Subject: Re: Syncing or changing BuildingGameObj 'IsDetroyed' state for clients
Posted by danpaul88 on Fri, 16 May 2014 15:01:35 GMT
View Forum Message <> Reply to Message

Jerad Gray wrote on Fri, 16 May 2014 15:48<snipped nested quotes>

Last time I ran into it was caused by changing the player character on spawn, but that is one of the rare occurrences where I had complete control and could have found a way around it besides just making a flag external to the script to check.

In the wild, I've seen it caused by things ranging from a separate timer doing clean up, to something a player does (for example once I saw a player die while at a PT (due to lag or a lockup which ever the case was they were still in the PT after their char died) and just as the player's character spawned they bought a character from the PT, the destroy flag on the scripts attached to the spawn character were called before the create due to the character being 'destroyed' and replaced with a new character.

Mind you, this is only a problem if your storing an array index, or a pointer in your script (playerId for efficiency, or a pointer to a list being the most common occurance).

Normally I just protect against it now by attaching a dummy script as the final step of create and then checking for the attached script on any classes that I've found can be called before create (Custom, Damaged, Killed, and Destroyed are the ones that come to mind) on any sensitive scripts (in all of ECW I've done this less than 50 times, and that's a good 4.27MB of code.


It's good practice to initialize any pointers to NULL in the constructor and then check != NULL in the Destroyed method.

The spawn character change should be safe in 4.1 since the lifecycle had some fixed applied at that time which should ensure the correct ordering of events.

---