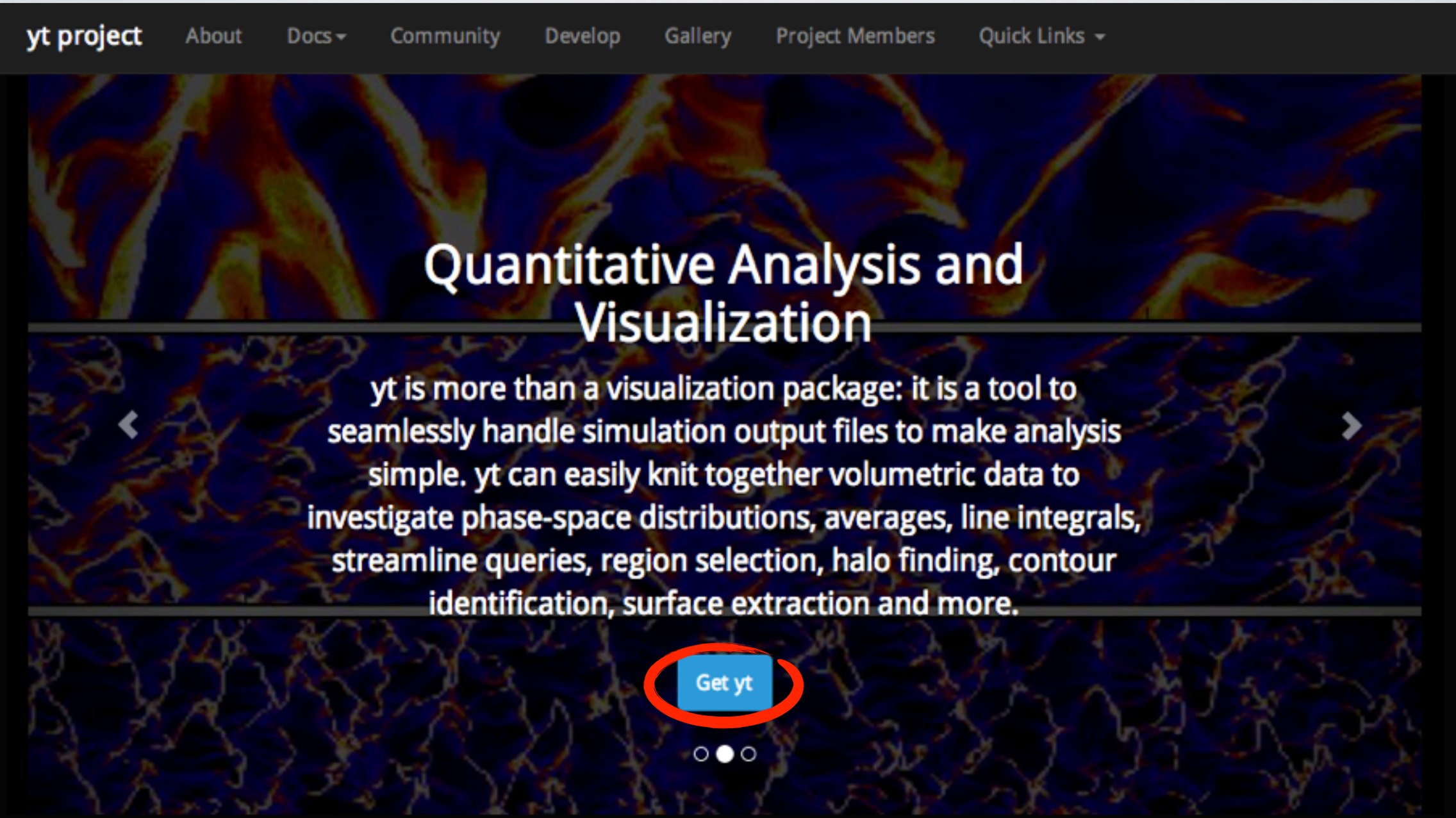
Install yt from yt-project.org

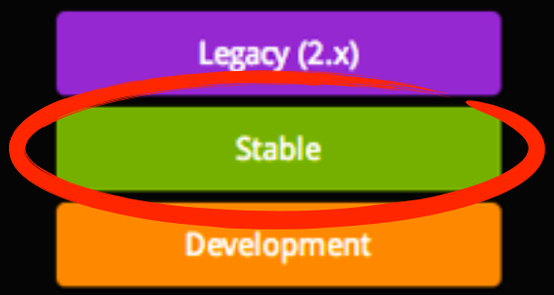


Install yt from yt-project.org

yt project | About | Docs | Community | Develop | Gallery | Project Members | Quick Links

can be upgraded and operated independently of the host operating system.

Usually getting yt is as simple as running the installation script. Simply download the legacy, stable, or development version of the install script and run it. You can do this using **wget** or **curl**, or even just right click and choose **Save As**. Carefully read the instructions the script prints to your terminal since there might be special instructions for your operating system.



Once you've downloaded it, just run:

```
$ bash install_script.sh
$ source YT_DEST/bin/activate
```

```
$ conda install yt
```

Get yt: from source.

If you are comfortable installing Python packages and have a build environment set up, you can install yt via **pip**:

```
$ pip install yt
```

If you would like to install the development version of yt, first clone the repository:

```
$ hg clone
https://bitbucket.org/yt_analysis/yt
```

And run the following command in the root source directory:

```
$ hg update yt
```

Then do the following:

```
$ python setup.py develop
```

INTRODUCTION TO ENZO

Britton Smith

OUTLINE

I. Available Modules

II. Problem Types

III. Enzo Resources

IV. Development

Image: Devin Silvia

ENZO METHOD PAPER

ENZO: AN ADAPTIVE MESH REFINEMENT CODE FOR ASTROPHYSICS

The Enzo Collaboration: Greg L. Bryan[1], Michael L. Norman[2,3], Brian W. O'Shea[4,5], Tom Abel[6, 20], John H. Wise[7], Matthew J. Turk[1], Daniel R. Reynolds[8], David C. Collins[9], Peng Wang[6], Samuel W. Skillman[10,11], Britton Smith[4], Robert P. Harkness[12], James Bordner[2], Ji-hoon Kim[13], Michael Kuhlen[14,15], Hao Xu[2], Nathan Goldbaum[15], Cameron Hummels[16], Alexei G. Kritsuk[2], Elizabeth Tasker[17], Stephen Skory[10], Christine M. Simpson[1], Oliver Hahn[18], Jeffrey S. Oishi[19], Geoffrey C So[2], Fen Zhao[20], Renyue Cen[21], and Yuan Li[1]

Draft version July 22, 2013

ABSTRACT

This paper describes the open-source code Enzo, which uses block-structured adaptive mesh refinement to provide high spatial and temporal resolution for modeling astrophysical fluid flows. The code is Cartesian, can be run in 1, 2, and 3 dimensions, and supports a wide variety of physics including hydrodynamics, ideal and non-ideal magnetohydrodynamics, N-body dynamics (and, more broadly, self-gravity of fluids and particles), primordial gas chemistry, optically-thin radiative cooling of primordial and metal-enriched plasmas (as well as some optically-thick cooling models), radiation transport, cosmological expansion, and models for star formation and feedback in a cosmological context. In addition to explaining the algorithms implemented, we present solutions for a wide range of test problems, demonstrate the code's parallel performance, and discuss the Enzo collaboration's code development methodology.

Keywords: methods: numerical — hydrodynamics

http://adsabs.harvard.edu/abs/2014ApJS..211...19B

WHAT IS ENZO?

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

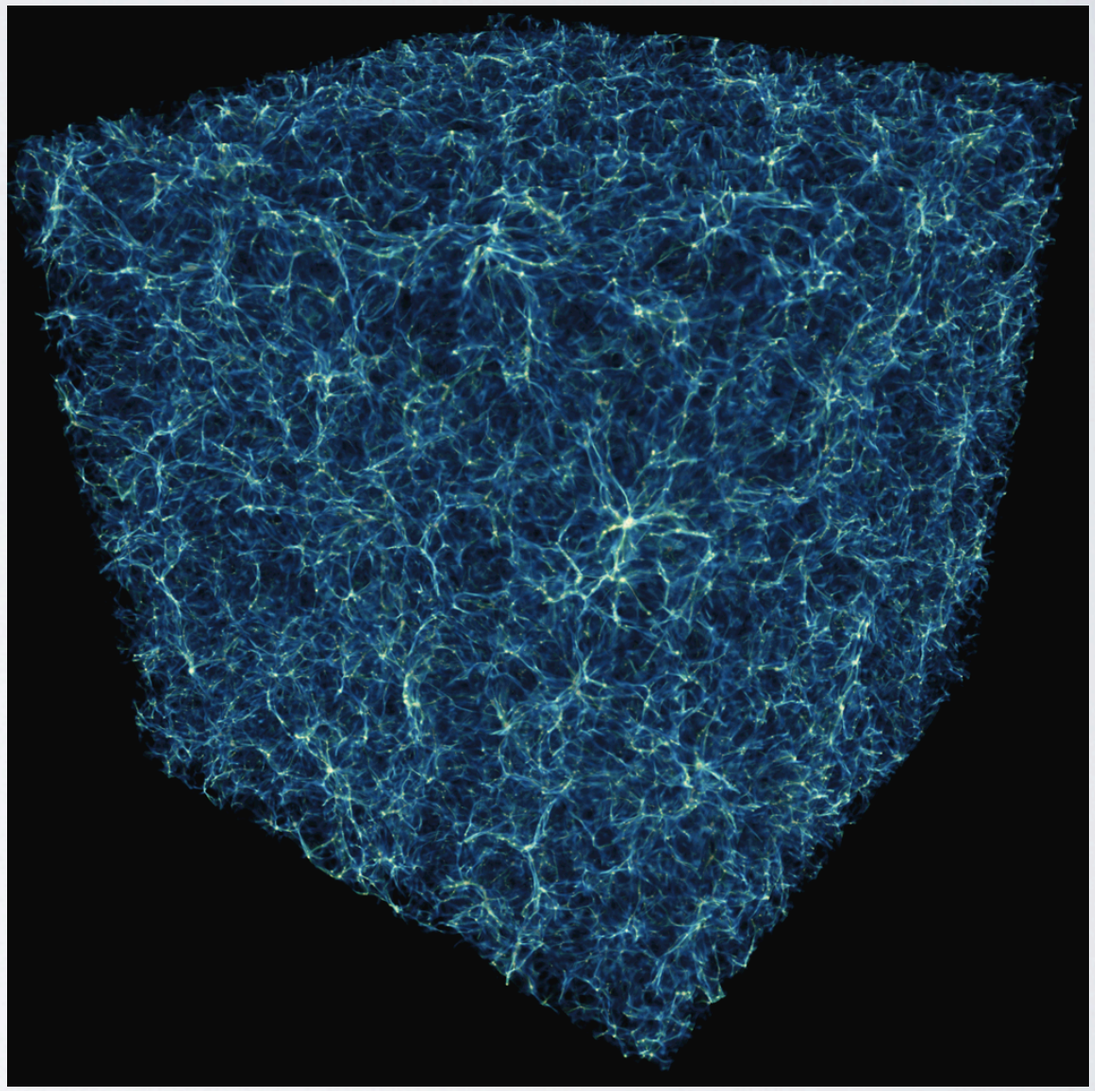
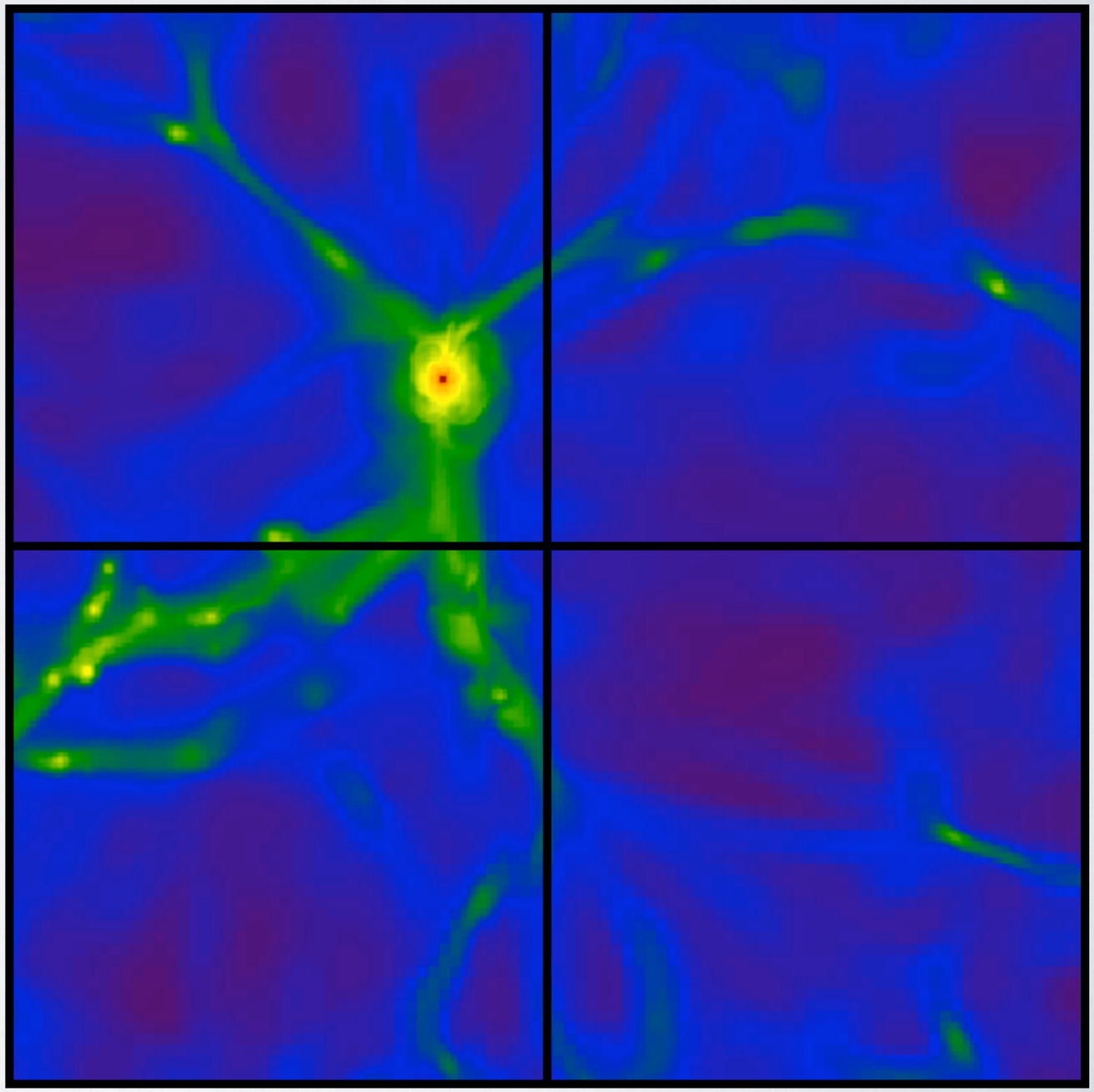


