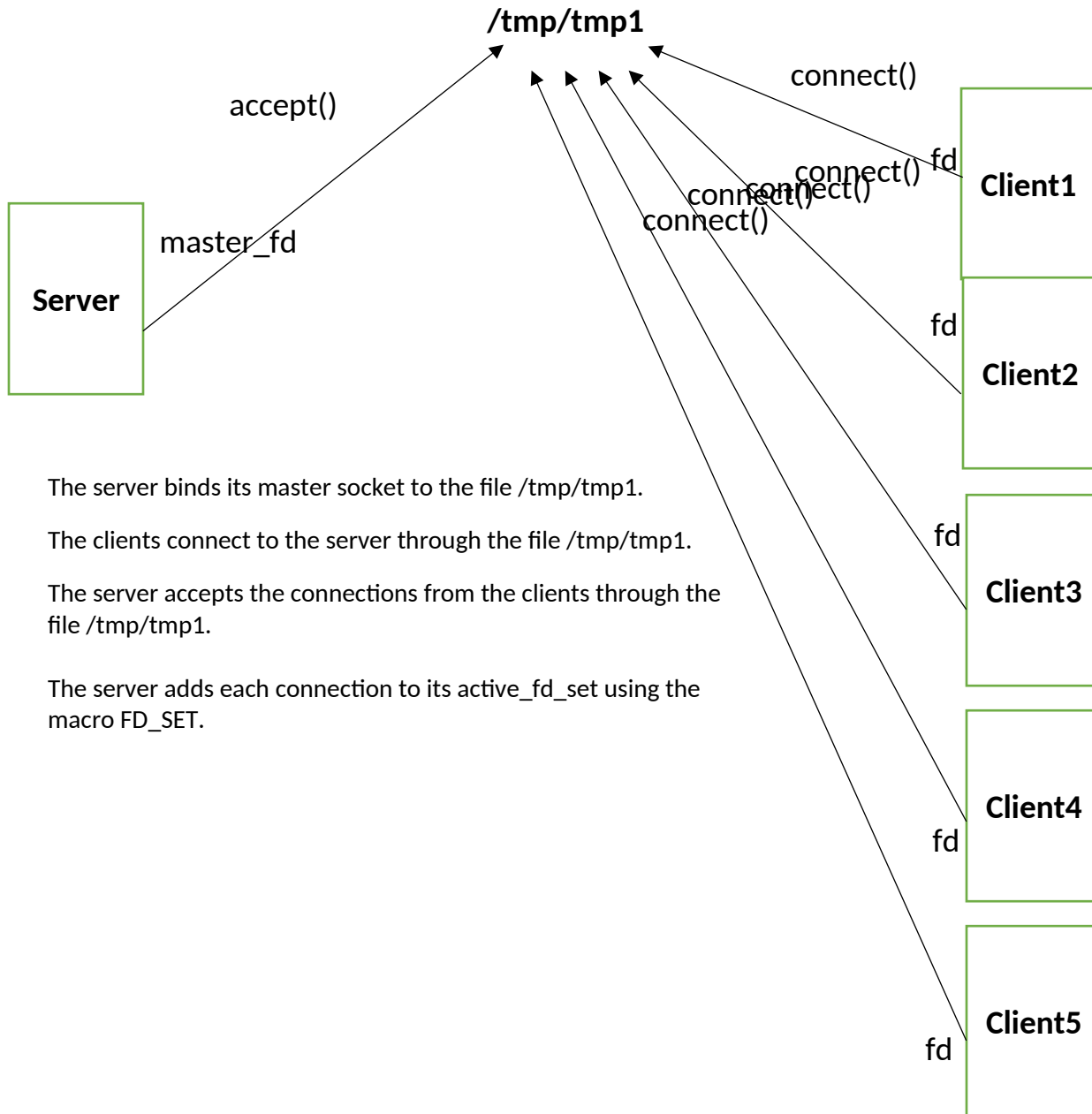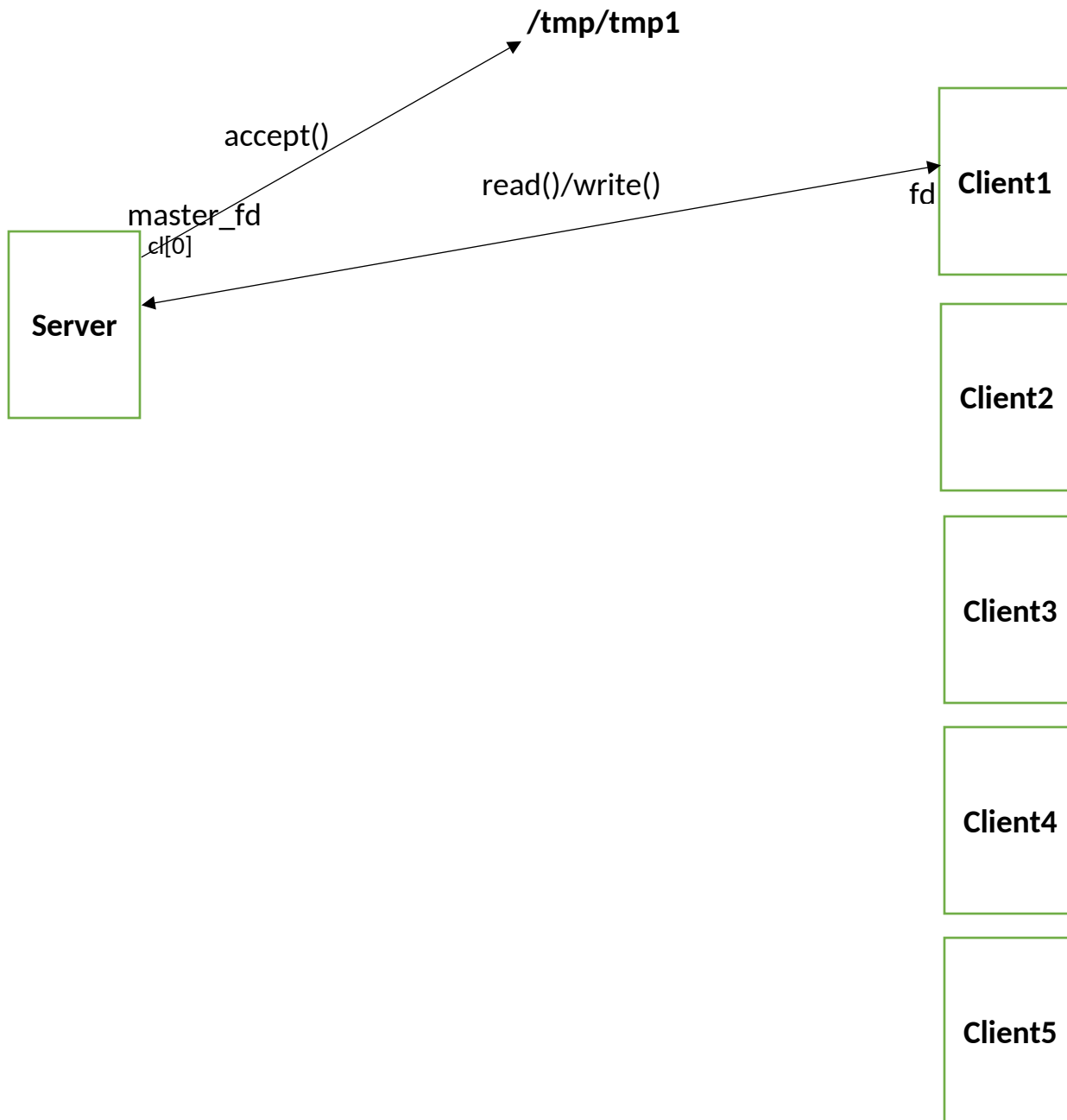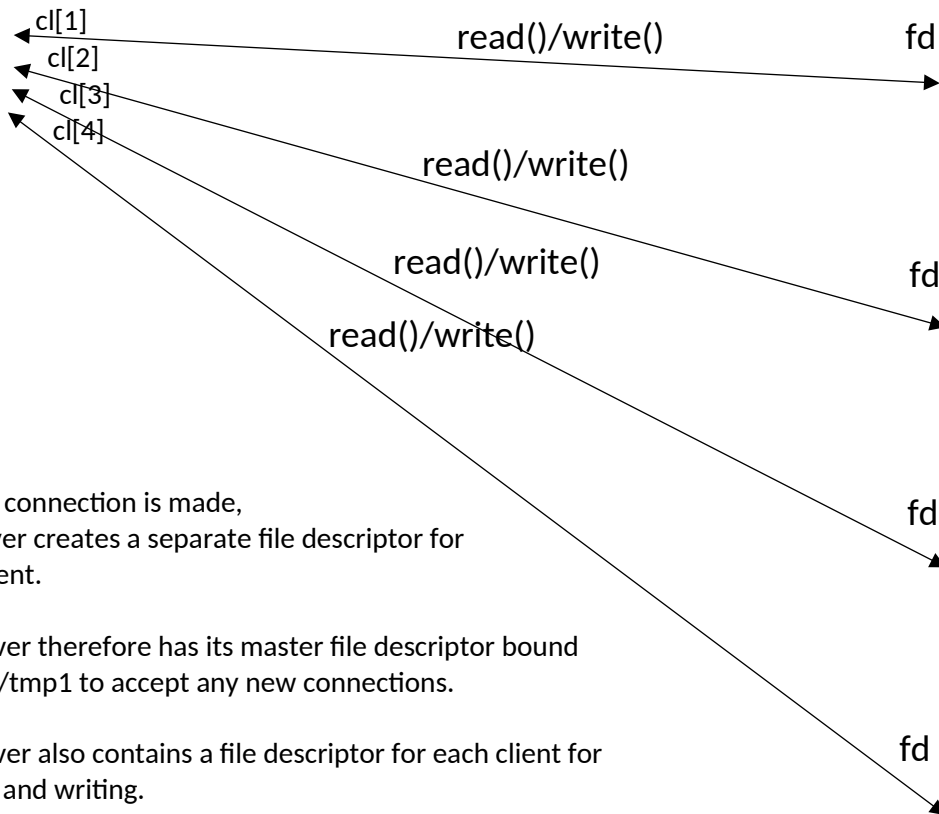read()/write()   fd

read()/write()

read()/write()   fd

read()/write()

fd

fd

As each connection is made,
the server creates a separate file descriptor for
each client.

The server therefore has its master file descriptor bound
to /tmp/tmp1 to accept any new connections.

The server also contains a file descriptor for each client for
reading and writing.

Each client has only one file descriptor, which is used for
reading and writing.

The server uses the select() function to see if a new client is requesting a connect,
or to determine which of the existing clients has sent it a message.

If a client sends the server a message, the server searches through its active_fd_set
using the macro FD_ISSET to find the client's file descriptor.