

INTRODUCTION TO ENZO

Britton Smith

OUTLINE

- I. Code Overview
- II. Obtaining
- III. Building
- IV. Running
- V. The Enzo Community

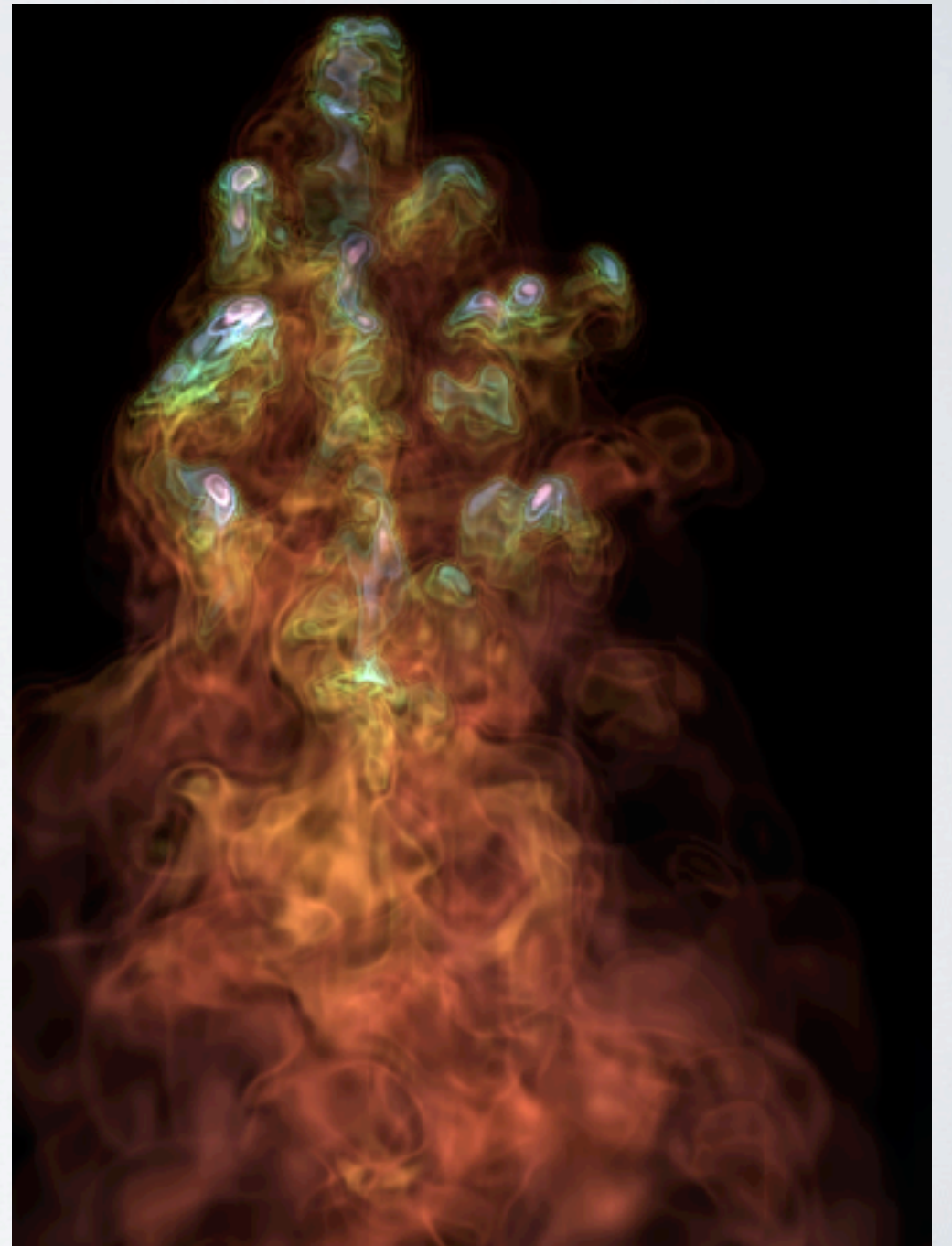


Image: Devin Silvia

WHAT IS ENZO?

Enzo is a cosmological,
adaptive-mesh refinement,
hydrodynamics + N-body
simulation code.

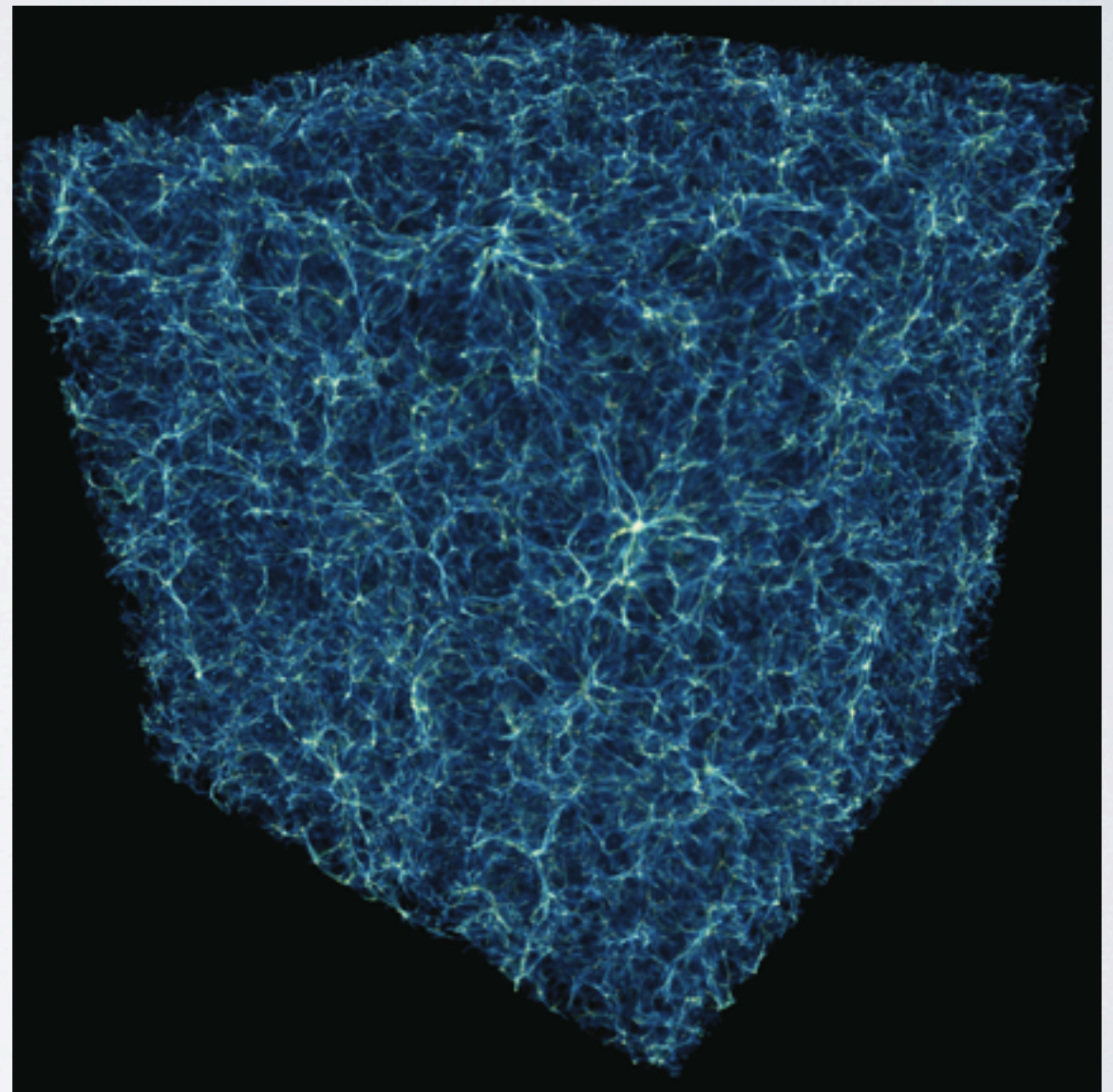
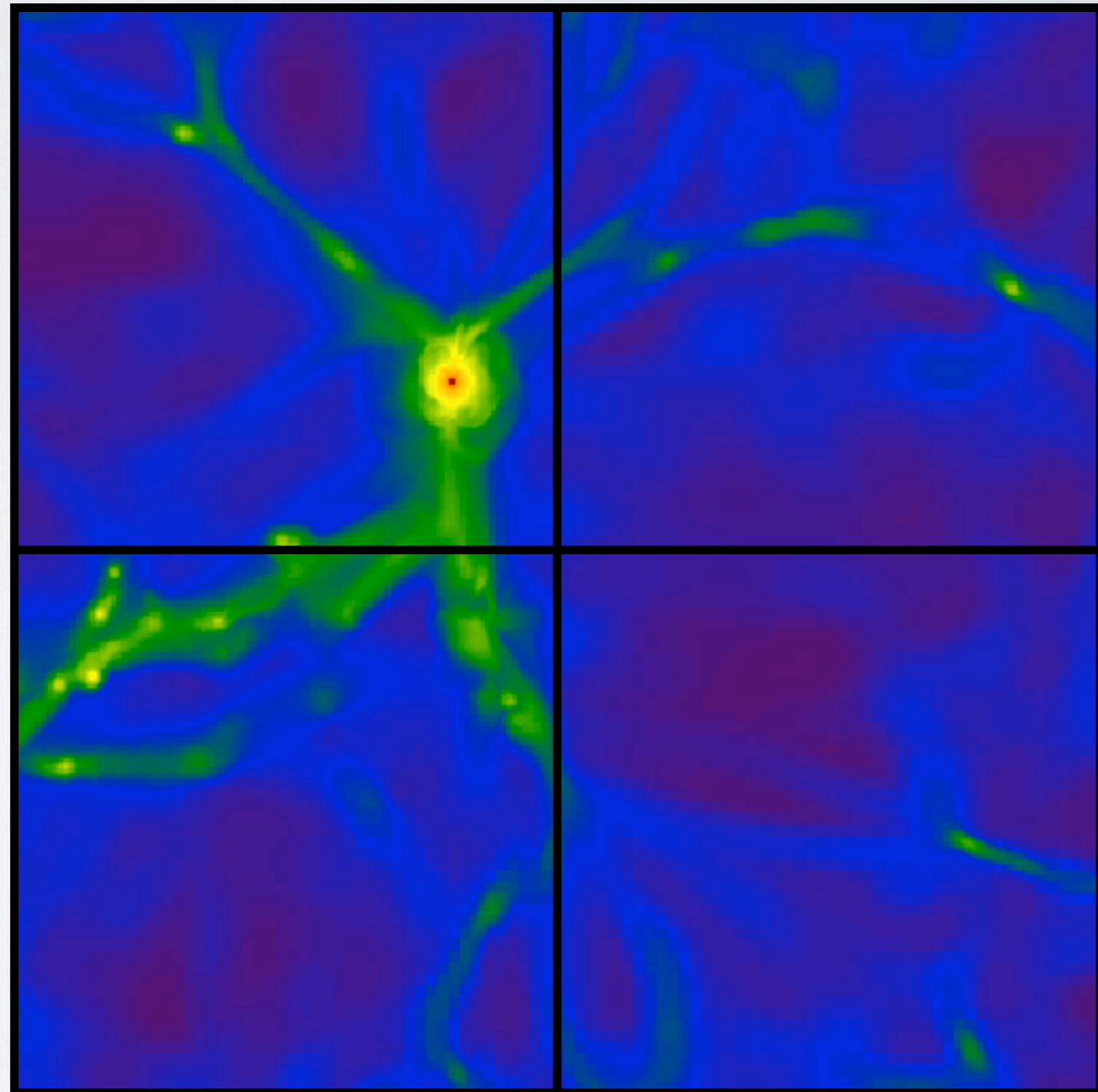


