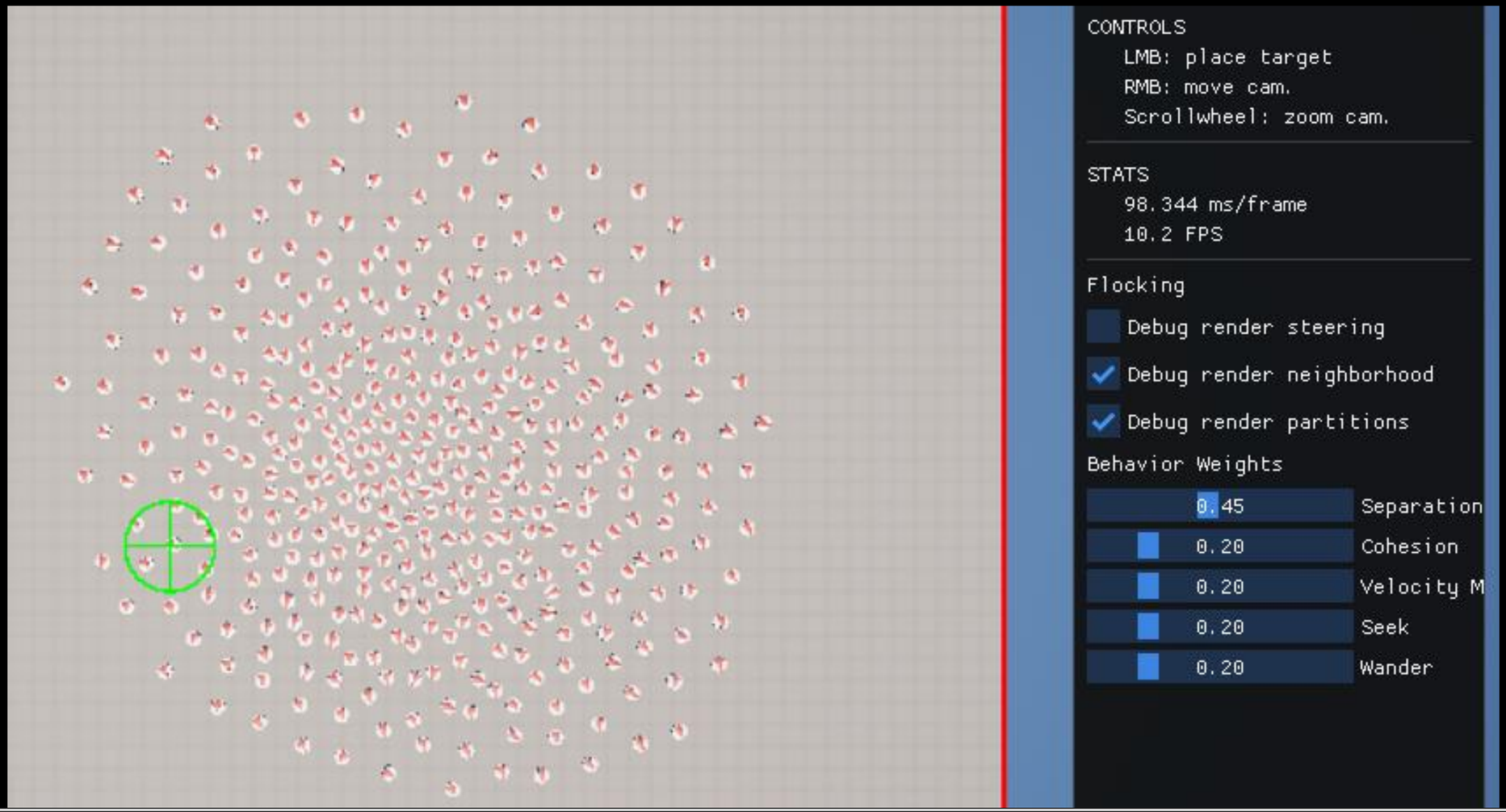
Spatial Partitioning

What the flock is going on with our frames?