Image: Eric Hallman, Brian O'Shea

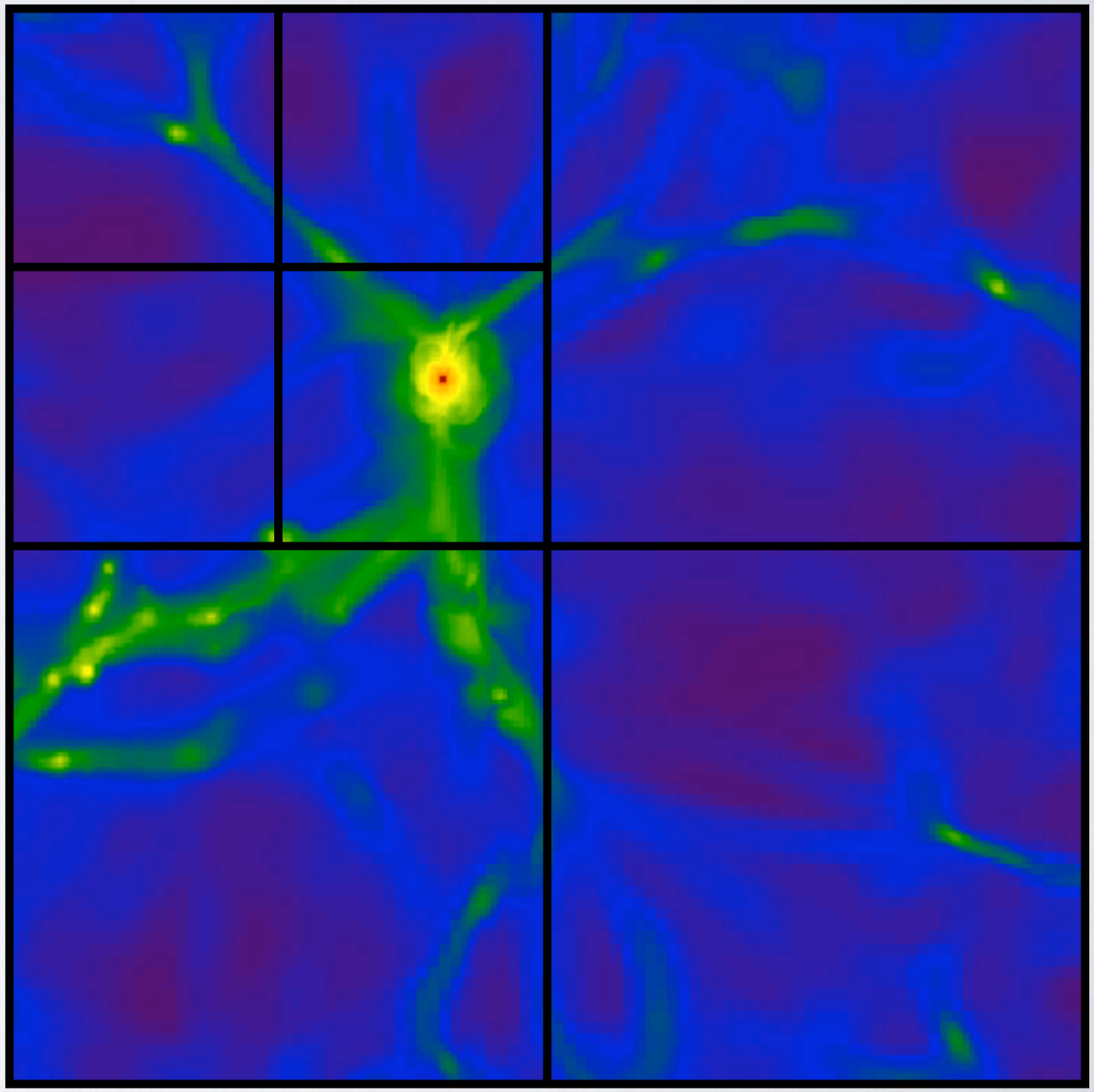
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



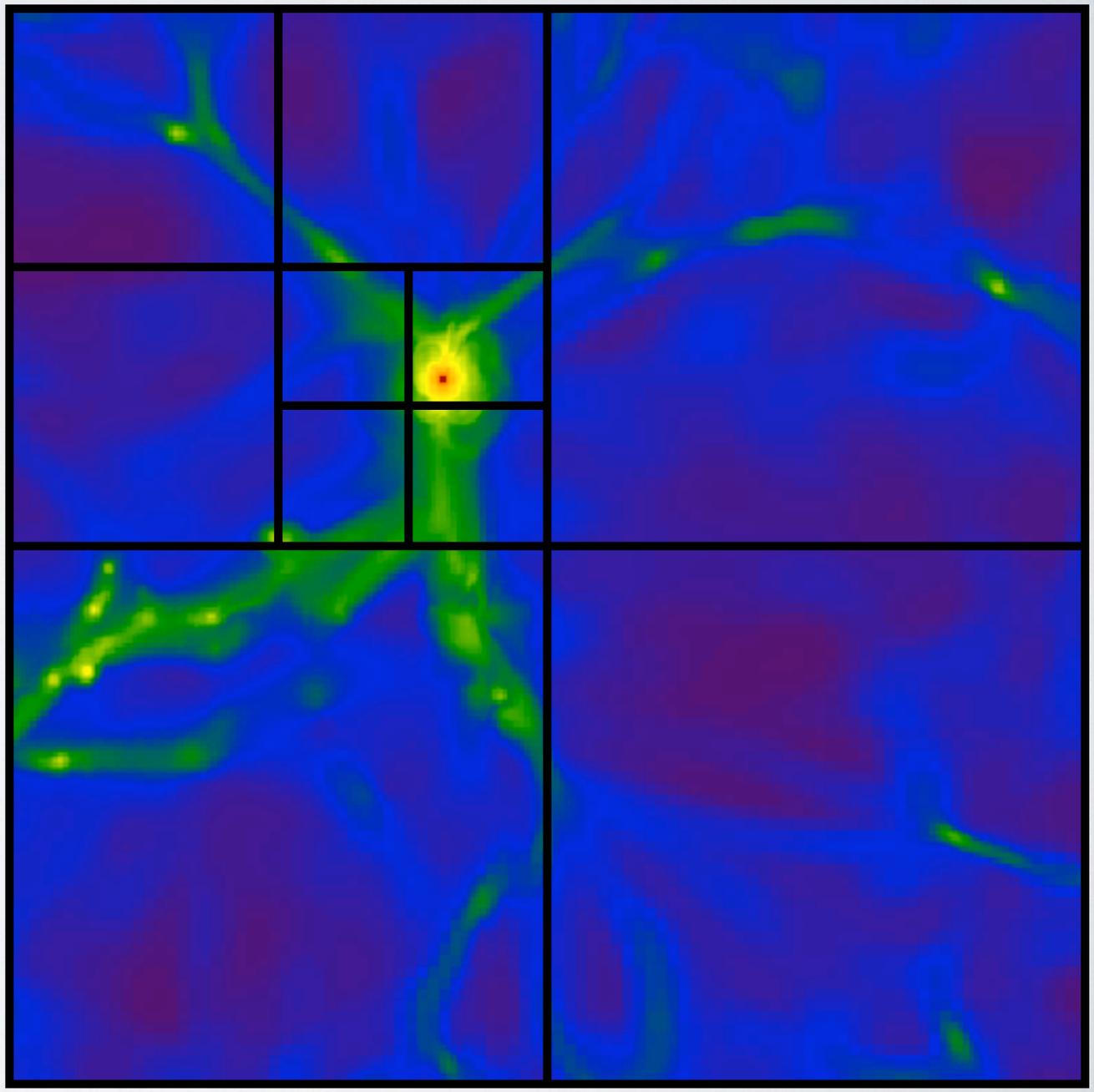
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



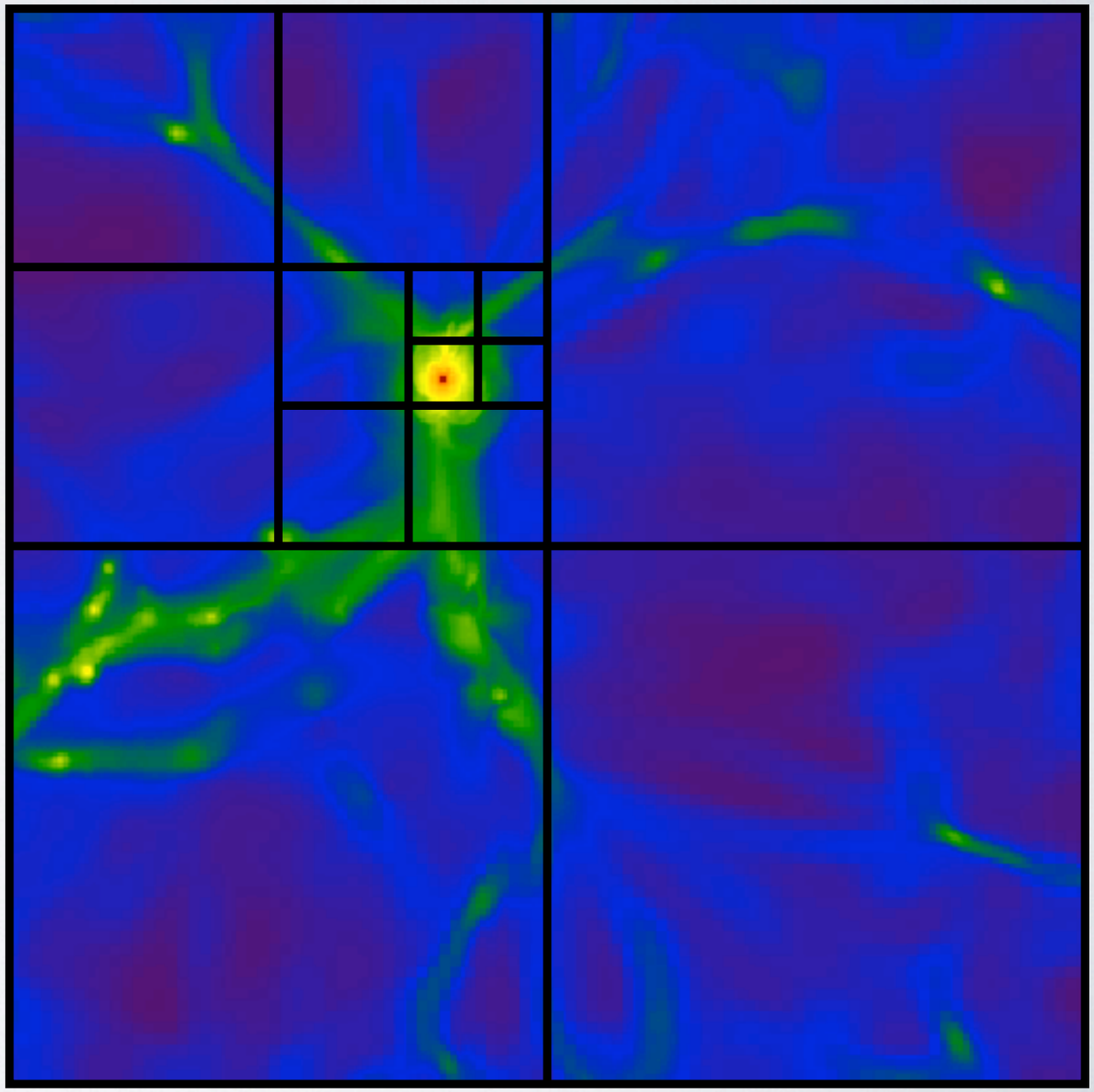
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



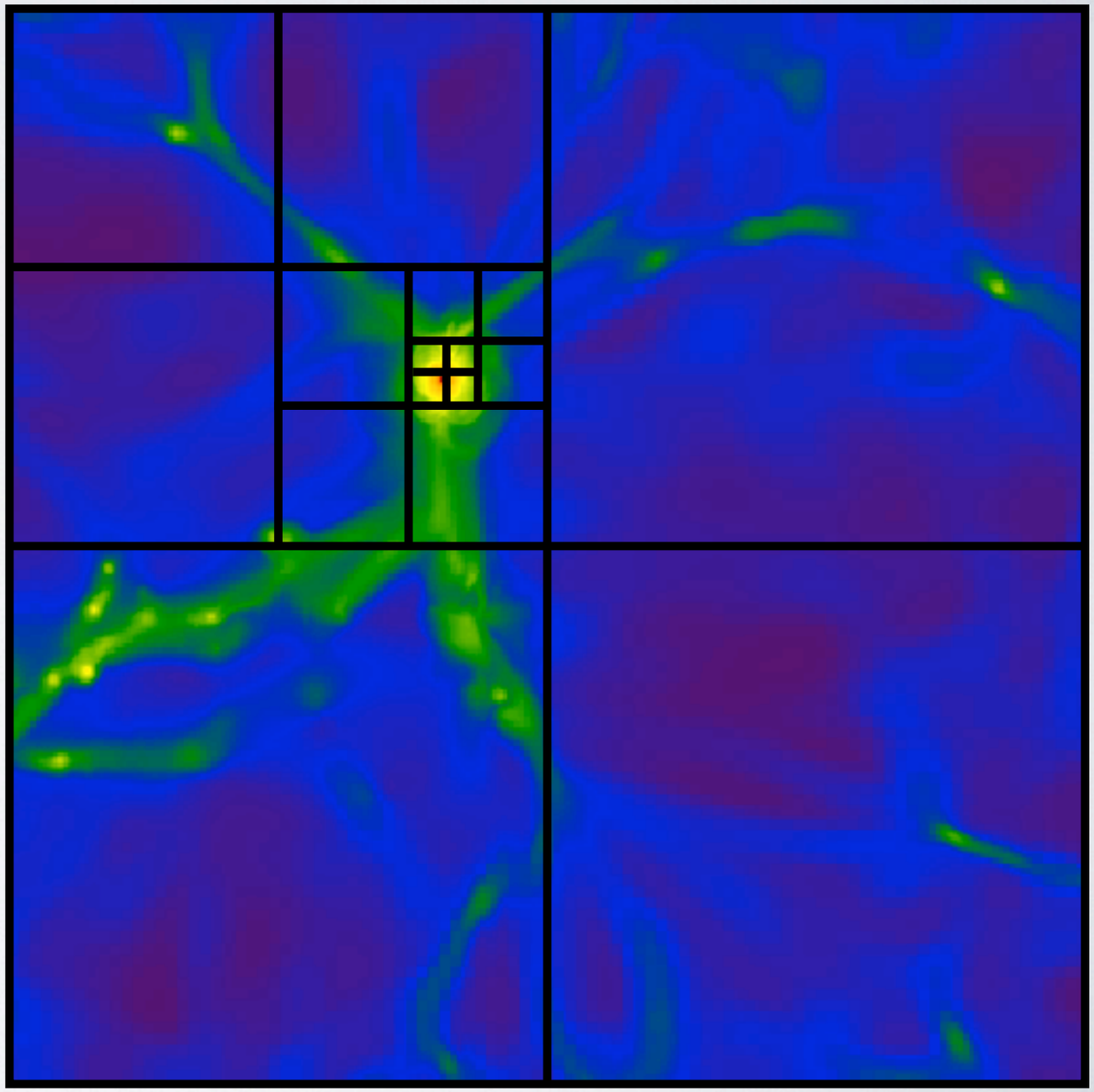
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



