On Deep Rock Galactic Spawn Mechanics

By Von

Introduction

When playing Deep Rock Galactic, various kinds of waves and swarms are spawned by the game on timers or when certain events are triggered. The timing and size of these spawning events are mostly hardcoded inside the game's code but they can be somewhat manipulated through AssemblyStorm's mod *Custom Difficulty*, which loads a JSON with parameters used by the game to decide which enemies will spawn and in which numbers. In order to understand existing difficulties or to create new ones, the players must familiarize themselves with terms such as *Rarity*, *Diversity*, *Difficulty*, etc, which at first don't mean anything because they are not used when playing. The purpose of this document is to explain the main mechanics used by the game's wave controllers when spawning enemies and to clarify the terms related to those mechanics.

Some supporting documents:

→ https://github.com/trumank/drg-custom-difficulties/blob/master/DATA.md is a list of enemy descriptor names, the internal names used by the game for each enemy. *Custom Difficulty* requires the internal name be used when making difficulties.

→ Quick And Dirty Guide to Custom Difficulty, recommended reading alongside this document

The basics

A succinct way to explain DRG's spawn mechanics is as follows: the game uses a system of points or *difficulties*[1], where every *event*, let it be Normal Waves, Encounters, Scripted Waves, etc, has a certain assigned budget which determines the size of the event. Every enemy has then a certain cost associated with it, or *DifficultyRating*, and the game spends the points in the budget by picking enemies from an *Enemy Pool* through a basic random sampling procedure and checking how many of those enemies can spawn based on their *DifficultyRating* and other parameters.

The explanation above is really simplified, but it helps having the big picture in mind. The following sections will explain the details.

The Enemy Pool

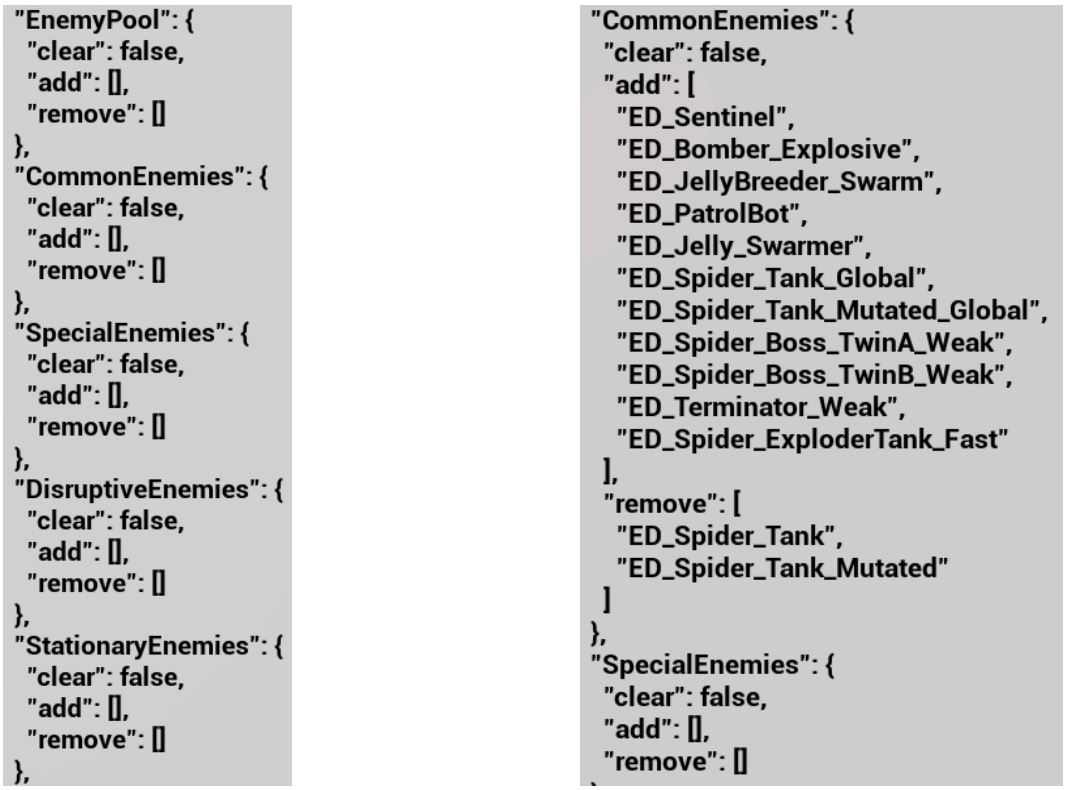
The *Enemy Pool* is the set of enemies that the game will pick from when deciding what to spawn. It uses enemies from three separate pools: the *Common Pool*, the *Disruptor Pool* and the *Special Pool*. The *Enemy Pool* is assembled **before every mission** based on the following procedure:

1. All enemies in the *Common Pool* are added to the *Enemy Pool*.
2. The *DisruptiveEnemyPoolCount* rolls a value between its min and max, and the resulting number of enemies are picked at random from the *Disruptor Pool* and added to the *Enemy Pool*.
3. If the number of enemies in the *Enemy Pool* is now smaller than the *MinPoolSize* parameter, enemies from the *Special Pool* are chosen at random until the number of enemies in the *Enemy Pool* is equal to *MinPoolSize*.