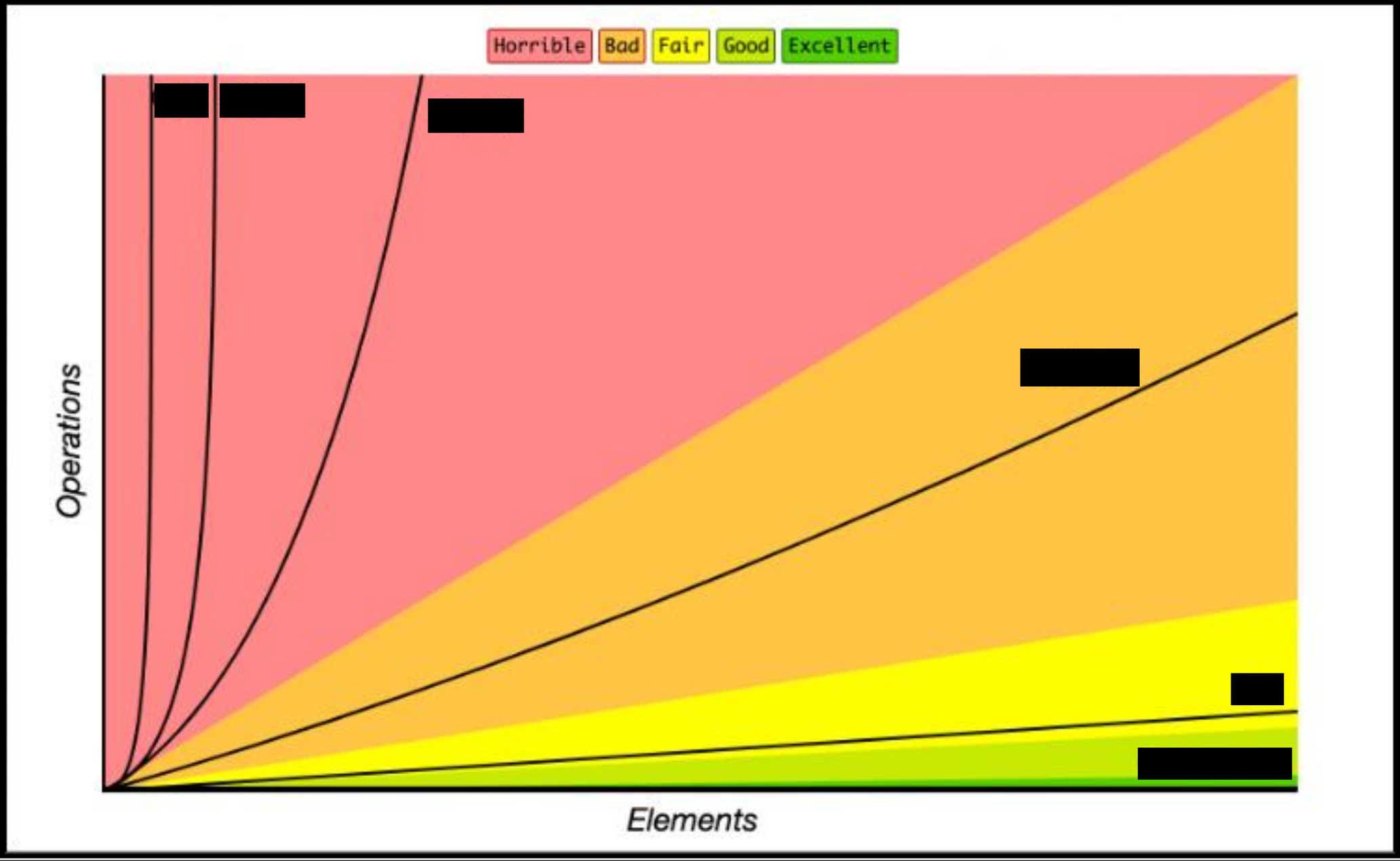
400 Agents



The Problem

For every agent we have, there’s a lot more work

```
for ( current : agents )
{
    for ( other : agents )
    {
        // check distance + add to neighborhood
    }
}
```


Horrible
Bad
Fair
Good
Excellent
Operations
Elements

The Problem

For every agent we have, there's a lot more work

```
for ( current : agents )
{
    for ( other : agents )
    {
        // check distance + add to neighborhood
    }
}
```

For every agent, every agent has an extra potential neighbor and there's another agent to check neighbors for:

100 Agents → 10,000 operations!

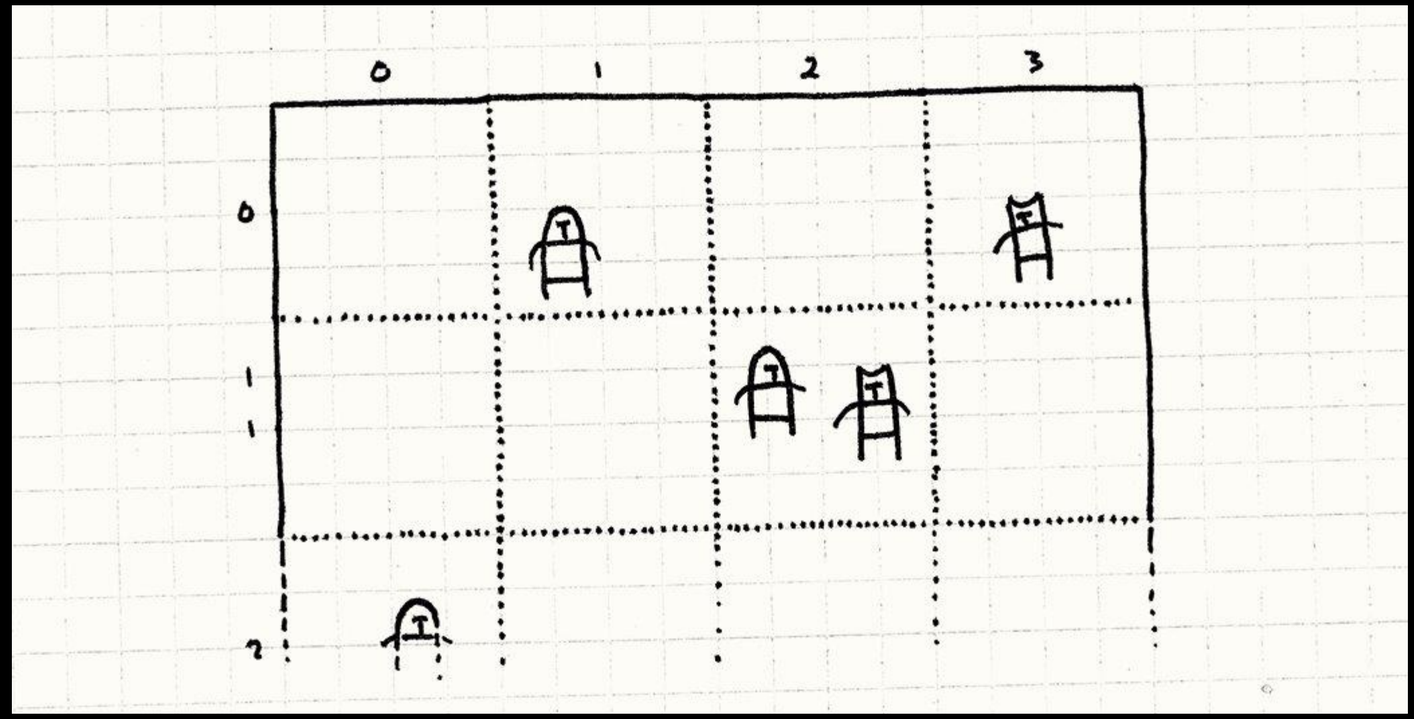
$O(n^2)$ complexity → *Not good!*

Spatial Partitioning!

AKA Space Partitioning

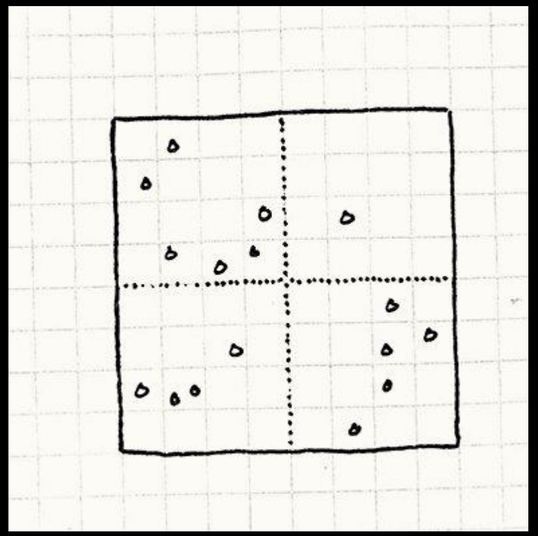
The Concept

- Dividing space into subsets, called **Partitions**
- Each **Partition** keeps track of its **own data**
  - *Such as agents in its area*
- Allows us to **access data** in a **spatially relevant way**, instead of having to evaluate the entire world
- Multiple methods exist! But fall into 2 categories



