

---

## Sparky Spawner Documentation

(with some code *additions/alterations* by [The Amethyst Dragon](#))

---

Implementation: Spawned encounters are added to an area by adding string variables to an individual area. Code for the Sparky Spawner system has already been added to "00000001enter" (generic Aenea area enter script). More than one encounter can be added to the same area, but each needs to have its own number.

### Basic Variable Entry:

encounter_01	string	v2, active hours, percentage chance, encounter entry
--------------	--------	--

**v2:** The identifier for version 2 of the database structure (this one never changes).

**active hours:** The hours this encounter will be active. {day, night, always, fullmoon, spring, springday, springnight, summer, summerday, summernight, autumn, autumnday, autumnnight, winter, winterday, winternight, or a range of hours (6-12)}

**percentage chance:** What chance the encounter has to spawn. {1 for super rare to 100 for always spawns}

**encounter entry:** This section comprises a larger collection of variables, all combined into a single string.

Several different types of encounter entries are supported:

**creature (or npc), resref, amount, circle location, offset, random walk, continue**

### Basic Creature Encounter Example:

encounter_01	string	v2, day, 75, creature, devourerbeetle, 1-6, random, , 1, 1
--------------	--------	--

- spawns only during the day
- 75% chance of spawning 1-6 devourer beetles
- spawns at a random location in the area
- beetles begin to wander around after spawning
- other encounters in the area will continue to spawn (if they are added)

**resref:** the blueprint **resref** of the creature being spawned

**amount:** number to be spawned {simple number (5) or a range of numbers (1-7)}

**circle:** Prefixing any location type with this will cause the spawns to form a circle around the reference point, with a radius of meters. {"circle: " = normal circle, "circleflip: " = same, but facing of spawned placeables is rotated 180 degrees}

**location:** where will the creature(s) be spawned?

{

**waypoint:** **waypoint tag** (uses the location of the waypoint)

**wp:** **waypoint tag** (uses the location of the waypoint)

**object:** **object tag** (uses the location of the object)

**location:** 0.0x 0.0y 0.0z @ 0.0 (uses provided vector coordinates) (zenith & facing coordinates optional)

**loc:** 0.0x 0.0y 0.0z @ 0.0 (uses provided vector coordinates) (zenith & facing coordinates optional)

**random** (uses a random location in the area)

**last** (uses the last (previous) location)

}

**offset:** If an offset is provided with either a waypoint, object, or location, each spawn will be created at a random distance up to x meters in a random direction from the reference location. {uses a float value such as **5.25** or else a blank " " space for no offset}

**random walk:** will creature(s) start to do actions on their their own? {**0** = no, **1** = random movement, **walk, spasm, threaten, laugh, talk, beg, worship, pray, conjure1, conjure2, dead** (can be raised), **corpse** (can't be raised)}

**continue:** continue to load encounters after this one fires? {**1** = yes, **0** = no}

**placeable (or prop), resref, amount, circle location, offset, continue**

**item, resref, amount, circle location, offset, continue**

**traps, strength, type, amount, size, detect DC, disarm DC, recoverable, continue**

**strength:** {**minor, average, strong, deadly, epic, random**}

**type:** {**spike, holy, acid, tangle, fire, electrical, gas, frost, negative, sonic, acid splash**, plus **random**}

**amount:** number of this type of trap in the area {**1, 2, 1-5**, etc.}

**size:** {**1.0** is the minimum size, **2.0** the normal size}

**detect DC:** {-**1** for undetectable, or the integer DC to detect the trap}

**disarm DC:** {-**1** for not disarmable, or the integer DC to disarm the trap}

**recoverable:** {**0** = unrecoverable, or **1** = recoverable}

#### Encounter Trap Example:

encounter_01	string	v2, always, 100, traps, epic, random, 1-4, 1.89, 19, 21, 1, 1
--------------	--------	---

**script, script name** (area runs specified script, not used in Aenea)

**table, table name** (Rolls d100, and checks the results on the specified encounter table.)

#### Encounter Table Example:

encounter_01	string	v2, night, 100, table, easycaves
table_easycaves_01	string	25, creature, devourerbeetle, 1-6, random, , 1, 1
table_easycaves_02	string	50, creature, troll, 1-2, random, , 1, 1
table_easycaves_03	string	75, creature, orc001, 3-8, random, , 1, 1
table_easycaves_04	string	100, creature, drggreen001, 1, random, , 1, 1

- encounter spawns 100% of the time

- calls one encounter from the table (1-6 devourer beetles, or 1-2 trolls, or 3-8 orcs, or 1 green dragon)

- other encounters after this one may spawn

**group**, **group name** (groups several encounter strings together)

### Encounter Group Example:

encounter_01	string	v2, fullmoon, 75, group, werewolves
group_werewolves_01	string	npc, werewolf001, 1, random, , 1, 1
group_werewolves_02	string	npc, werewolf002, 1-2, last, , 1, 1
group_werewolves_03	string	npc, werewolf003, 1-2, last, , 1, 1
group_werewolves_04	string	npc, werewolf004, 1-6, last, , 1, 1

- active only during the night of a full moon
- 75% chance of spawning
- 1st werewolf spawns at a random location in the area
- other werewolves spawn at same location as the first
- all begin random wandering upon spawning
- other encounters will continue loading after these

**Copied Creatures** from another area (to save on storing creatures plus their custom items in the palette)

**v2**, **copy**, **area tag**, **object tag**, **new tag**, **amount**, **location**, **offset**, **action**, **continue**

### Example Copy Entry

encounter_01	string	v2, copy, area473, direwolf001, direwolfghost, 1, random, , 1, 1
encounter_02	string	v2, copy, area473, direwolf001, , 1, random, , 1, 1

**area tag**: tag of the area that the "original" creature is kept in

**object tag**: tag of the "original" creature

**new tag**: want to give the copy its' own new tag? put it here or else leave this blank

**amount**: how many copies?

**location**: same as above

**offset**: same as above

**action**: same as **random walk** above

**continue**: same as above