Most difficulties assign a high number to *MinPoolSize* so that all missions have all the enemies described in the difficulty, but this procedure allows to have very different swarm compositions from mission to mission if that is what the difficulty designer

[1] Really unfortunate name.

wishes. As an example, Septic Spreaders are in the *Special Pool* in vanilla's Hazard 5, and the *MinPoolSize* is a low number such that they do not spawn in all missions. Enemies can be added to the *Common Pool*, the *Disruptor Pool* and the *Special Pool* as desired with *Custom Difficulty*:

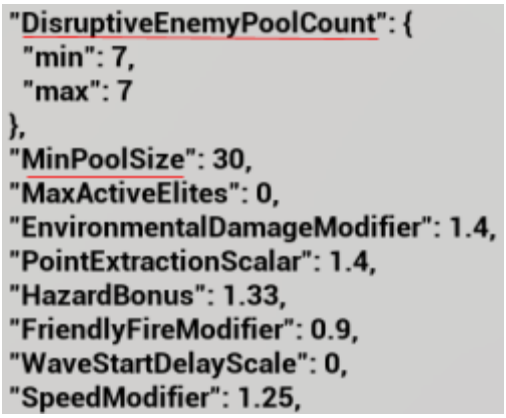


```
"EnemyPool": {
  "clear": false,
  "add": [],
  "remove": []
},
"CommonEnemies": {
  "clear": false,
  "add": [],
  "remove": []
},
"SpecialEnemies": {
  "clear": false,
  "add": [],
  "remove": []
},
"DisruptiveEnemies": {
  "clear": false,
  "add": [],
  "remove": []
},
"StationaryEnemies": {
  "clear": false,
  "add": [],
  "remove": []
},
```

```
"CommonEnemies": {
  "clear": false,
  "add": [
    "ED_Sentinel",
    "ED_Bomber_Explosive",
    "ED_JellyBreeder_Swarm",
    "ED_PatrolBot",
    "ED_Jelly_Swarmer",
    "ED_Spider_Tank_Global",
    "ED_Spider_Tank_Mutated_Global",
    "ED_Spider_Boss_TwinA_Weak",
    "ED_Spider_Boss_TwinB_Weak",
    "ED_Terminator_Weak",
    "ED_Spider_ExploderTank_Fast"
  ],
  "remove": [
    "ED_Spider_Tank",
    "ED_Spider_Tank_Mutated"
  ]
},
"SpecialEnemies": {
  "clear": false,
  "add": [],
  "remove": []
```

Left: the pools described above inside Custom Difficulty. Right: some enemy descriptors added and removed from the Common Pool as an example.

as well as changing the *MinPoolSize* and the *DisruptiveEnemyPoolCount* parameters. Once the game has assembled the *Enemy Pool* with the procedure above, the other pools have no effect during the mission.



The Vanilla Pools

By default Custom Difficulty uses Hazard 5 values for everything, including the pools. The following table shows the vanilla pool distribution for reference:

Pool	**Enemies**
Common Pool	Grunts, Praetorians
Disruptor Pool	Menaces, Wardens, Bulk Detonators, Stingtails, Shellbacks, Grabbers, Goo Bombers, Stalkers
Special Pool	Swarmers, Exploders, Web Spitters, Acid Spitters, Septic Spreaders, Mactera Spawns and Trijaws
Stationary Pool	Breeders, Swarmer Nexii, Spitballers, Cave Leech, Barragers, Vartok Scalebramble
Constant Pressure Pool	Swarmers, Web Spitters, Grunts, Guards, Slashers
Encounter Pool	Oppressors, Swarmers, Web Spitters, Grunts, Guards, Slashers, Exploders, Wardens, Spitballers, Nemesis, Goo

	Bombers, Breeders, Acid Spitter, Menace, Cave Leech, Trijaw, Mactera Spawn, Brundle, Grabber, Prospector, Shellbacks

See the sections below for an explanation on Stationary, Constant Pressure and Encounter Pools.