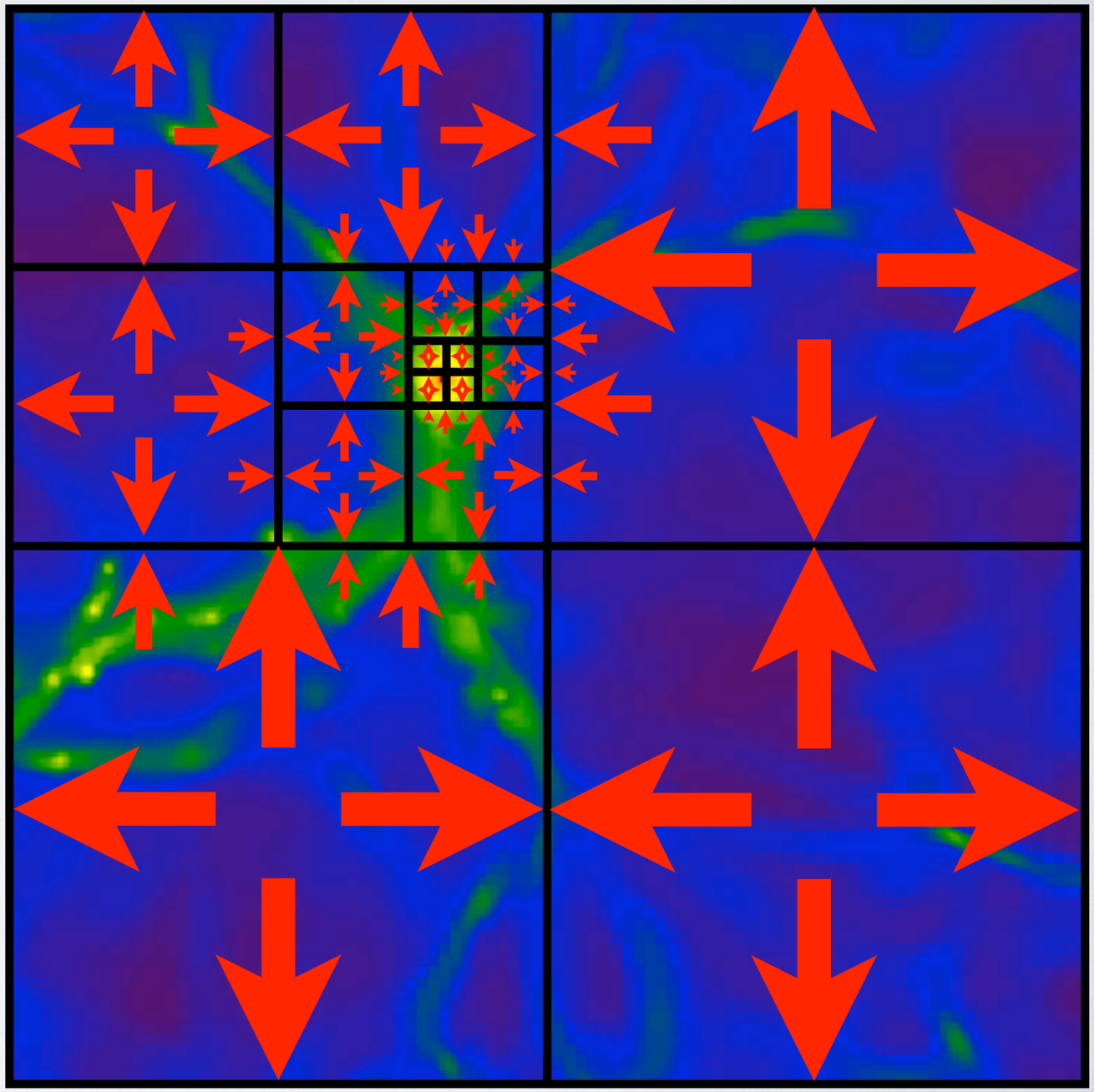
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



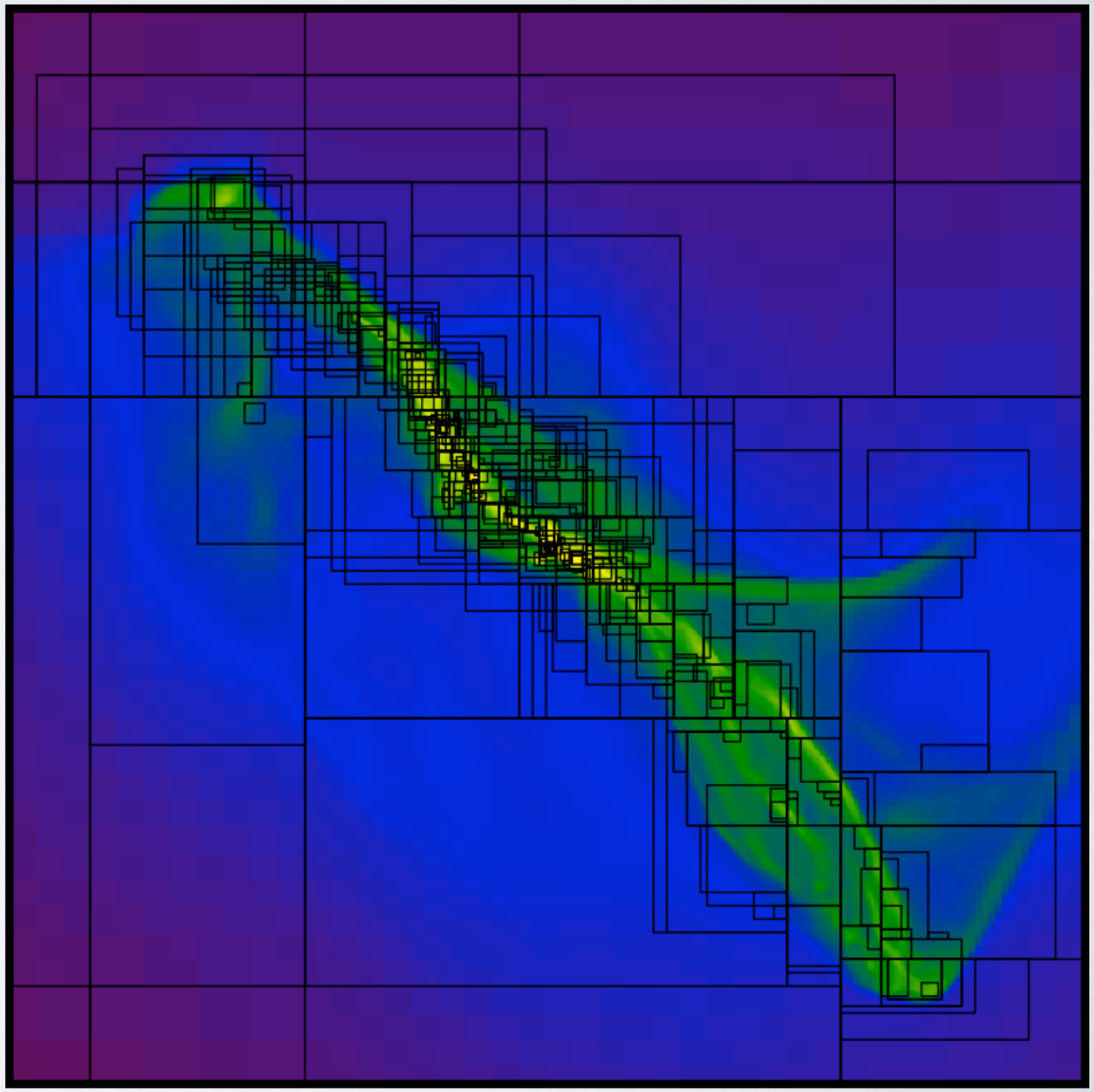
AMR:
ADAPTIVE-MESH REFINEMENT

- 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



ADAPTIVE-MESH REFINEMENT
EXTREME DYNAMIC RANGE

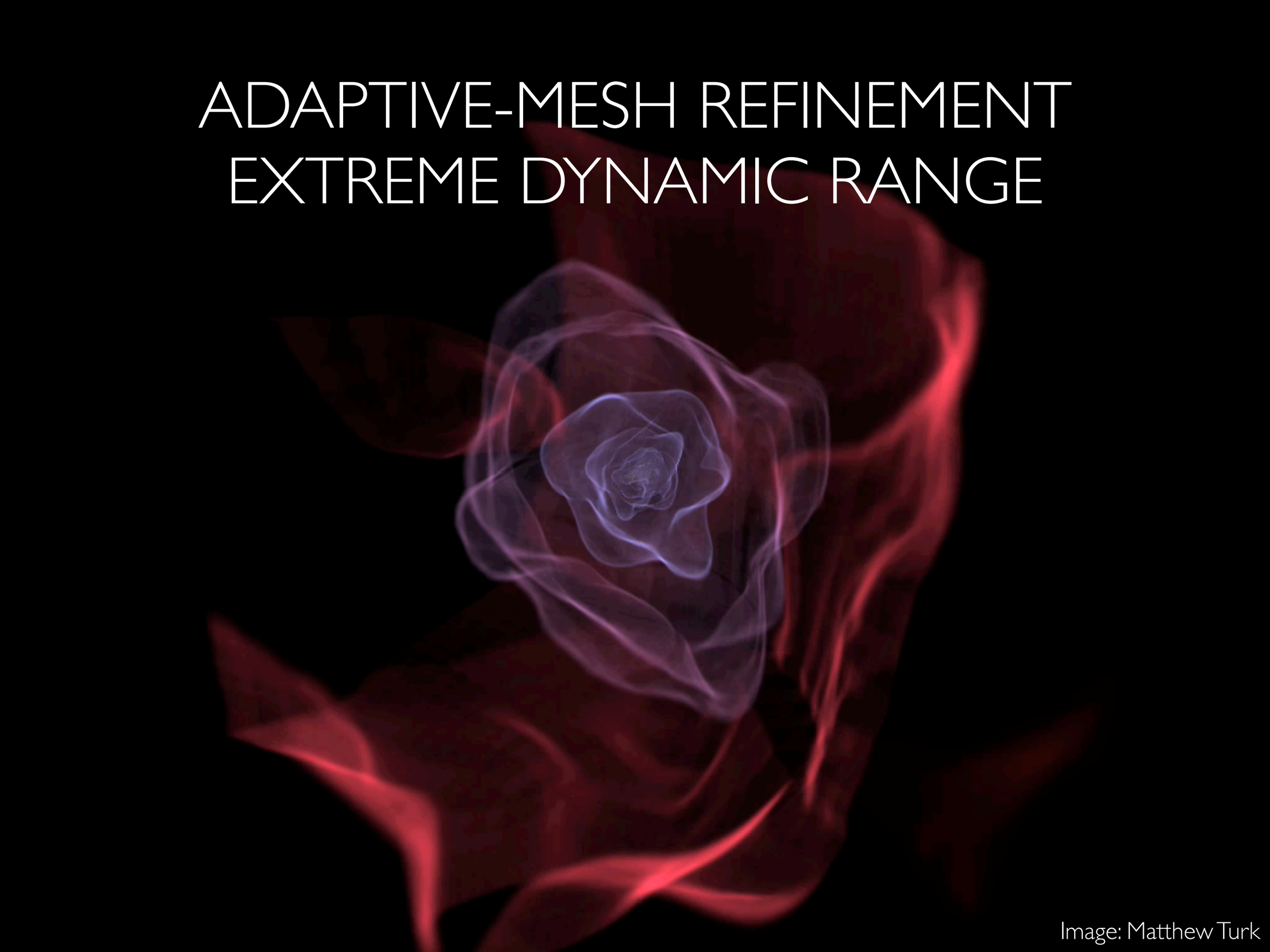


Image: Matthew Turk

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)

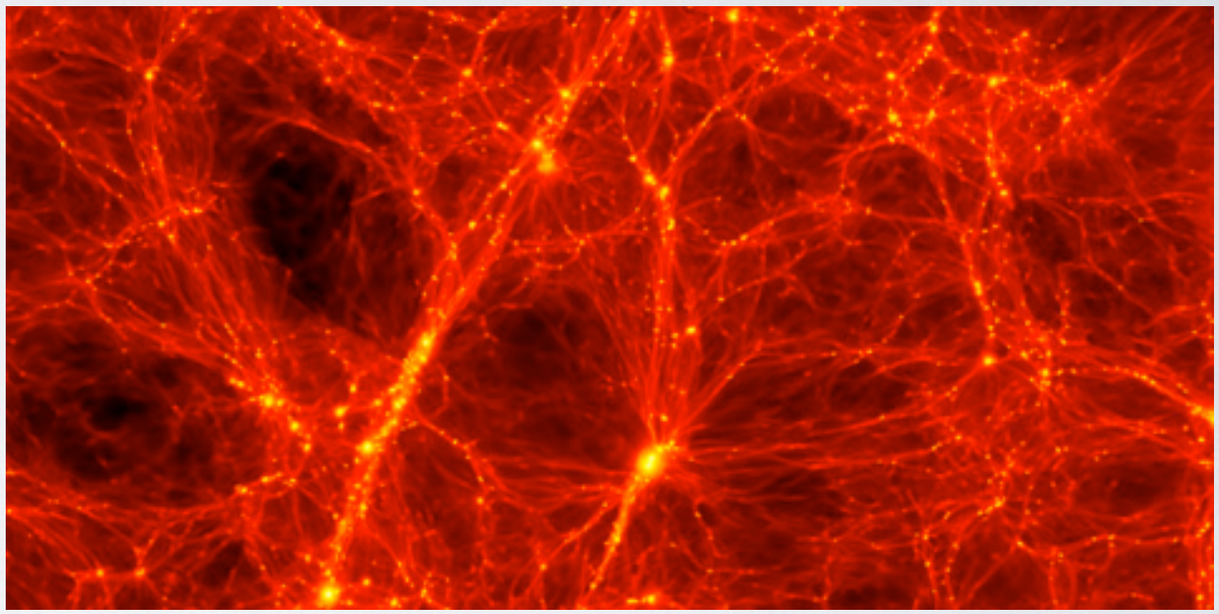


Image: Michael Norman et. al.

