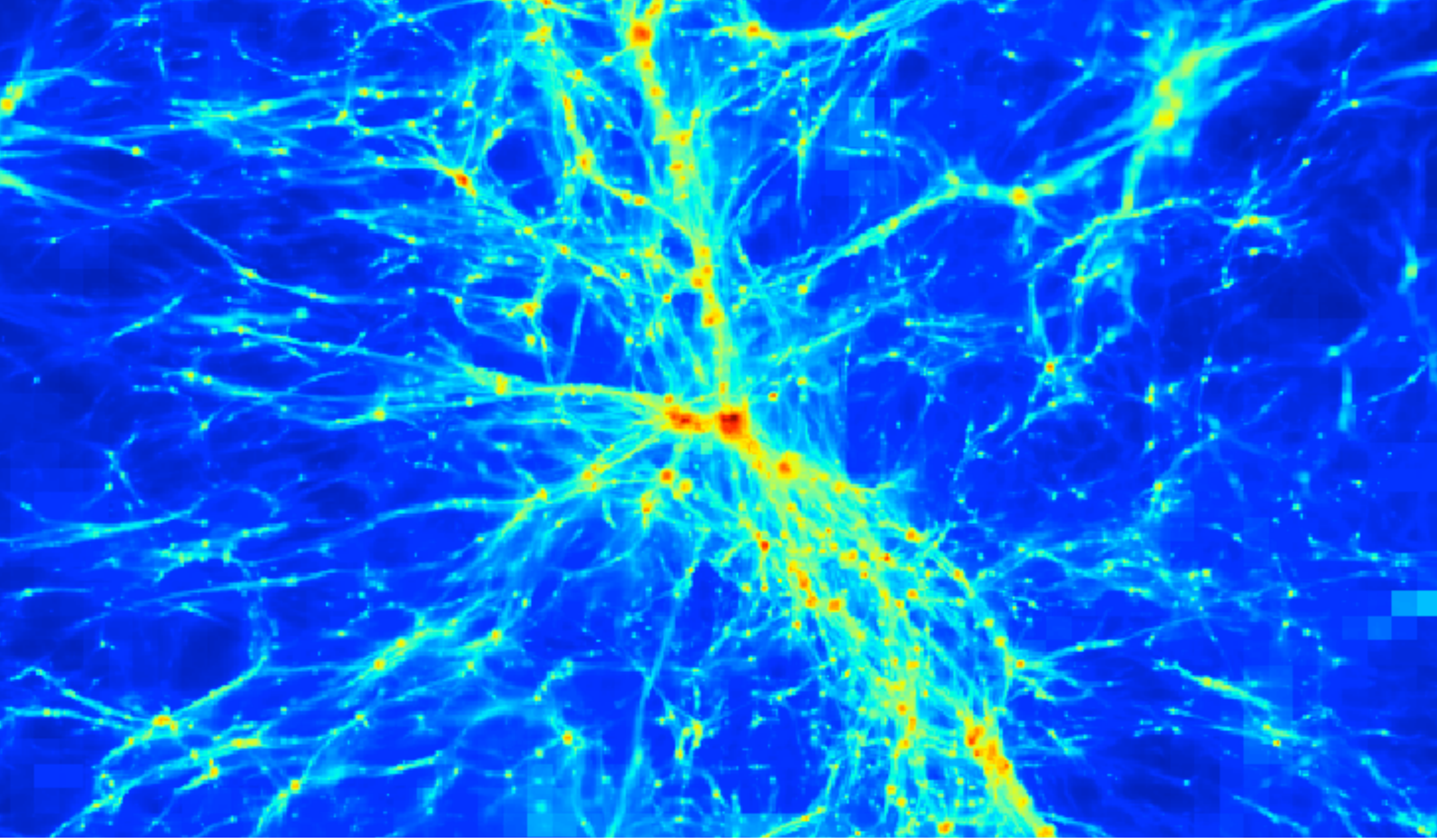


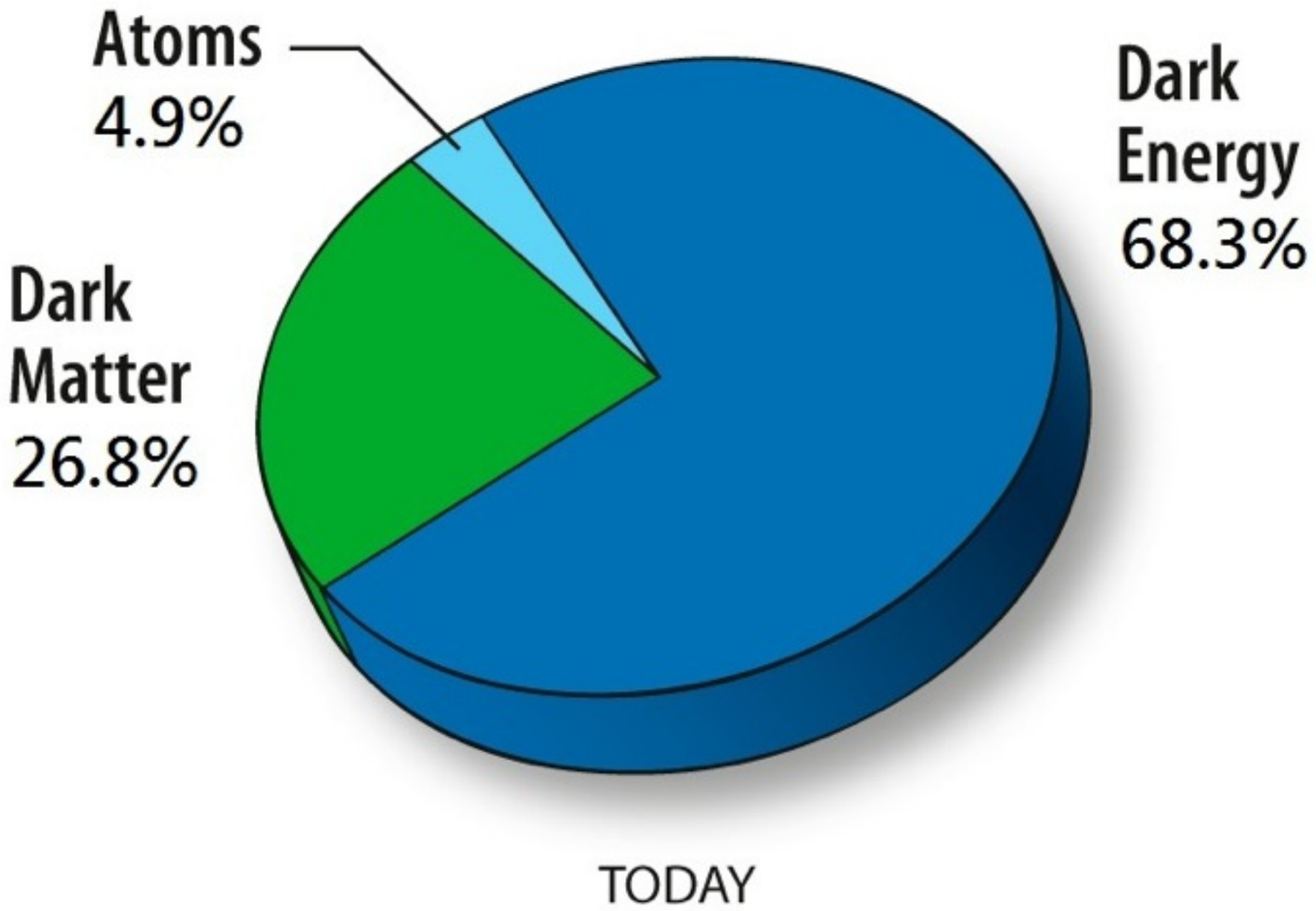
Cosmology Simulations with Enzo

John Wise (Georgia Tech)
Enzo Workshop
北海道大学 – 19 Nov 2014

Outline

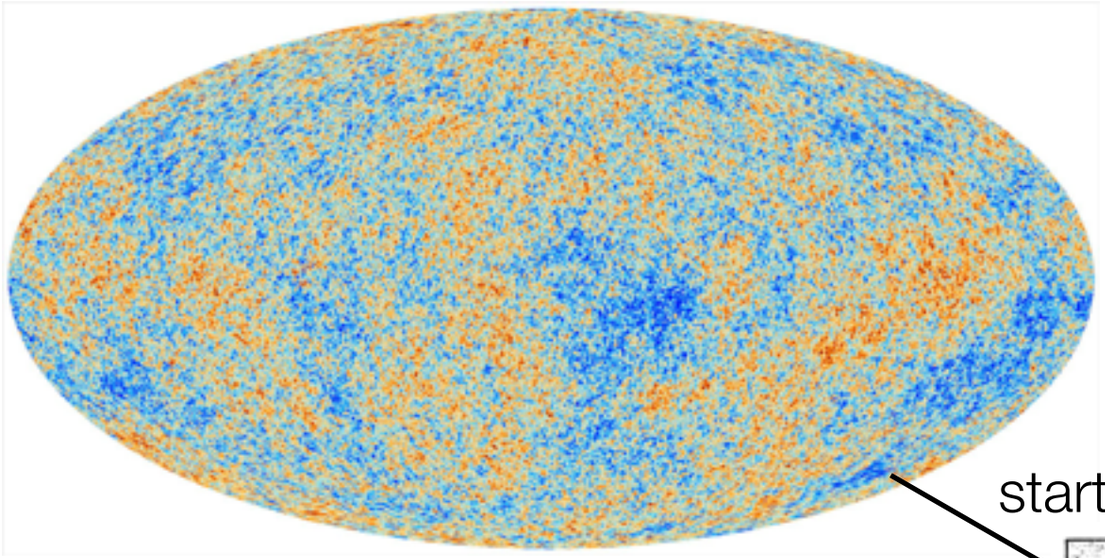
- Introduction to unigrid cosmology simulations
- Introduction to nested grid cosmology simulations
- Using different non-equilibrium chemistry models
