
Subject: Re: Turret and GT not shooting

Posted by [danpaul88](#) on Thu, 14 Jun 2012 08:06:37 GMT

[View Forum Message](#) <> [Reply to Message](#)

I have found the best way to handle 'stop shooting at stuff you can no longer see' is to record the last time you 'saw' the enemy (via the Enemy_Seen) event and have a timer that fires regularly to check that against the current time. If you have not 'seen' the enemy you are trying to shoot at for more than 2 or 3 seconds you should probably stop shooting at them.

Alternatively if you change;

```
params.AttackCheckBlocked = false;  
to  
params.AttackCheckBlocked = true;
```

I believe you will get an ActionComplete and it will stop firing if it can no longer shoot at the target, but I am not 100% sure what AttackCheckBlocked does so you will have to experiment (this is the sort of thing I would like to document so let me know your results if you do try this)

reborn wrote on Wed, 13 June 2012 22:57

Very nice documentation, DP. As a community, we tend to lack proper documentation, good job!

I am actually working on documenting all of the TT stuff in that fashion but there's an awful lot of it so it's slow going. Jonwil has done a bit too.

Another example;
Toggle Spoiler

File Attachments

1) [ar_miner_ai_docs.jpg](#), downloaded 865 times

int	oreFieldValue
int	oreLoadLevel
int	oreValue
bool	useAI
bool	animating
int	aiState

Detailed Description

Author:

Daniel Paul (danpaul88@yahoo.co.uk)

This class contains the basic logic for the ore miners used in Apocalypse Rising and supports both AI and player controlled miners. It is used in the **dp88_AR_Ore_Field_Zone** script zone with the **dp88_AR_Ore_Field_Zone** script attached to it, which determines the value of each ore 'unit' and the time it takes to mine an ore unit from an ore field whilst ore units are 'collected'.

The miner can hold up to a configurable number of ore units, each of which takes a configurable amount of time to mine from an ore field whilst ore units are 'collected'.

Miners can deposit ore at any time as long as they have at least one unit mined, with the total value of the deposit determined by the **orePerLoadLevel** parameter. Ore deposits are achieved by entering a script zone with the **dp88_AR_Ore_Deposit_Zone** attached to it.

When the AI is enabled the miner will use the pathfind grid to locate the nearest ore field and drive to it to begin mining. It will also locate an ore deposit zone and drive there to unload it's ore. This will be repeated ad-infinitum until the miner is destroyed. When using AI you should place pathfind blockers around the problematic areas and re-generate pathfind data to make it work.

Parameters:

enableAI	Specifies that this miner should use its AI to mine autonomously. 1 to enable AI, 0 to disable AI.
loadLevels	Specifies the total number of ore units this miner can hold at any time
orePerLoadLevel	Amount of ore mined in each unit, this is multiplied by the oreValue parameter in the ore field script to determine the value of an ore unit
timePerLoadLevel	The amount of time, in seconds, it takes to mine one ore unit from an ore field
unloadtime	The amount of time, in seconds, it takes to unload all ore units at an ore deposit zone
aiStartDelay	If the miner AI is enabled this specifies the initial delay before starting the first AI action. This can be overridden by the aiState parameter.
dockAnimation	An optional animation to be played when depositing ore at a refinery
dockSound	An optional sound effect to be played when depositing ore at a refinery
miningAnimation	An optional animation to be played in a loop whilst mining in an ore field
miningSound	An optional sound effect to be played each time the ore load level increases

Note:

Because this script is designed to act as a base class for both types of AR miner it is not named after either. However, because there is no additional logic required for the Soviet War Miner there isn't actually a derived class. LevelEdit is really an instance of this base class script.