Image: Eric Hallman, Brian O'Shea

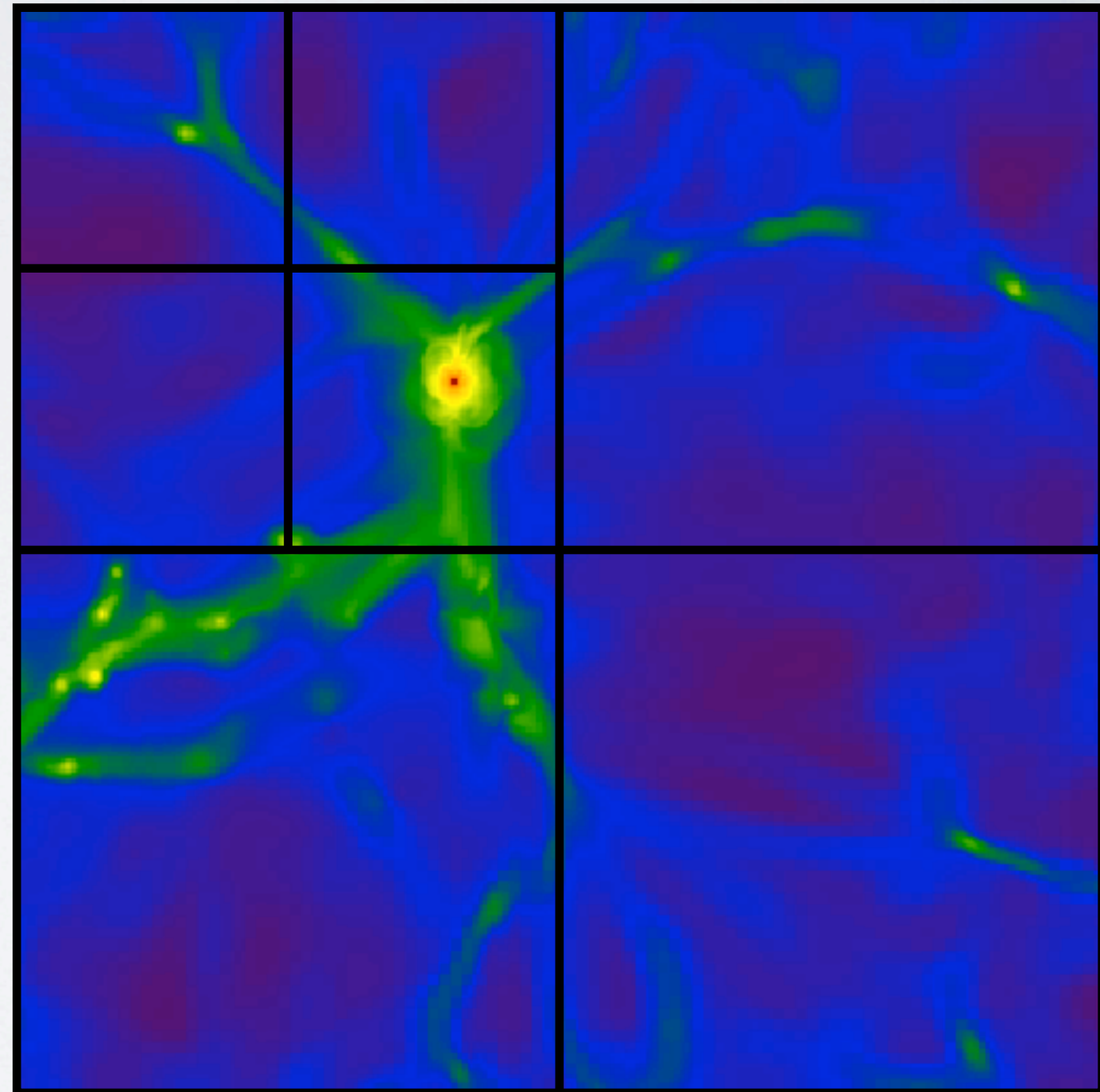
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



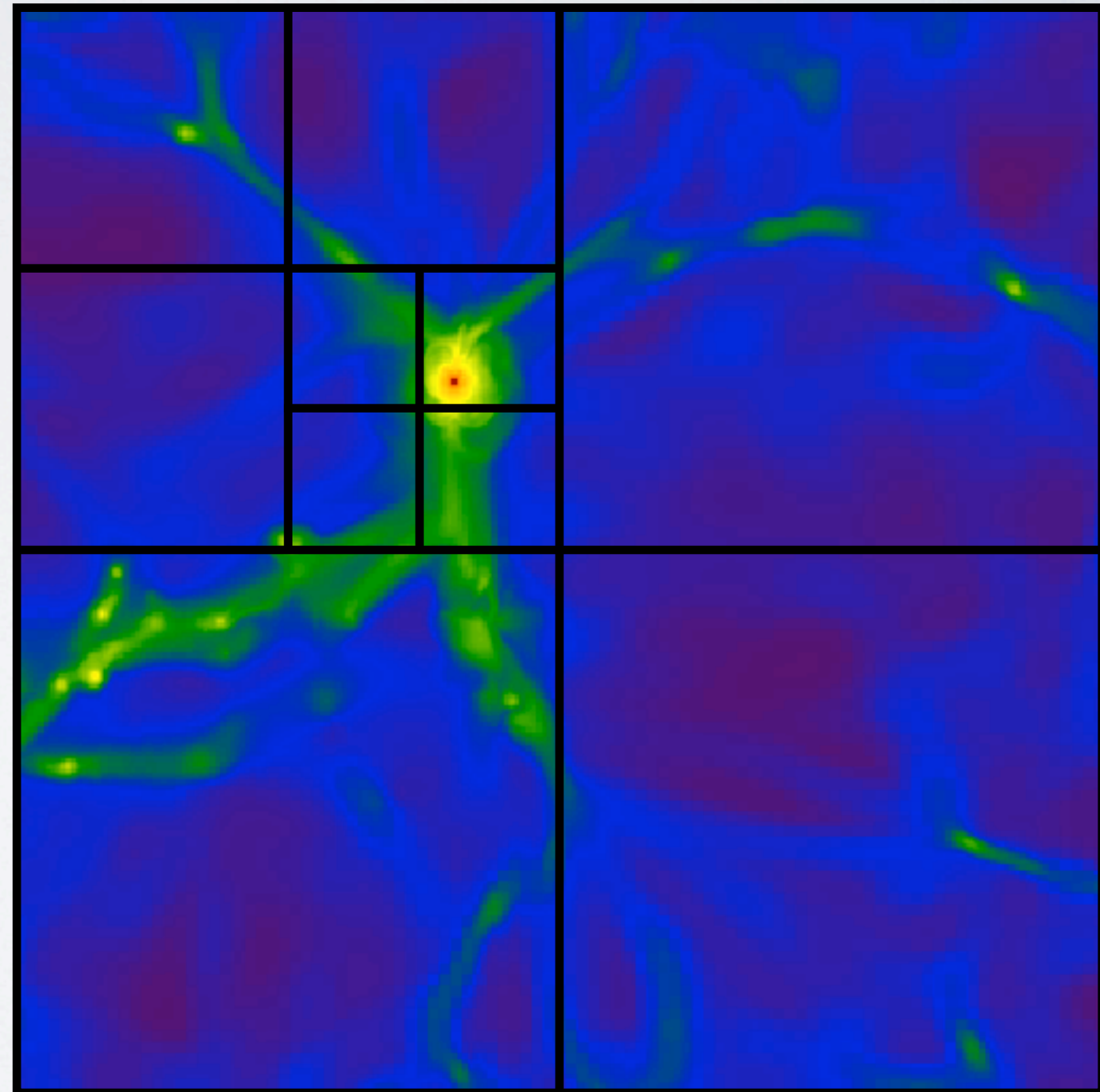
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



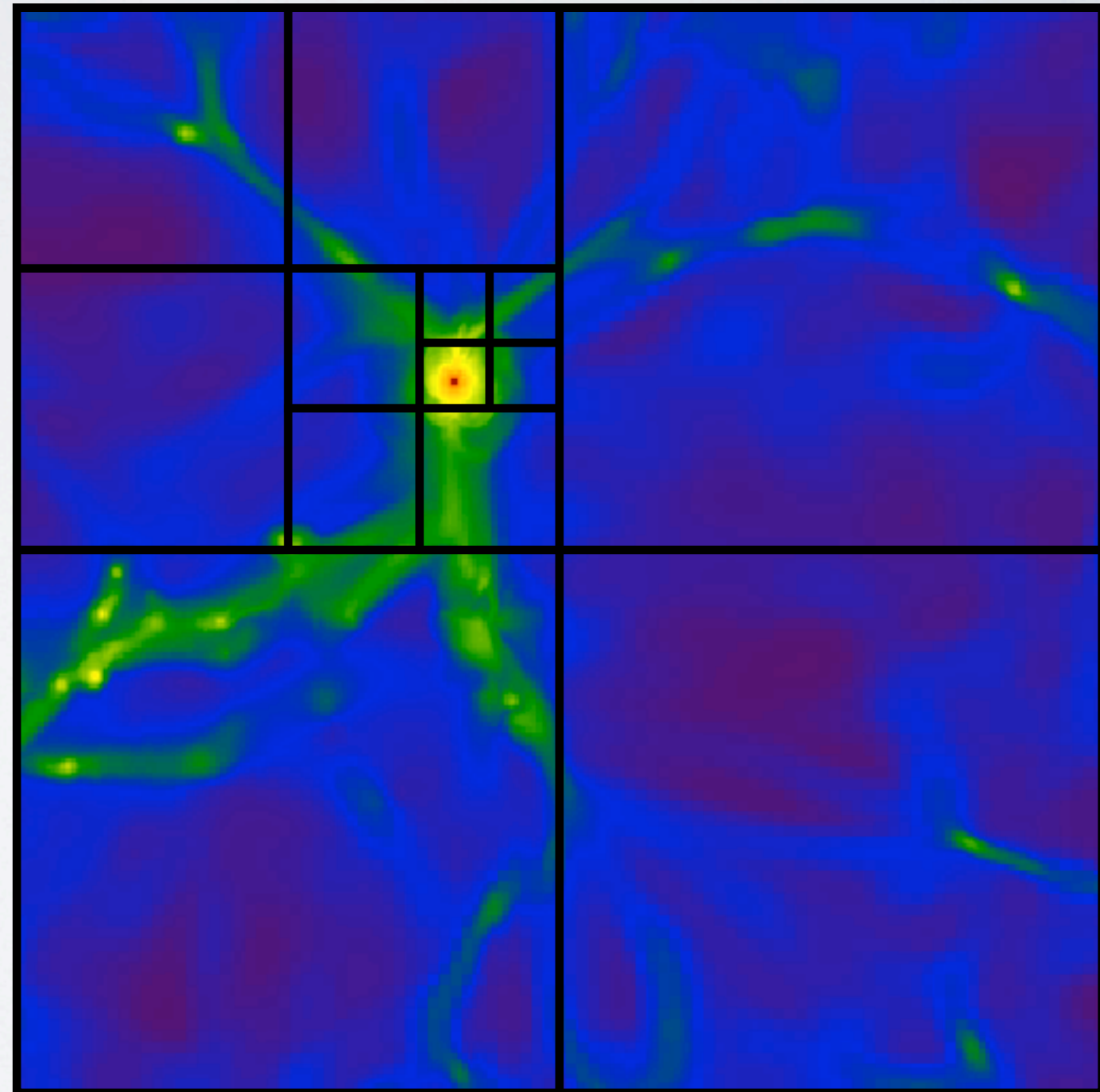
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



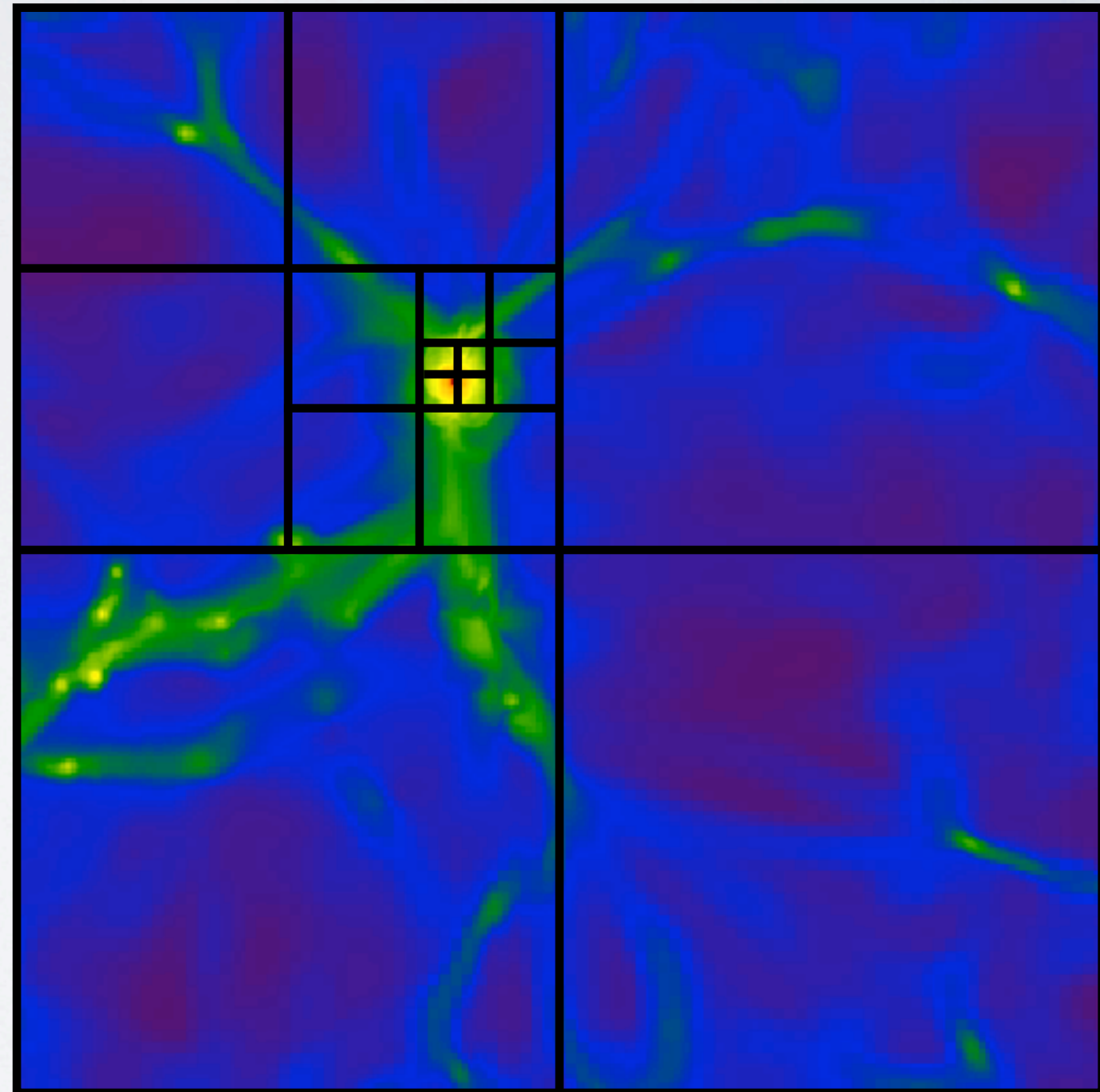
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