- Including radiative cooling & star formation/feedback
- Including radiative transfer

A universe in a box



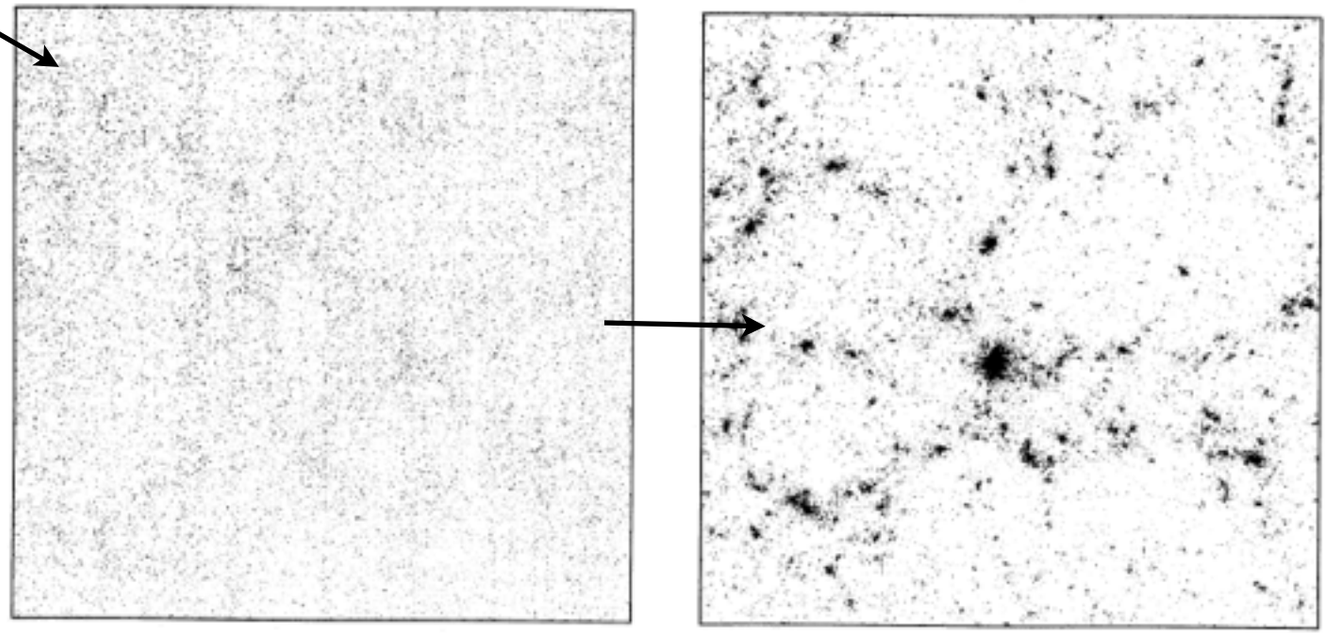
Planck Collaboration (2014)

A universe in a box



z ~ 1080

Linearly evolve to a
starting redshift (z ~ 50–200)

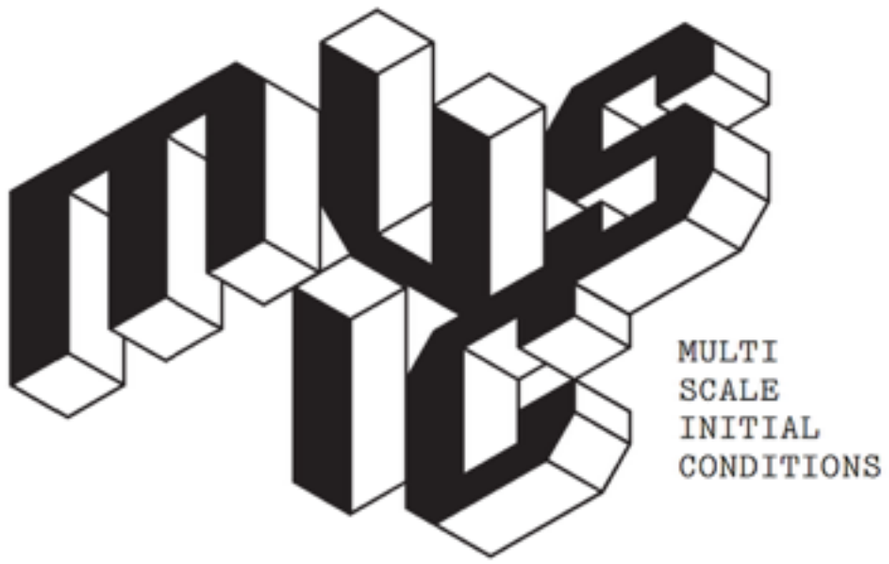


White et al. (1985)

