Subject: Re: Request of my TT colleagues
Posted by StealthEye on Tue, 17 Nov 2009 20:10:44 GMT
View Forum Message <> Reply to Message

reborn wrote on Tue, 17 November 2009 15:05StealthEye wrote on Tue, 17 November 2009 08:01This requires for client and serverside modifications. It's not going to be separated from TT.

The quick&dirty fix is indeed to make some big object (or an object far down in the ground), because then the coordinates being sent will allow lower (negative) values. If you make the map "high enough" it will fix the problem. Note that you will have to do the map changes on both server and client.

I realise that the first thing you stated was that a client and server fix was needed for this. However, if all that is needed for a dirty fix is to have an object low down on the z-axis, then would an object created on the level loaded event on the server be sufficient?

I don't really understand what causes the problem, let alone the possible solution, just asking... Nope, it needs to be part of the map. Basically, what the game does when loading a map is determine its bounds. It sets the network encoders to these bounds. That allows a few bits or bytes to be saved for position updates.

Consider a level with bounds (-300, 300) (-300, 300) and (-10, 30) in the X Y and Z direction respectively. A position like (150, -300, 10) would be sent though the netcode like (.5, 0, .33), basically giving the positions on the axes relative to their bounds. This works perfectly for positions. However, for targeting, the game computes the target vector relative to the player's position. So if you shoot straight down this could be something like (0, 0, -20) (assuming you're 20m from the ground), not depending on the player's position. This target vector however, is sent though the same encoder, which means it is rounded to the level's bounds: (0, 0, -10). That's not necessarily the direction you wanted to shoot in.

If you understood what I tried to say, then it's clear that it can be solved by either changing the way things are sent (like TT does) or enlarging the map's bounds by placing an object low in the Z axis. The bounds are only computed for map meshes (I think) or at least only for things that are available at map load time when the bounds are determined, so there's no way to make that happen without clientside changes.