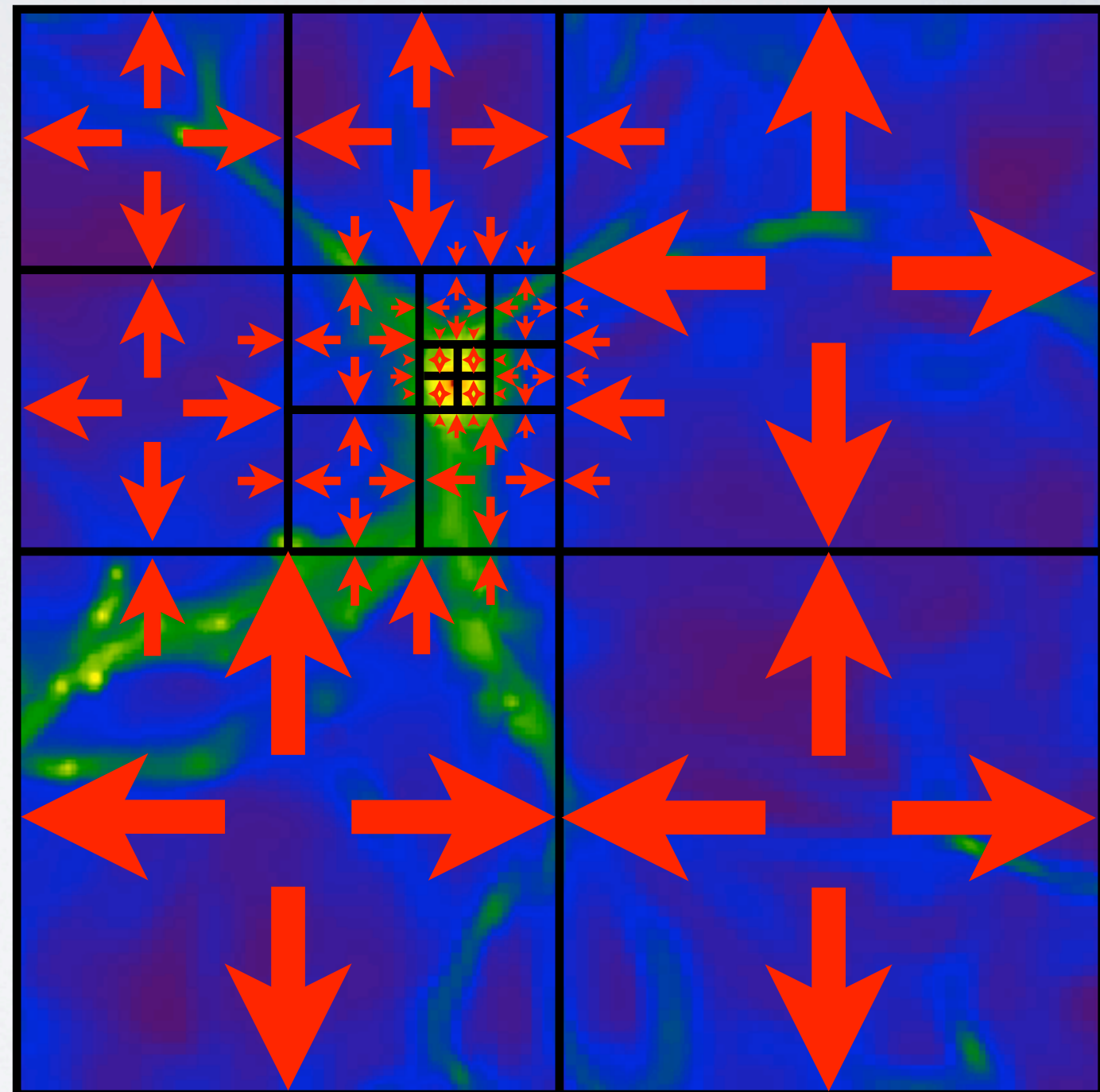
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



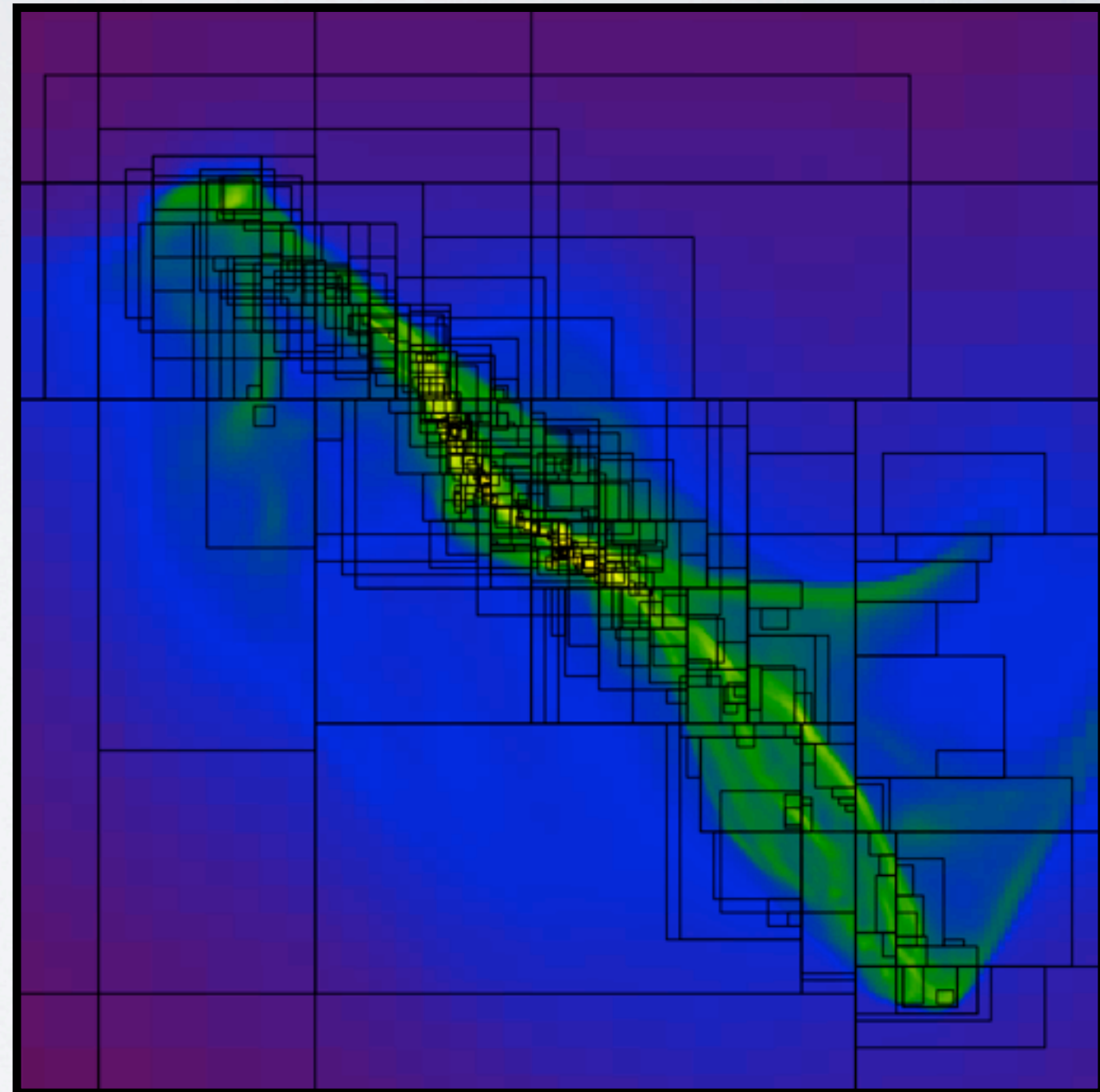
AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria



AMR: ADAPTIVE-MESH REFINEMENT

- create and destroy grid patches dynamically (block-structured)
- grids at multiple resolutions
- multiple refinement criteria:
 - density (gas or dark matter)
 - gradients, shocks
 - cooling time
 - Jeans length
 - refine regions around particles
- easy to create new criteria





Credit: Matthew Turk

GRAVITY

- dark matter treated as collision-less particles
- adaptive particle-mesh method
 - solve Poisson eqn.: $\nabla^2\phi = 4\pi G\rho$
 - particles interpolated onto grid to create density field, then gas densities added
 - multigrid relaxation for refined grids
- advantage: very fast!
- disadvantage: force res. is $2\Delta x$ (not great)

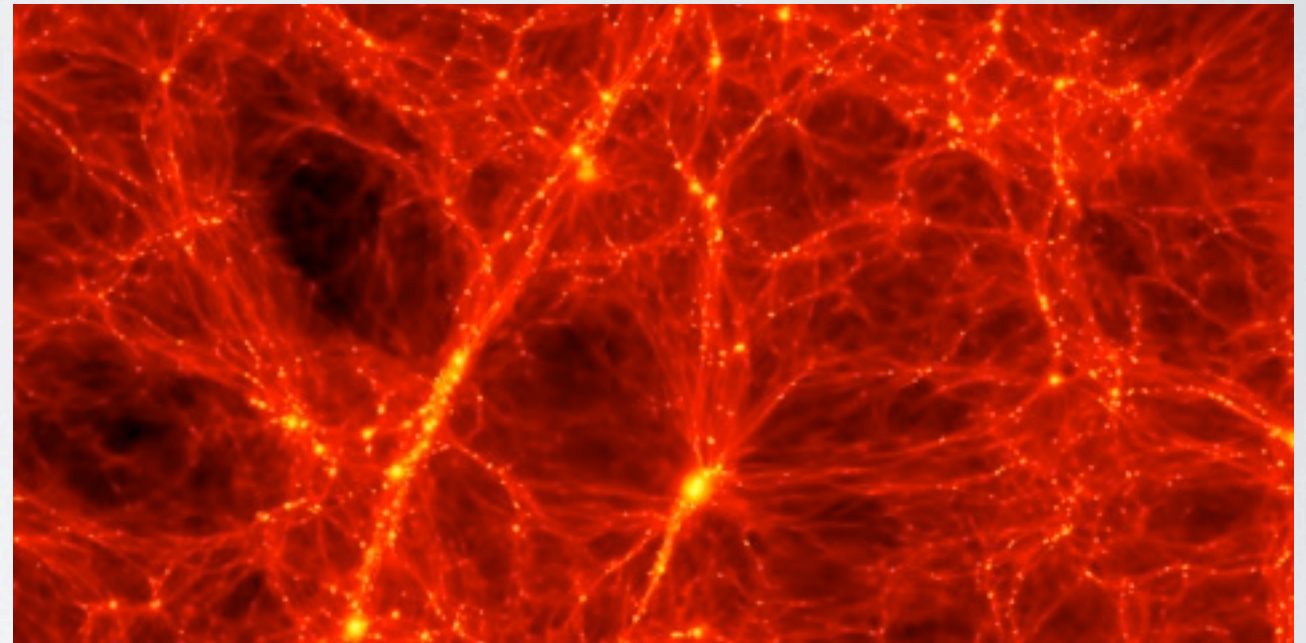


Image: Michael Norman et. al.