Use simulations to evolve in
the non-linear regime

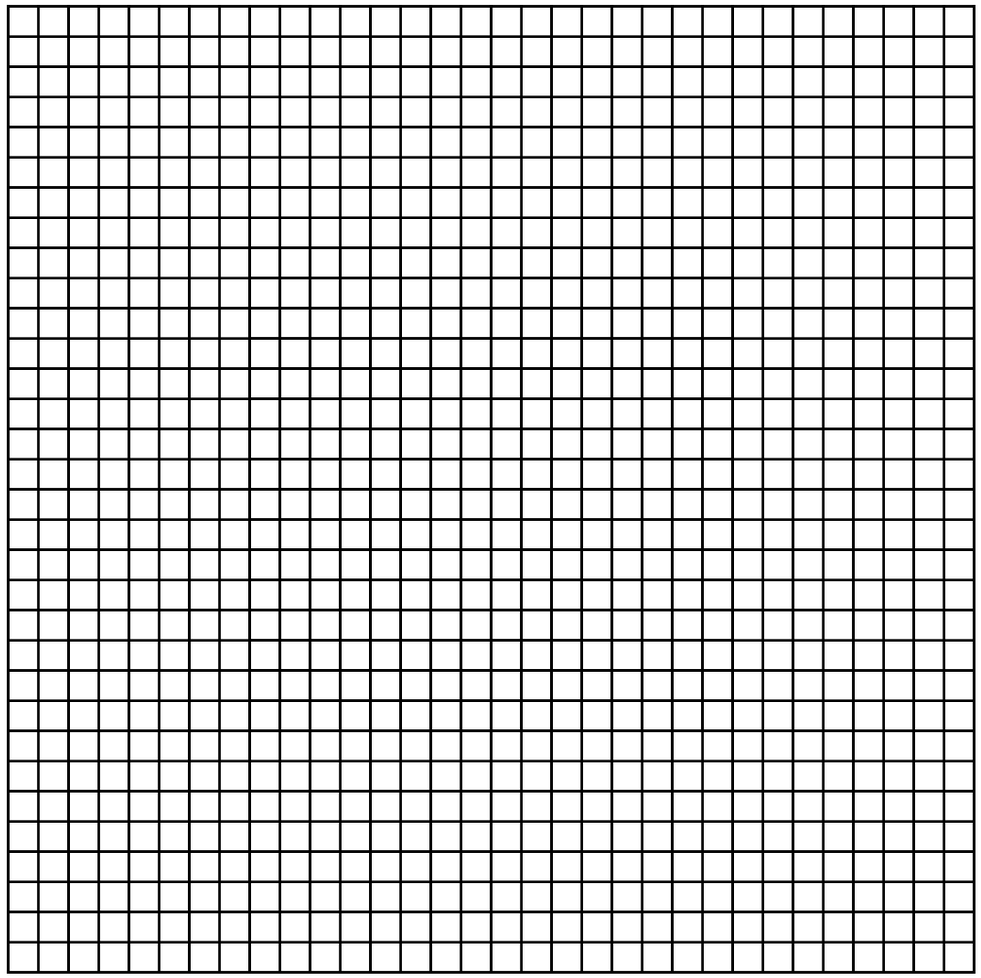
Unigrid Initial Conditions

- Two programs are provided with Enzo to generate cosmology initial conditions: **inits** (src/inits) or **mpgrafic** (src/mpgrafic)
- However, we will be using the program **MUSIC** (Hahn & Abel 2011) that is more robust and efficient.
  - http://www.phys.ethz.ch/~hahn/MUSIC/index.html
  - https://bitbucket.org/ohahn/music
- 10 initial condition files in HDF5 file format
  - Baryon Density: 1 x N x N x N
  - (x,y,z) Baryon Velocity: 1 x N x N x N
  - (x,y,z) Particle Velocity: 1 x N x N x N
  - (x,y,z) Particle Displacement: 1 x N x N x N



Unigrid Initial Conditions with MUSIC

- MUSIC: MUlti-Scale Initial Conditions
- Designed specifically for nested grid “zoom-in” simulations.
- Improves errors by two orders of magnitude compared to previous codes.
- Zel’dovich approximation and 2LPT
- Multi-code support (e.g. GADGET, ART, Tipsy, RAMSES, Enzo)
- Publicly available, OpenMP parallel



Unigrid Initial Conditions with MUSIC

- **Requirements:**
  - **GNU Scientific Library (GSL)**
  - **FFTW 2 or 3**
  - **HDF5**

Installing MUSIC

- Download the source code (on conival)
  - `hg clone https://bitbucket.org/ohahn/music`
  - `cp ~guest01/music/Makefile music/`
  - `cd music`
  - `make`
  - `export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/home/guest01/local/lib`

Grab MUSIC and Enzo parameter files

- On conival,
  - `cd ~`
  - `tar xf ~guest01/sapporo_cosmo.tar`
- On your laptop
  - `wget http://www.physics.gatech.edu/~jw254/scpics/sapporo_cosmo.tgz`
  - `tar xf sapporo_cosmo.tgz`

Initial Conditions Parameters

- Cosmology Parameters: box size, Hubble constant, mass-energy fractions (Ω_Λ, Ω_m, Ω_b), initial redshift

- Power spectrum parameters: power spectrum type, σ_8, random seed

```
[setup]
boxlength       = 5
zstart          = 50
```

units of comoving Mpc/h

level = 5 → 32^3 resolution

```
[cosmology]
Omega_m         = 0.3175
Omega_L         = 0.6825
Omega_b         = 0.049
H0              = 67.11
sigma_8         = 0.8344
nspec           = 0.9624
transfer        = eisenstein

[random]
seed[5]         = 201310160
```

Initial Conditions Parameters – Grid parameters

```
[setup]
boxlength       = 5
zstart          = 50
levelmin        = 5
levelmin_TF     = 5
levelmax        = 5
baryons         = yes
use_2LPT        = yes
use_LLA         = yes
periodic_TF     = yes
```

Initial redshift

Base resolution = 2^n cells across

Finest resolution of zoom-in region

Generate baryon ICs

2LPT = yes; Zel'dovich = no

```
MUSIC% ~/codes/music/MUSIC sapporo_cosmo.conf

                                this is music! versi

 - Opening log file 'sapporo_cosmo.conf_log.txt'
 - Selecting transfer function plug-in 'eisenste
 - starting at a=0.0196078
 - Selecting region generator plug-in 'box'...
 - WARNING: The selected transfer function does
            distinct amplitudes for baryon and DM
            Perturbation amplitudes will be ident
-------------------------------------------------
 - Grid structure:
     Level   5 :   offset = (    0,     0,     0)
                   size = (   32,    32,    32)
-------------------------------------------------
 - Finest level :
                    extent =  5 x 5 x 5 h-3 Mpc**3
                 mtotgrid =  1.10141e+13 h-1 M_o
            particle mass =  2.84249e+08 h-1 M_o
          baryon mass/cell =  5.1874e+07 h-1 M_o
                        dx =  0.15625 h-1 Mpc
-------------------------------------------------
 - Selecting output plug-in 'enzo'...
=================================================
   GENERATING WHITE NOISE
-------------------------------------------------
```

```
 - Density calculation took 0.170149s with 4 threads.
 - Top grid mean density is off by -4.1901e-17, correcting...
 - Global density extrema:
 -    minimum: delta=-0.583318 at (0.765625,0.953125,0.515625)
 -         shifted back at (0.765625,0.953125,0.515625)
 -    maximum: delta=0.664508 at (0.359375,0.796875,0.921875)
 -         shifted back at (0.359375,0.796875,0.921875)
-------------------------------------------------
 - Invoking unigrid FFT Poisson solver...
-------------------------------------------------
 - Invoking unigrid FFT Poisson solver...
=================================================
   COMPUTING BARYON DENSITY
-------------------------------------------------
 - Using real-space transfer function kernel.
   - Computing transfer function kernel...
 - Deconvolving density kernel...
 - Performing noise convolution on level  5 ...
 - Fetching kernel for level 5
   - Performing density convolution... (32, 32, 32)
 - Density calculation took 0.1516s with 4 threads.
 - Top grid mean density is off by -4.19006e-17, correcting...
-------------------------------------------------
 - Invoking unigrid FFT Poisson solver...
-------------------------------------------------
 - Invoking unigrid FFT Poisson solver...
 - Top grid mean density is off by 0.0175262, correcting...
=================================================
 - Wrote output file 'sapporo_cosmo'
     using plugin 'enzo'...
 - Done!
```

Set up *Enzo* parameter file

- MUSIC creates a basic but non-functional parameter file, `parameters.txt`.
- The user must add the relevant AMR, hydro, chemistry, etc. parameters.
- See `run/Cosmology/Hydro/AMRCosmology/AMRCosmology.enzo` for an example.
- For this workshop, we have provided a set of example parameter files for each lab exercise.

Run the simulation

- Move parameter file and initial conditions into the same directory (see `Adiabatic` directory)

- Run enzo!

