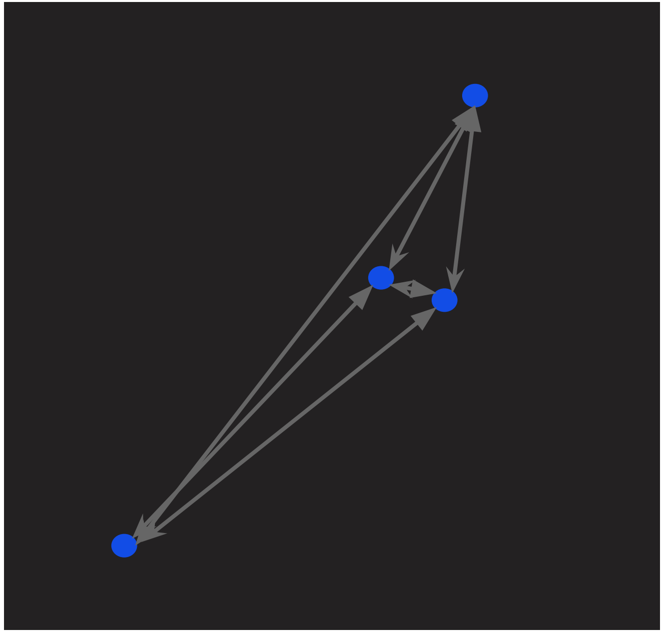
# Barnes-Hut simulation of a 2 galaxy collision


> Marnix Rebergen
> Sarwan Peiter
> Mario Gely

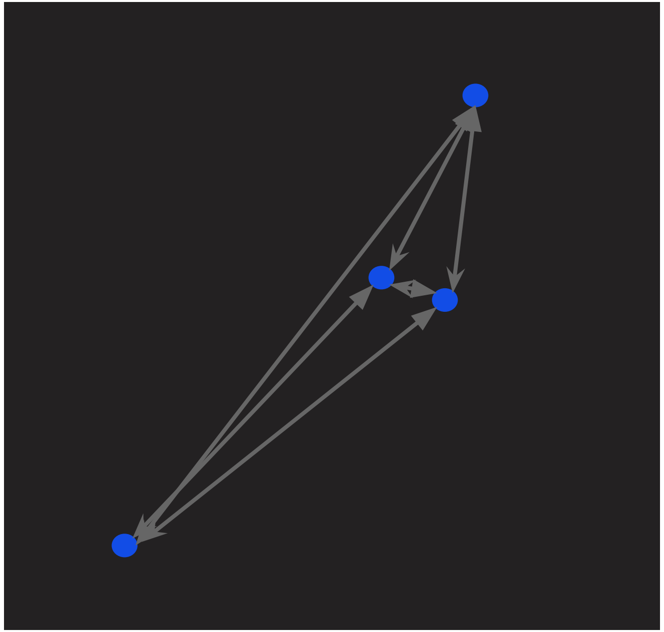
# Solving the N-Body Problem

We want to solve the equations of motion for a model galaxy with >10k stars



# Solving the N-Body Problem

Similar to Molecular Dynamics, but in this case the forces involved are long-range, and always attractive



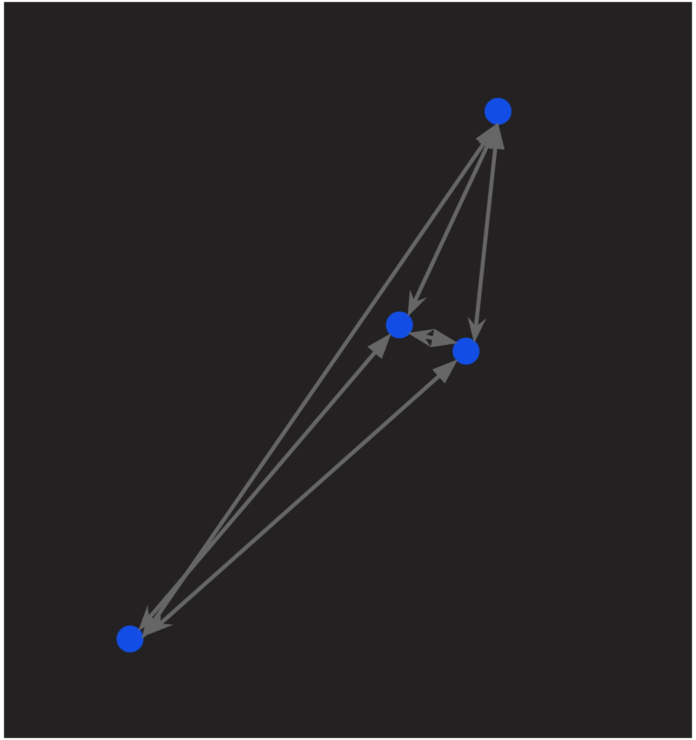
# Solving the N-Body Problem

The lack of repulsion causes problems when combined with the singularity in the gravitational force at $r = 0$