HYDRODYNAMICS

Multiple Hydro Methods

- Piecewise Parabolic Method
 - fits state variables to 3rd order parabolic
 - nonlinear Riemann solver for excellent shock capturing
 - can be unstable with cosmology or cooling (now more stable thanks to improvements by John Wise)
- Zeus
 - less accurate, more diffusive
 - extremely robust (excellent for cosmology)

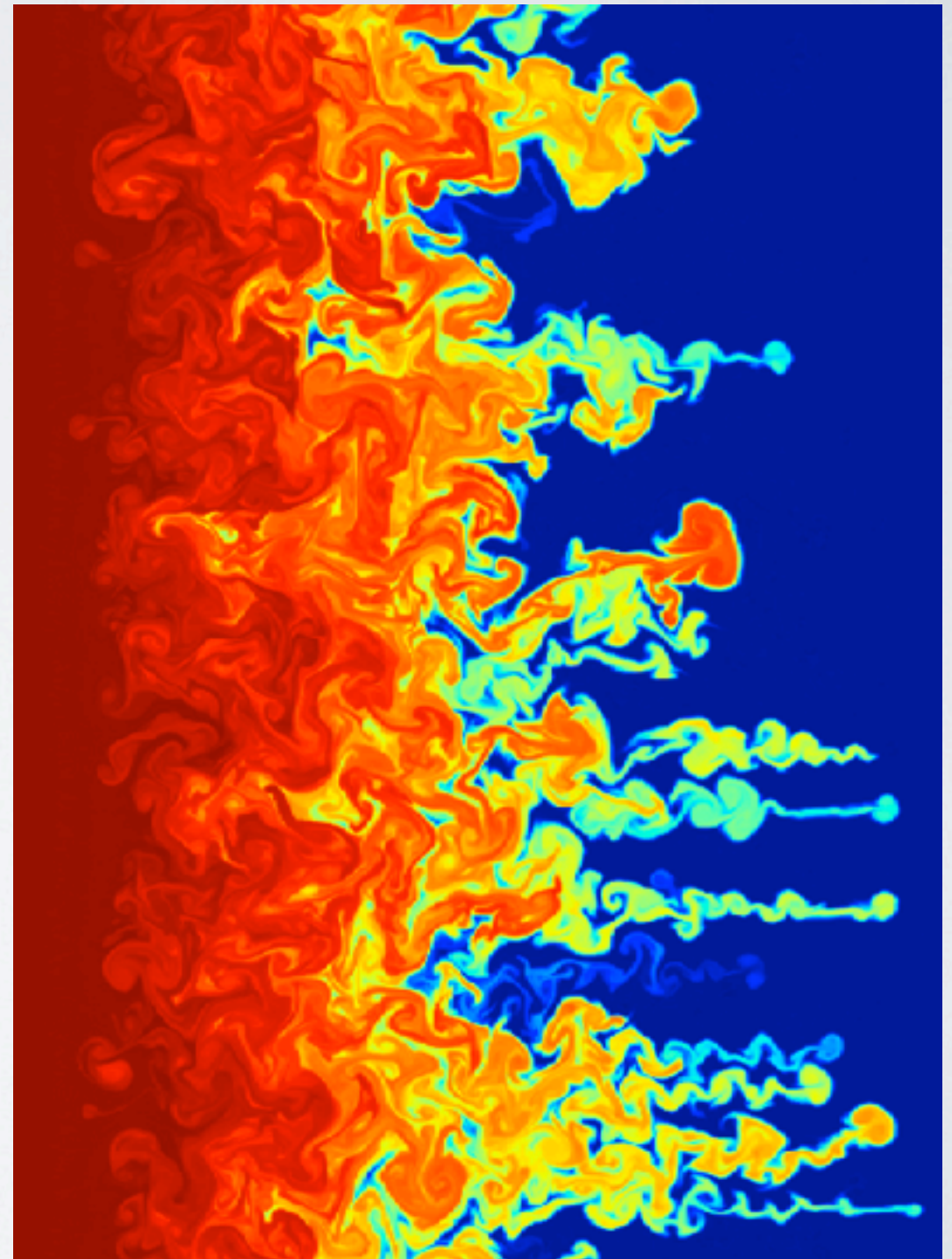


Image: Nick Earl

HYDRODYNAMICS

Multiple Hydro Methods

- MUSCL
 - 2nd order accurate Godunov solver
 - 2nd order Runge-Kutta time integration
 - multiple Riemann solvers and interpolation methods available
- MHD using MUSCL
 - uses MUSCL framework above
 - hyperbolic divergence cleaning method to ensure $\nabla \cdot \mathbf{B} = 0$

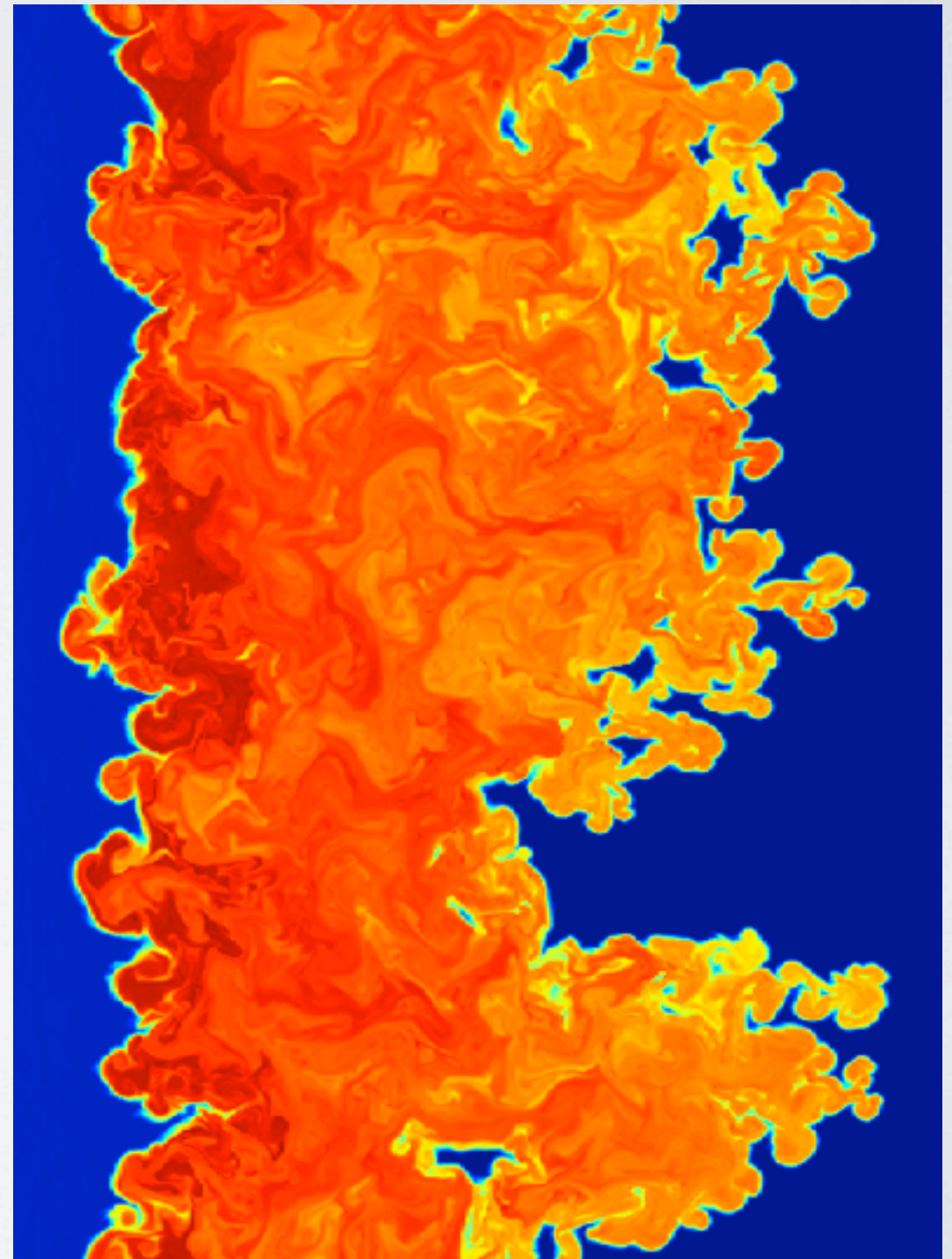


Image: Nick Earl

RADIATIVE TRANSFER

Two Rad.Transfer Methods

- Adaptive Ray Tracing
 - radiation from discrete sources (star and black hole particles)
 - adaptive ray splitting and merging
 - fully coupled to chemistry network
- Flux Limited Diffusion
 - treats radiation like a fluid
 - couple to atomic chemistry
 - highly scalable
 - unigrid and AMR versions available

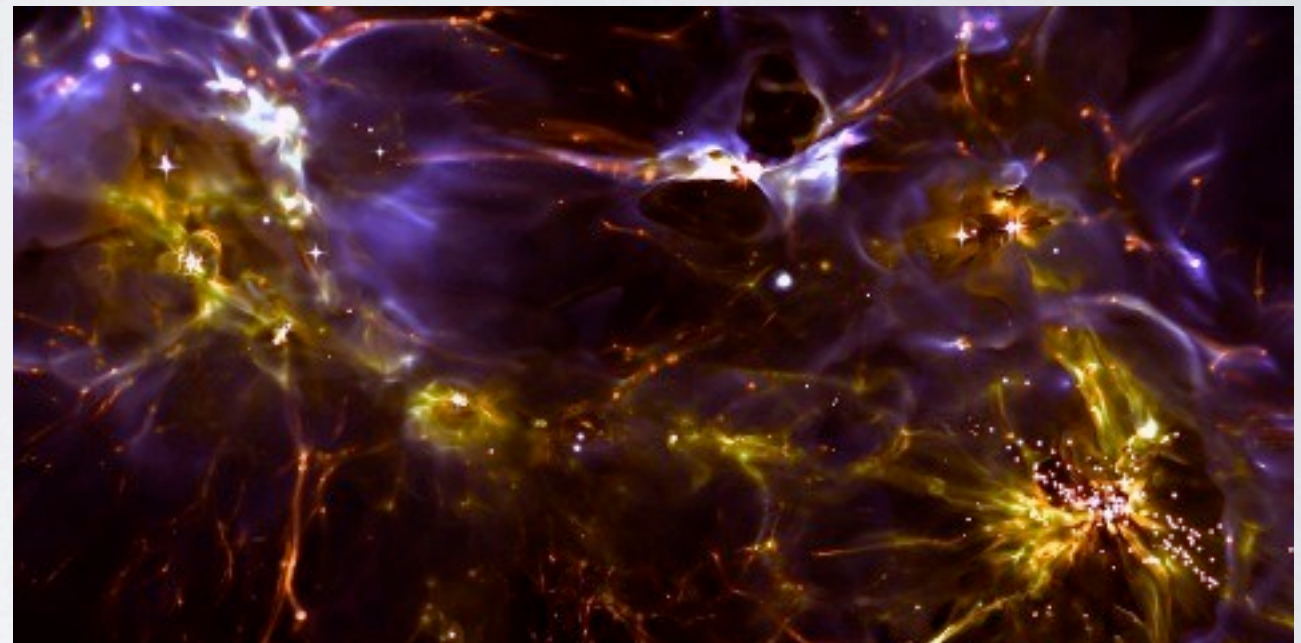


Image: John Wise

RADIATION BACKGROUNDS

- spatially smooth, time-dependent radiation fields
- UV metagalactic, ionizing backgrounds mimic effects of photo-heating and ionization during Reionization
- Lyman-Werner soft UV fields represent radiation from first stellar sources and photo-dissociate H_2

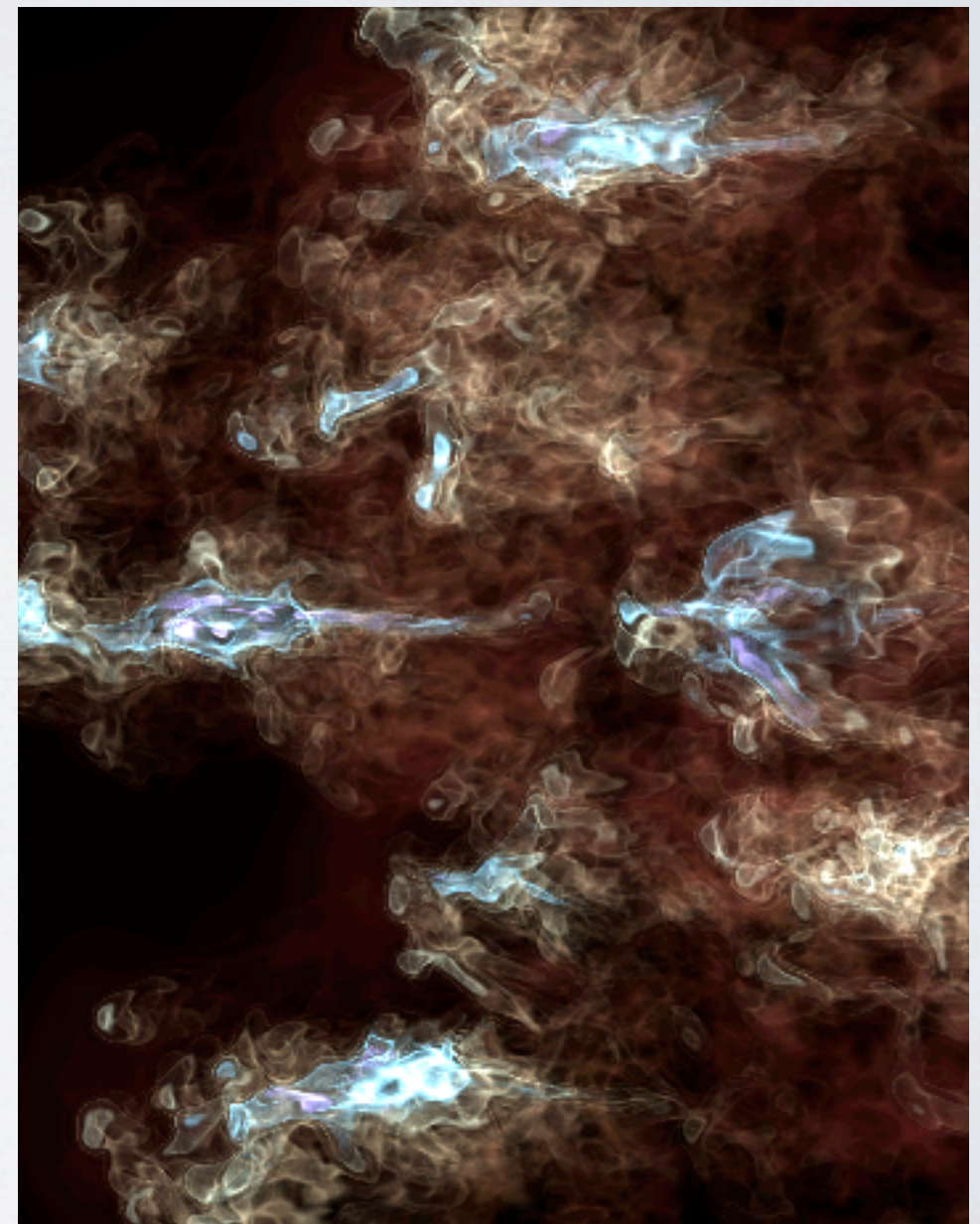


Image: Devin Silvia

CHEMISTRY AND COOLING

- Non-equilibrium primordial chemistry
 - $\text{H}, \text{H}^+, \text{H}^-, \text{He}, \text{He}^+, \text{He}^{++}, \text{H}_2, \text{H}_2^+, \text{D}, \text{D}^+, \text{HD}, \text{e}^-$
 - H_2 chemistry: 2-body, 3-body channels, chemical heating/cooling
- Metal cooling
 - simple tabulated rates ($T > 10^4 \text{ K}$)
 - atomic fine-structure lines
 - multidimensional Cloudy tables
 - density, metallicity, temperature, electron fraction, background redshift
 - new tables can be made (different abundance patterns, input spectra)