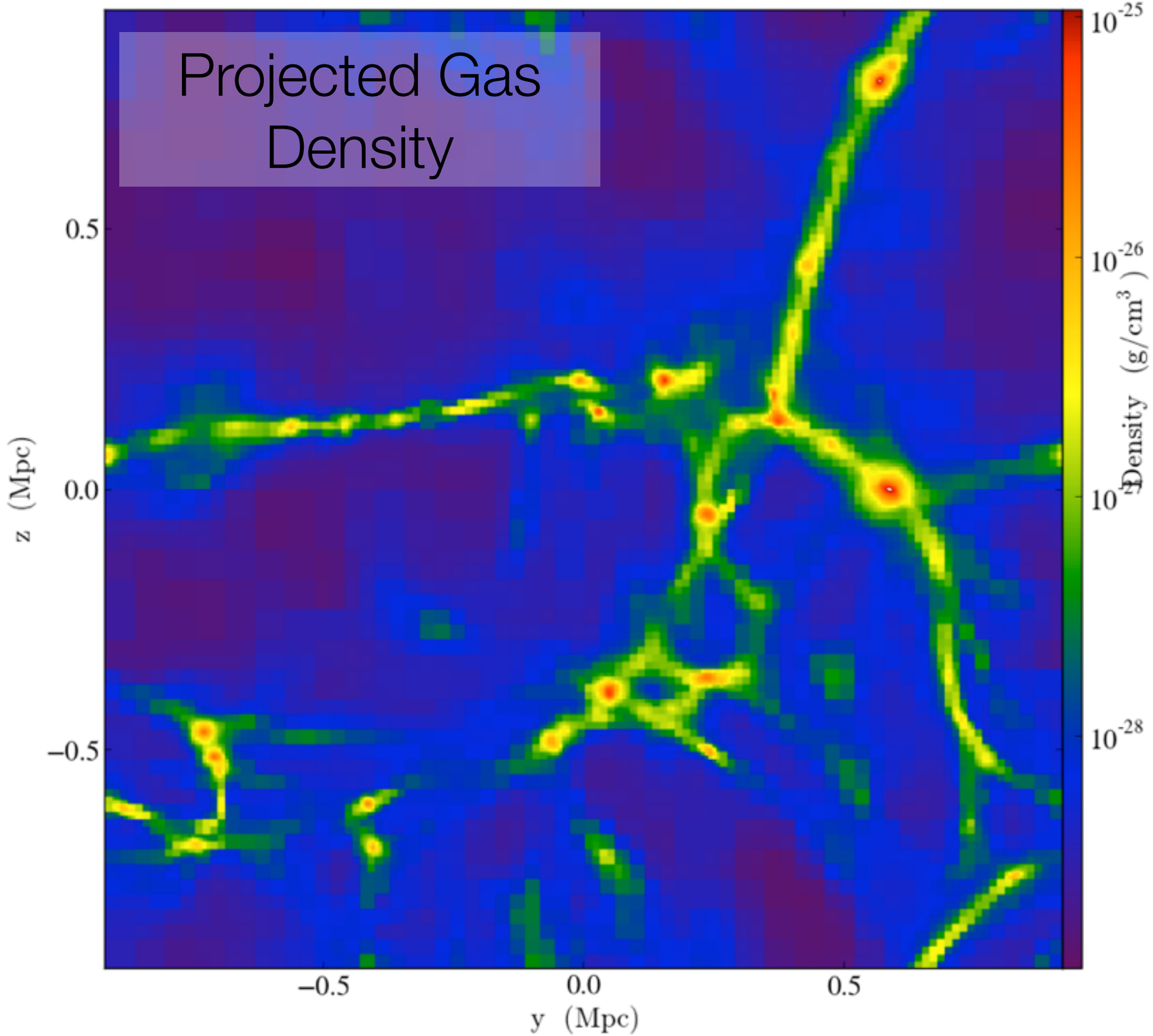
```
sapporo_cosmo% ./enzo.exe adiab.enzo
```

- Add the “-d” option for debugging information

- Add the “-r” option for restarting from a snapshot.

```
sapporo_cosmo% ./enzo.exe -d -r DD0010/DD0010
```

Projected Gas Density



Nested Grid Cosmology Simulations

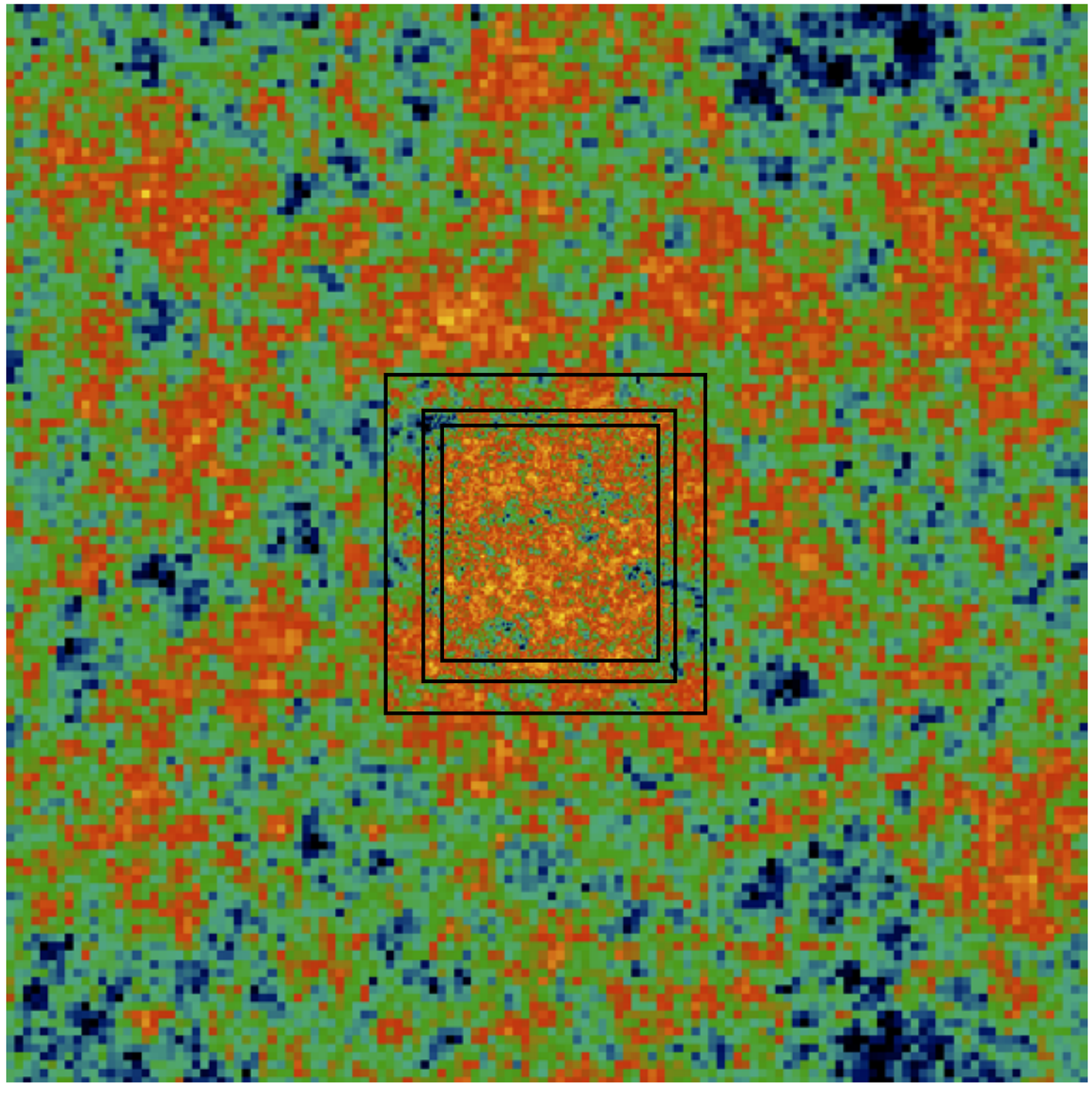
aka Zoom-in calculations

1. *N*-Body Simulation

2. Locate region of interest, e.g. most massive halo

3. Generate nested grid initial conditions

4. Run simulation!



N-body Simulation

- Usually run at the same resolution as the production level-0 resolution
- Create initial conditions with MUSIC and use_baryons=no
- Key parameters

```
CosmologySimulationOmegaBaryonNow       = 0.0
CosmologySimulationOmegaCDMNow          = 0.3175
CosmologySimulationParticleVelocity1Name = ParticleVelocities_x
CosmologySimulationParticleVelocity2Name = ParticleVelocities_y
CosmologySimulationParticleVelocity3Name = ParticleVelocities_z
CosmologySimulationParticleDisplacement1Name = ParticleDisplacements_x
CosmologySimulationParticleDisplacement2Name = ParticleDisplacements_y
CosmologySimulationParticleDisplacement3Name = ParticleDisplacements_z
CosmologySimulationNumberOfInitialGrids = 1
CosmologySimulationCalculatePositions = 1
```

```
RefineBy                    = 2    // refinement factor
#CellFlaggingMethod          = 2 4    // use baryon and DM mass
CellFlaggingMethod          = 4    // use DM mass for refinement
MinimumEfficiency           = 0.4  // fraction efficiency
MinimumOverDensityForRefinement = 4. // times the initial dens
```

Halo finding

- On the last output, we need to find the cosmological halos.

