

PROGETTO DI SISTEMI OPERATIVI (client-server multithread game)

Studente: Dimitrios Ieronymakis

Repository:

<https://github.com/OhGreat/client-server-multithread-game.git>

Compilazione:

make

Avviare server:

./so_game_server images/maze.pgm images/maze.ppm

Avviare client:

./so_game_client 127.0.0.1 images/arrow-right.ppm

Descrizione generale server:

Avviato il server, il main process crea le strutture necessarie per comunicare con i client (come socket per i vari thread, strutture per tenere traccia degli utenti, World, semafori etc.).

In secondo luogo vengono creati due thread:

-wup_sender_thread: si occupa di mandare in broadcast il WorldUpdatePacket a tutti i client ogni tot. millisecondi.

-cl_thread: riceve i VehicleUpdate dai client sulla porta CL_UP_RECV_PORT e aggiorna il veicolo sul WorldUpdatePacket che il wup_sender_thread manda ai client.

Infine, il main si mette in ascolto sulla porta SERVER_TCP_PORT e in loop aspetta di ricevere una richiesta di connessione da i client. Quando un client si connette, viene creato il

-client_handler_thread: interagisce con il client scambiandosi tutte le informazioni necessarie per la connessione, (come id, texture e mappa di gioco), poi aspetta le richieste del client delle texture degli altri giocatori e gliela manda.

Descrizione generale client:

Avviato il client, il main process crea le strutture necessarie per comunicare con il server e inizia a richiedere al server tutti i dati necessari per connettersi al gioco (es. Id, mappa, elevation_map etc.). Dopo essersi scambiati tutti i dati necessari, il client crea il mondo, il veicolo e due threads:

-cl_up_thread: manda le ClientUpdate al server su CL_UP_RECV_PORT in modalita' udp.

-wup_receiver_thread: che come suggerisce anche il nome si occupa di ricevere il WorldUpdatePacket dal server, aggiornare i veicoli nel gioco e in caso di mancata texture di un giocatore, la richiede al server tramite la funzione unknown_veh_handler. Il tutto in modalita' udp su CLIENT_WUP_RECEIVER_PORT.

Creati i threads, il client chiama WorldViewer_runGlobal e il gioco comincia.