Spawning Enemies During Events

Spawn Budget

Each event in the game has a points budget *B*, which is made of a base value called the *Difficulty* and a multiplier coefficient called the *EnemyCountModifier*:

$$Budget = Difficulty \cdot EnemyCountModifier$$

The *difficulty* of each event is almost always hard-coded by the game, with the exception of *Encounters*, *NormalWaves* and *Stationaries* (see the *Quick And Dirty Guide to Custom Difficulty* linked in the introduction). The *EnemyCountModifier* is a global parameter that will generally affect all events and it's modifiable through *Custom Difficulty* with a bin based on the number of players:

```
"EnemyCountModifier": [
 2.09,
 2.31,
 2.97,
 3.63
],
```

EnemyCountModifier is the main tool to control wave sizes in *Custom Difficulty* difficulties. As an example, the following table shows the *EnemyCountModifier* for some commonly played hazards:

Number of Players	Hazard 5	Hazard 6x2	Lx2
1	0.85	1.9	2.3
2	0.85	2.1	2.5
3	1.25	2.7	2.9
4	1.5	3.3	3.5

Spawning enemies during events: Diversity, Rarity, SpawnAmountModifier, DifficultyRating, MinSpawnCount, MaxSpawnCount

The budget from the previous section is calculated for every event and is used by the game's wave controller to calculate the number of enemies that will spawn. We talked before about how the game assembles a pool of enemies called the *Enemy Pool* before every mission: enemies are taken from this pool following a certain procedure and their cost, or *DifficultyRating*, is used to compute how many will spawn. However, there are a bunch of other parameters involved that are able to manipulate the number of spawns of a certain type. The following procedure is how the game picks and computes the amount enemies that will spawn during a certain event with budget B:

1. The game rolls a *Diversity* bin and a value between min and max is picked at random, say, N.
2. N enemies will be picked from the *Enemy Pool* using weighted sampling without replacement. The weights are 1/*Rarity* for each enemy in the pool.
3. The amount of enemies to spawn A for each descriptor is then computed as follows:

$$A = \frac{B}{DifficultyRating} \cdot \left(\frac{SpawnAmountModifier}{M} \right)$$

where M = the sum of SpawnAmountModifiers of all enemies in the Enemy Pool (in other words, the game is normalizing all SpawnAmountModifiers to keep the sum equal to 1 and preserve the relative weights), and the DifficultyRating and SpawnAmountModifier are relative to each enemy.

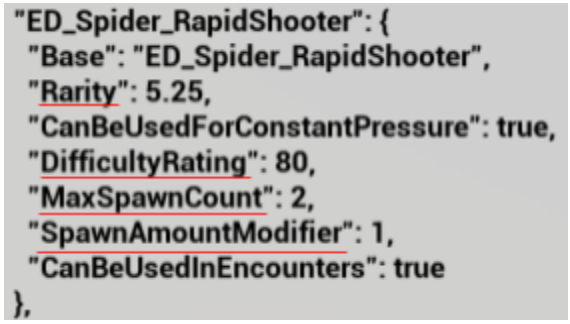
4. If MinSpawnCount · DifficultyRating > B, no enemies will spawn for that descriptor, and the enemy is removed from the pool so the points can be used for all other descriptors.
5. If MinSpawnCount · DifficultyRating < B and MinSpawnCount > A, the game will spawn MinSpawnCount enemies of that descriptor. The total amount of points used, equal to MinSpawnCount · DifficultyRating, is subtracted from the budget B for the calculation of the other spawns.
6. If MaxSpawnCount < A, the game will spawn MaxSpawnCount or sometimes MaxSpawnCount + 1 enemies of that descriptor. The points difference between A and MaxSpawnCount (or MaxSpawnCount + 1) is discarded and doesn't go to any other descriptor.
7. If 4, 5 and 6 are false, the game will spawn A enemies of that descriptor, rounded to the nearest integer.

Some observations:

→ Because of the sampling without replacement, the probability of picking each enemy converges to 1 as *N* approaches the number of enemies in the pool. In particular, if the diversity roll *N* is equal or bigger than the number of enemies in the pool, rarity has no effect and all enemies will be picked for that event.

→ If there is a descriptor with a MinSpawnCount · DifficultyRating very similar to *B*, that descriptor will eat almost all the points if picked first and will not spawn if picked after another descriptor has used some points. It is unknown how the

game decides the order of the spawns once the pool is assembled, but tests seem to indicate a random behavior.

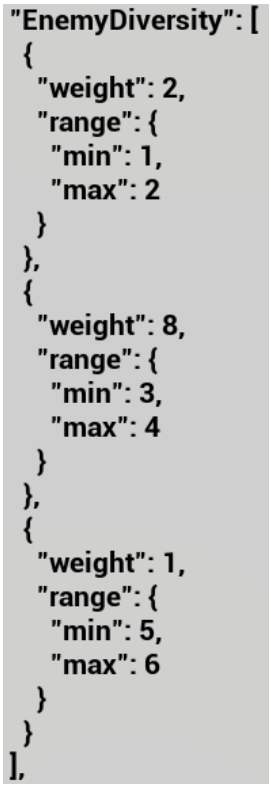


Example of an enemy descriptor inside Custom Difficulty with some of the parameters explained in the text.