- Enzo has a built-in halo finder. Run it with

```
sapporo_cosmo% ./enzo.exe -F -d -r DD0020/DD0020
```

- Will put the halo list sorted by mass in the FOF/ directory.

```
# Time      = 15.921139
# Redshift = 12.802003
# Number of halos = 83
#
# Column 1.  Center of mass (x)
# Column 2.  Center of mass (y)
# Column 3.  Center of mass (z)
# Column 4.  Halo number
# Column 5.  Number of particles
# Column 6.  Halo mass [solar masses]
# Column 7.  Virial mass [solar masses]
# Column 8.  Stellar mass [solar masses]
# Column 9.  Virial radius (r200) [kpc]
# Column 10. Mean x-velocity [km/s]
# Column 11. Mean y-velocity [km/s]
# Column 12. Mean z-velocity [km/s]
# Column 13. Velocity dispersion [km/s]
# Column 14. Mean x-angular momentum [Mpc * km/s]
# Column 15. Mean y-angular momentum [Mpc * km/s]
# Column 16. Mean z-angular momentum [Mpc * km/s]
# Column 17. Spin parameter
#
# datavar lines are for partiview.  Ignore them if you're not partiview.
#
datavar 0 halo_number
datavar 1 number_of_particles
datavar 2 halo_mass
datavar 3 virial_mass
datavar 4 stellar_mass
groups_00137.dat lines 1-29/122 4%
```

Terminal — 80×30 — ⌘3

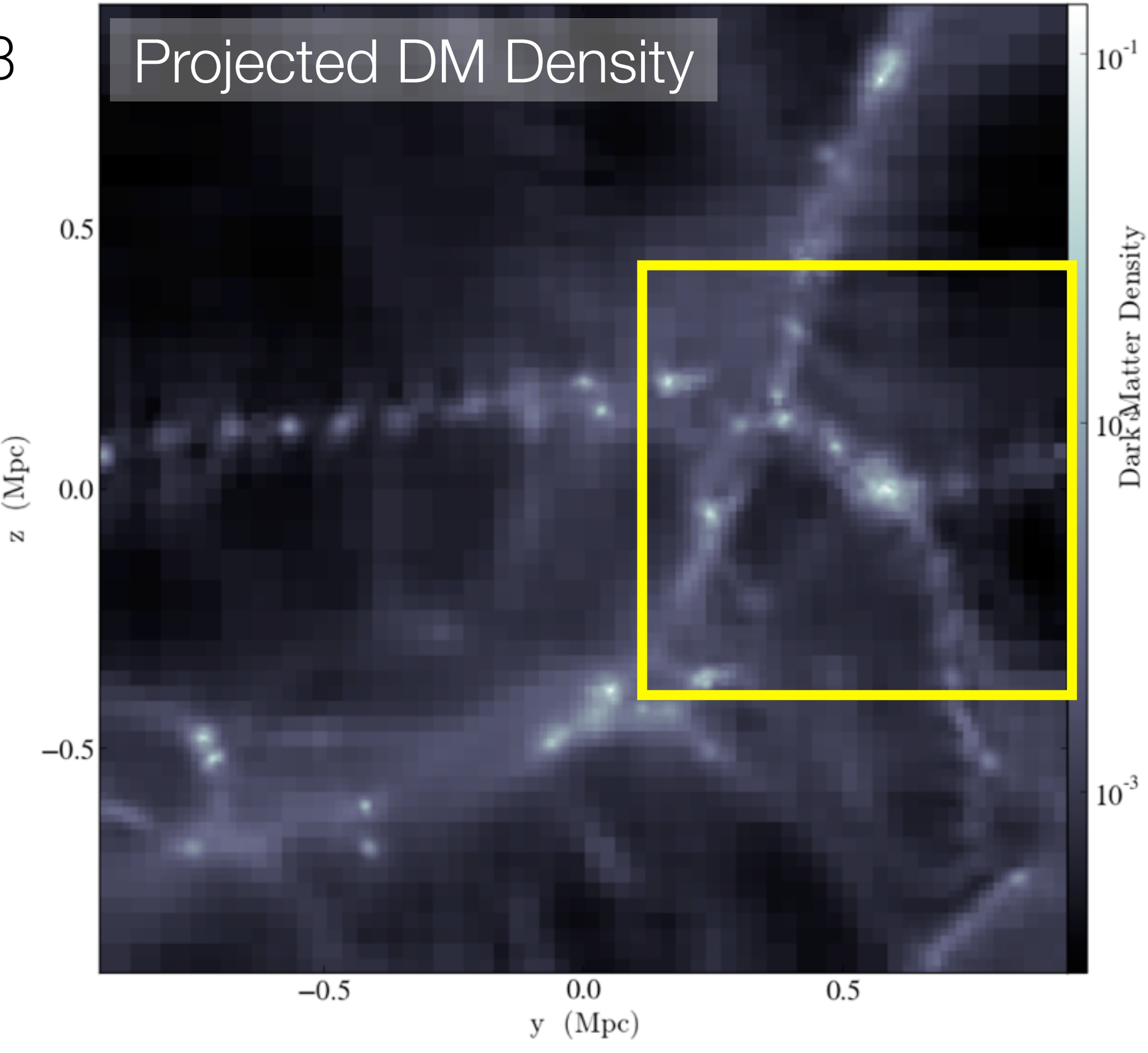
```
datavar 0 halo_number
datavar 1 number_of_particles
datavar 2 halo_mass
datavar 3 virial_mass
datavar 4 stellar_mass
datavar 5 virial_radius
datavar 6 x_velocity
datavar 7 y_velocity
datavar 8 z_velocity
datavar 9 velocity_dispersion
datavar 10 x_angular_momentum
datavar 11 y_angular_momentum
datavar 12 z_angular_momentum
datavar 13 spin
```

Position

Mass

```
 0.76364088   0.29754913   0.37248853            0         2060    571700.88
 447925.16            0   0.11323795   0.84548277  -0.41380891    1.0349354
5.341588 2.408008e-05 3.4065459e-05 -6.1198812e-06   0.11691254
  0.46177065   0.61586523    0.6905852            1         1390    385758.56
 372992.47            0   0.10630372    2.0898418   0.23426482   -1.659736    4
.6082692 5.5966311e-06 1.264129e-05 -3.7420041e-06  0.041144241
  0.84373128   0.67770523   0.27474043            2          415     115172.2
 70213.453            0  0.060892154  -0.90891731  -0.64614326    2.0012767    2
.8843138 2.717258e-06 -1.8763769e-06 -7.576411e-06  0.078942753
  0.51242203   0.62618536   0.67746228            3          372    103238.98
 95745.844            0  0.067207284  -0.48392525   0.49669054   -2.3942778    2
.9666159 7.2602943e-06 3.2799553e-06 -2.7749318e-06  0.060778547
  0.77068067   0.30860633   0.33854923            4          330        91583
 81314.617            0   0.06338729   -1.6947181  -0.76077771    3.8264091    2
groups_00137.dat lines 25-44/122 11%
```

z = 3


Projected DM Density
0.5
0.0
−0.5
z (Mpc)
−0.5
0.0
0.5
y (Mpc)
10^{-1}
10^{-2}
10^{-3}
Dark Matter Density

Nested Grid Initial Conditions with MUSIC

- Find halo of interest.