Therefore a softening parameter $\varepsilon > 0$ is introduced:

$$\mathbf{F} \propto -\frac{\mathbf{r}}{(r^2 + \epsilon^2)^{3/2}}$$

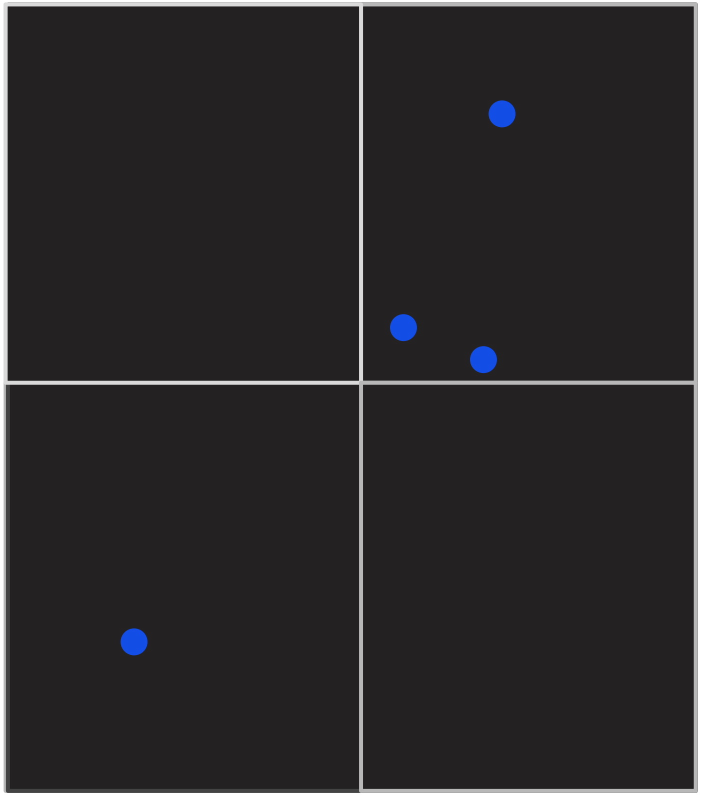
# Direct Calculation



> O(N^2) operations

> Long range potential: using a cutoff is inaccurate

# The Barnes-Hut Algorithm (2D)

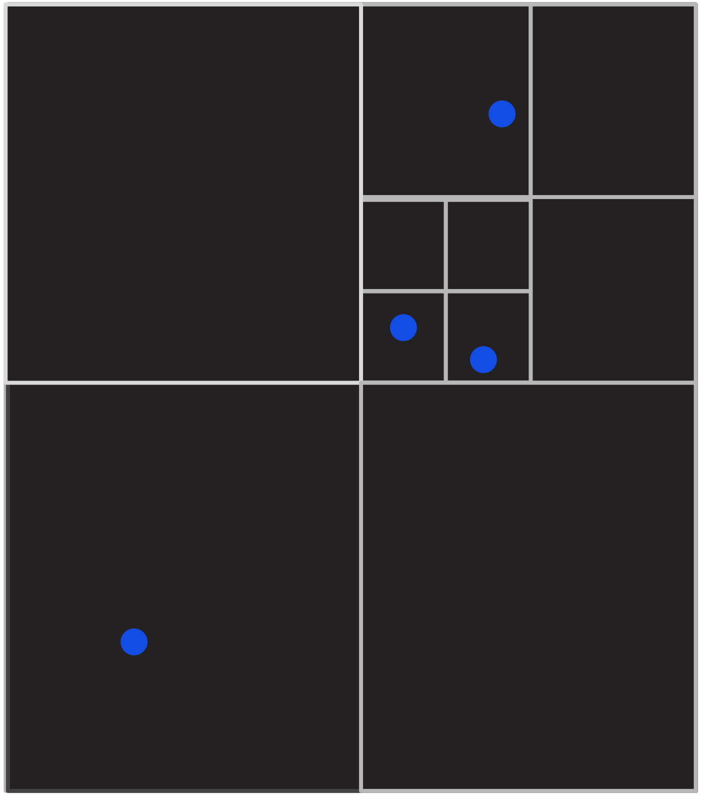


>Create a square cell large enough to contain the entire system