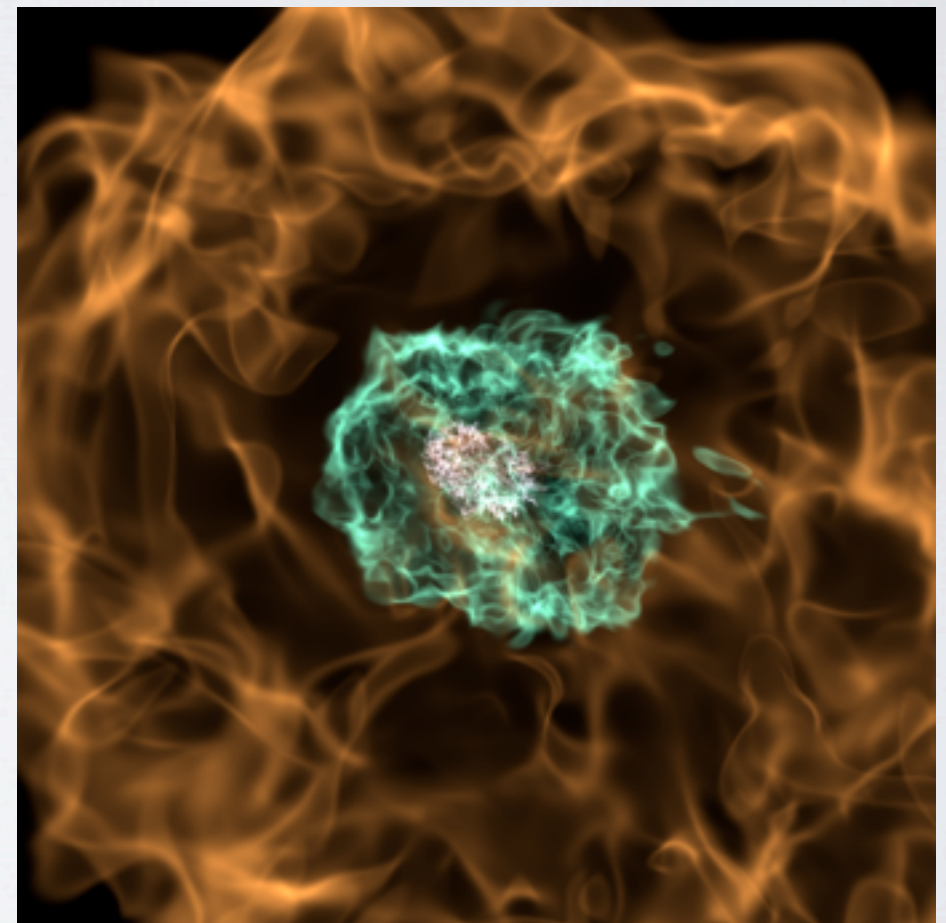
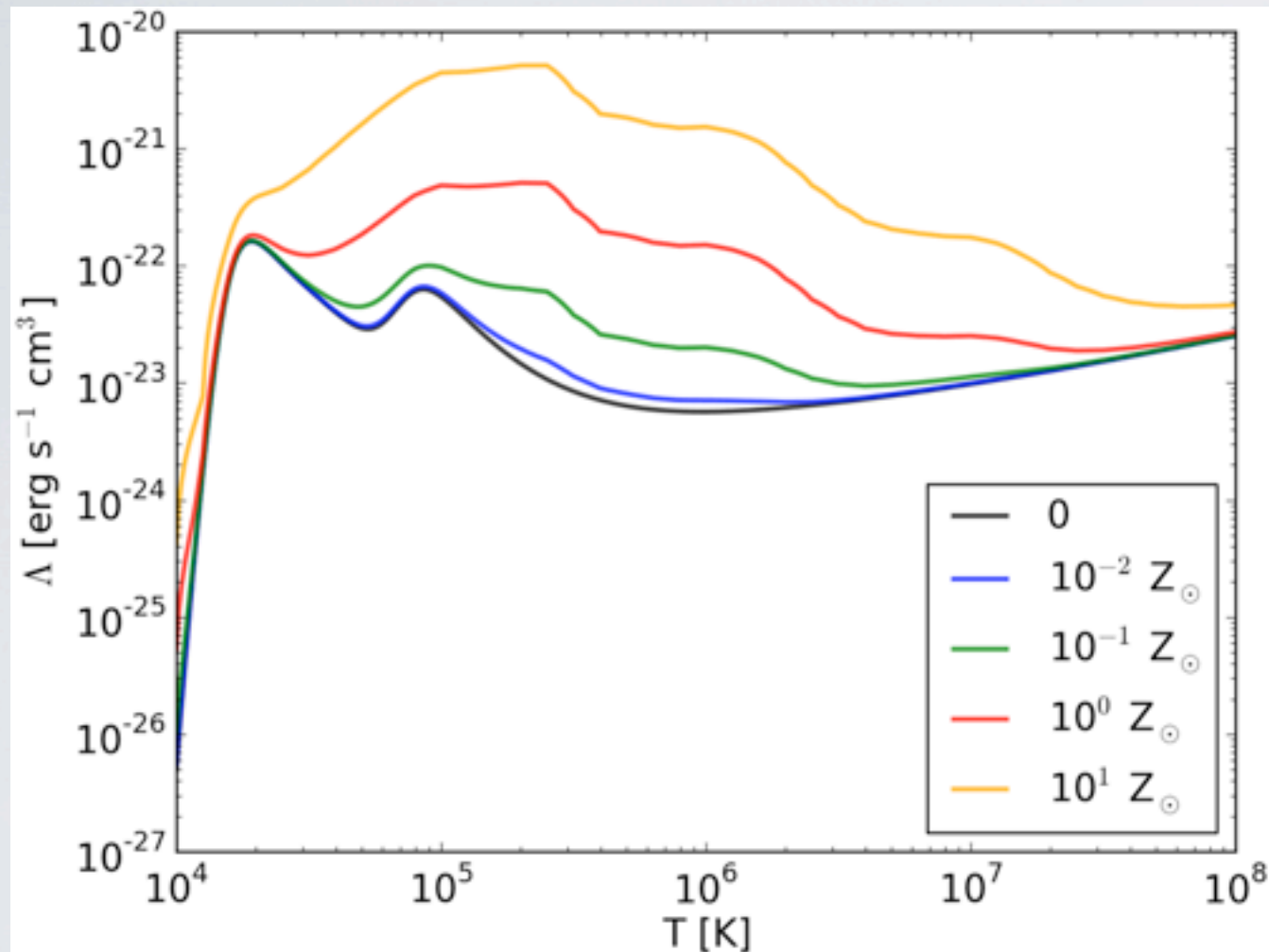
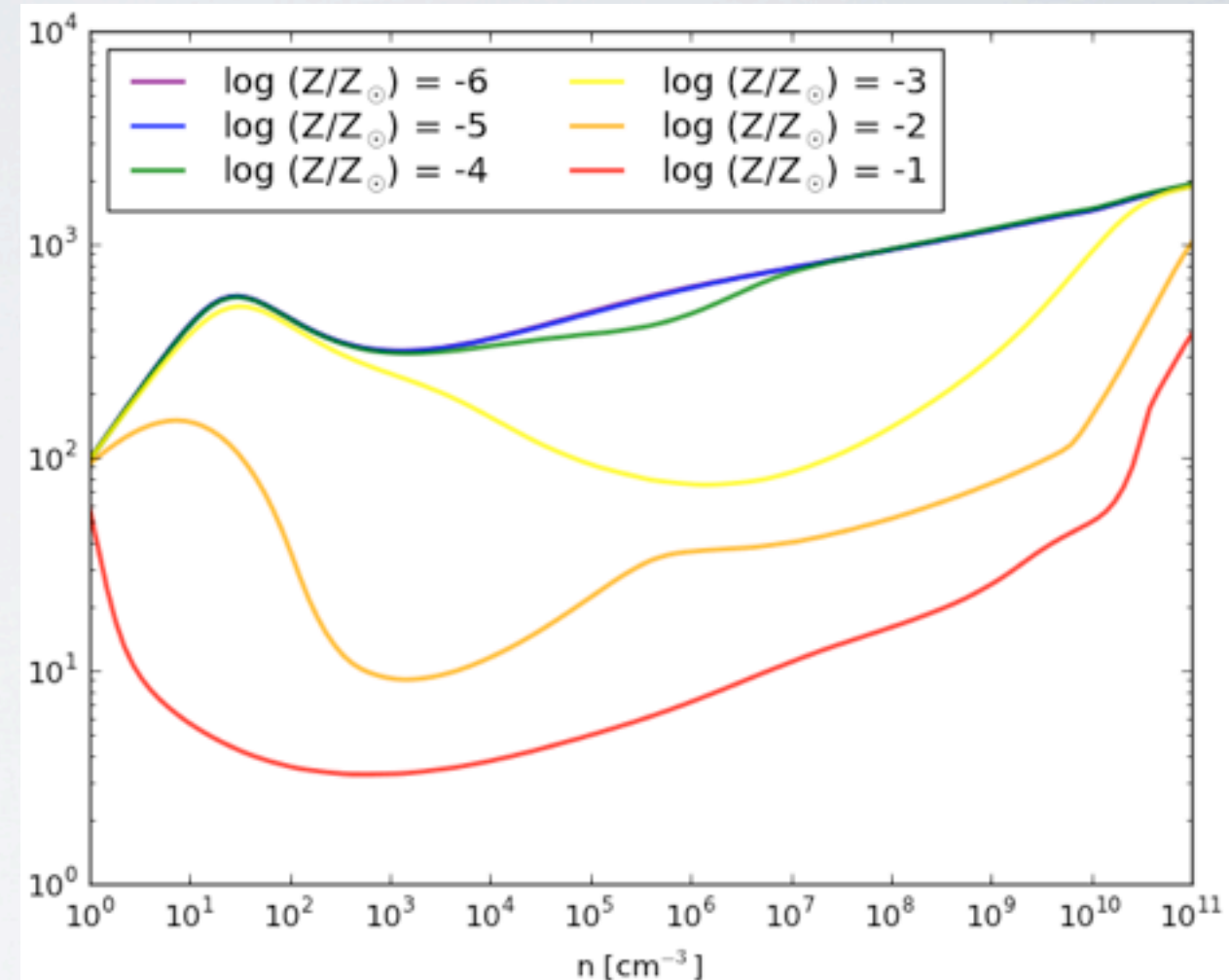


Image: Matthew Turk

CHEMISTRY AND COOLING



High temperature cooling rates at various metallicities.



One-zone model of gas collapse at various metallicities.

