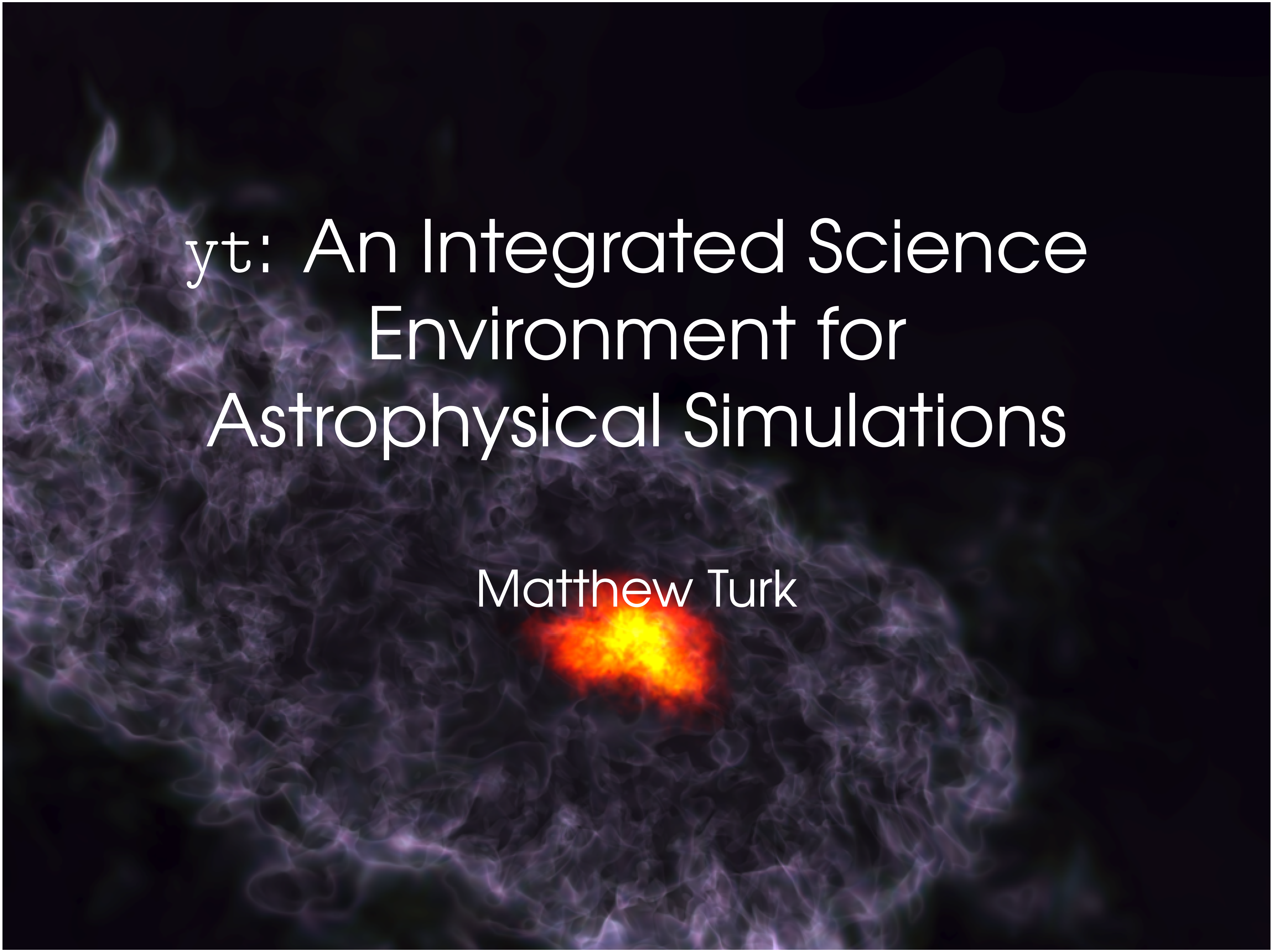


`yt`: An Integrated Science Environment for Astrophysical Simulations

Matthew Turk

There is only one sky.