>Divide this cell into 4 sub cells

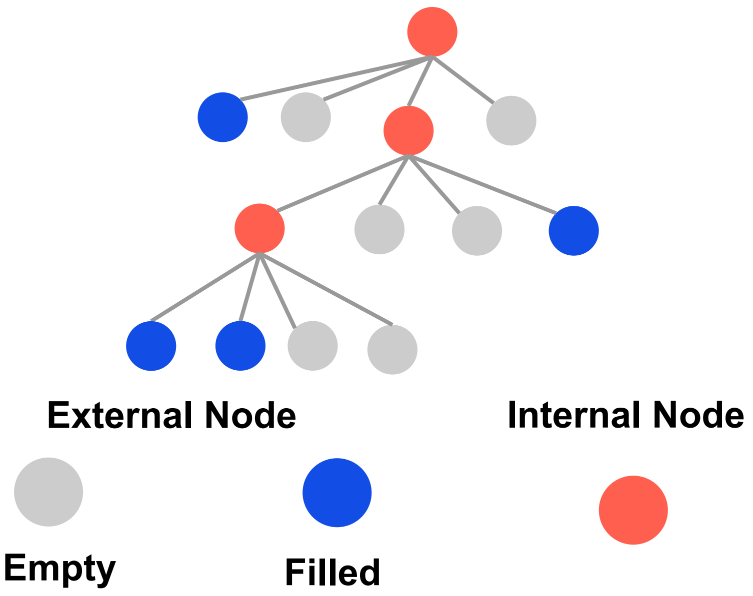
> Continue dividing the cells until each contains 1 or 0 particles

# The Barnes-Hut Algorithm (2D)

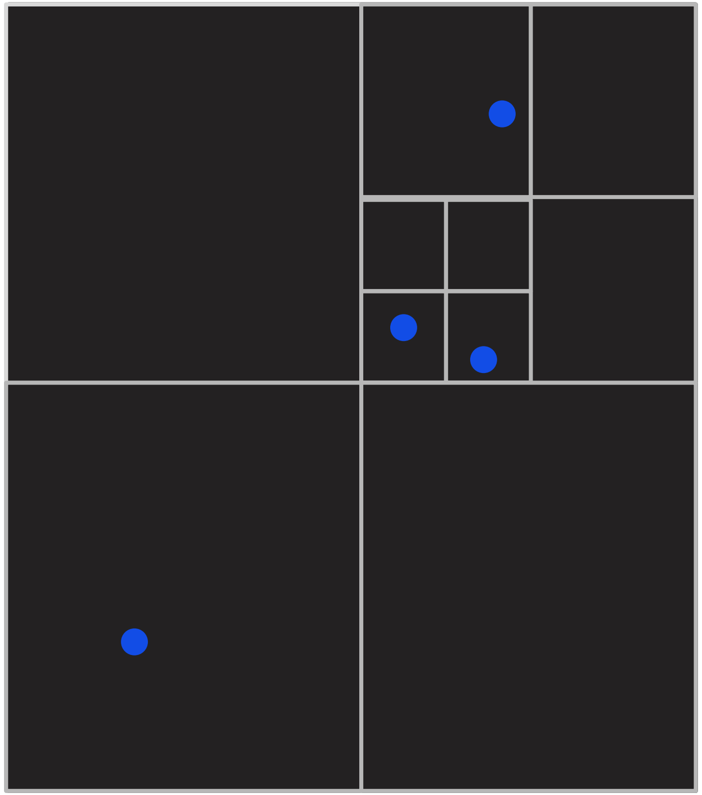


> Each cell is represented by a node

> Together the nodes form a (quad)tree structure



# The Barnes-Hut Algorithm (2D)

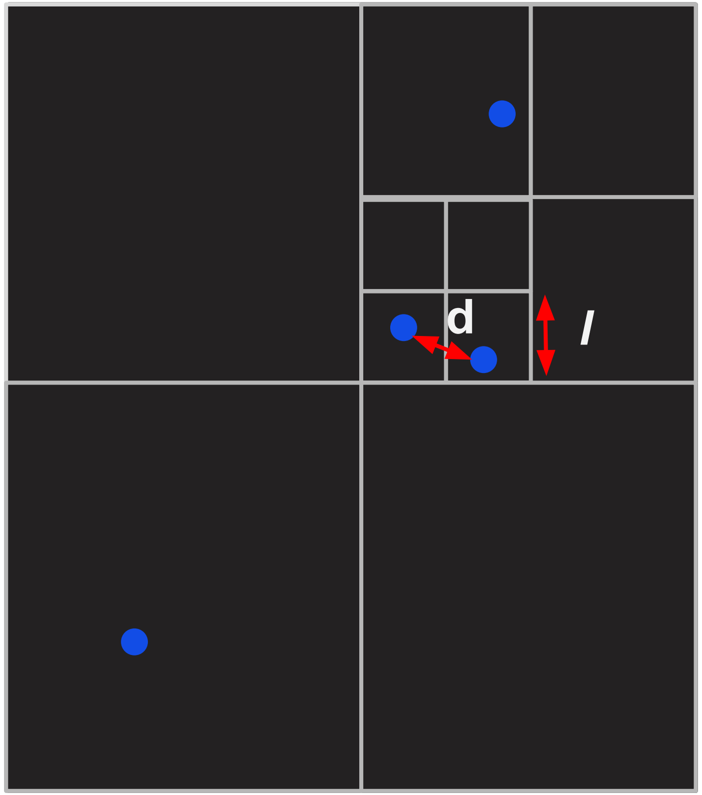


The internal nodes represent virtual particles:

> Mass equal to the total mass in the cell

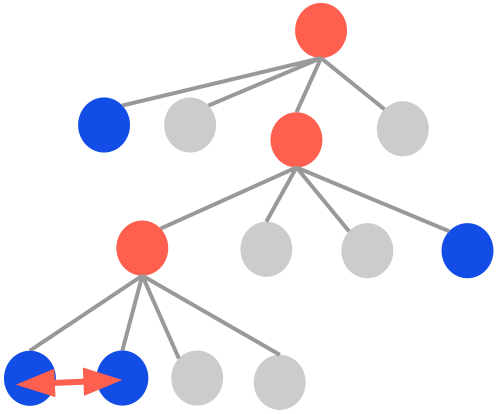
> Located at cell's center of mass

# The Barnes-Hut Algorithm (2D)

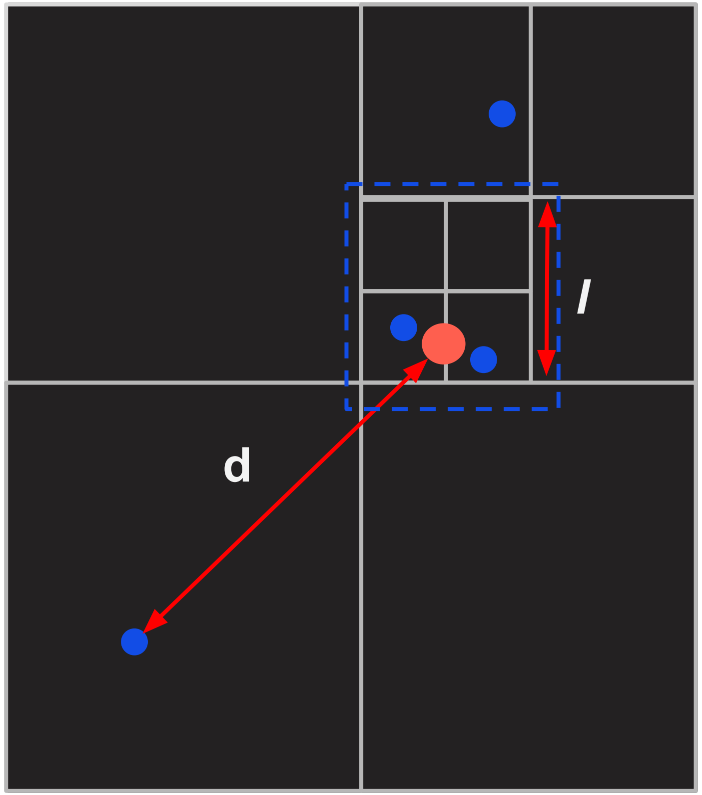



Distance criterion:

> For pair satisfying: $l/d<\theta$~0.5 the interactions are calculated directly

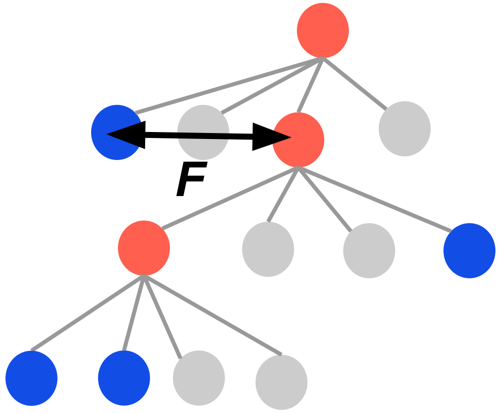


# The Barnes-Hut Algorithm (2D)

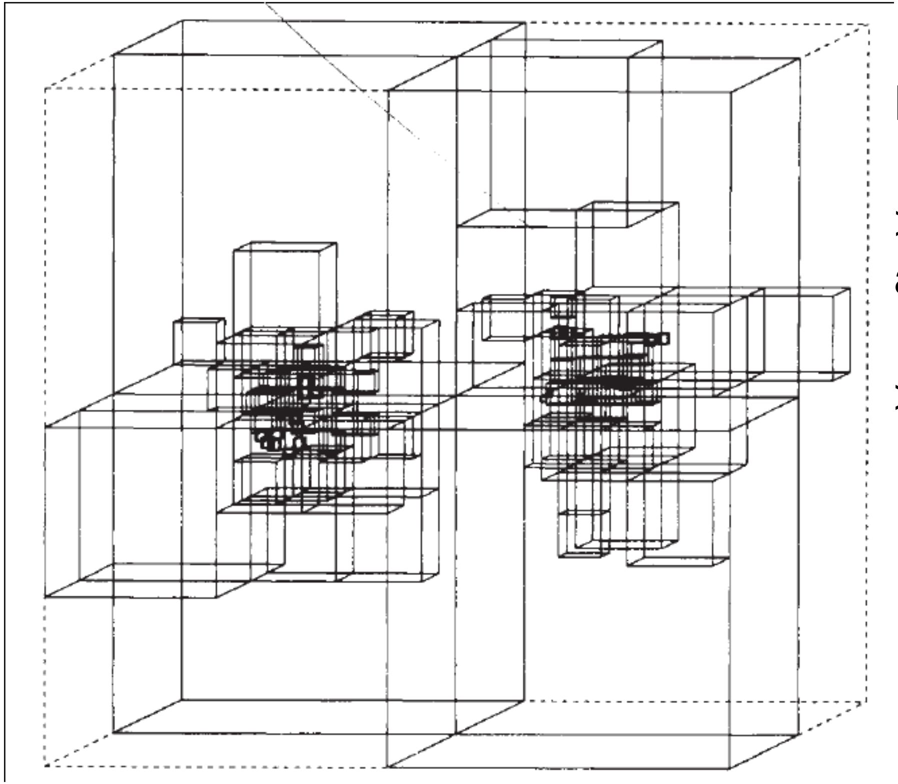



When $l/d > \theta$ :

> Distant particles are represented by the virtual particle at the internal node



# The Barnes-Hut Algorithm (3D)



> Image: Barnes and Hut (1986)

In 3D the BH algorithm works the same:

> Space is now divided into cubes. These are then divided into 8 octants

> This results in an octree structure

# The Barnes-Hut Algorithm

This algorithm reduces the number of operations from O(N^2) to O(N log(N)) compared to the direct method