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 (much better now)
- Zeus
  - less accurate, more diffusive
  - extremely robust (excellent for cosmology)

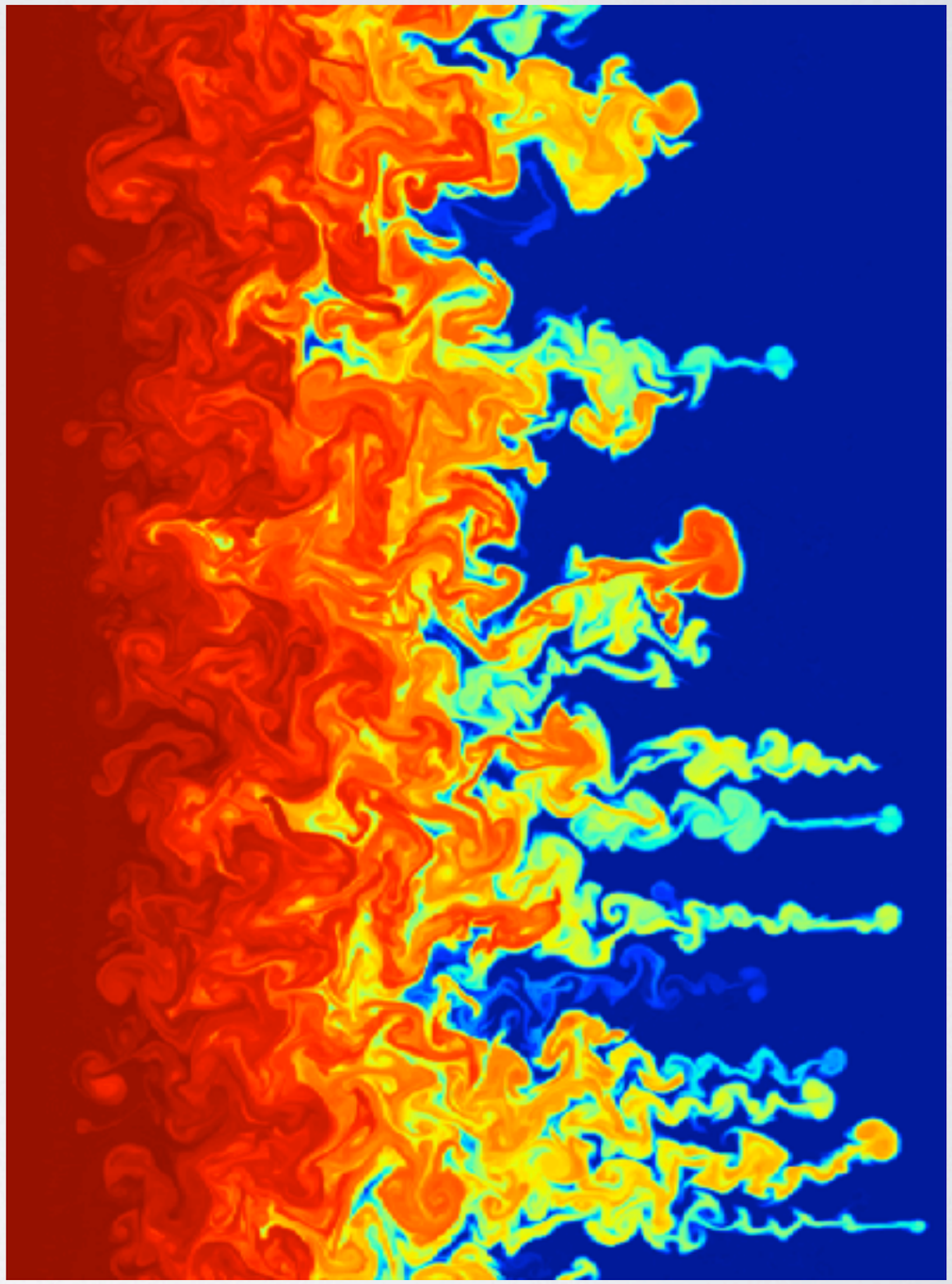


Image: Nick Earl

HYDRODYNAMICS

Multiple Hydro Methods

- MUSCL
  - 2nd order accurate Godunov solver
  - 2nd order Runge-Kutta time integration
  - multiple Riemann solvers and interpolation methods available

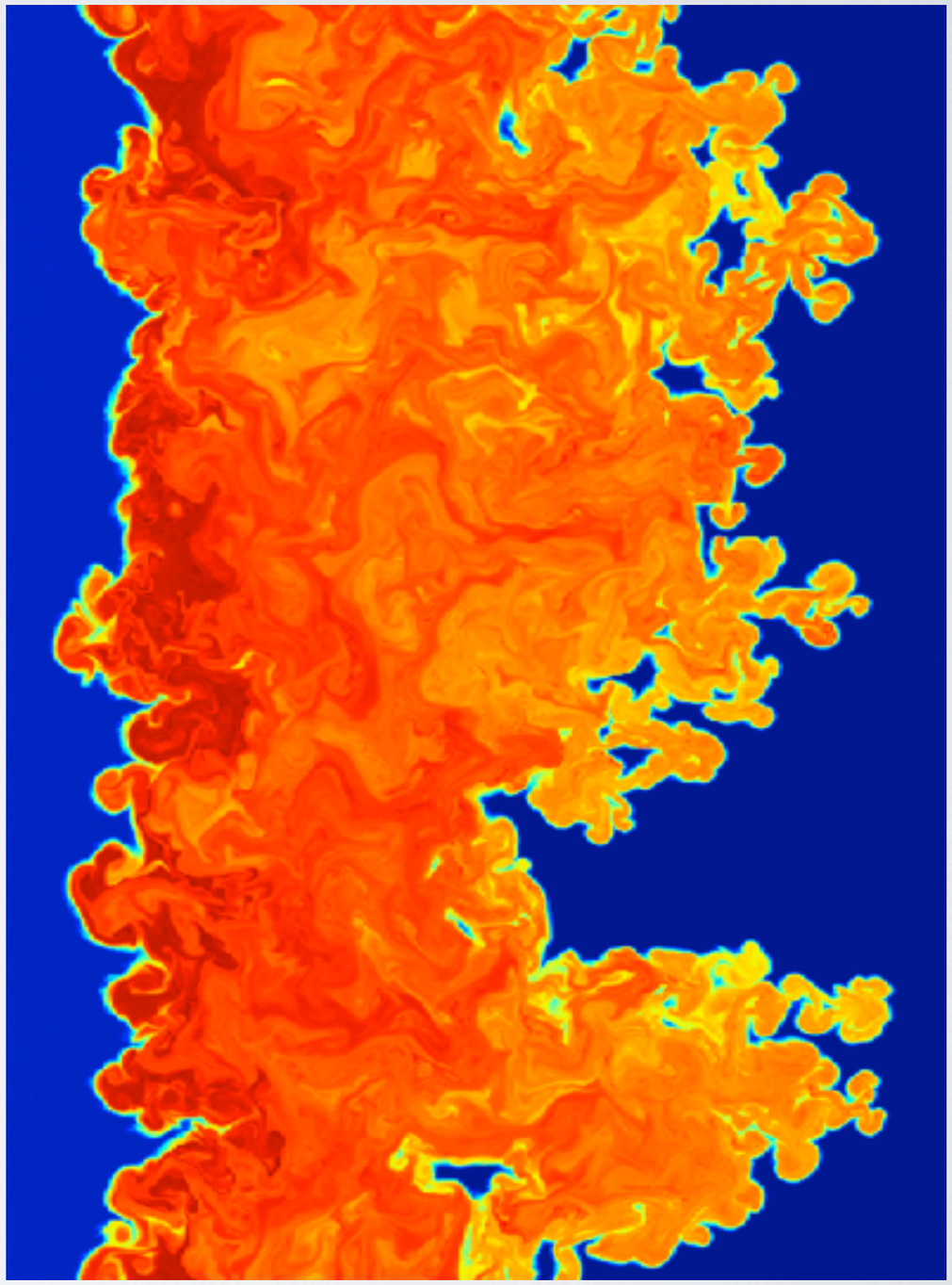
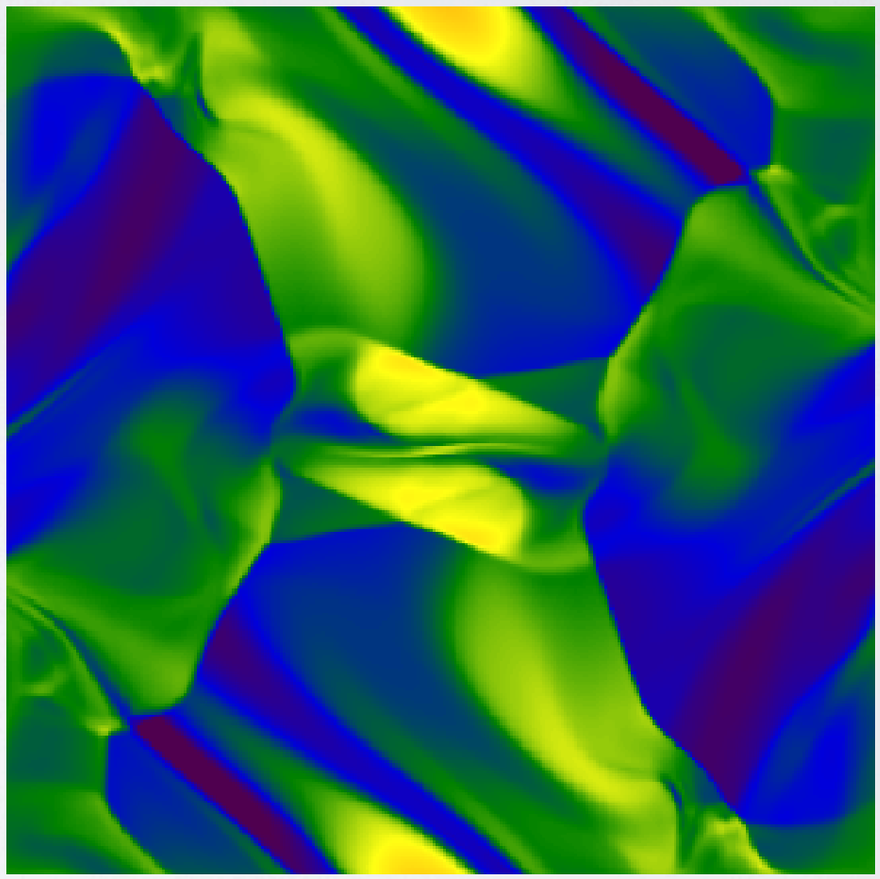


Image: Nick Earl

MHD

Multiple Hydro Methods

- Dedner
  - uses MUSCL framework above
  - hyperbolic divergence cleaning method to ensure $\nabla \cdot B = 0$
  - uses cell-centered B field
- Constrained Transport
  - magnetic field updated as the curl of the electric field
  - preserves $\nabla \cdot B = 0$ (since $\nabla \cdot (\nabla \times \mathbf{F}) = 0$)
  - needs face and edge-centered fields: more complicated



Enzo Method Paper

RADIATIVE TRANSFER

Two Rad. Transfer Methods

- Adaptive Ray Tracing
  - radiation from discreet 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

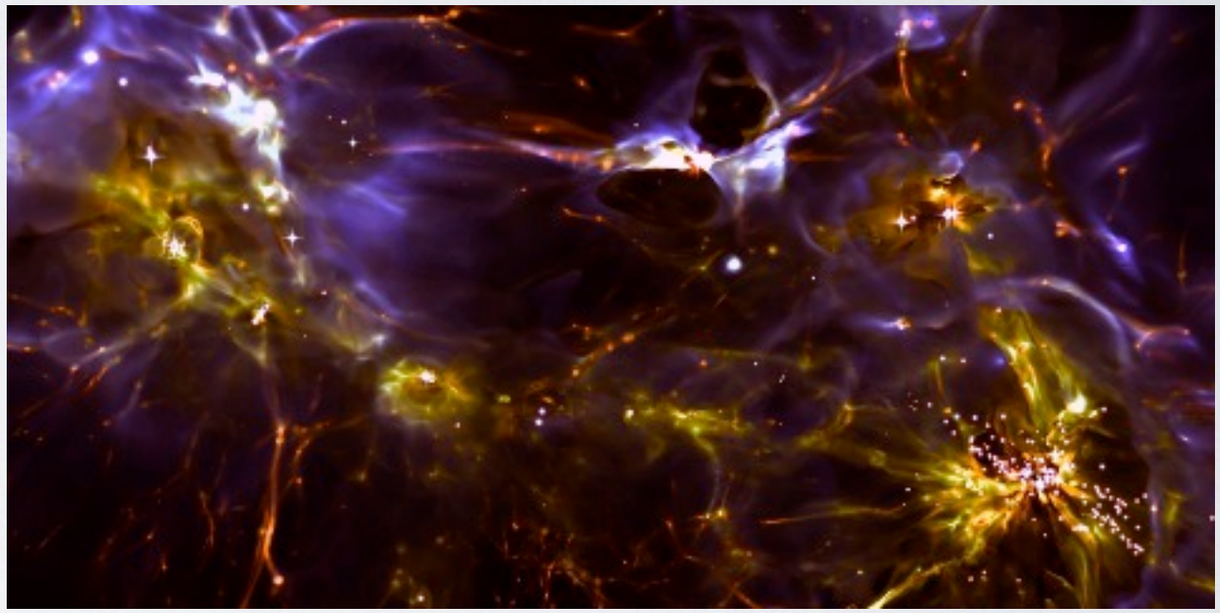


Image: John Wise

RADIATION BACKGROUNDS

- spatially smooth, time-dependent radiation fields
- UV metagalactic, ionizing backgrounds for photo-heating and ionization during Reionization (multiple models)
- Lyman-Werner soft UV fields represent radiation from first stellar sources and photo-dissociate H_2