- We want to re-sample the Lagrangian volume of this region with a higher resolution.

- We can use yt to find this volume. Sample script can be found at

- https://bitbucket.org/MatthewTurk/yt.lagrangian_volume

    - All particles have unique IDs, so we can find the particles in some region in the final output.

    - Then we find the bounding box of the same particles in the initial output.

MUSIC: Nested Grid Parameters

```
[setup]
boxlength       = 5
zstart          = 50
levelmin        = 5
levelmin_TF     = 5
levelmax        = 7
padding         = 2
overlap         = 0
ref_center      = 0.3, 0.81, 0.5
ref_extent      = 0.25, 0.25, 0.25
align_top       = yes
baryons         = no
use_2LPT        = yes
use_LLA         = yes
periodic_TF     = yes
```

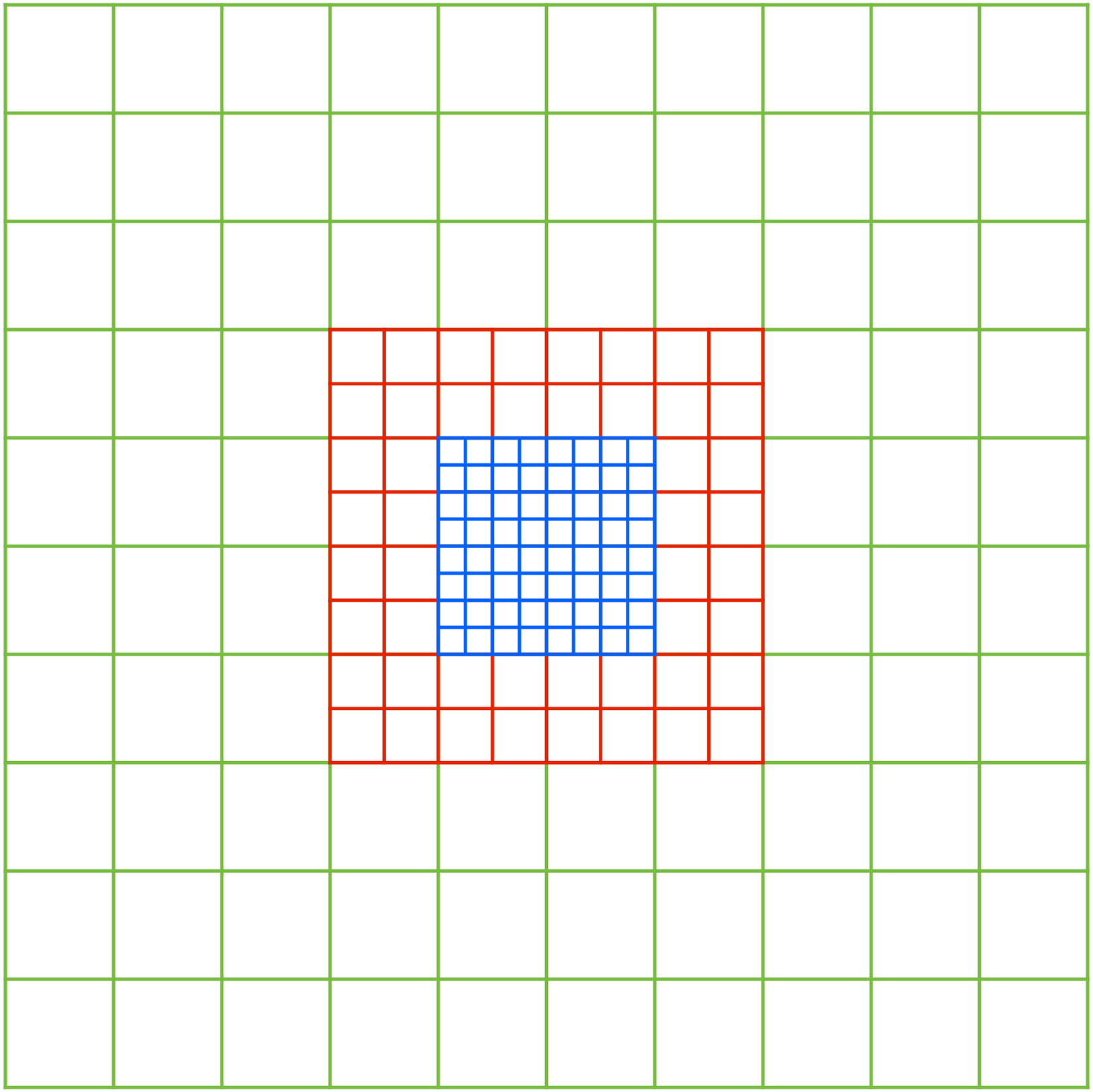
Resolution of the finest nested grid.
Effective resolution = $2^7 = 128$

Center of the Lagrangian volume
Width of the Lagrangian volume

```
[random]
seed[5]         = 201310160
seed[6]         = 301310160
seed[7]         = 401310160
```

Optional: New random seeds for nested grids.

padding = 2



with MUSIC

```
sapporo_cosmo% ~/codes/music/MUSIC sapporo_cosmo_nested.conf

                                                this is mus

 - Opening log file 'sapporo_cosmo_nest
 - Selecting transfer function plug-in
 - starting at a=0.0196078
 - Selecting region generator plug-in '
 - refinement region is 'box', w/ bound
        left = [0.175000,0.685000,0.373
       right = [0.425000,0.935000,0.626
-----------------------------------------
 - Domain will be shifted by (6, -9, 0)

 - Grid structure:
    Level   5 :   offset = (    0,
                    size = (   32,
    Level   6 :   offset = (    9,
                    size = (   26,
    Level   7 :   offset = (    2,
                    size = (   40,
-----------------------------------------
 - Finest level :
                   extent =  1562.5 x 1
                mtotgrid =  3.36123e+1
           particle mass =  5.25192e+0
                      dx =  39.0625 h-
-----------------------------------------
 - Selecting output plug-in 'enzo'...
=========================================
```

```
- Top grid mean density is off by -6.5921e-05, correcting...
- Global density extrema:
-   minimum: delta=-0.730190 at (0.449219,0.378906,0.425781) (level=7)
-        shifted back at (0.261719,0.660156,0.425781)
-   maximum: delta=0.986598 at (0.582031,0.621094,0.488281) (level=7)
-        shifted back at (0.394531,0.902344,0.488281)
- Density extrema on finest level:
-   minimum: delta=-0.730190 at (0.449219,0.378906,0.425781)
-        shifted back at (0.261719,0.660156,0.425781)
-   maximum: delta=0.986598 at (0.582031,0.621094,0.488281)
-        shifted back at (0.394531,0.902344,0.488281)
-------------------------------------------------------------
- Invoking multi-grid Poisson solver...
- Converged in 5 steps to 3.89154e-06
- Poisson solver took 0.742935s with 4 threads.
- Computing 2LPT term....
- Solving 2LPT Poisson equation
------------------------------------------------------------
- Invoking multi-grid Poisson solver...
- Converged in 5 steps to 2.05267e-06
- Poisson solver took 0.775615s with 4 threads.
  - Performing hybrid Poisson step... (80, 80, 80)
- velocity component 0 : sigma = 3.0004
- velocity component 1 : sigma = 1.79269
- velocity component 2 : sigma = 1.84676
============================================================
  COMPUTING DARK MATTER DISPLACEMENTS
------------------------------------------------------------
  - Performing hybrid Poisson step... (80, 80, 80)
============================================================
- Wrote output file 'IC.2'
    using plugin 'enzo'...
- Done!
```

1 Nested Grid

Unigrid
Smoothed DM Density Slice

Setting up the Nested Grid Parameter File

```
CosmologySimulationNumberOfInitialGrids  = 3
CosmologySimulationGridDimension[1]      =          26          26          26
CosmologySimulationGridLeftEdge[1]       =     0.28125     0.34375     0.28125
CosmologySimulationGridRightEdge[1]      =      0.6875        0.75      0.6875
CosmologySimulationGridLevel[1]          = 1
CosmologySimulationGridDimension[2]      =          40          40          40
CosmologySimulationGridLeftEdge[2]       =      0.3125       0.375      0.3125
CosmologySimulationGridRightEdge[2]      =       0.625      0.6875       0.625
CosmologySimulationGridLevel[2]          = 2
#
# region allowed for further refinement
#
RefineRegionLeftEdge                     = 0.3125 0.375 0.3125
RefineRegionRightEdge                    = 0.625 0.625 0.625
```