This makes it possible to calculate the interactions for much larger systems

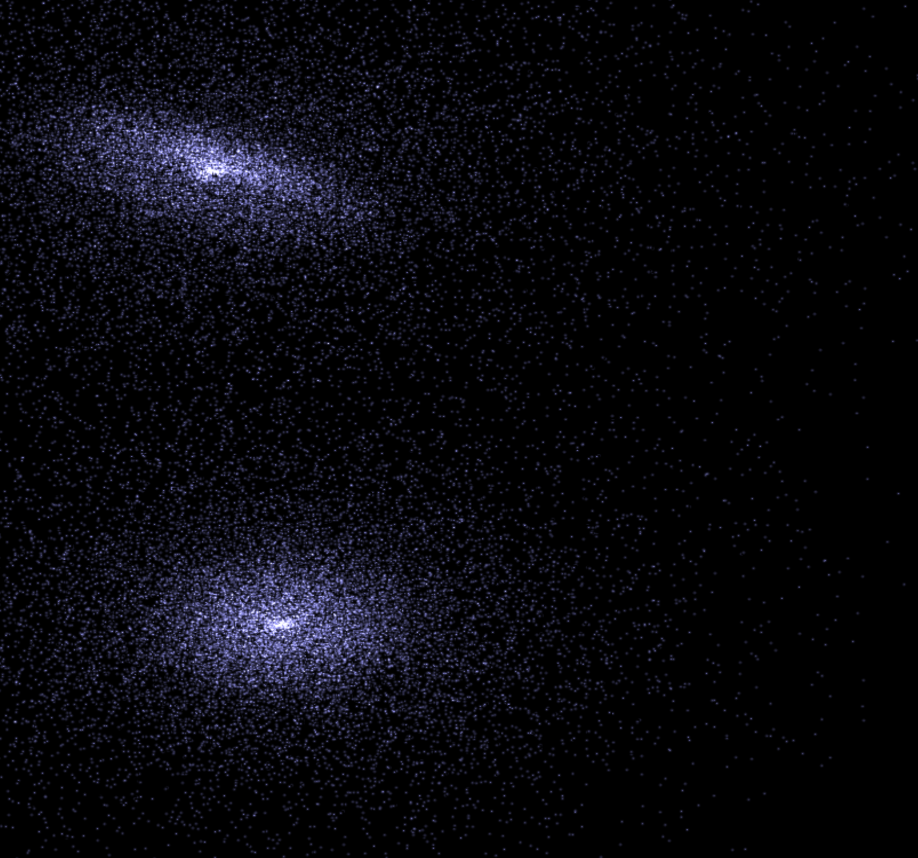
# Implementation: overview

3D simulation

Verlet time integration

Fortran and C++ versions

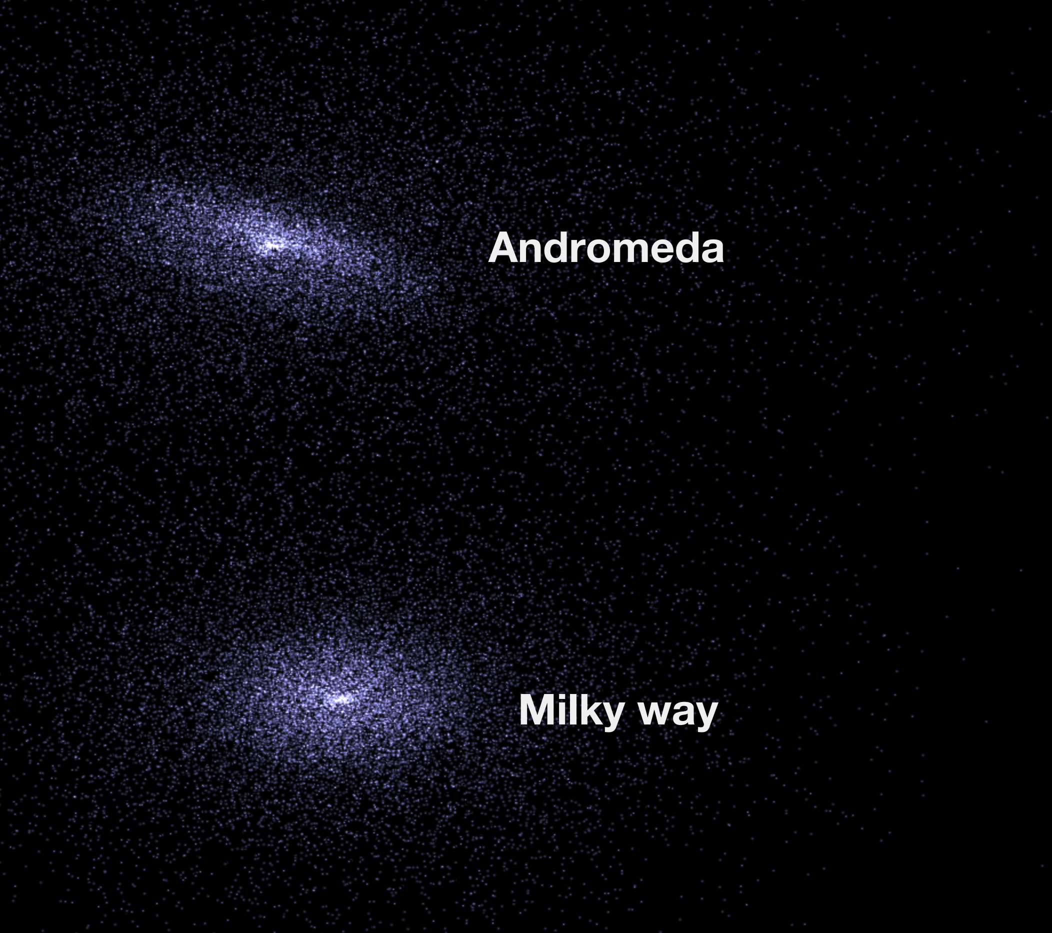
OpenGL



# Implementation: Initial Conditions

**Initial conditions: Data for Collision Milky Way and Andromeda**

**A total of**
**N = 81920 Stars**



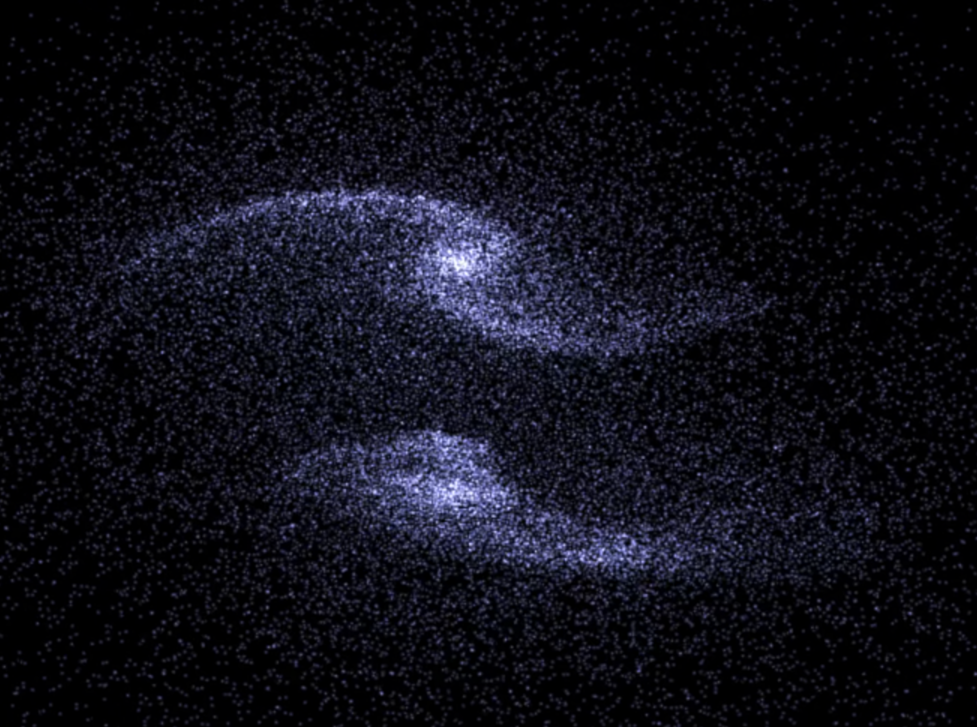