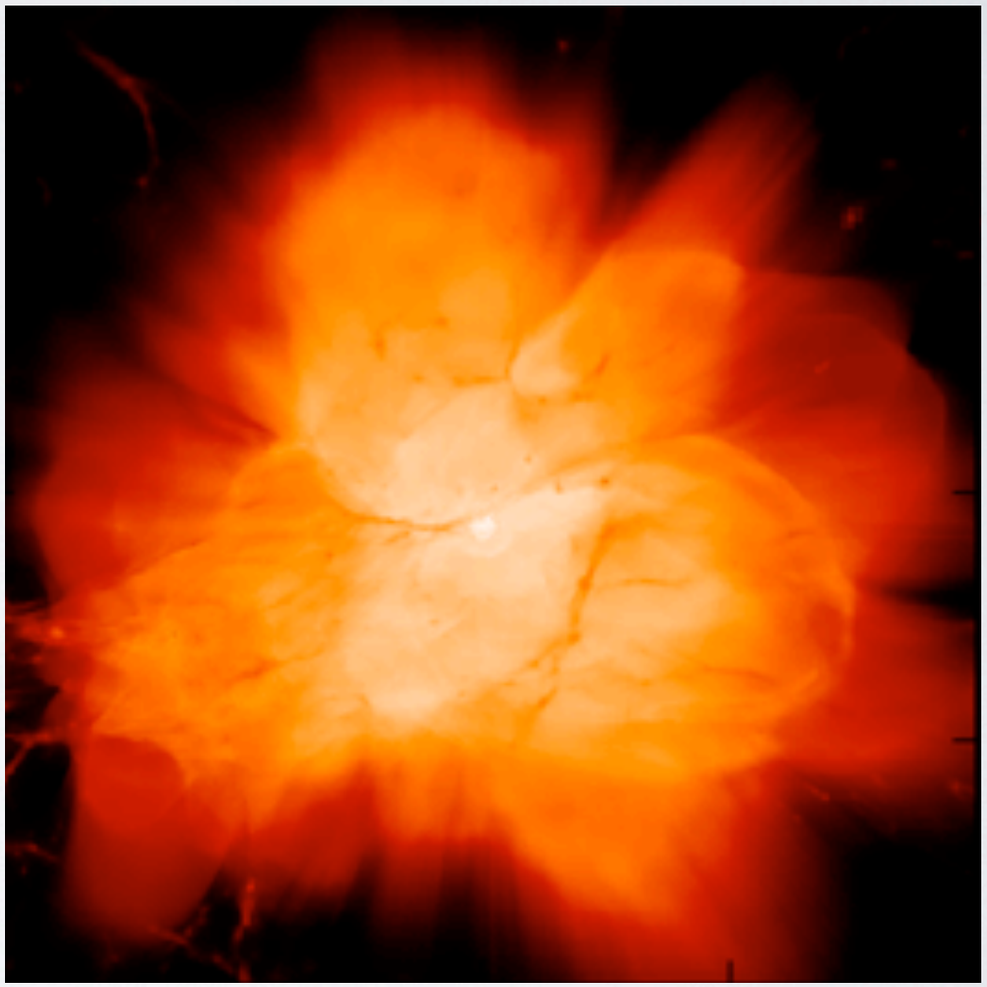
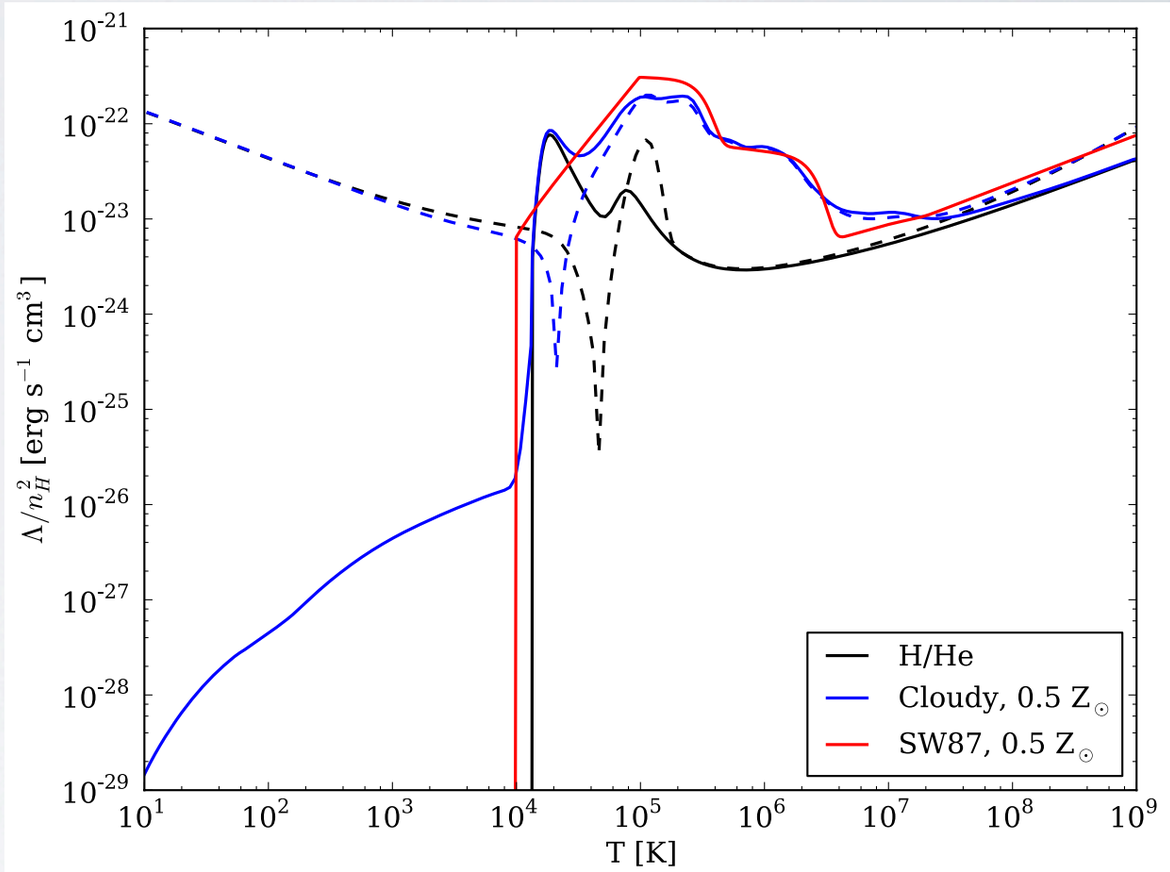


Image: Britton Smith

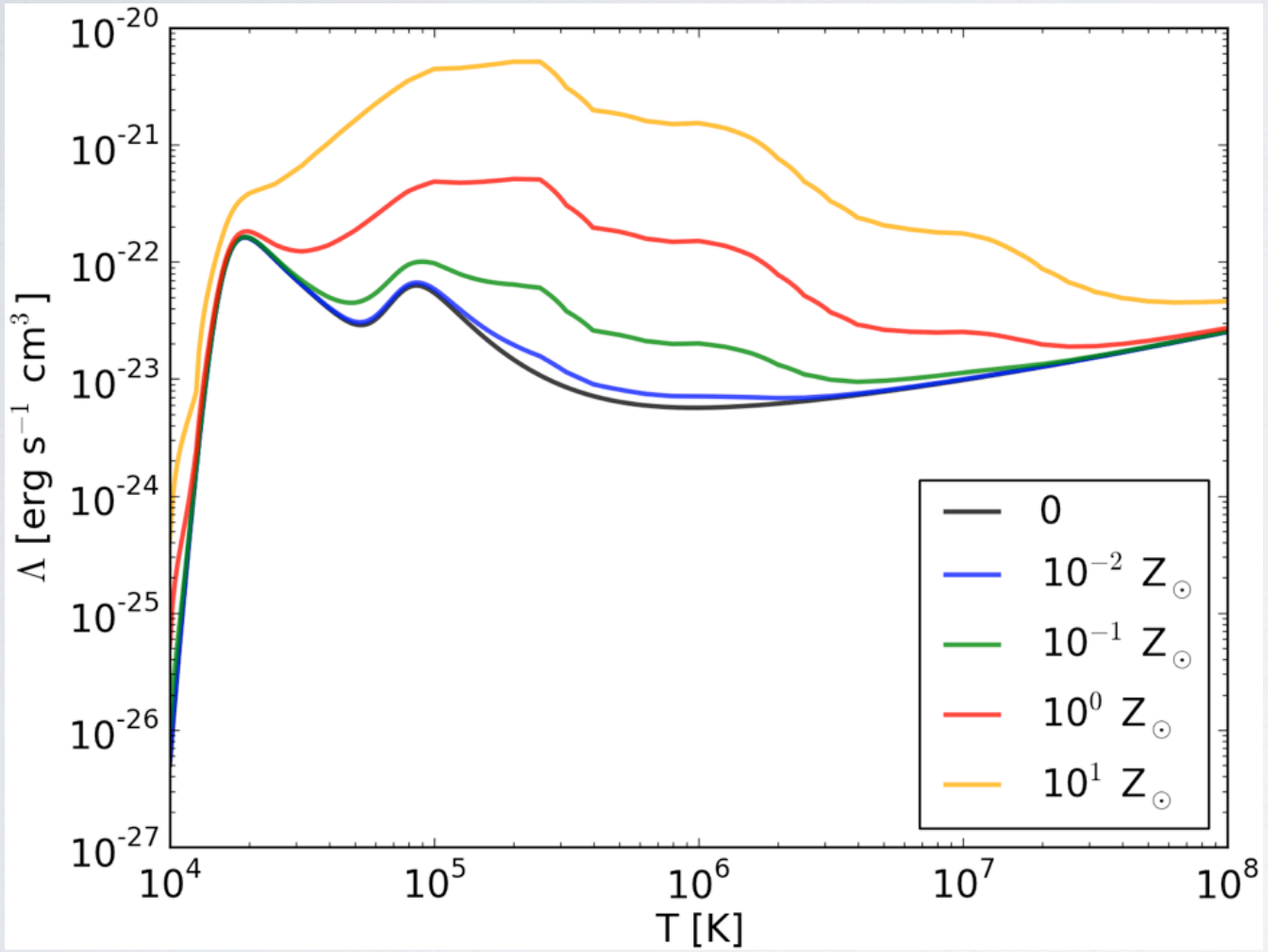
CHEMISTRY AND COOLING

- Non-equilibrium primordial chemistry
  - H, H^+, H^-, He, He^+, He^{++}, H_2, H_2^+, D, D^+, HD, e^-
  - H_2 chemistry: 2-body, 3-body channels, dust grains, chemical heating/cooling
- Metal cooling
  - simple tabulated rates ($T > 10^4$ K)
  - atomic fine-structure lines
  - Cloudy tables: density, metallicity, temperature, electron fraction, background redshift

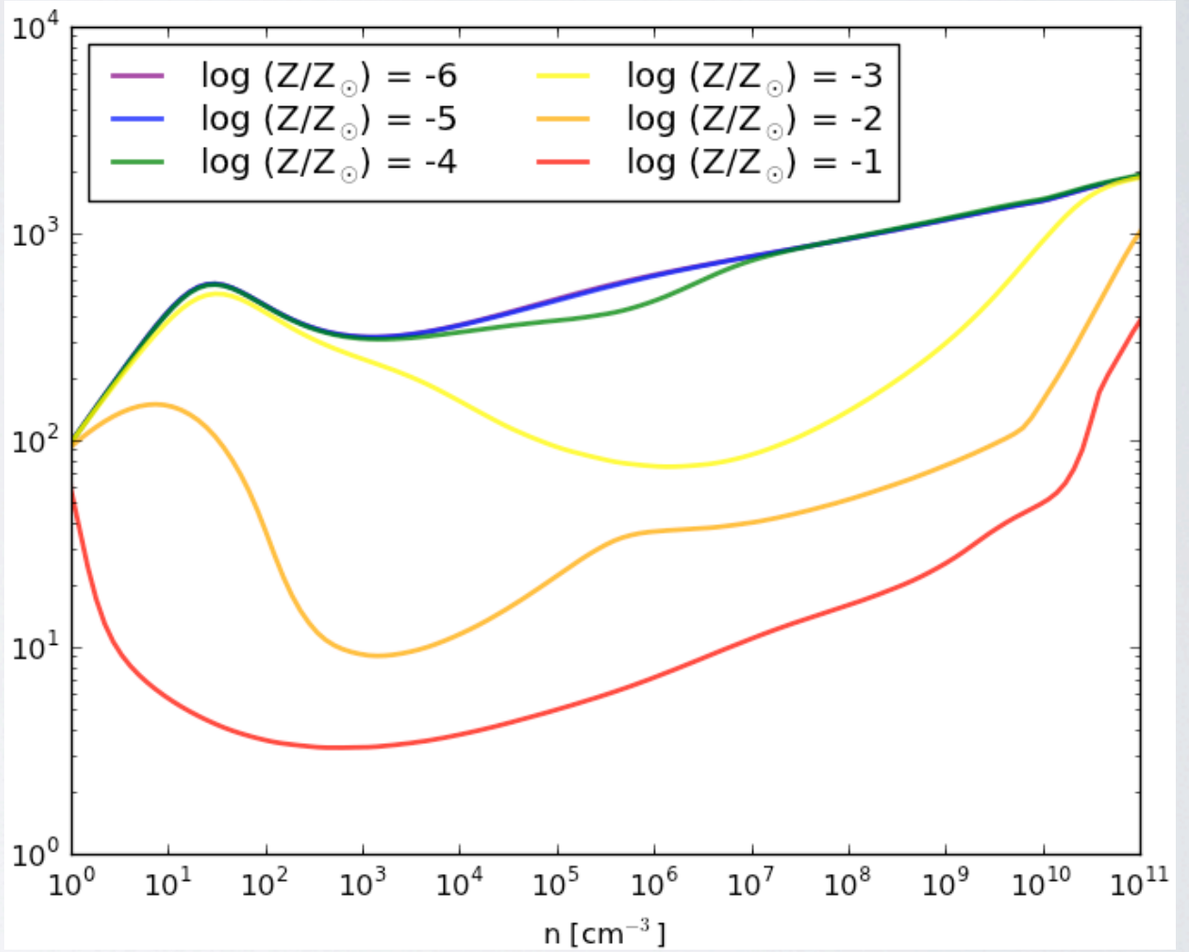


Enzo Method Paper

CHEMISTRY AND COOLING



High temperature cooling rates at various metallicities.