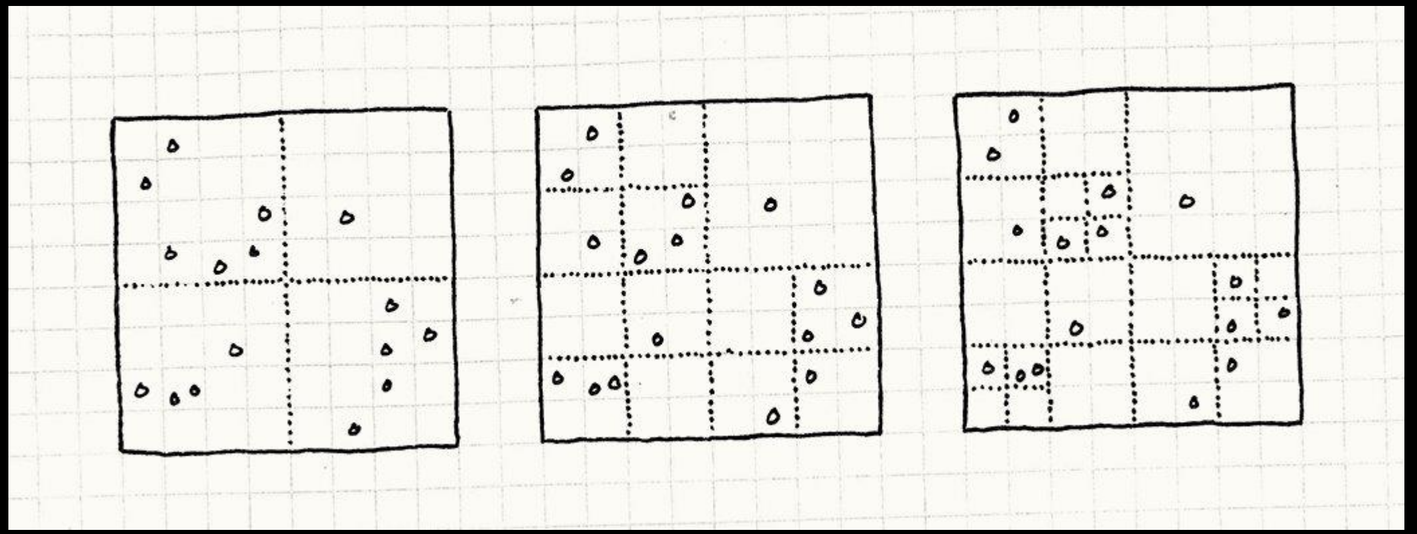
Read more here

Flat vs Hierarchical Partitioning



Flat Partitioning

- Uniform grid (or hashmap)
- Constant memory size
- Non adaptive
- Easy to update & query
  - Since we can know the cell we're in based on our position
  - Query → **O(1)**
- Expensive for large spaces



Hierarchical Partitioning

- Tree structure (Quadtree, Octree,…)
- More memory efficient for uneven distributions
- Adaptive
- Can adapt to large "sparse" spaces
- Fast spatial queries **O(log n)**

Alternative use case

In large worlds, its possible we'll have many agents scattered over a large area

When too far away, we may want to simplify our simulation. So, we avoid making redundant calculations