Nested Grid Dimensions and Boundaries

Refine region restricts AMR grids within these coordinates. Should equal or be smaller than the finest nested grid

Overdensity criteria

- **Very important:** Must divide the baryon and DM mass refinement criteria by a factor of 8^n, where n := maximum initial level
- e.g. Refine on an overdensity of 3 with 2 initial grids → 3.0/64 = 0.046875
- If not done, the hierarchy will remain under-resolved, compared with the desired results.

```
#
#  set grid refinement parameters
#
StaticHierarchy           = 0    // dynamic hierarchy
MaximumRefinementLevel    = 8
MaximumGravityRefinementLevel    = 8
MaximumParticleRefinementLevel   = 8
RefineBy                  = 2    // refinement factor
CellFlaggingMethod        = 2 4 6   // use baryon mass for refinement
#CellFlaggingMethod        = 4    // use DM mass for refinement
MinimumEfficiency         = 0.3  // fraction efficiency
MinimumOverDensityForRefinement = 0.5 0.5 // times the initial density refers to top g
rid: divide by 8 for each additional level
MinimumMassForRefinementLevelExponent = -0.2 0.0
```

Run enzo! (see directory: `DarkMatter-Nested`)

- Running on a single core, outputting standard output to display.

```
sapporo_cosmo% ./enzo.exe -d nbody_nested.enzo
```

- Running on 4 cores, piping standard output to the file, estd.out.

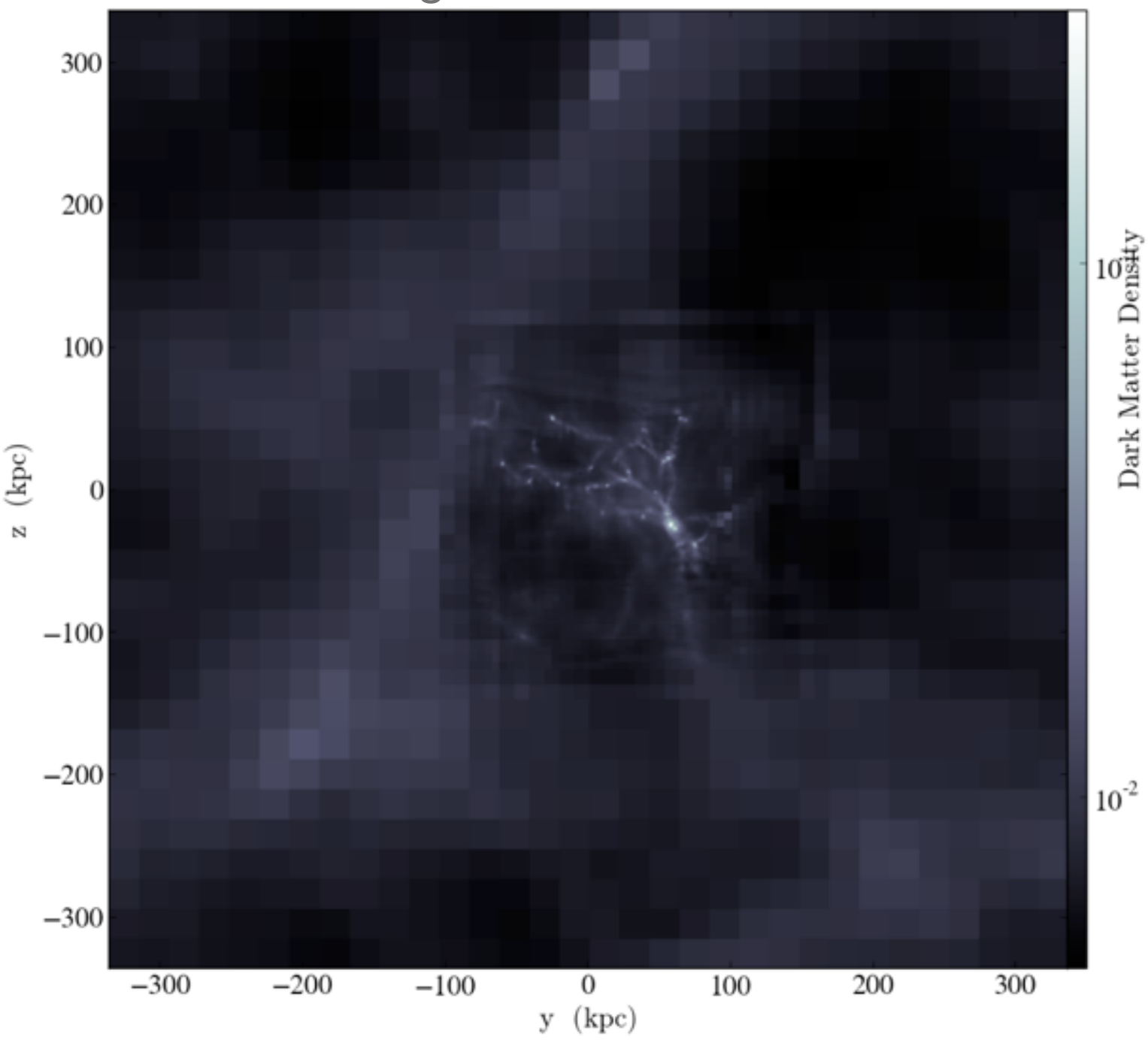
```
sapporo_cosmo% mpirun -n 4 ./enzo.exe -d nbody_nested.enzo >& estd.out
```

- This simulation takes significantly longer than the simulation with no nested grids. On my laptop, it took 20 minutes on 2 cores.

Differences between simulations

- Same spatial resolution but higher mass resolution!