ACTIVE PARTICLES

- act on the grid by adding or removing gas, energy, and momentum
- non-radiating star particles
 - form in dense, collapsing, cooling gas
 - inject thermal energy, metals into nearby grid cells
- radiating star and black-hole particles form the same way and emit radiation
- sink particles accrete nearby gas like collapsing protostars

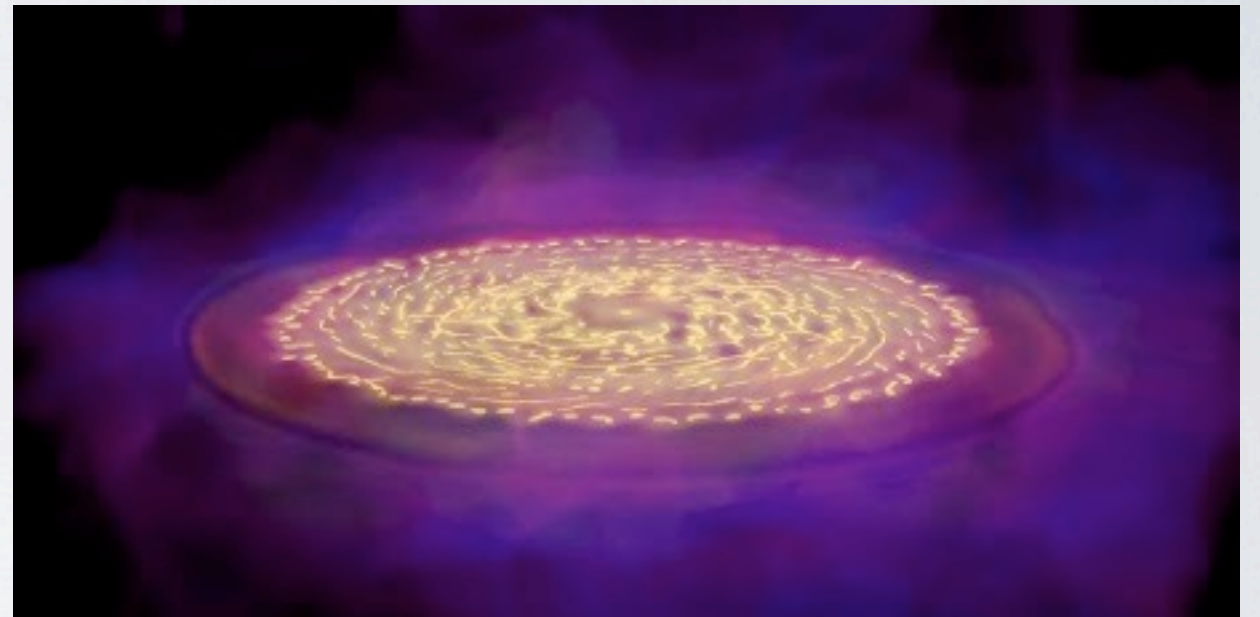
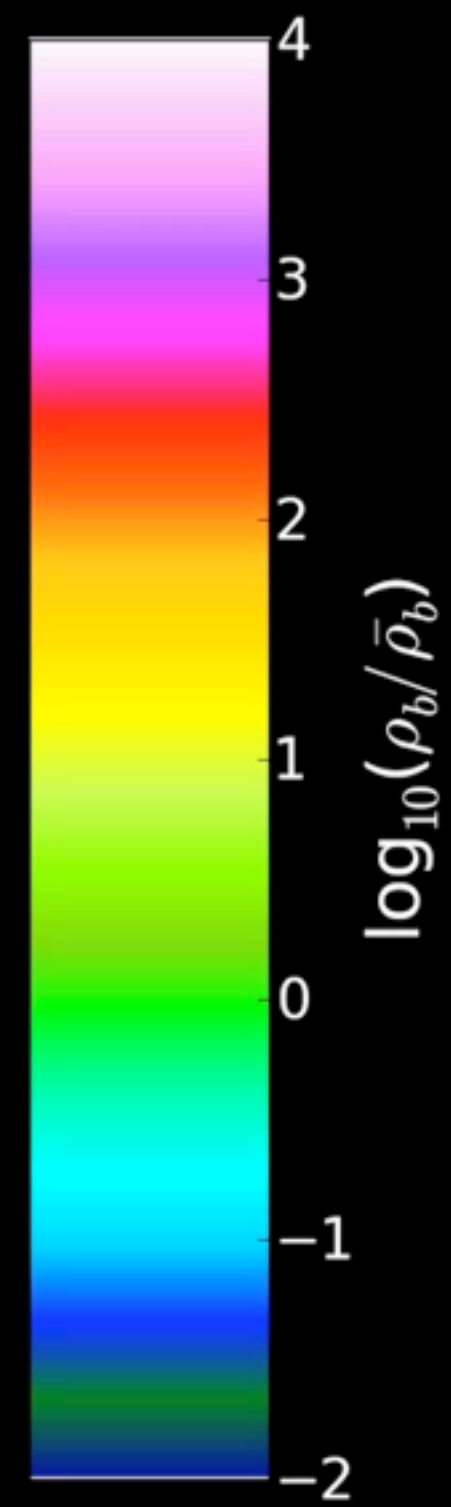
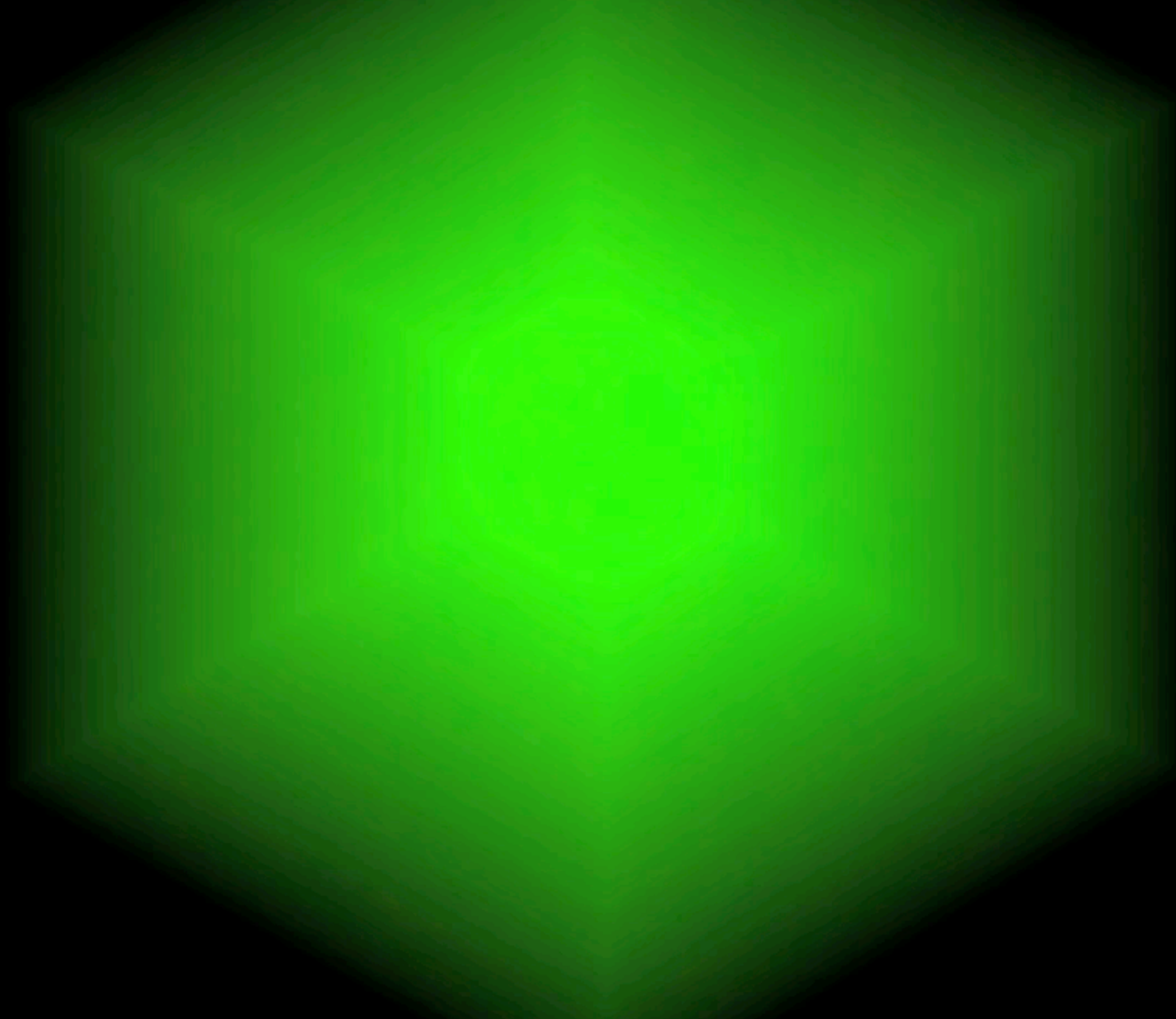


Image: Elizabeth Tasker

$t = 0.018 \text{ Gyr}$

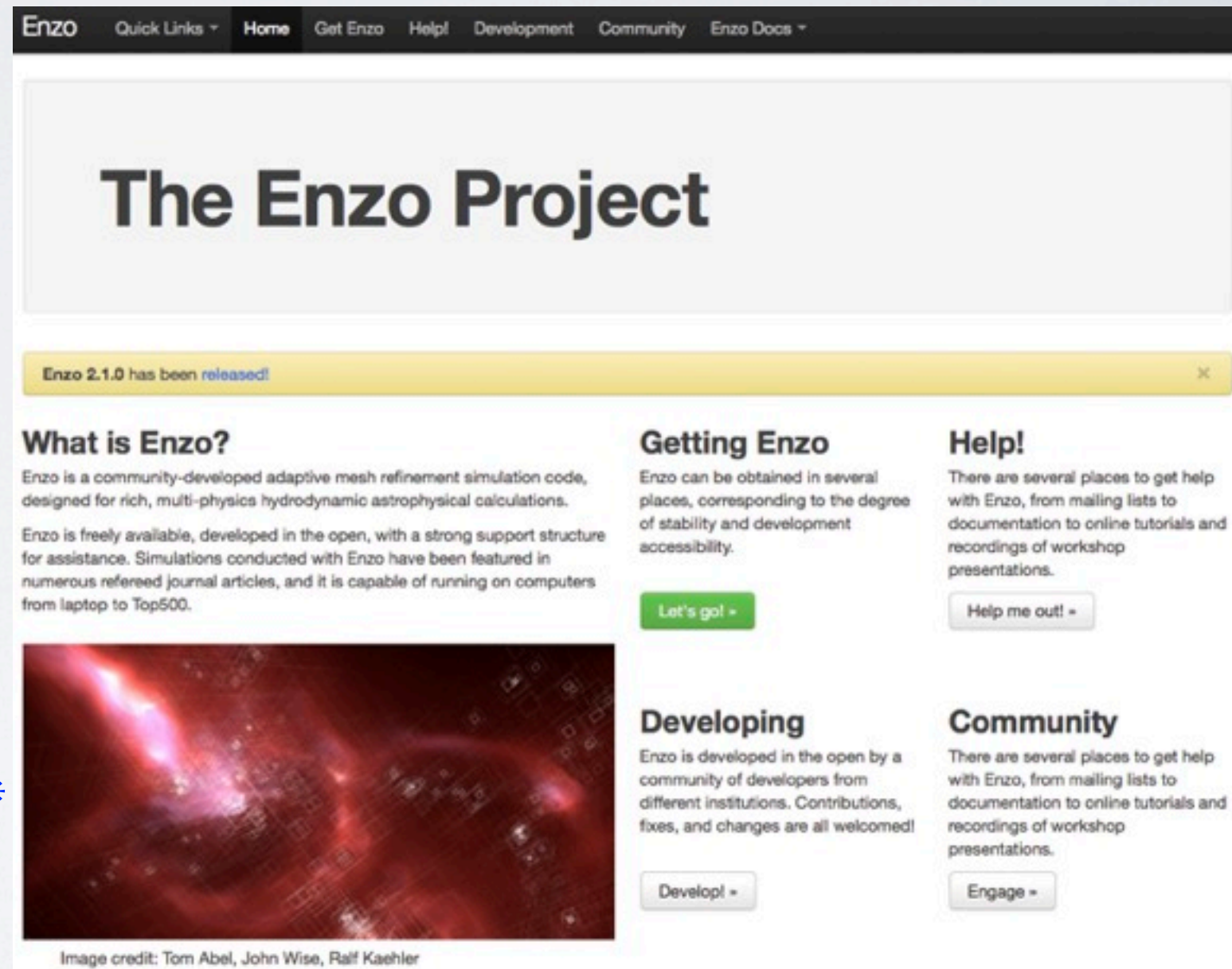


$z = 99.000$

GETTING ENZO

enzo-project.org

- Go to **enzo-project.org!**
 - links to the source code
 - documentation
 - community resources
- Two major versions:
 - Stable: v2.1.1 (Oct 27, 2011)
 - ~1-2 major updates per year
 - periodic bug-fixes
 - Development ***recommended***
 - ~1-2 updates per month
 - changes reviewed by developers



DEPENDENCIES

- hdf5 (Hierarchical Data Format) - version 1.8.x
- mpi (Message Passing Interface) - OpenMPI suggested
- Mercurial - version control system

RECOMMENDED METHOD

1. Go to yt-project.org

- download and run install script
- installs hdf5 and mercurial

2. Download and install OpenMPI (open-mpi.org)

1. ./configure

2. make

3. make install

The screenshot shows the Yt Project website. On the left is a dark sidebar with white text links: HOME, GET YT, EXAMPLES, COMMUNITY, DEVELOP, HELP!, DOCS, BLOG, and HUB. The main content area has a header 'THE YT PROJECT' with the subtitle 'ASTROPHYSICAL SIMULATION ANALYSIS AND VIZ'. Below this is a large visualization of a star formation simulation (Orion) with a color bar on the right labeled 'log density (g cm⁻³)' ranging from -18 to -22. A text box over the visualization reads 'COMMON ANALYSIS LANGUAGE' and 'ORION simulation of star formation by S. Offner et al.'. Below the visualization, it states 'DETAILED DATA ANALYSIS AND VISUALIZATIONS, WRITTEN BY WORKING ASTROPHYSICISTS AND DESIGNED FOR PRAGMATIC ANALYSIS NEEDS.' At the bottom are three columns: 'DATA-DRIVEN' (Inspect your data), 'COMMUNITY' (Participants welcome!), and 'FREE SOFTWARE' (Open Source, Open Science). Each column has a brief description of Yt's capabilities and philosophy.

