# Level System

IDEA FOR HOW TO HANDLE LEVELS

Levels/rooms can be an entity, a Room Entity with a sequence component

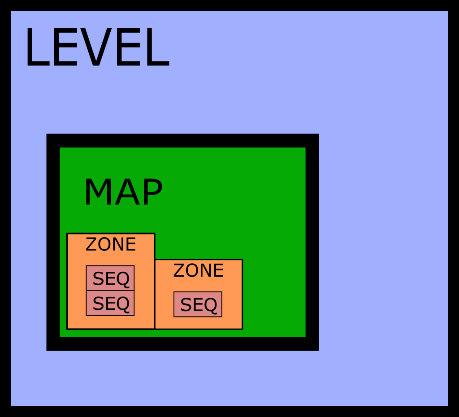
OR

Levels can have a list of sequences.

Each sequence is a room/sequence/event that is triggered by the player colliding with a trigger/sensor field and these are resolved via some criteria.

Say the sequence type is a narrative sequence that has an audio file play when the player crosses a certain point or trigger. It would have a reference to an audio file and it would send a game sequence event signal to any who give a shit. The game would note that this sequence has been completed somewhere so that it doesn’t get activated more than once

Say the sequence type is a combat sequence. The player crosses a trigger and the combat sequence has a reference to a list of enemies that it spawns.



Level, map, zone, sequence

Levels will be files, and these files will contain data

Each level file will have a map object or a list of map objects

Within each map will be zones that hold sequences

And sequences are where the magic of levels really happens

## Sequences

Sequences will be game objects that can be of varying types; combat, narration, etc.

The type of the sequence will determine its completion criteria

There will be an ordering for each sequence, which will just be a number that will determine when they are initiated

Sequences can contain spawndata and other relevant data to the sequence type

## ~~Zones~~

~~Zones can be found at different parts of a map and each zone contains at least one sequence.~~

~~Zones are also characterized by a type~~

~~Zones are the equivalent of rooms and each zone is activated by a trigger and each zone has its own completion criteria.~~

## Maps

The map files will hold particular pieces of level data. Maps are what you traverse through and clear in order to pass onto the next level. Levels can contain one map or a series of maps.

### How it’s going to work

The map file will contain a layer ~~for zones and a layer~~ for triggers and each trigger will have a special tag.

When the level is loaded, the map file will be loaded first, and the map and walls will be created. Then the level class will get the triggers from the map file and will create them as well. The level class will go through the level file and create the will go through the list of sequences in the