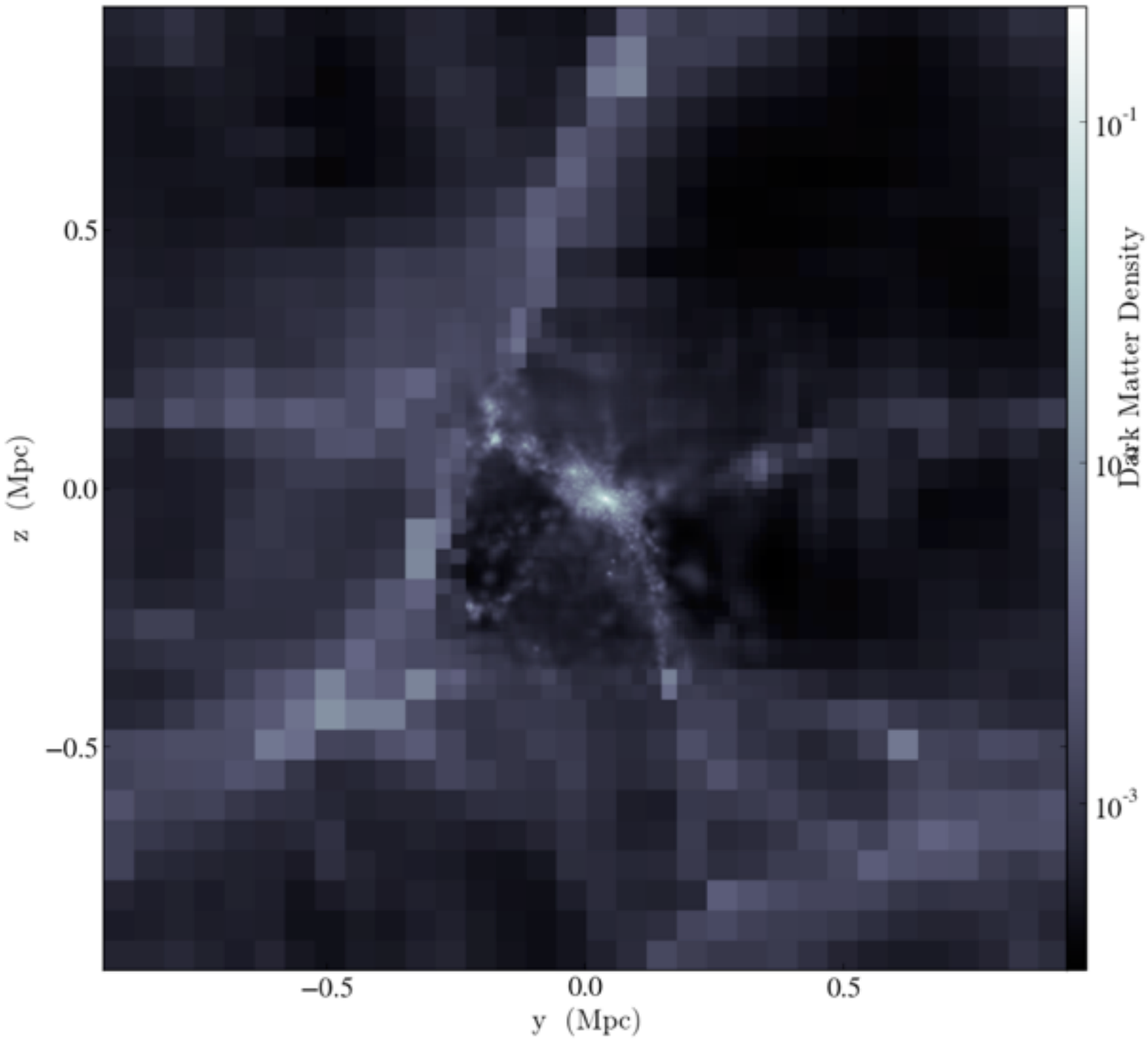
z = 10



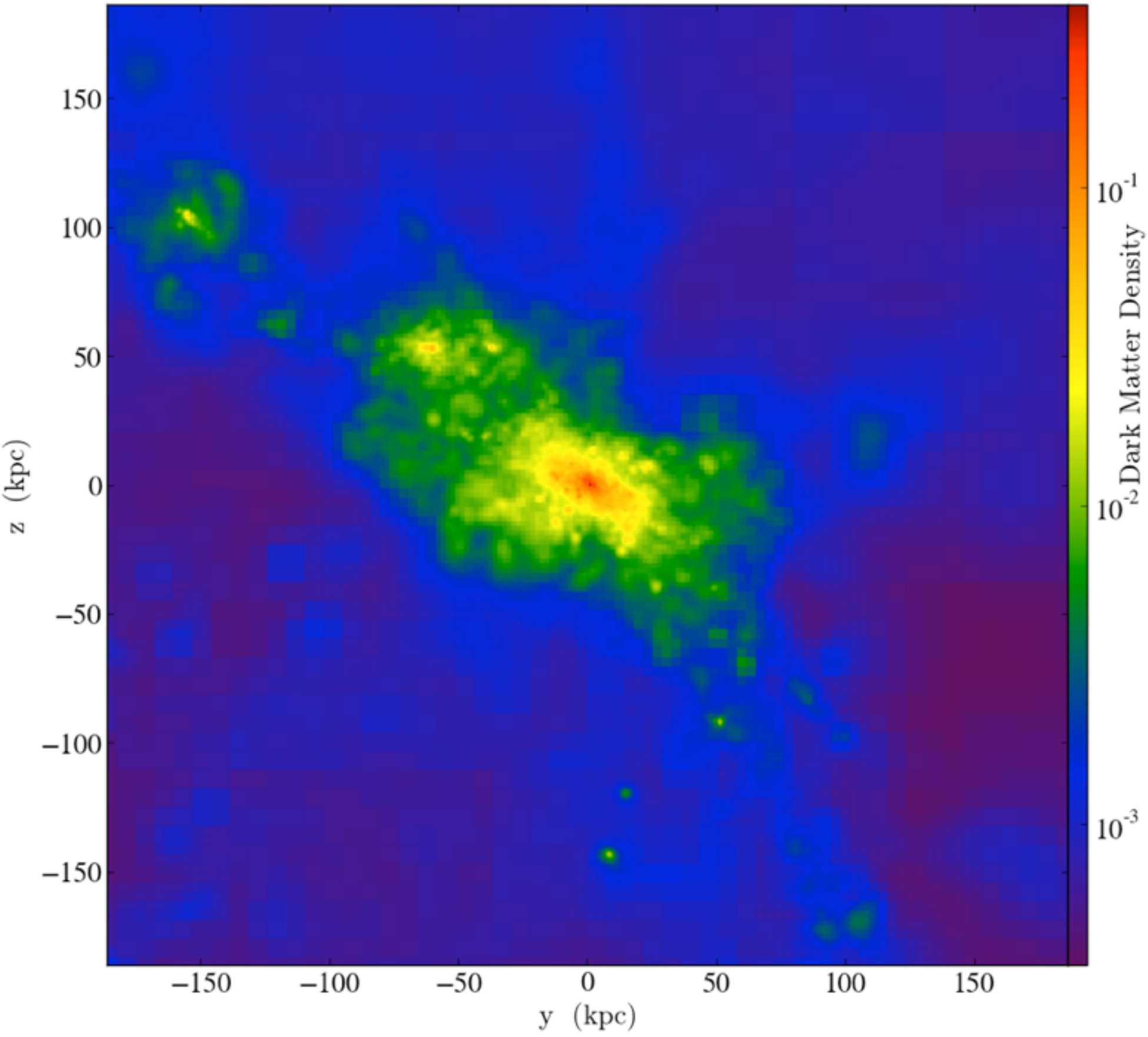
Differences between simulations

- Same spatial resolution but higher mass resolution!

z = 3



z = 3


150
100
50
0
−50
−100
−150
z (kpc)
−150
−100
−50
0
50
100
150
y (kpc)
10^{-1}
10^{-2}
10^{-3}
Dark Matter Density

Different chemistry and cooling models

- **MultiSpecies == 0:** Equation of state with an adiabatic index ϒ
    - **RadiativeCooling == 1:** Cooling from a tabulated cooling curve (copy input/cool_rates.in to working directory)

Non-equilibrium chemistry

- **MultiSpecies == 1:** 6-species (H, H^+, He, He^+, He^{++}, e^-)
- **MultiSpecies == 2:** 9-species (H, H^+, He, He^+, He^{++}, e^-, H^-, H_2, H_2^+); i.e. +molecular hydrogen cooling
- **MultiSpecies == 3:** 12-species (H, H^+, He, He^+, He^{++}, e^-, H^-, H_2, H_2^+, D, D^+, HD); +HD cooling

Different chemistry and cooling models

- **MetalCooling == 1:** Tabulated cooling table from Glover & Jappsen (2007) & Sutherland & Dopita (1993) – input/metal_cool.dat.
  - $T < 10^4$ K: fine-structure metal-line cooling
  - $T > 10^4$ K: collisional ionization equilibrium
- **MetalCooling == 2:** Explicitly calculated metal cooling rates (Cen)
- **MetalCooling == 3:** Tabulated cooling table from CLOUDY (Smith et al. 2008)

3 more parameters for **large** nested grid runs

- i.e. >256^3 nested grids
- `ParallelRootGridIO = 1`: Root grids are tiled before I/O, and each processor reads/writes their own tile
- `PartitionNestedGrids = 1`: Same thing as above but for nested grids
- `UnigridTranspose = 2`: More parallel-efficient transpose for the FFT done for the gravity solver.

Summary

You should now know **how to:**

- Create cosmological unigrid and nested grid initial conditions with MUSIC. For more, see
  - http://www.phys.ethz.ch/~hahn/MUSIC/index.html
  - https://bitbucket.org/ohahn/music
  - https://bitbucket.org/ohahn/music/downloads/MUSIC_Users_Guide.pdf
- Run cosmological regular and nested grid runs (*N*-Body and *N*-Body + gas)
  - http://enzo.readthedocs.org/en/latest/tutorials/RunCosmologySimulation.html
  - However, these instructions are for the *inits* initial conditions generator that is included with Enzo.