Clients/Server Communications

I - States

Here is a description of the different states of both server and clients.

a) GS_WAIT

This is the state server and clients take when they are created. Server leaves it and takes the GS_INITGAME state when two clients are connected. Then it sends MSG_INITGAME message to clients. They turn into GS_INITGAME state when they receive it.

Server ignores all messages from clients, except connection messages.

b) GS_INITGAME

Server and clients take this state only once. Clients ask the server to create teams and players with MSG_TEAMINFO and MSG_PLAYERCREATE messages. If server accepts, it creates the instance and sends an acknowledgment to clients. Then, they create the corresponding team or player.

When a client is ready, it sends the MSG_INITGAME message back to the server. When both clients are ready, server turns into GS_DRAWKICKER state.

c) GS_DRAWKICKER

The server enters this state at the beginning of the first half-time (and the third half-time if any). A coach is drawn and gets the choice to either kick or receive the ball. When the client has chosen, it sends a MSG_DRAWKICKER message back to the server, then server turns into GS_INITKICKOFF state.

d) GS_INITKICKOFF

The server enters this state before a kick-off. It removes ball and players from field and sends a MSG_INITKICKOFF message to the defender team.

The client enters this state too; it has to place its players. It sends a MSG_INITKICKOFF message back to the server who checks the placement. If it is valid, it sends the message to the receiving team.

When both teams are correctly placed, server sends a MSG_INITKICKOFF message again, to the kicking team. the client has to place the ball. When it is done, the server resolves the kick-off. If ball goes out of the field or in the defender zone, server turns into GS_TOUCHBACK state and sends a MSG_GIVEBALL message to the receiving team. If the ball finally lands in the attacker zone, game begins: server enters in GS_COACH1 or GS_COACH2 state and sends a MSG_NEWTURN message.

e) GS_TOUCHBACK

The ball has not landed in the correct zone. Client turns into this state when it receives a MSG_GIVEBALL message. Then it sends it back with the id of the player to give the ball to.

If the player can carry the ball (on the field, standing, with hands…), the servers send the message back, as an acknowledgment. Then, servers enters in GS_COACH1 or GS_COACH2 state and sends a MSG_NEWTURN message.

f) GS_COACH1/GS_COACH2

This is the turn of the corresponding coach. Server waits for actions from clients. It checks the client is allowed to realize it and send and MSG_ILLEGAL message or resolve the action. If it is its turn, the client can declare or realize actions; else it can just chat or ask for an illegal procedure.

g) GS_REROLL

The current team can use a team re-roll. When it has chosen, it sends a MSG_REROLL message to the server. Then, server sends the message back as an acknowledgment and resolves the action

h) GS_BLOCK

A client has to choose a block dice to resolve the block. When it has chosen, it sends a MSG_BLOCKDICE or MSG_REROLL message to the server. If the message is valid, it sends the message back and resolves the action.

The other client stays into GS_COACH1 or GS_COACH2 state.

i) GS_PUSH

The client enters this state when it can choose a square to push a player, as a result of a block action. The server enters this state too. When it has chosen, it sends an ACT_BLOCKPUSH message to the server. If the message is valid, it sends the message back and resolves the action.

The other client stays into GS_COACH1 or GS_COACH2 state.

j) GS_FOLLOW

The client can choose if its player follows or not after a block. The server enters this state too. When it has chosen, it sends a MSG_FOLLOW message to the server. If the message is valid, it sends the message back and resolves the action.

The other client stays into GS_COACH1 or GS_COACH2 state.

k) GS_SKILL

A client can choose to use or not a skill. When it has chosen, it sends a MSG_SKILL message to the server. If the message is valid, it sends the message back and resolves the action.

The other client stays into GS_COACH1 or GS_COACH2 state.

II - Mechanisms

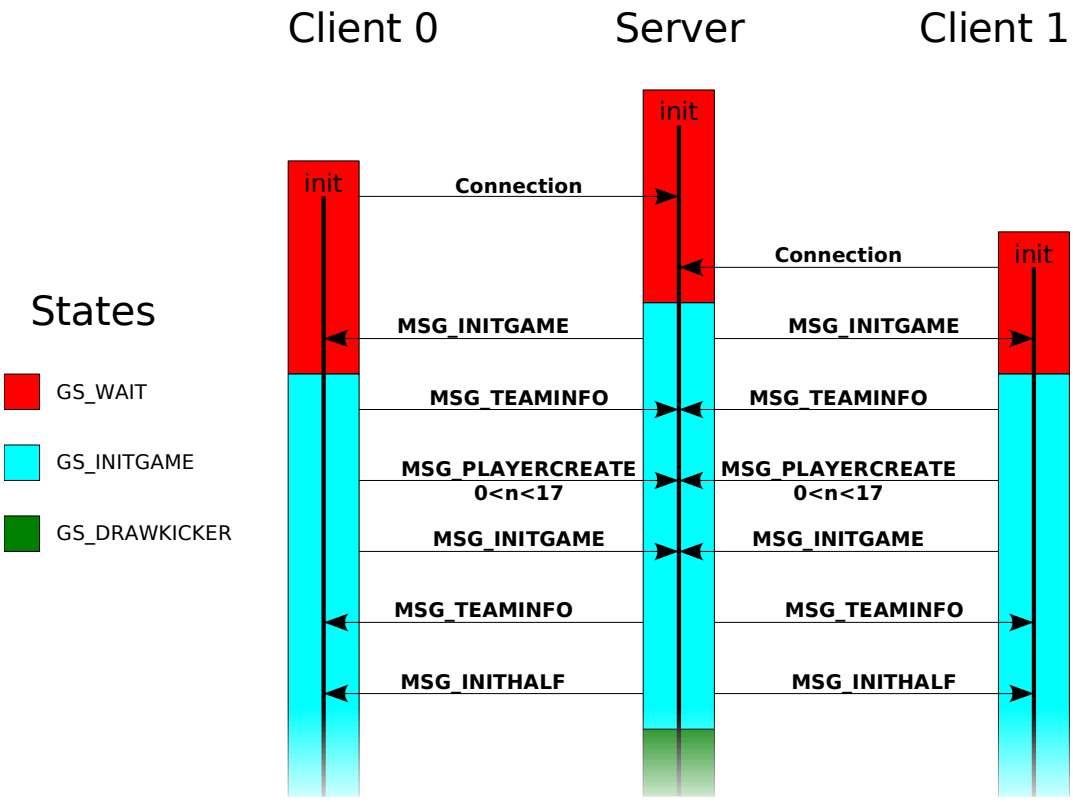
Here are explained some of the main mechanisms of the game.

a) Initialization – global aspect

Initialization of the game uses many messages; it will be explained in different parts. First, server waits for clients. When both are connected, all needed instances are created (ball, players…). Clients and server are in GS_INITGAME state. Then, the server initializes the first half-time.

b) Initialization – teams

When server is in GS_INITGAME state, it waits for the clients to declare their team and players. When it receives an INFO message, it collects what it needs and resends it to both clients. When a client is ready, it sends back the MSG_INITGAME message.



c) Kick-off

When server is in GS_INITKICKOFF state, it waits for the client who kicks the ball to place his players. When he has finished, he sends a MSG_INITKICKOFF message. Server checks the placement is valid, and in this case, it sends a MSG_INITKICKOFF message to the client who receives the ball, he can place his players. Then he sends a MSG_INITKICKOFF message and server checks again the placement.

When both teams are correctly placed, server sends a MSG_INITKICKOFF message to the client who kicks the ball. Hehas to choose which square he aims by sending a MSG_BALLPOS message.

If the placement is valid, the server resolves the kick-off.