NETWORK-PROTOCOL

Chats:

1: Global // can be written on Chat-GUI or commandLine

2: Lobby //(chat only receives messages from server, no client to client chat within Lobby)

3: Direct //Whisper Chat , start with @name

Player is uniquely identified (internally) by the upmost 64bit of the SHA(IP:Systime)

All networking commands start with a three letter long signature. Then an eight digit long hexadecimal number follows which encodes the length of the whole command. Finally a message or some parameters follow. A standard networking command looks like this: Txt cx 00000014hello

//no = number, b = bit

MSG-types:

Bel no 64b //"Bell": PingPong

Ckm no 64b // command outside of main game, catches messages about who has won

Fld no 64b // is used to send the chessboard information from the server to all clients.

/*is used to request information. A client can request a player list, a lobby list or a highscore list. (status on commandLine)*/
Get no 64b

Lob no 64 b // broadcast messages to lobbyplayers

Mov no 64b //"Move": is used to send chess moves from the clients to the server.

```
Nam no 64b                                  //"Name": is used to change the player name.

QUI no 64b                                  //someone quits over commandLine or closes main GUI-Panel, user logout

Snd no 64b                                  // is used to send information form the server to all clients.

/* is used to send text messages. These messages will all be displaced in the chat window.*/
Txt no 64b
```

@ = only SRV as receiver //Server

# = only CLT as receiver //Client

¬= no confirmation of reception

type Bel: @

response: Bel no 64b Ans (=echo)

type Cmk: #

message: someone has won

response: chessboard is cleared and all player go back to Lobby

type Fld: #

message: any kind of update for chessboard

response: sends information to chessboard from server to all clients in the specified lobby, updates chessboard

type Get: @

input: “#” lobbies, status, players, highscores

response: displays specified list on Chat and defined GUI

type Lob: @#

message: enter lobby or leave lobby

response: adds/deletes player from list in the specified lobby (@)

updates chessboard (if game has already started) (#), sends message to all players in the same lobby who has left (received by #)

type Mov: @

input: "* [0-17] [0-17] [0-17] [0-17]"

response: make a move or sends error message

type Nam: @#

message: change my name

response: changes name if it is unique (@), sends confirmation or error message (received by #)

type QUI: @#

message: player wants to log out

response: all client-sided threads closed (#), deletes the thread from list (@)

type Snd: @#

message: send message to everyone

response: displays message on chat

type Txt: @#

message: any text that hasn’t been mentioned yet. (incl. broadcast)

input: “Txt cx (TEXT-len) 32b TEXT”

response: displays message on chat