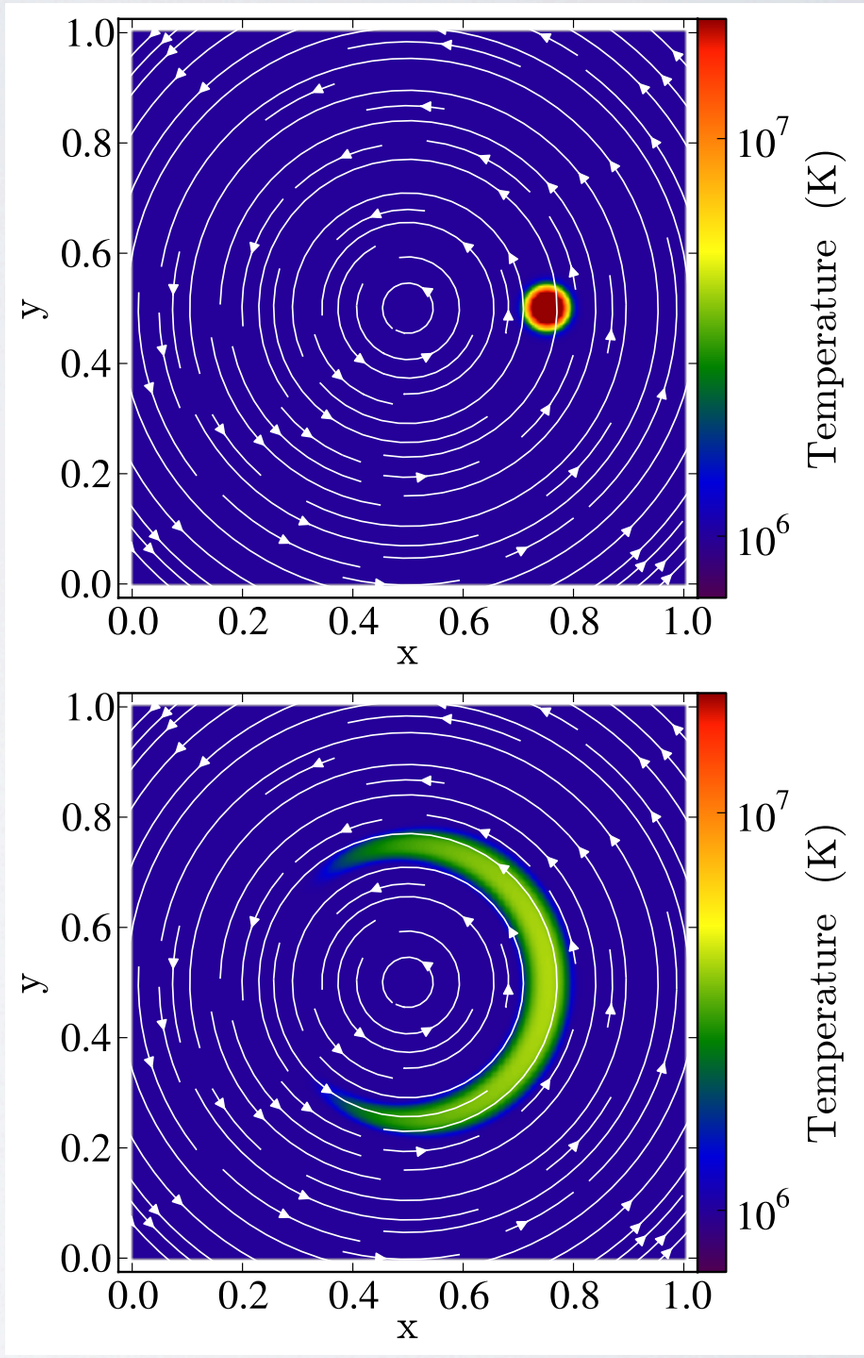


One-zone model of gas collapse at various metallicities.

THERMAL CONDUCTION

Spitzer Conduction

- heat transfer through electron Coulomb interactions
- $\kappa \sim T^{5/2}$, with saturation for sharp temperature gradients
- significant for $T > 10^7$ K (galaxy clusters)
- isotropic (hydro only) and anisotropic (MHD) available
- explicit solver: $dt \sim dx^2\, n / \kappa$: short timesteps!



Enzo Method Paper

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: John Wise

TRACER PARTICLES

- can be placed anywhere in a simulation
- used to trace hydrodynamic flow
- output field values in which particles exist
- output separately from main dataset: can be output with higher frequency

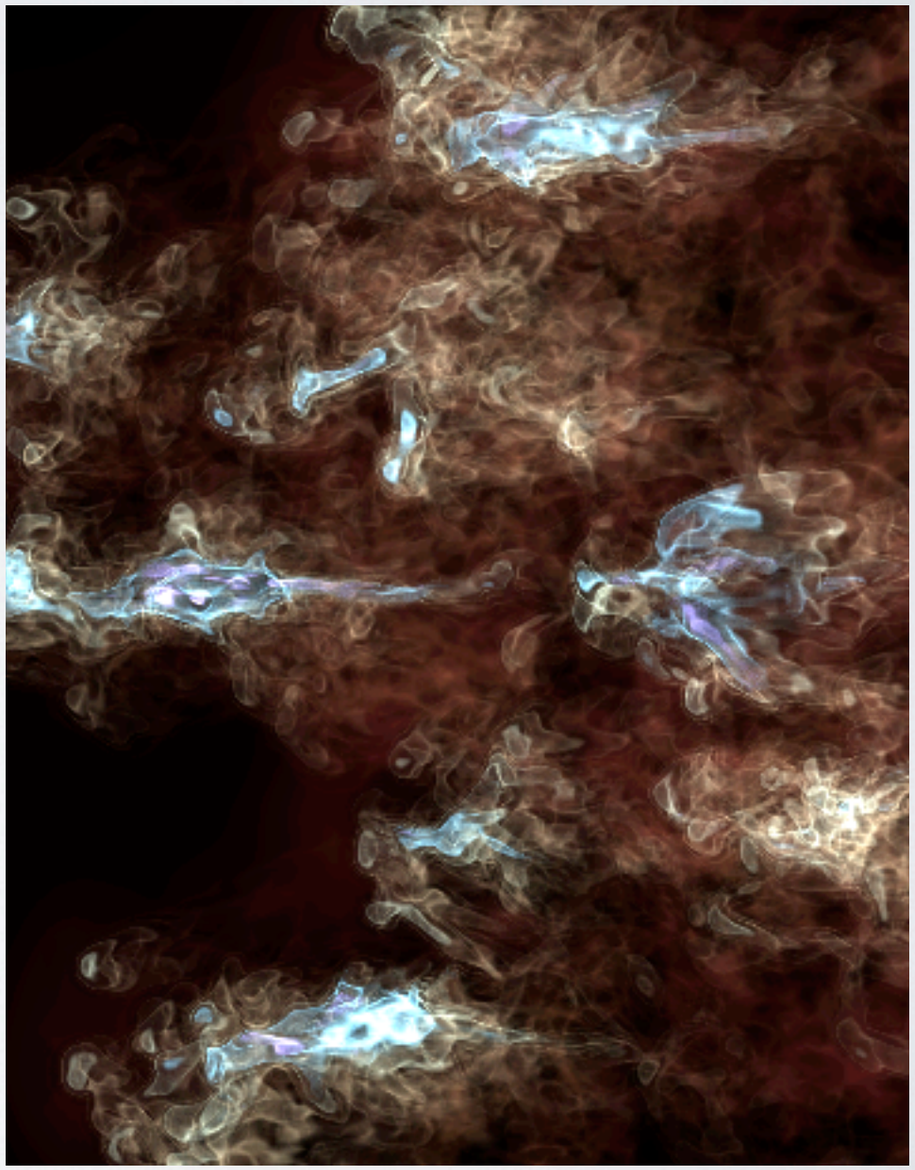
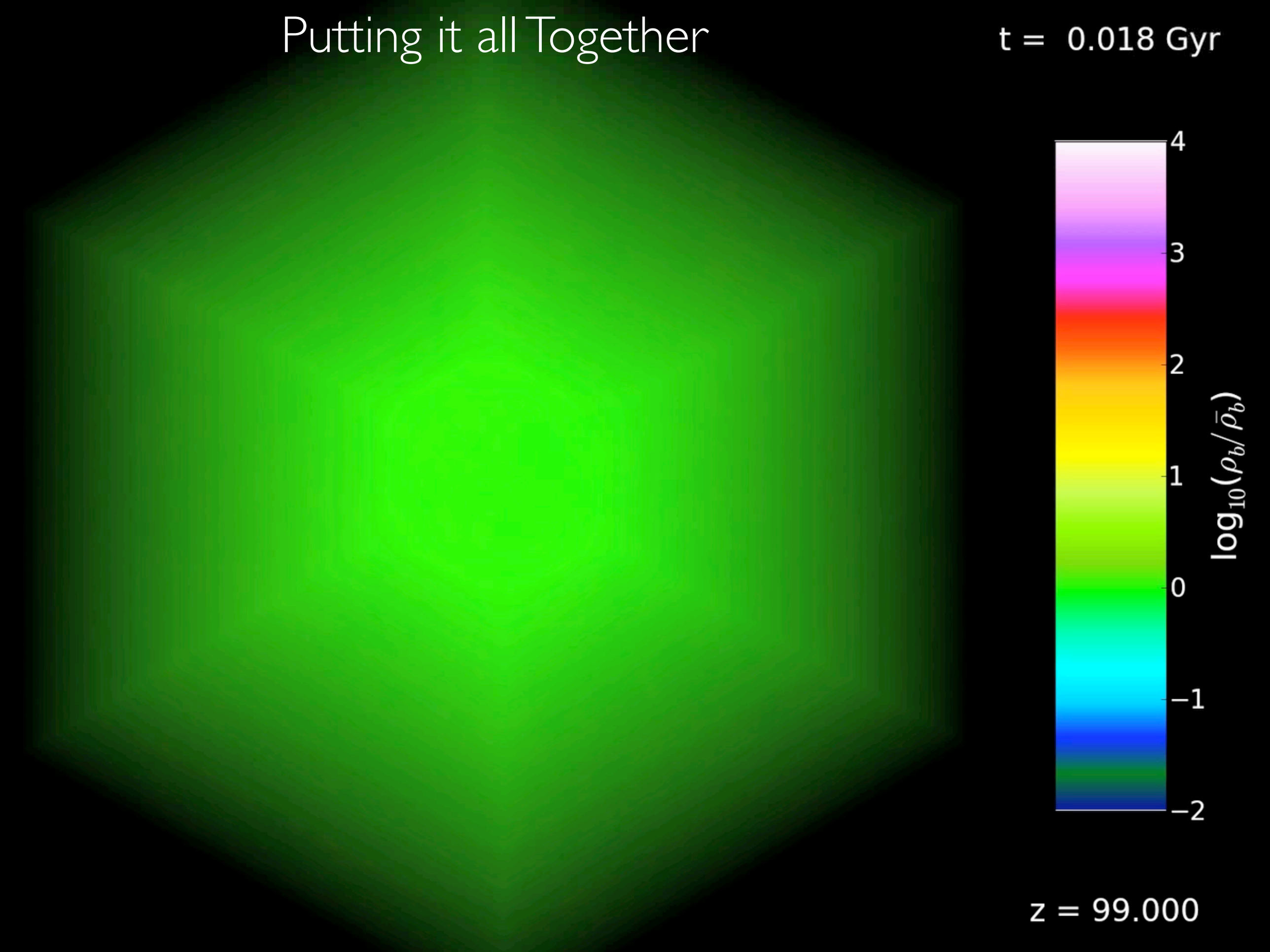


Image: Devin Silvia

Putting it all Together



PROBLEM TYPES

- Need external initial conditions files
  - Cosmology
  - Turbulence
- Enzo initializes everything
  - spheres: rotating, collapsing, colliding
  - galactic disks
  - shock tubes
  - cloud crushing
  - gravity, hydro tests
  - many, many more

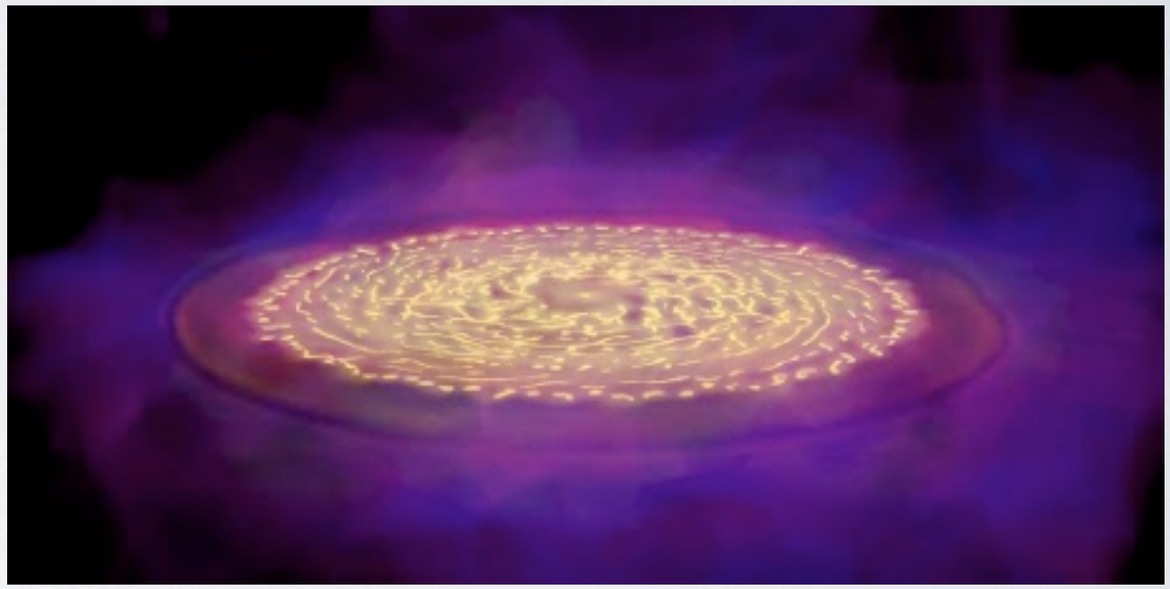


Image: Elizabeth Tasker

RESOURCES

- Enzo Webpage: enzo-project.org
- Documentation
- Email List
- IRC Channel
- yt Webpage: yt-project.org
- Mercurial tutorial: hginit.com

