

# Chunk System

## *Visual Representacion of how the chunk system works*

At the start of the development of the game, we foresaw a problem that was the overload when creating the map, and the amount of objets that live in the RAM, that constantly upgrade themself. A way of preventing this issue, was the use of chunks. Where we divide the world in parts of land (this being the chunks), and the chunk that the player is interacting is the only part that gets upgraded.


(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
(1,3)	(2,3)	(3,3)	(4,3)	(5,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)

*(a visual representation of how  
chunks separate the map, each box  
being a chunk)*


The chunk system only render chunks where the camera is, and its adjacents

### Clarifications:


- **Camera**
- **Render Chunk**

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
(1,3)	(2,3)		(4,3)	(5,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)


To prevent the chunks to live even after the camera is far away from them (or the camera is two chunks away), this objet chunk will be deleted

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
(1,3)	(2,3)		(4,3)	(5,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)

1

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
(1,3)	(2,3)	(3,3)		(5,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)

2

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)
(1,3)	(2,3)	(3,3)		(5,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)

3

Its important to remember, that the hole system is implemented in the view, because the model still loads every object despite the camera not seeing them

## A few more examples

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)	(6,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)	(6,2)
(1,3)	(2,3)	(3,3)	(4,3)	(5,3)	(6,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)	(6,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)	(6,5)
(1,6)	(2,6)	(3,6)	(4,6)	(5,6)	(6,6)

(1,1)	(2,1)	(3,1)	(4,1)	(5,1)	(6,1)
(1,2)	(2,2)	(3,2)	(4,2)	(5,2)	(6,2)
(1,3)	(2,3)	(3,3)	(4,3)	(5,3)	(6,3)
(1,4)	(2,4)	(3,4)	(4,4)	(5,4)	(6,4)
(1,5)	(2,5)	(3,5)	(4,5)	(5,5)	(6,5)
(1,6)	(2,6)	(3,6)	(4,6)	(5,6)	(6,6)