HOME
GET YT
EXAMPLES
COMMUNITY
DEVELOP
HELP!
DOCS
BLOG
HUB

THE YT PROJECT
ASTROPHYSICAL SIMULATION ANALYSIS AND VIZ

COMMON ANALYSIS LANGUAGE
ORION simulation of star formation by S. Offner et al.

log density (g cm⁻³)
-18
-20
-22

DETAILED DATA ANALYSIS AND VISUALIZATIONS, WRITTEN BY WORKING ASTROPHYSICISTS AND DESIGNED FOR PRAGMATIC ANALYSIS NEEDS.

DATA-DRIVEN
Inspect your data
yt is designed to provide a consistent, cross-code interface to analyzing and visualizing

COMMUNITY
Participants welcome!
yt is composed of a friendly community of users and developers. We want to make it

FREE SOFTWARE
Open Source, Open Science
yt is developed completely in the open, released under the GPL license. The developers are

VERSION CONTROL WITH MERCURIAL

- Distributed version control
 - no need for a central repository
 - changes can be pushed from any repository to any repository
 - merging changes from multiple branches is easy (at least easier)
- Mercurial tutorial: <http://hginit.com>

GETTING ENZO WITH MERCURIAL

Check out a copy of Enzo (clone the repository):

```
hg clone https://bitbucket.org/enzo/enzo-dev
```

creates a directory on your computer called “enzo-dev”

Update your repository with the latest changes:

```
hg pull <source>
```

← pulls changes into the local repository

```
hg update
```

← updates the working copy with the latest changes

Add your new changes:

```
hg commit
```

← adds changes to the local repository

```
hg push <destination>
```

← pushes changes to another repository

COMPILING

```
cd enzo-dev  
./configure  
cd src/enzo
```

- Choose the make file that is right for your machine.
- Type “**ls Make.mach.***” to see the available options.
- For Mac OS, choose “Make.mach.darwin”.

COMPILING

- Edit Make.mach.darwin
- Change the following variables to the correct values:
 - LOCAL_MPI_INSTALL
 - LOCAL_HDF5_INSTALL
- Type:

```
make machine-darwin  
make
```
- If everything is right, Enzo is now compiled!

COMPILE OPTIONS

- Enzo has many additional compile options.
- Type: **make show-config** to see the current settings.
- Type: **make help-config** for a description of each parameter.
- Example: **make opt-high** to compile with basic optimizations. **Recommended!**
- Enzo must be recompiled after options are changed.

EXTRA TIPS

- Custom make files can be stored the .enzo directory in your home directory.
- Compiler settings can be saved with:
make save-config-<keyword>
- Reload custom settings with:
make load-config-<keyword>
- Settings files saved in ~/.enzo/Make.settings.<keyword>

COMPILING ON CONIVAL

```
[guest13@Conival ~]$ ls
enzo-dev  mpiexamples  workshop
[guest13@Conival ~]$ cd enzo-dev/
[guest13@Conival enzo-dev]$ cd src/enzo
[guest13@Conival enzo]$ cp ~/workshop/Make.mach.conival .
[guest13@Conival enzo]$ cd ../../
[guest13@Conival enzo-dev]$ ./configure
Configure complete.
[guest13@Conival enzo-dev]$ cd src/enzo
[guest13@Conival enzo]$ make machine-conival

*** Execute 'gmake clean' before rebuilding executables ***

MACHINE: conival
MACHINE-NAME: conival
[guest13@Conival enzo]$ make opt-high

*** Execute 'gmake clean' before rebuilding executables ***

CONFIG_OPT [opt-{warn,debug,cudadebug,high,aggressive}] : high
[guest13@Conival enzo]$ make

Updating DEPEND
make: [dep] Error 1 (ignored)
Compiling enzo.C
Compiling acml_st1.F
Compiling AdiabaticExpansionInitialize.C
Compiling AdjustRefineRegion.C
Compiling AdjustMustRefineParticlesRefineToLevel.C
Compiling AMRH5writer.C
Compiling AnalysisBaseClass.C
Compiling AnalysisBaseClass_HDF5Utils.C
```

Move into enzo directory.

Move into source directory.

Copy the make file into the current directory.

Move back to the top directory.

Run ./configure.

Move back to the source directory.

Type this to select this make file for compiling.

Configure to compile with high optimization.

Compile the code.

RUNNING A SIMULATION

- Simulations are configured with a parameter file.
- Run a new simulation:

```
mpirun -np <#> ./enzo.exe -d <parameter_file>
```

- Restart a simulation:

```
mpirun -np <#> ./enzo.exe -d -r <dataset>
```

- Many sample parameter files in [enzo-dev/run](#)

GETTING HELP

The screenshot shows the Enzo Project website. The navigation bar at the top includes links for 'Enzo', 'Quick Links', 'Home', 'Get Enzo', 'Help!', 'Development', 'Community', and 'Enzo Docs'. Red circles highlight 'Quick Links' and 'Enzo Docs', with a red arrow labeled 'Documentation' pointing to both. A yellow banner below the navigation bar states 'Enzo 2.1.0 has been released!'. The main content area is divided into four columns: 'What is Enzo?', 'Getting Enzo', 'Help!', and 'Community'. The 'Help!' column is circled in red, with a red arrow labeled 'More help!' pointing to it. The 'Help!' column contains text about where to get help and a 'Help me out!' button. The 'Getting Enzo' column has a 'Let's go!' button. The 'Developing' column has a 'Develop!' button. The 'Community' column has an 'Engage' button. A large image of a simulated galaxy cluster is shown in the bottom left, with the credit 'Image credit: John Wise' below it.

Enzo Quick Links Home Get Enzo Help! Development Community Enzo Docs

Documentation

The Enzo Project

Enzo 2.1.0 has been released!

What is Enzo?
Enzo is a community-developed adaptive mesh refinement simulation code, designed for rich, multi-physics hydrodynamic astrophysical calculations.
Enzo is freely available, developed in the open, with a strong support structure for assistance. Simulations conducted with Enzo have been featured in numerous refereed journal articles, and it is capable of running on computers from laptop to Top500.

Getting Enzo
Enzo can be obtained in several places, corresponding to the degree of stability and development accessibility.
[Let's go! »](#)

Help!
There are several places to get help with Enzo, from mailing lists to documentation to online tutorials and recordings of workshop presentations.
[Help me out! »](#)

Developing
Enzo is developed in the open by a community of developers from different institutions. Contributions, fixes, and changes are all welcomed!
[Develop! »](#)

Community
There are several places to get help with Enzo, from mailing lists to documentation to online tutorials and recordings of workshop presentations.
[Engage »](#)

Image credit: John Wise

GETTING HELP

Enzo Quick Links ▾ Home Get Enzo **Help!** Development Community Enzo Docs ▾

Help!

Places To Go

- [Current Documentation](#)
- [Live Chat](#)
- [Users' Mailing List](#)
- [Dev Mailing List](#)
- [Bug Tracker](#)
- [Workshop Videos](#)

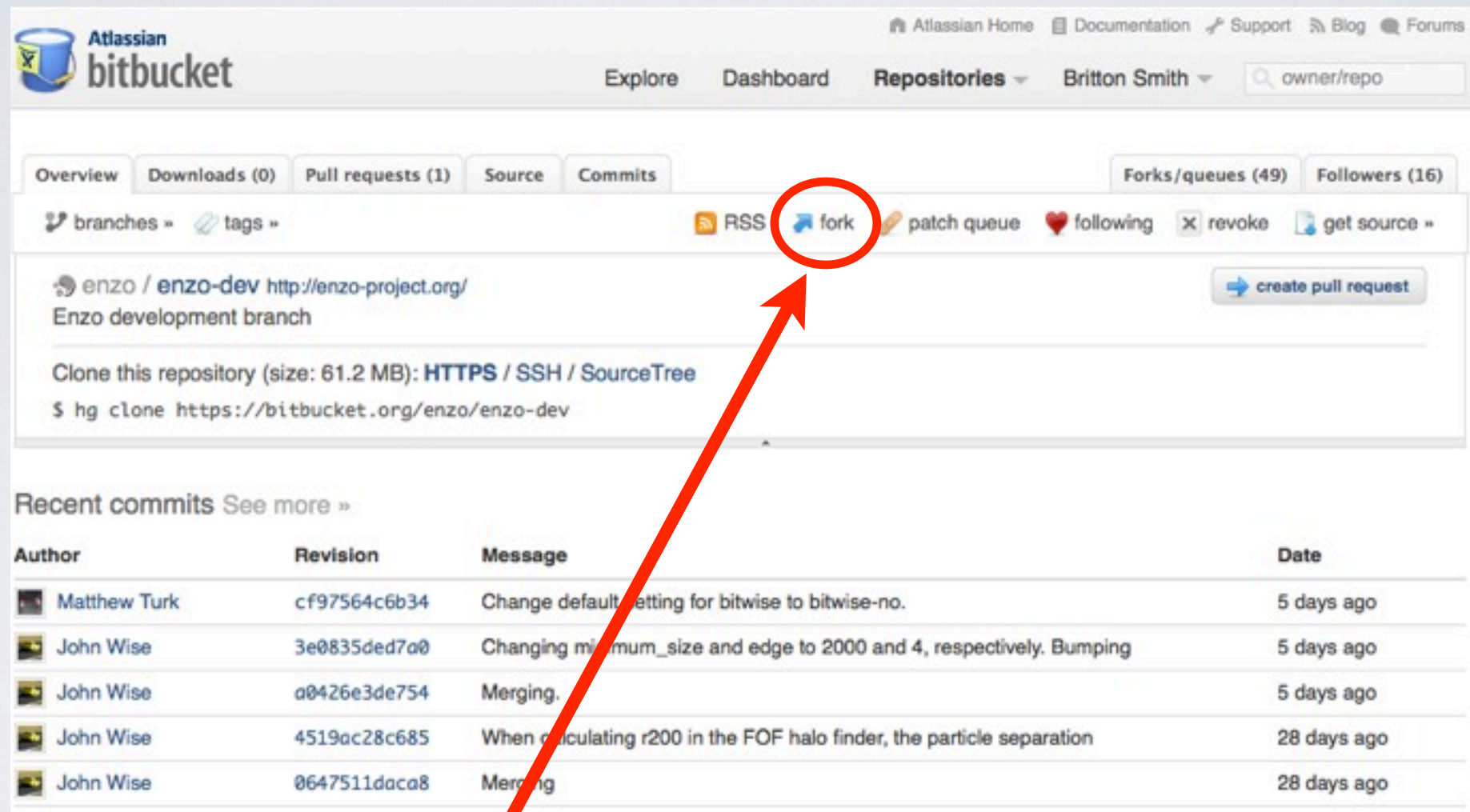
Using Enzo effectively can be a bit challenging at times — but there is plenty of assistance available, thanks to the broad community of users and developers. There are a number of tutorials in the [documentation](#), including example problems in the current source tree. For more narrative descriptions of how to use Enzo, there are a number of [videos](#) from the 2010 Enzo Users' Workshop.

But, in addition to prepared materials, there are also several forums available to get in touch with people who may be able to help you out with problems using or developing Enzo. There's both a [user-centric](#) mailing list as well as a mailing list more suited for [development issues](#). If this still doesn't help, we are tracking known issues in a public [bug tracker](#).

There are also usually a couple people in the channel `#enzo` on `irc.freenode.org`. You can get there with an IRC client like Adium, or we even have a [website](#) that will connect you *right in your browser!*

- mailing lists for users and developers
- IRC channel for live help
- workshop videos

DEVELOPMENT



Atlassian bitbucket

Explore Dashboard Repositories Britton Smith owner/repo

Overview Downloads (0) Pull requests (1) Source Commits Forks/queues (49) Followers (16)

branches » tags » RSS fork patch queue following revoke get source »

enzo / enzo-dev <http://enzo-project.org/>
Enzo development branch

Clone this repository (size: 61.2 MB): [HTTPS](#) / [SSH](#) / [SourceTree](#)
\$ hg clone https://bitbucket.org/enzo/enzo-dev