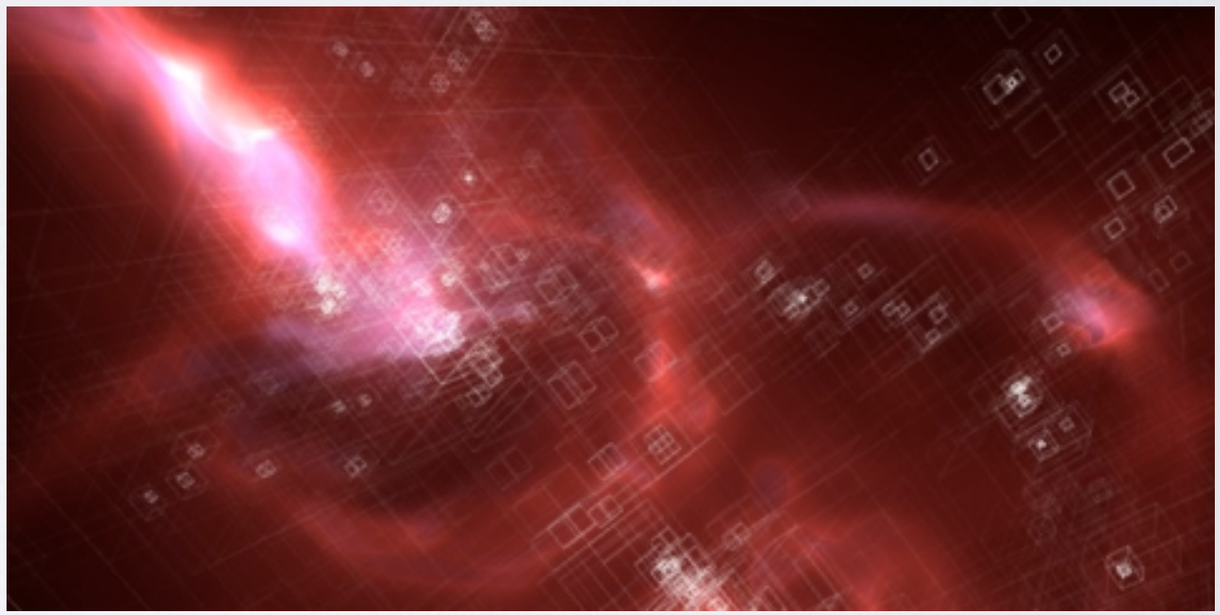


Image: Abel, Wise, Kahler

enzo-project.org

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

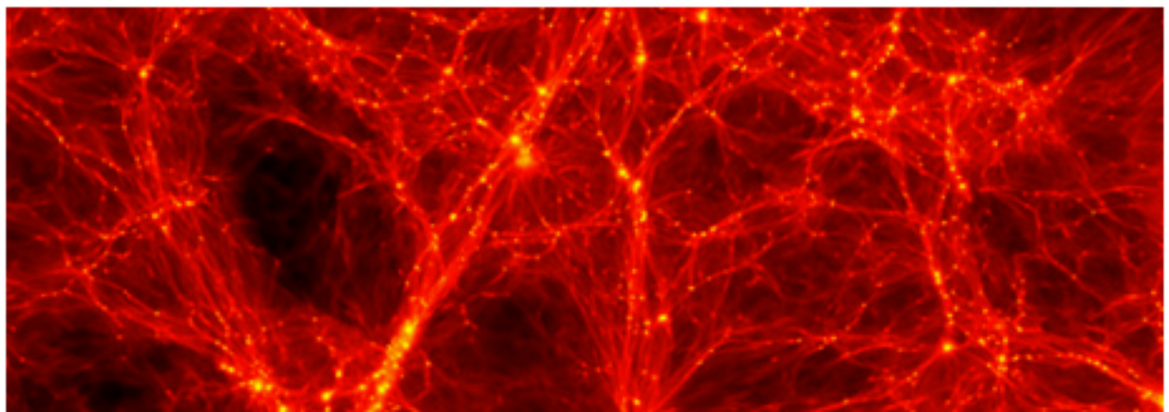
The Enzo Project

Aug 8 2013: Enzo 2.4 has been released. View the Release Notes!

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!

Community

There are several places to get help with Enzo, from mailing lists to documentation to online tutorials and recordings of workshop

enzo-project.org

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

Documentation

The Enzo Project

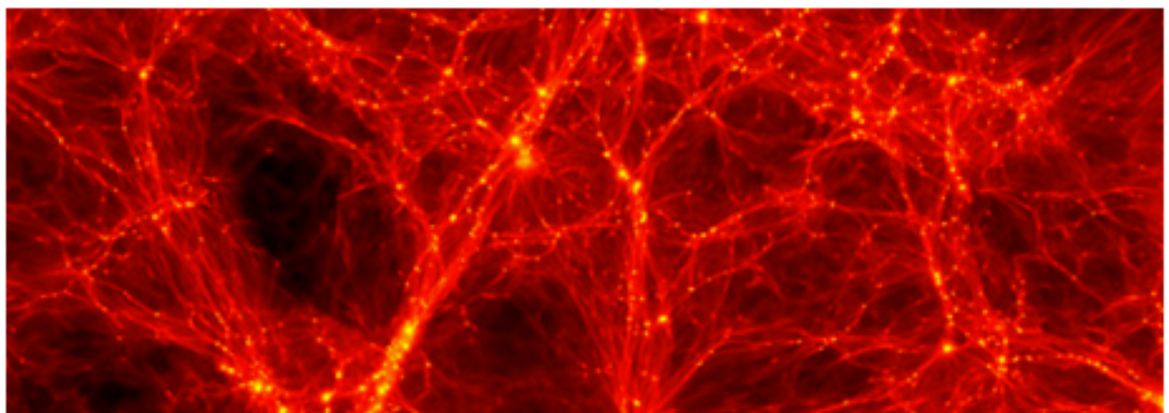
Help!

Aug 8 2013: Enzo 2.4 has been released. View the Release Notes!

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!

Community

There are several places to get help with Enzo, from mailing lists to documentation to online tutorials and recordings of workshop

DOCUMENTATION

Enzo 2.4 documentation

Welcome to Enzo's documentation!

This is the development site for Enzo, an adaptive mesh refinement (AMR), grid-based hybrid code (hydro + N-Body) which is designed to do simulations of cosmological structure formation. Links to documentation and downloads for all versions of Enzo from 1.0 on are available.

Enzo development is supported by grants AST-0808184 and OCI-0832662 from the National Science Foundation.

- Enzo Public License
- Getting Started with Enzo
  - Obtaining and Building Enzo
  - How to run an Enzo test problem
  - How to run a cosmology simulation
  - Sample inits and Enzo parameter files
  - Writing Enzo Parameter Files
  - Data Analysis Basics
  - Controlling Enzo data output
- User Guide
  - Executables, Arguments, and Outputs
  - Running Enzo
  - Measuring Simulation Progress
  - Running Enzo with CUDA
  - Running Enzo with Grackle

TABLE OF CONTENTS

Enzo Public License

Getting Started with Enzo

User Guide

Enzo Parameter List

Physics Modules in Enzo

Developer's Guide

Reference Information

Presentations Given About Enzo

DOCUMENTATION

Enzo 2.4 documentation

NEXT

Welcome to Enzo's documentation!

This is the development site for Enzo, an adaptive mesh refinement (AMR), grid-based hybrid code (hydro + N-Body) which is designed to do simulations of cosmological structure formation. Links to documentation and downloads for all versions of Enzo from 1.0 on are available.

Enzo development is supported by grants AST-0808184 and OCI-0832662 from the National Science Foundation.

- Enzo Public License
- Getting Started with Enzo
  - Obtaining and Building Enzo
  - How to run an Enzo test problem
  - How to run a cosmology simulation
  - Sample inits and Enzo parameter files
  - Writing Enzo Parameter Files
  - Data Analysis Basics
  - Controlling Enzo data output
- User Guide
  - Executables, Arguments, and Outputs
  - Running Enzo
  - Measuring Simulation Progress
  - Running Enzo with CUDA
  - Running Enzo with Grackle

TABLE OF CONTENTS

Enzo Public License

Getting Started with Enzo

User Guide

Enzo Parameter List

Physics Modules in Enzo

Developer's Guide

Reference Information

Presentations Given About Enzo

SEARCH

Go

Enter search terms or a module, class or function name.

on the internet or
on your computer

EMAIL LIST

Quick Help

Send to:

enzo-users@googlegroups.com

Archives:

http://groups.google.com/group/enzo-users

IRC CHANNEL

Live Help

Come say hello!
http://enzo-project.org/irc.html

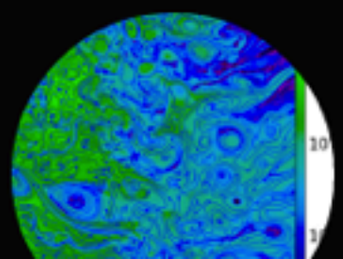
yt-project.org

yt project | About | Docs | Community | Develop | Gallery | Project Members | Quick Links

Volumetric Data Analysis

yt is a python package for analyzing and visualizing volumetric, multi-resolution data from astrophysical simulations, radio telescopes, and a burgeoning interdisciplinary community.

Get yt

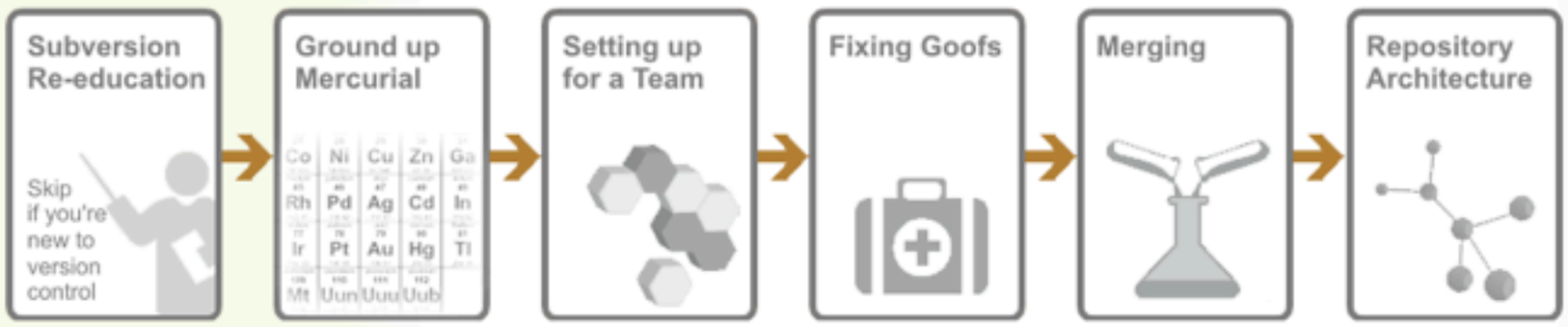


hginit.com

80
Hg
Init
200.59

Hg Init: a Mercurial tutorial

Mercurial is a modern, open source, distributed version control system, and a compelling upgrade from older systems like Subversion. In this user-friendly, six-part tutorial, Joel Spolsky teaches you the key concepts. Also, Fog Creek offers free monthly webinars that go over the basics of Mercurial.



DEVELOPMENT

Bitbucket Dashboard Teams Repositories Create owner/repository

Overview

HTTPS https://brittonsmith@bitbucket.or Share

Last updated	2014-11-05
Website	http://enzo-project.org/
Language	C++
Access level	Admin (revoke)

33 Branches	12 Tags
102 Forks	35 Watchers

Edit README

= ENZO =

ENZO IS AN OPEN SOURCE CODE. We encourage you to take it, inspect it, use it, and contribute back any changes you have made. We strive to make the the Enzo community a community of *developers*.

