

idTarget_Spawn

An entity to spawn other entities. Primarily used for idAI2 entities.

It should be noted that most of these parameters can be excluded unless needed. An **idTarget_Spawn** can be simplified down to this, as an example, if all the player wishes to do is be able to spawn in demons with the teleport **anim/fx** and no other frills.

When copying vanilla spawn targets (rather than from this wiki) you may notice the entities to be spawned listed redundantly in under both "EntityDefs" and "Targets". The EntityDefs list is what you want. "Targets" does not do what you expect; it deletes listed entities on map load. This is not typically an issue because listed entities are attached to the inactive spawn_target_layer and therefore not spawned in the map initially, but in rare cases where a level would be set up otherwise, listing a target may cause issues.

Usage

```
entity {
    entityDef example_target_spawn_eternal {
        inherit = "target/spawn";
        class = "idTarget_Spawn";
        expandInheritance = false;
        poolCount = 0;
        poolGranularity = 2;
        networkReplicated = false;
        disableAIPooling = false;
        edit = {
            flags = {
                noFlood = true;
            }
        }
        spawnConditions = {
            maxCount = 0;
            reuseDelaySec = 0;
            doBoundsTest = false;
            boundsTestType = "BOUNDSTEST_NONE";
            fovCheck = 0;
            minDistance = 0;
            maxDistance = 0;
```

```

    neighborSpawnerDistance = -1;
    LOS_Test = "LOS_NONE";
    playerToTest = "PLAYER_SP";
    conditionProxy = "";
}

spawnEditableShared = {
    groupName = "";
    deathTrigger = "";
    coverRadius = 0;
    maxEnemyCoverDistance = 0;
}

entityDefs = {
    num = 0;
}

conductorEntityAIType = "SPAWN_AI_TYPE_ANY";
initialEntityDefs = {
    num = 0;
}

spawnEditable = {
    spawnAt = "";
    copyTargets = false;
    additionalTargets = {
        num = 0;
    }
    overwriteTraversalFlags = true;
    traversalClassFlags = "CLASS_A";
    combatHintClass = "CLASS_ALL";
    spawnAnim = "";
    aiStateOverride = "AIOVERRIDE_TELEPORT";
    initialTargetOverride = "";
}

portal = "";

targetSpawnParent = "example_spawn_parent";

disablePooling = false;

spawnPosition = {
    x = 209.669998;
    y = -10.990000;
    z = -23.95;
}
}

```

```
}  
}
```

Similarly, a spawn target can be simplified to the following example.

```
entity {  
  [entityDef ai_target_spawn_mod_1 {  
    [inherit = "target/spawn";  
    [class = "idTarget_Spawn";  
    [expandInheritance = false;  
    [poolCount = 0;  
    [poolGranularity = 2;  
    [networkReplicated = false;  
    [disableAIPooling = false;  
    [edit = {  
      [flags = {  
        [noFlood = true;  
      }  
      [entityDefs = {  
        [num = 1;  
        [item[0] = {  
          [name = "example_ai_gargoyle_1";  
        }  
      }  
      [targetSpawnParent = "mod_master_spawn_parent";  
      [spawnEditable = {  
        [overwriteTraversalFlags = true;  
        [traversalClassFlags = "CLASS_A";  
        [combatHintClass = "CLASS_ALL";  
        [aiStateOverride = "AIOVERRIDE_TELEPORT";  
      }  
      [spawnPosition = {  
        [x = 209.669998;  
        [y = -10.990000;  
        [z = -23.95;  
      }  
    }  
  }  
}
```

Sometimes you might want an AI to spawn with an animation rather than through the generic teleportation effect.

This target spawn makes a Mancubus spawn in with its jump animation - like it is jumping from below a ledge.

```
entity {
  EntityDef example_mancubus_traversal_spawn {
    inherit = "target/spawn";
    class = "idTarget_Spawn";
    expandInheritance = false;
    poolCount = 0;
    poolGranularity = 2;
    networkReplicated = false;
    disableAIPooling = false;
    edit = {
      flags = {
        noFlood = true;
      }
      spawnConditions = {
        maxCount = 0;
        reuseDelaySec = 0;
        doBoundsTest = false;
        boundsTestType = "BOUNDSTEST_NONE";
        fovCheck = 0;
        minDistance = 0;
        maxDistance = 0;
        neighborSpawnerDistance = -1;
        LOS_Test = "LOS_NONE";
        playerToTest = "PLAYER_SP";
        conditionProxy = "";
      }
      spawnEditableShared = {
        groupName = "";
        deathTrigger = "";
        coverRadius = 0;
        maxEnemyCoverDistance = 0;
      }
      entityDefs = {
        num = 1;
        item[0] = {
```

```

    name = "example_ai_heavy_mancubus_fire"; // Can only spawn Mancubi
}

conductorEntityAIDType = "SPAWN_AI_TYPE_ANY";

initialEntityDefs = {
    hum = 0;
}

spawnEditable = {
    spawnAt = "";
    copyTargets = false;
    additionalTargets = {
        hum = 0;
    }
    overwriteTraversalFlags = true;
    traversalClassFlags = "CLASS_A";
    combatHintClass = "CLASS_ALL";
    spawnAnim = "animweb/characters/monsters/mancubus_fire/traversal/jump_forward_500_up_500"; //
    Animation to use
    aiStateOverride = "AIOVERRIDE_PLAY_ENTRANCE_ANIMATION"; // Sets to play into animation
    instead of teleport
    initialTargetOverride = "";
}

portal = "";
targetSpawnParent = "";
disablePooling = false;
spawnPosition = { // Be sure to set spawn position to where the AI's start position will be
    x = -69.55;
    y = 395.78;
    z = 102.83;
}

spawnOrientation = { // Be sure to set the spawn rotation
    mat = {
        mat[0] = {
            x = 0.99994820356;
            y = -0.0052357506938;
        }
        mat[1] = {
            x = 0.0052359499969;
            y = 0.99998629093;
        }
    }
}

```



The **spawnAnim** parameter varies in syntax based on what demon AI is being spawned.

Revision #9

Created 8 September 2021 17:34:59 by elizabethany

Updated 31 March 2023 02:57:36 by Alby