Diversity

There's two diversity parameters that can be changed in Custom Difficulty: *EnemyDiversity* and *StationaryEnemyDiversity*. *EnemyDiversity* is the field affecting most of the spawn events in the game: normal waves, defense objective waves, egg waves, etc. Please see the *Quick And Dirty Guide to Custom Difficulty* linked in the introduction for a complete list.

StationaryEnemyDiversity doesn't do anything right now, since it's hard-coded to 4 by the game. There are mods that can increase the value, which is useful to add other stationeries into the pool through other mods.



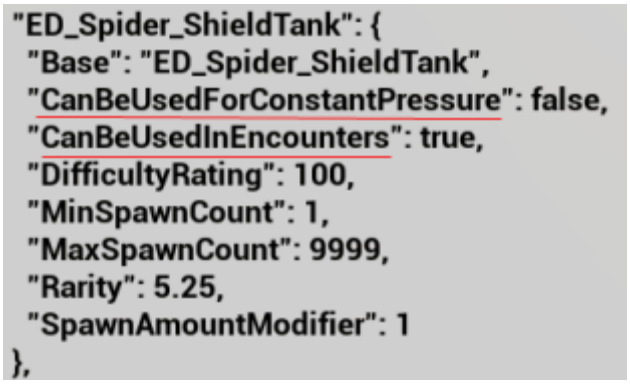
Enemy diversity bins inside Custom Difficulty as an example.

The Constant Pressure Pool

It is possible to assign enemy descriptors to a separate pool called *the Constant Pressure Pool*. Enemies in this pool will spawn during certain events in the game, for example, during announced swarms, during pipeline repair in Refinery missions, in Point Extraction missions, and some others.

The Encounter Pool

Encounters are spawn events that happen on certain missions when reaching new caves. The most familiar example are perhaps mining missions, where a small wave spawns after breaching a new room. It is possible to assign descriptors to a separate *Encounter Pool* so enemies will spawn during these events.



ShieldTank descriptor (Oppressor) stating that it won't be appearing in Constant Pressure events but will appear during Encounters.

Stationary Enemies

Stationaries such as breeders, spitballers and swarmer nexii follow the same spawning logic as described before, and have a separate *StationaryDifficulty* and a *StationaryEnemyDiversity* parameters, and a separate *Stationary Pool* that the game will pick from. The obvious difference is that stationary enemies do not respawn during events, so their numbers are computed at the start of the mission and placed in the map only once. It is possible to add stationary enemy descriptors to the Enemy Pool or to the other pools described before; this will cause them to spawn during game events in the same way as non-stationary enemies.
As commented before, without additional mods, changing the StationaryEnemyDiversity doesn't do anything since it's hard-coded to 4 by the game.

Veteran Enemies

Veteran enemies is how the game calls the relation between certain descriptors:

1. *Glyphid Guards* and *Slashers* are the veterans of the *Glyphid Grunt*.
2. *Mactera Trijaws* and *Brundles* are the veterans of the *Mactera Spawn*.
3. *Glyphid Oppressors* are the veterans of the *Glyphid Praetorian*.

The way veterans appear during events is by promoting the base descriptor to a veteran with a random chance. Inside Custom Difficulty, these chances are called *VeteranNormal* (affects Grunts and Mactera) and *VeteranLarge* (affects Oppressors). Of course, the veteran variants can be added by themselves to whatever pool is desired.

```
"VeteranNormal": [
 {
  "weight": 1,
  "range": {
   "min": 0.5,
   "max": 0.8
  }
 }
],
"VeteranLarge": [
 {
  "weight": 1,
  "range": {
   "min": 0.2,
   "max": 0.2
  }
 }
```

Veteran enemies are the source of a few unexpected quirks within the spawn system:

1. Encounters don't seem to go through the veteran upgrade process, so unless the veteran variants are added directly to the *Encounter Pool*, encounters will consist only of the base variants even if the veteran chances are very high.
2. Some veteran enemies, notably trijaws, seem to make it to the spawn pool even if they are not added directly to any pool and the VeteranNormal chance is zero.

Biome Variants

Some enemies only appear in certain biomes by default (the following table ignores critters):

Biome	**Special enemies**

Dense Biozone	Naedocyte Shockers
Glacial Strata	Naedocyte Shockers
Hollow Bough	Wasp Nest, Stabber Vine
Salt Pits	Younglings
Sandblasted Corridors	Nayaka Trawler

The descriptors corresponding to these enemies can be added to the spawn pools with CD so they spawn in every biome. In addition to these, Glacial Strata has frost versions of the Grunts, Praetorians and Goo Bombers, and the Radioactive Exclusion Zone has a radioactive Praetorian variant.