== RESOURCES ==

Enzo's main webpage is:

- http://enzo-project.org

Enzo is developed in the open on bitbucket.org:

- https://bitbucket.org/enzo/enzo-dev

Documentation, including instructions for compilation, can be found at:

- http://enzo-project.org/docs/2.4/

Please subscribe to the Enzo Users' mailing list at:

- https://groups.google.com/forum/#!forum/enzo-users

If you are interested in Enzo development, you may want to sign up for the Enzo Developer's mailing list as well:

- https://groups.google.com/forum/#!forum/enzo-dev

Invite users to this repo

Send invitation

Recent activity

Minor FLD solver update to skip linear …
Pull request #254 commented on in enzo/enzo-dev
Sam Skillman · 2014-11-06

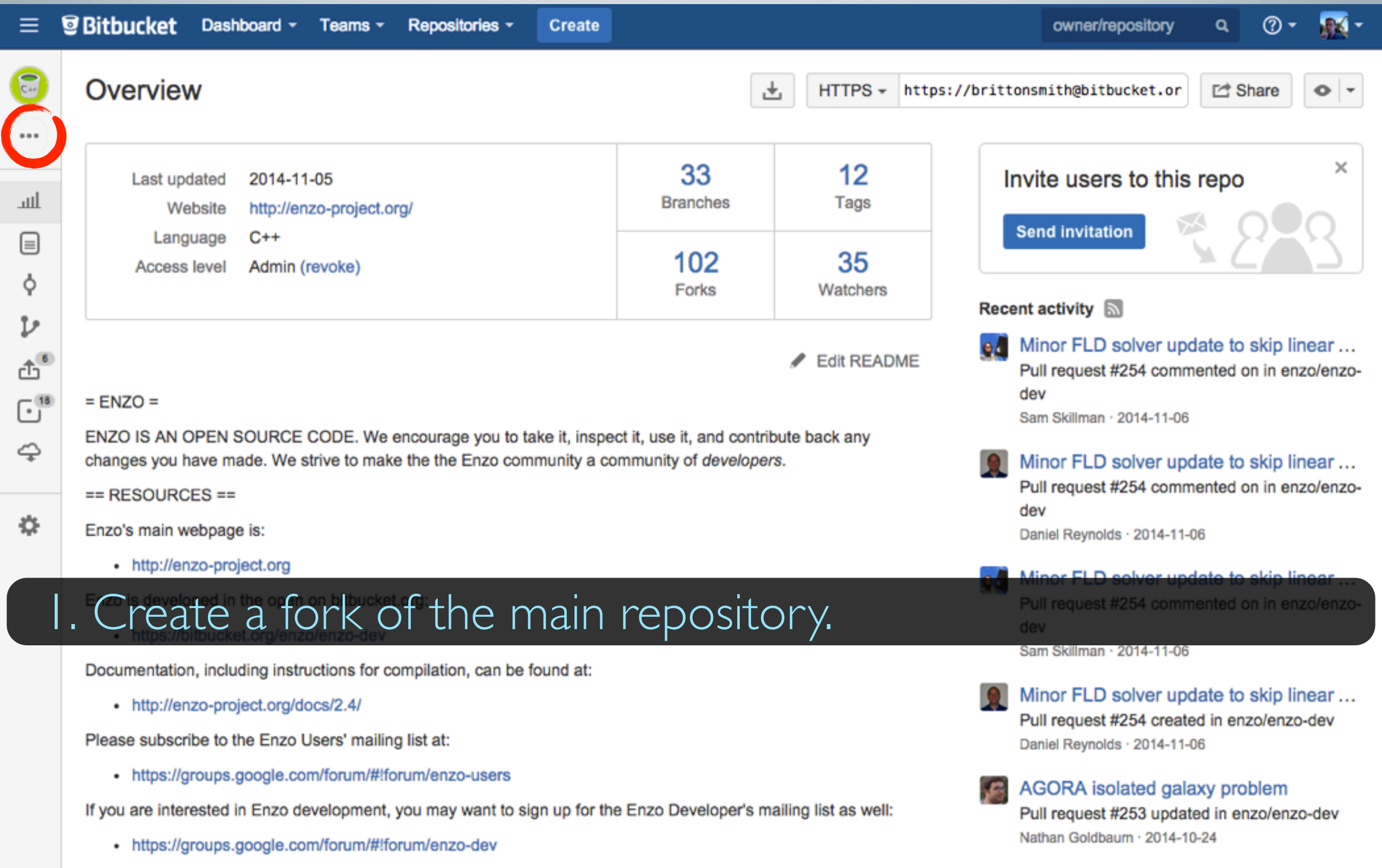
Minor FLD solver update to skip linear …
Pull request #254 commented on in enzo/enzo-dev
Daniel Reynolds · 2014-11-06

Minor FLD solver update to skip linear …
Pull request #254 commented on in enzo/enzo-dev
Sam Skillman · 2014-11-06

Minor FLD solver update to skip linear …
Pull request #254 created in enzo/enzo-dev
Daniel Reynolds · 2014-11-06

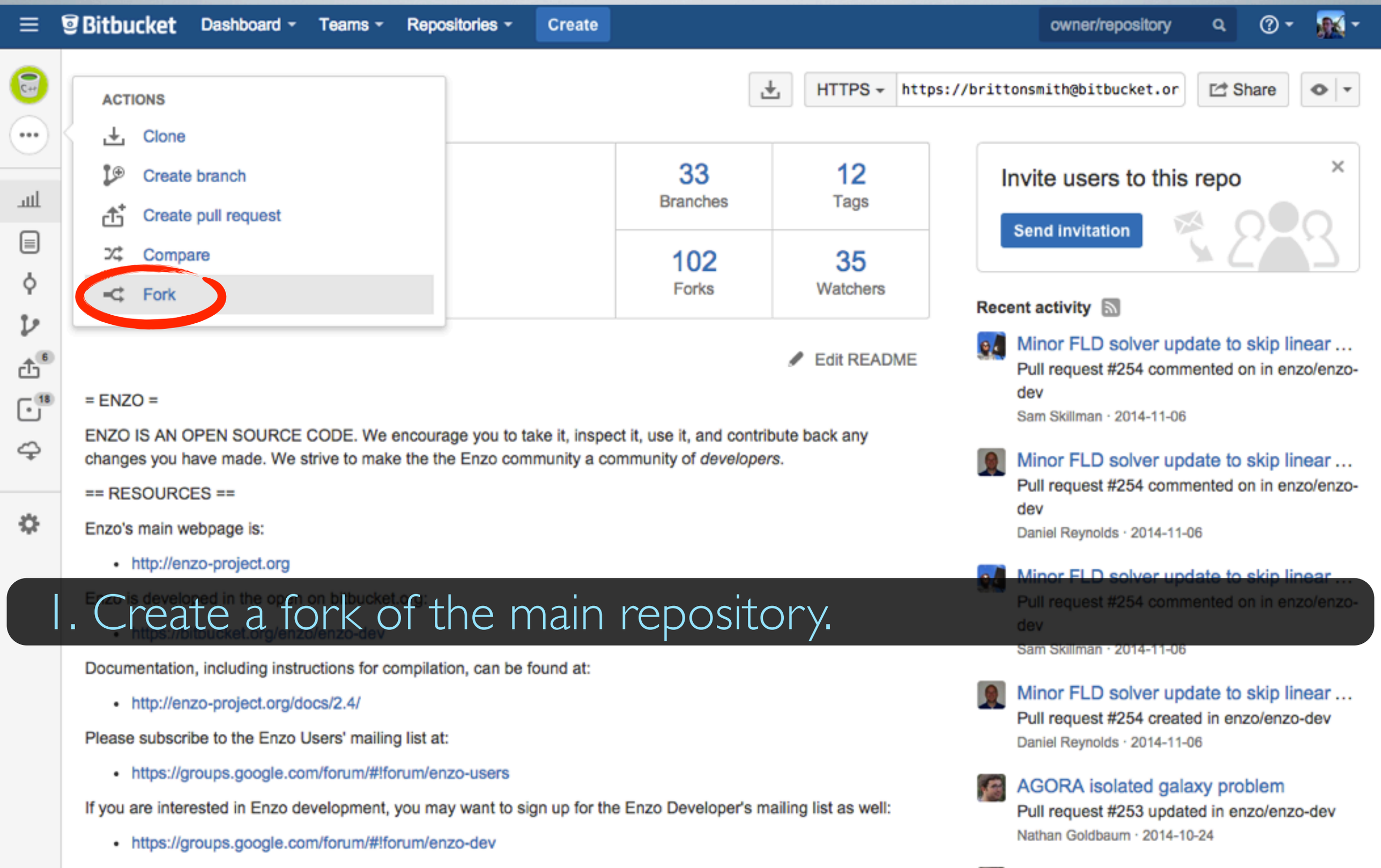
AGORA isolated galaxy problem
Pull request #253 updated in enzo/enzo-dev
Nathan Goldbaum · 2014-10-24

DEVELOPMENT



1. Create a fork of the main repository.

DEVELOPMENT



1. Create a fork of the main repository.

DEVELOPMENT

Bitbucket Dashboard Teams Repositories Create owner/repository

ACTIONS
Clone
Create branch
Create pull request
Compare
Fork

HTTPS https://brittonsmith@bitbucket.or Share

33 Branches
12 Tags
102 Forks
35 Watchers

Invite users to this repo
Send invitation

Recent activity

```
grackle:~[1] hg clone https://bitbucket.org/brittonsmith/enzo-dev
warning: bitbucket.org certificate with fingerprint 24:9c:45:8b:9c:aa:ba:55:4e:01:6d:
eck hostfingerprints or web.cacerts config setting)
destination directory: enzo-dev
requesting all changes
adding changesets
changesets [
                                    <=>
```

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

= ENZO

ENZO IS ... changes

== RESO

Enzo's ...

- http://enzo-project.org

Enzo is developed in the open on bitbucket.org:

- https://bitbucket.org/enzo/enzo-dev

Documentation, including instructions for compilation, can be found at:

- http://enzo-project.org/docs/2.4/

Please subscribe to the Enzo Users' mailing list at:

- https://groups.google.com/forum/#!forum/enzo-users

If you are interested in Enzo development, you may want to sign up for the Enzo Developer's mailing list as well:

- https://groups.google.com/forum/#!forum/enzo-dev

Minor FLD solver update to skip linear ... on in enzo/enzo-

Minor FLD solver update to skip linear ...
Pull request #254 commented on in enzo/enzo-dev
Sam Skillman · 2014-11-06

Minor FLD solver update to skip linear ...
Pull request #254 created in enzo/enzo-dev
Daniel Reynolds · 2014-11-06

AGORA isolated galaxy problem
Pull request #253 updated in enzo/enzo-dev
Nathan Goldbaum · 2014-10-24

DEVELOPMENT

Bitbucket Dashboard Teams Repositories Create owner/repository

Pull requests

#254 OPEN enzo-dev_reynol… week-of-code → week-of-code

Merge Edit Decline Approve 0

Minor FLD solver update to skip linear solve when possible

Overview Commits Activity

Author Daniel Reynolds

Reviewers

Watch this pull request

Learn about pull requests

Description Updated gFLDSplit_Evolve() routine to skip the solve if possible (useful for reionization runs before stars are made).

Comments (3)

Sam Skillman

This ever so slightly breaks the tests in this way:

```
Items are not equal: Output times not equal.
 ACTUAL: 0.70000038133545
```

http://srs.slac.stanford.edu/hudson/job/enzo-dev/141/testReport/

Reply • Delete • Create task • 2014-11-06

Daniel Reynolds AUTHOR

Interesting! Actually it's not the dt, TimeUnits bit, since that happens at the end of the linear solve anyway. So if I skip the linear solve, I still need to scale the units back to their normalized values.

Instead, it looks like in this test problem, the first few time steps meet the criteria for skipping the call to the HYPRE solver. Eventually, the species get enough out of equilibrium so that the solves kick in.

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

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.

DEVELOPMENT

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

Many people have contributed to the development of Enzo -- here's just a short list of the people who have recently contributed, in alphabetical order:

* Tom Abel tabel@stanford.edu
* James Bordner jobordner@ucsd.edu
* Greg Bryan gbryan@astro.columbia.edu
* Renyue Cen cen@astro.princeton.edu
* Dave Collins dcollins@physics.ucsd.edu
* Brian Crosby crosby.bd@gmail.com
* Nathan Goldbaum goldbaum@ucolick.org
* Oliver Hahn hahn@phys.ethz.ch
* Robert Harkness harkness@sdsc.edu
* Elizabeth Harper-Clark h-clark@astro.utoronto.ca
* Cameron Hummels chummels@gmail.com
* Ji-hoon Kim mornkr@ucolick.org
* Daegene Koh dkoh30@gatech.edu
* Alexei Kritsuk akritsuk@ucsd.edu
* Michael Kuhlen mqk@astro.berkeley.edu
* Eve Lee elee@cita.utoronto.ca
* Yuan Li yuan@astro.columbia.edu
* Michael Norman mlnorman@ucsd.edu
* JS Oishi jsoishi@gmail.com
* Brian O'Shea oshea@msu.edu
* Pascal Paschos ppaschos@minbari.ucsd.edu
* Carolyn Peruta perutaca@msu.edu
* Alex Razoumov razoumov@gmail.com
* Dan Reynolds reynolds@smu.edu
* Munier Salem msalem@astro.columbia.edu
* Christine Simpson csimpson@astro.columbia.edu
* Samuel Skillman samskillman@gmail.com
* Stephen Skory s@skory.us
* Britton Smith brittonsmith@gmail.com
* Geoffrey So gsiisg@gmail.com
* Elizabeth Tasker tasker@astro1.sci.hokudai.ac.jp
* Matthew Turk matthewturk@gmail.com
* Rick Wagner rwagner@physics.ucsd.edu
* Peng Wang pengw@slac.stanford.edu
* John Wise jwise@physics.gatech.edu
* Hao Xu haxu@ucsd.edu
* Fen Zhao fenzhao@stanford.edu

DEVELOPERS

Many people have contributed to the development of Enzo -- here's just a short list of the people who have recently contributed, in alphabetical order:

* Tom Abel	tabel@stanford.edu
* James Bordner	jobordner@ucsd.edu
* Greg Bryan	gbryan@astro.columbia.edu
* Renyue Cen	cen@astro.princeton.edu
* Dave Collins	dcollins@physics.ucsd.edu
* Brian Crosby	crosby.bd@gmail.com
* Nathan Goldbaum	goldbaum@ucolick.org
* Oliver Hahn	hahn@phys.ethz.ch
* Robert Harkness	harkness@sdsc.edu
* Elizabeth Harper-Clark	h-clark@astro.utoronto.ca
* Cameron Hummels	chummels@gmail.com
* Ji-hoon Kim	mornkr@ucolick.org
* Daegene Koh	dkoh30@gatech.edu
* Alexei Kritsuk	akritsuk@ucsd.edu
* Michael Kuhlen	mqk@astro.berkeley.edu
* Eve Lee	elee@cita.utoronto.ca
* Yuan Li	yuan@astro.columbia.edu
* Michael Norman	mlnorman@ucsd.edu
* JS Oishi	jsoishi@gmail.com
* Brian O'Shea	oshea@msu.edu
* Pascal Paschos	ppaschos@minbari.ucsd.edu
* Carolyn Peruta	perutaca@msu.edu
* Alex Razoumov	razoumov@gmail.com
* Dan Reynolds	reynolds@smu.edu
* Munier Salem	msalem@astro.columbia.edu
* Christine Simpson	csimpson@astro.columbia.edu
* Samuel Skillman	samskillman@gmail.com
* Stephen Skory	s@skory.us
* Britton Smith	brittonsmith@gmail.com
* Geoffrey So	gsiisg@gmail.com
* Elizabeth Tasker	tasker@astro1.sci.hokudai.ac.jp
* Matthew Turk	matthewturk@gmail.com
* Rick Wagner	rwagner@physics.ucsd.edu
* Peng Wang	pengw@slac.stanford.edu
* John Wise	jwise@physics.gatech.edu
* [illegible] Xu	haxu@ucsd.edu
* Fen Zhao	fenzhao@stanford.edu

- You!

THANK YOU