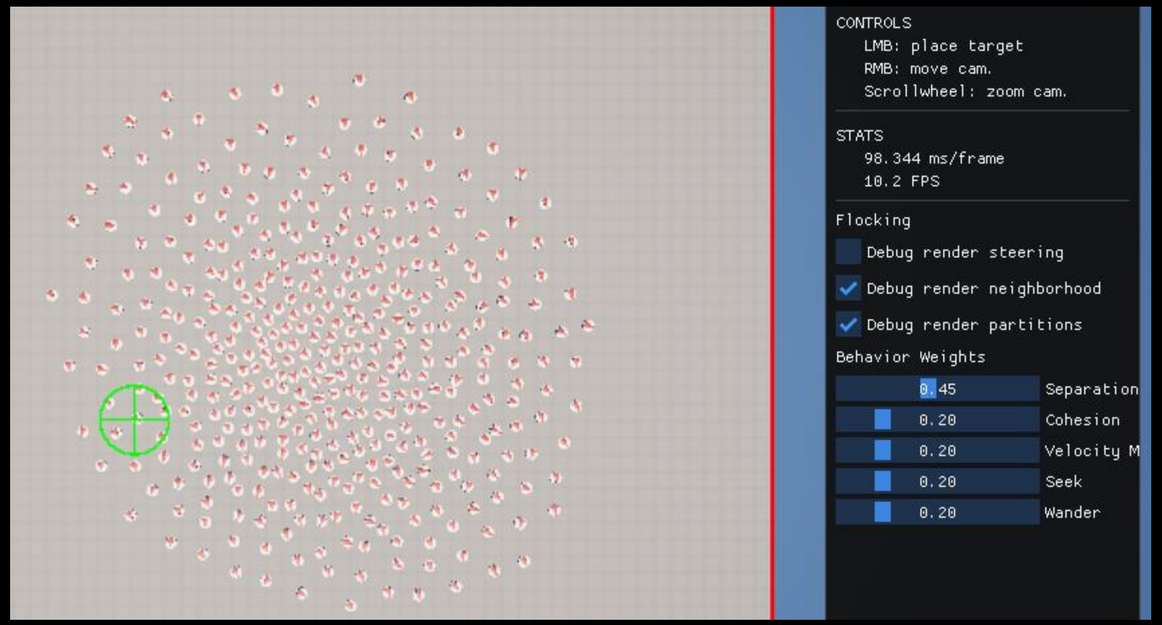
Like LOD's for agent behavior

Only data in the current partition (or nearby partitions) will be loaded/considered relevant

What the flock is going on with our frames?

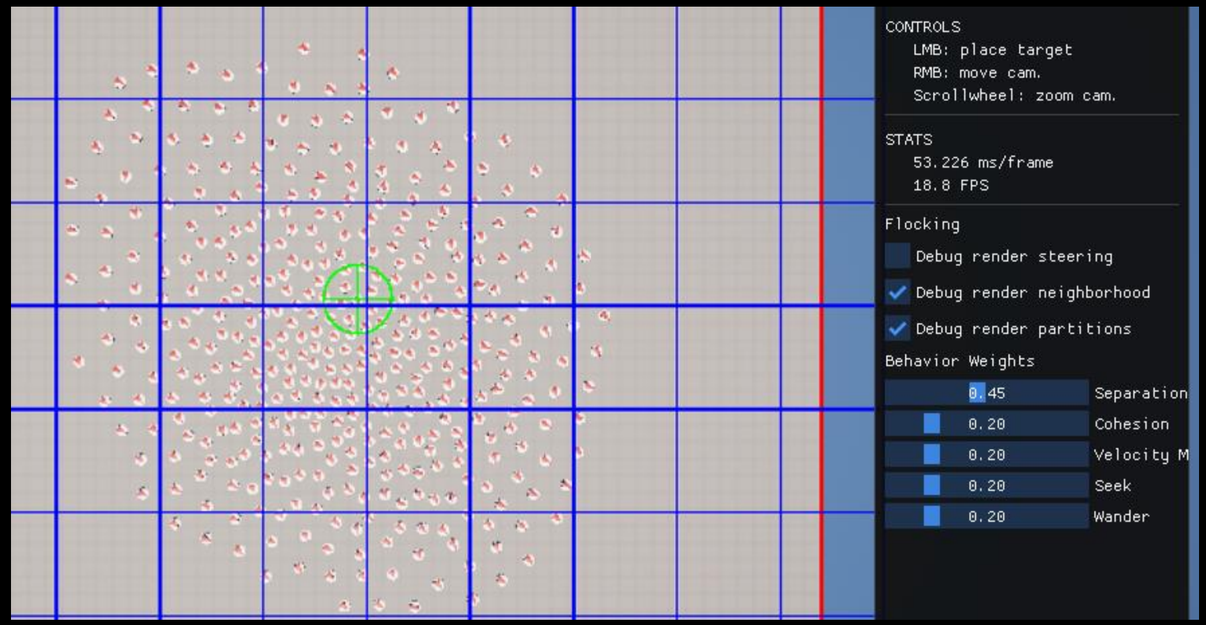
No Spatial Partitioning

400 agents, **10FPS**



With Spatial Partitioning

400 agents, **19FPS** *(+90%!!!)*



Result

insert epic battle music

Extra Assignment

Implement one of the following:

- Implement a Hierarchical partitioned space in the flocking assignment

- Profile the flocking assignment in various situations. Test in release mode, write a short document explaining your findings. Make sure to apply proper performance profiling methods