> Data: Dubinski et al. 1995

# Results

N = 81920
θ = 0.5
ε = 0.025
dt = 0.02

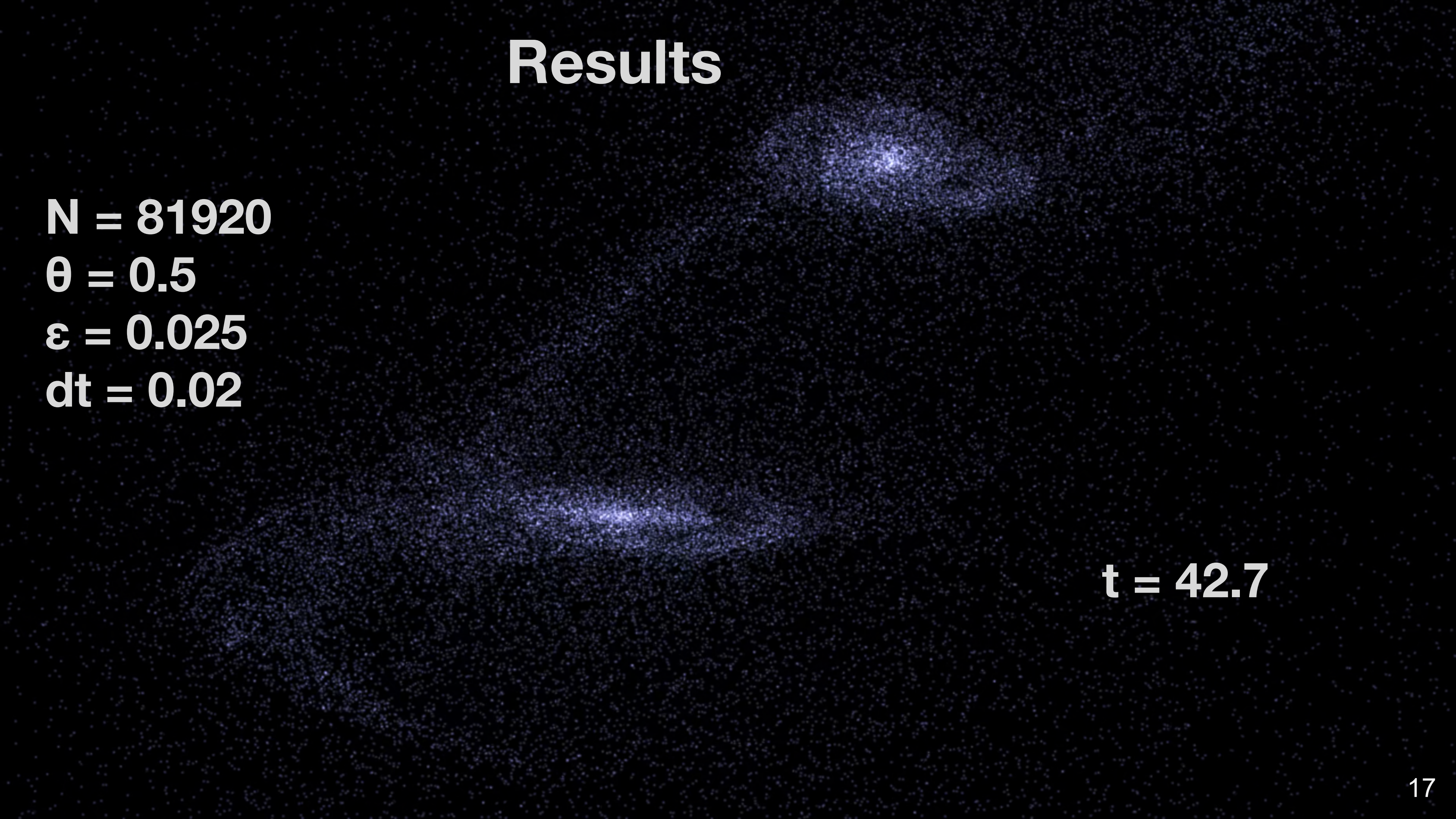
# Results

N = 81920
θ = 0.5
ε = 0.025
dt = 0.02



t = 21.3

# Results

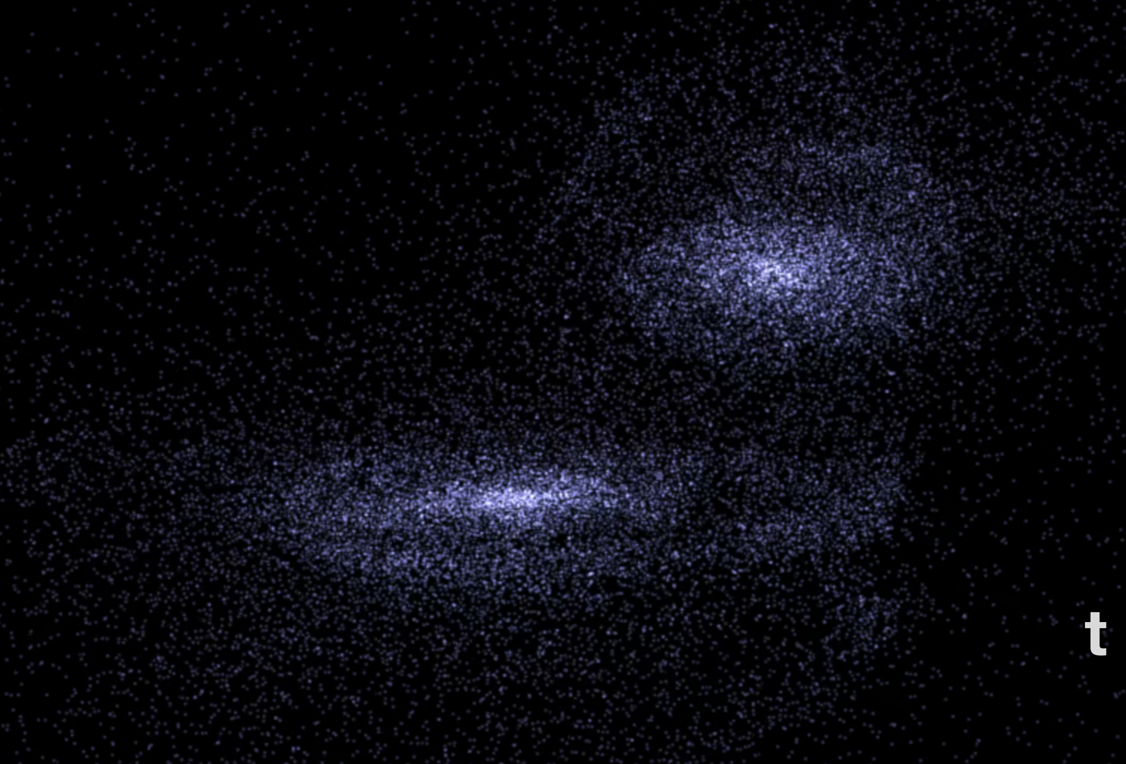


N = 81920
θ = 0.5
ε = 0.025
dt = 0.02

t = 42.7

# Results

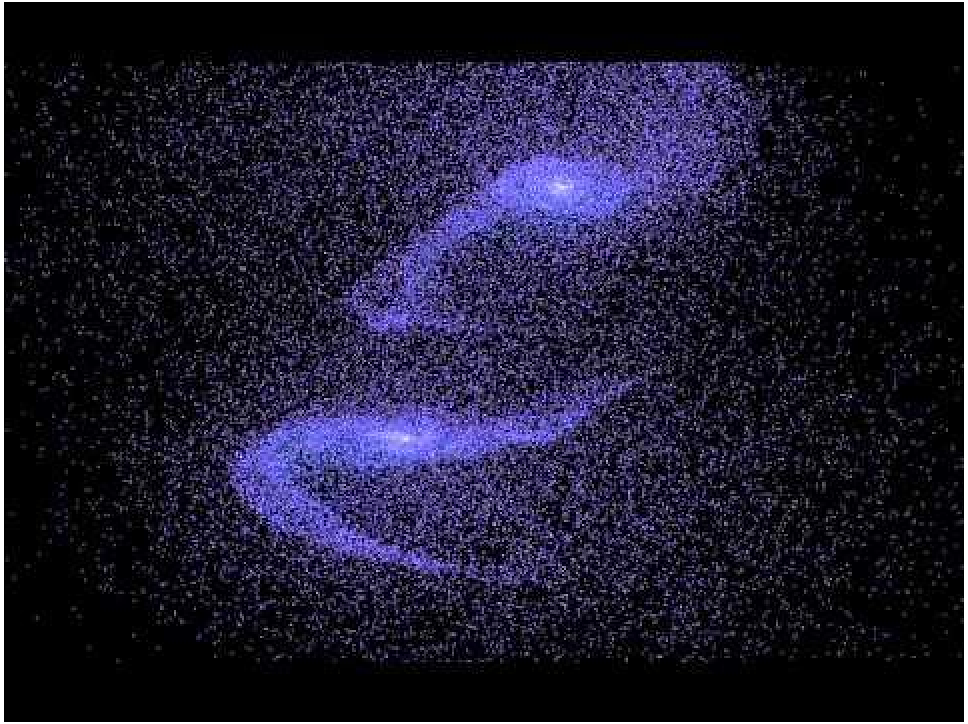
N = 81920
θ = 0.5
ε = 0.025
dt = 0.02



t = 60.5

# Results: animation

**N = 81920**
**θ = 0.5**
**ε = 0.025**
**dt = 0.02**



# References


> Barnes, Josh, and Piet Hut. "A hierarchical O (N log N) force-calculation algorithm." Letters to Nature, (1986): 446-449.

> Aarseth, S. J., M. Henon, and Roland Wielen. "A comparison of numerical methods for the study of star cluster dynamics." Astronomy and Astrophysics 37 (1974): 183-187.