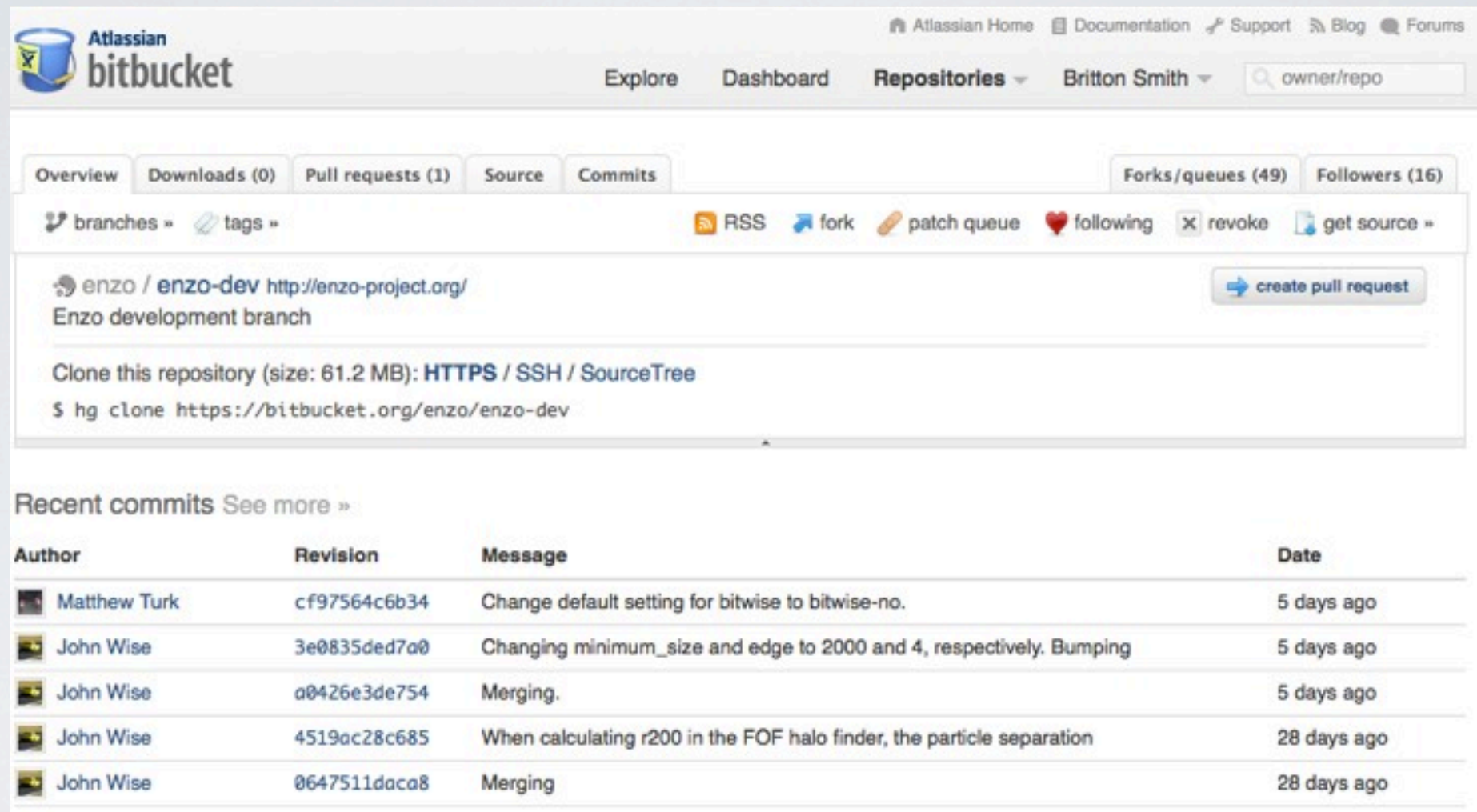
create pull request

Recent commits See more »

Author	Revision	Message	Date
Matthew Turk	cf97564c6b34	Change default setting for bitwise to bitwise-no.	5 days ago
John Wise	3e0835ded7a0	Changing minimum_size and edge to 2000 and 4, respectively. Bumping	5 days ago
John Wise	a0426e3de754	Merging.	5 days ago
John Wise	4519ac28c685	When calculating r200 in the FOF halo finder, the particle separation	28 days ago
John Wise	0647511daca8	Merging	28 days ago

I. Create a fork of the main repository.

DEVELOPMENT

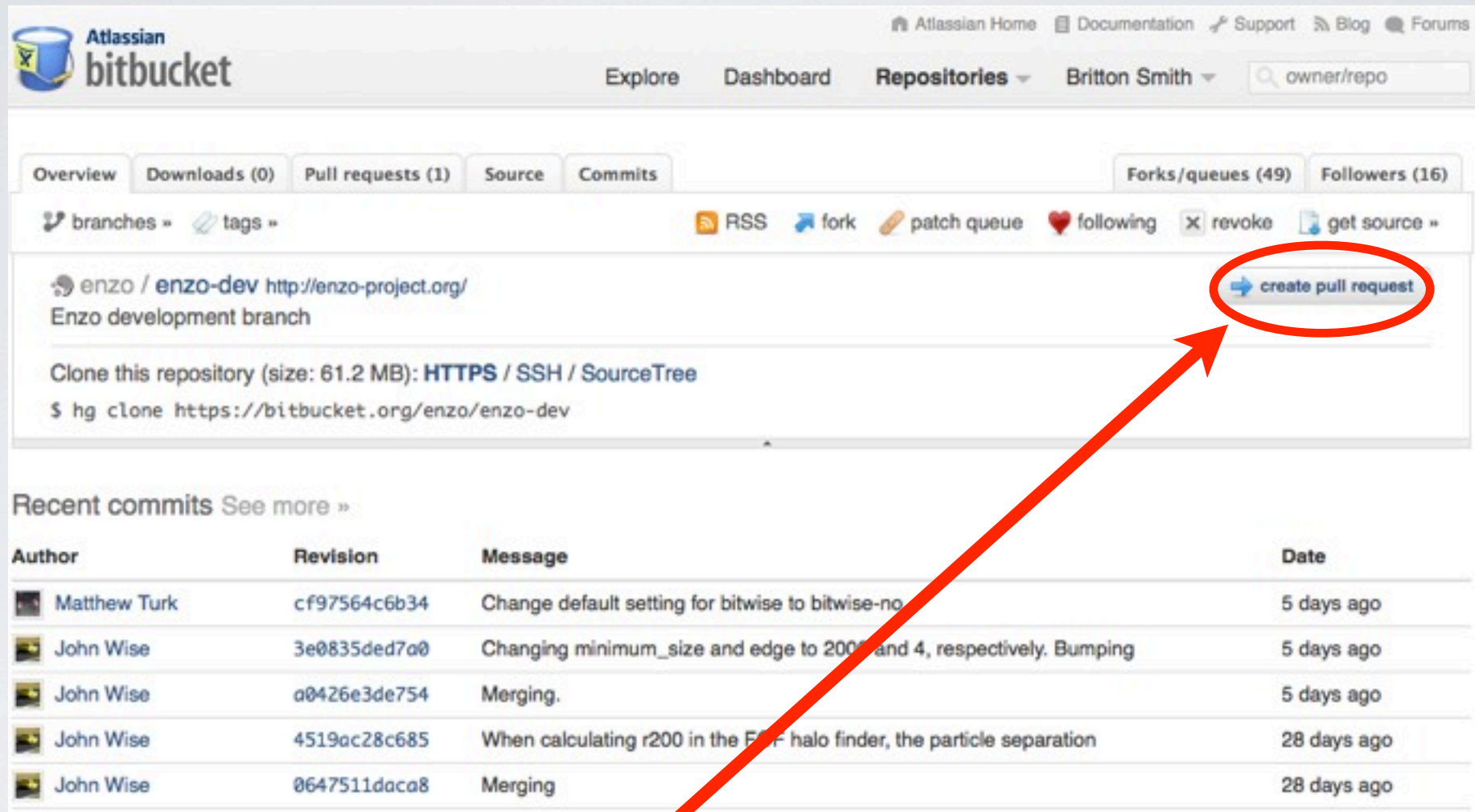


The screenshot shows the Bitbucket web interface for the 'enzo-dev' repository. The top navigation bar includes links for Atlassian Home, Documentation, Support, Blog, and Forums. The main header shows the repository name 'enzo / enzo-dev' and a search bar. Below the header, there are tabs for Overview, Downloads (0), Pull requests (1), Source, and Commits. The 'Overview' tab is active, displaying the repository's name, a 'create pull request' button, and cloning instructions. The 'Recent commits' section shows a table of the latest changes.

Author	Revision	Message	Date
Matthew Turk	cf97564c6b34	Change default setting for bitwise to bitwise-no.	5 days ago
John Wise	3e0835ded7a0	Changing minimum_size and edge to 2000 and 4, respectively. Bumping	5 days ago
John Wise	a0426e3de754	Merging.	5 days ago
John Wise	4519ac28c685	When calculating r200 in the FOF halo finder, the particle separation	28 days ago
John Wise	0647511daca8	Merging	28 days ago

1. Create a fork of the main repository.
2. Clone your fork, commit changes, push them to your fork.

DEVELOPMENT



Atlassian bitbucket

Explore Dashboard Repositories Britton Smith owner/repo

Overview Downloads (0) Pull requests (1) Source Commits Forks/queues (49) Followers (16)

branches » tags » RSS fork patch queue following revoke get source »

enzo / enzo-dev <http://enzo-project.org/>
Enzo development branch

Clone this repository (size: 61.2 MB): [HTTPS](#) / [SSH](#) / [SourceTree](#)
\$ hg clone https://bitbucket.org/enzo/enzo-dev

Recent commits [See more »](#)

Author	Revision	Message	Date
Matthew Turk	cf97564c6b34	Change default setting for bitwise to bitwise-no	5 days ago
John Wise	3e0835ded7a0	Changing minimum_size and edge to 2000 and 4, respectively. Bumping	5 days ago
John Wise	a0426e3de754	Merging.	5 days ago
John Wise	4519ac28c685	When calculating r200 in the FOF halo finder, the particle separation	28 days ago
John Wise	0647511daca8	Merging	28 days ago

1. Create a fork of the main repository.
2. Clone your fork, commit changes, push them to your fork.
3. Issue a “pull request”.

DEVELOPMENT

The screenshot shows the Bitbucket interface for a repository named 'enzo-dev'. The top navigation bar includes links for 'Atlassian Home', 'Documentation', 'Support', 'Blog', and 'Forums'. Below this, the repository name 'enzo / enzo-dev' is displayed along with its URL 'http://enzo-project.org/'. The repository is described as the 'Enzo development branch'. A 'create pull request' button is visible. Below the repository information, there is a section for 'Recent commits' with a 'See more »' link. The commits are listed in a table with columns for Author, Revision, Message, and Date.

Author	Revision	Message	Date
Matthew Turk	cf97564c6b34	Change default setting for bitwise to bitwise-no.	5 days ago
John Wise	3e0835ded7a0	Changing minimum_size and edge to 2000 and 4, respectively. Bumping	5 days ago
John Wise	a0426e3de754	Merging.	5 days ago
John Wise	4519ac28c685	When calculating r200 in the FOF halo finder, the particle separation	28 days ago
John Wise	0647511daca8	Merging	28 days ago

1. Create a fork of the main repository.
2. Clone your fork, commit changes, push them to your fork.
3. Issue a “pull request”.
4. Other developers review changes, make comments, accept.

DEVELOPERS

== DEVELOPERS ==

Many people have contributed to the development of Enzo -- here's just a short list of the people who have recently contributed.

- * Greg Bryan gbryan@astro.columbia.edu
- * Tom Abel tabel@stanford.edu
- * Michael Norman mlnorman@ucsd.edu
- * John Wise jwise@physics.gatech.edu
- * Dan Reynolds reynolds@smu.edu
- * Michael Kuhlen mquk@astro.berkeley.edu
- * Matthew Turk matthewturk@gmail.com
- * Brian O'Shea oshea@msu.edu
- * Robert Harkness harkness@sdsc.edu
- * Alexei Kritsuk akritsuk@ucsd.edu
- * Elizabeth Tasker taskere@mcmaster.ca
- * Dave Collins dcollins@physics.ucsd.edu
- * Britton Smith brittonsmith@gmail.com
- * Elizabeth Harper-Clark h-clark@astro.utoronto.ca
- * Peng Wang pengw@slac.stanford.edu
- * Fen Zhao fenzhao@stanford.edu
- * James Bordner jobordner@ucsd.edu
- * Pascal Paschos ppaschos@minbari.ucsd.edu
- * Stephen Skory sskory@physics.ucsd.edu
- * Rick Wagner rwagner@physics.ucsd.edu
- * Renyue Cen cen@astro.princeton.edu
- * Alex Razoumov razoumov@gmail.com
- * Cameron Hummels chummels@astro.columbia.edu
- * JS Oishi jsoishi@gmail.com
- * Christine Simpson csimpson@astro.columbia.edu
- * Samuel Skillman samskillman@gmail.com

DEVELOPERS

== DEVELOPERS ==

Many people have contributed to the development of Enzo -- here's just a short list of the people who have recently contributed.

- * Greg Bryan gbryan@astro.columbia.edu
- * Tom Abel tabel@stanford.edu
- * Michael Norman mlnorman@ucsd.edu
- * John Wise jwise@physics.gatech.edu
- * Dan Reynolds reynolds@smu.edu
- * Michael Kuhlen mqk@astro.berkeley.edu
- * Matthew Turk matthewturk@gmail.com
- * Brian O'Shea oshea@msu.edu
- * Robert Harkness harkness@sdsc.edu
- * Alexei Kritsuk akritsuk@ucsd.edu
- * Elizabeth Tasker taskere@mcmaster.ca
- * Dave Collins dcollins@physics.ucsd.edu
- * Britton Smith brittonsmith@gmail.com
- * Elizabeth Harper-Clark h-clark@astro.utoronto.ca
- * Peng Wang pengw@slac.stanford.edu
- * Fen Zhao fenzhao@stanford.edu
- * James Bordner jobordner@ucsd.edu
- * Pascal Paschos ppaschos@minbari.ucsd.edu
- * Stephen Skory sskory@physics.ucsd.edu
- * Rick Wagner rwagner@physics.ucsd.edu
- * Renyue Cen cen@astro.princeton.edu
- * Alex Razoumov razoumov@gmail.com
- * Cameron Hummels chummels@astro.columbia.edu
- * JS Oishi jsoishi@gmail.com
- * Christine Simpson csimpson@astro.columbia.edu
- * Samuel Skillman samskillman@gmail.com

• You?

THANK YOU