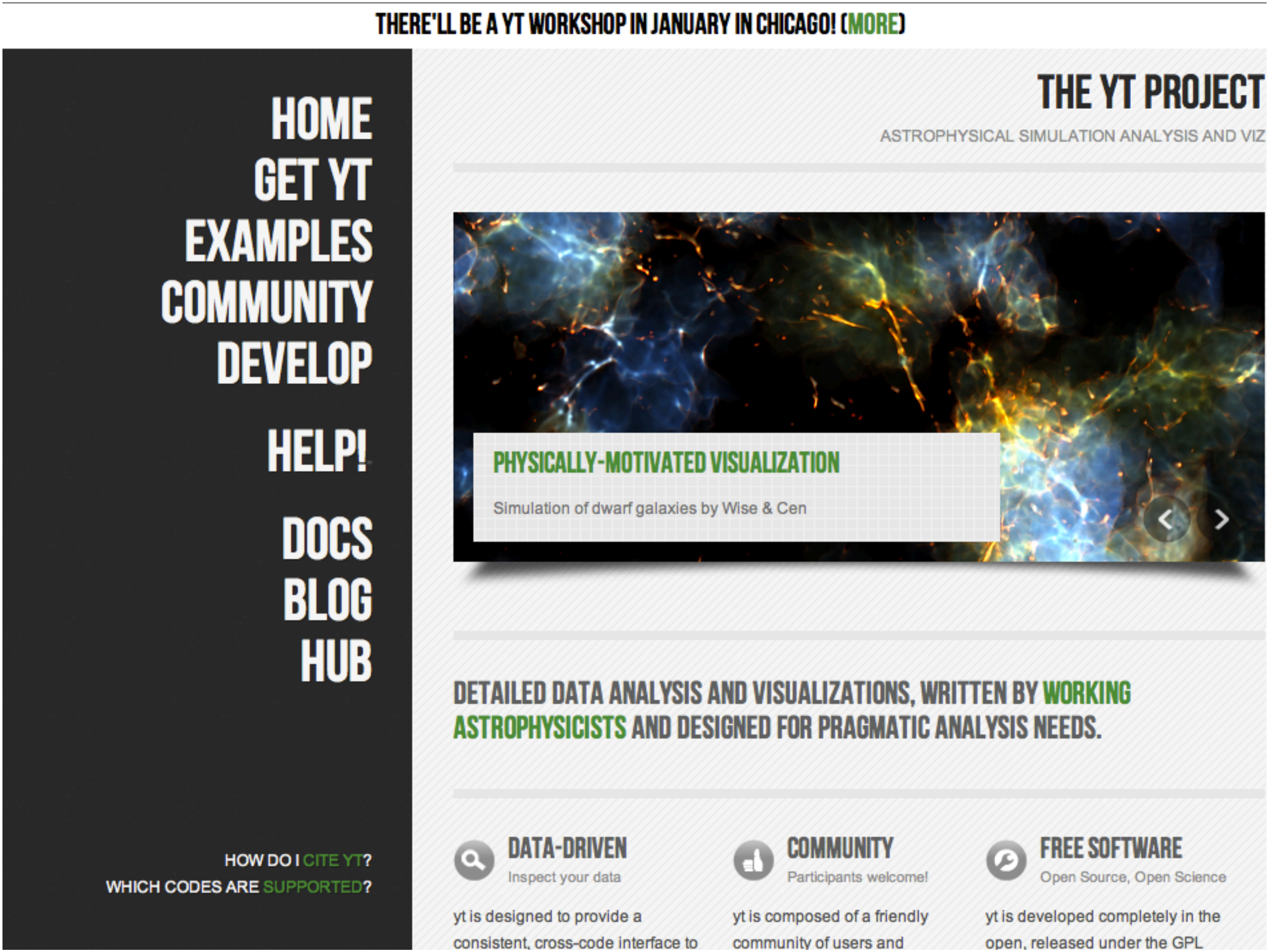
(but there are many simulation codes)

Different methods,
data structures,
assumptions,
IO methods,
units,
variable names,
...

aimeric@acapulco

Analysis.

astro-ph/1011.3514
yt-project.org



THERE'LL BE A YT WORKSHOP IN JANUARY IN CHICAGO! (MORE)

THE YT PROJECT

ASTROPHYSICAL SIMULATION ANALYSIS AND VIZ

PHYSICALLY-MOTIVATED VISUALIZATION

Simulation of dwarf galaxies by Wise & Cen

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

HOW DO I CITE YT?

WHICH CODES ARE SUPPORTED?

DATA-DRIVEN

Inspect your data

yt is designed to provide a consistent, cross-code interface to

COMMUNITY

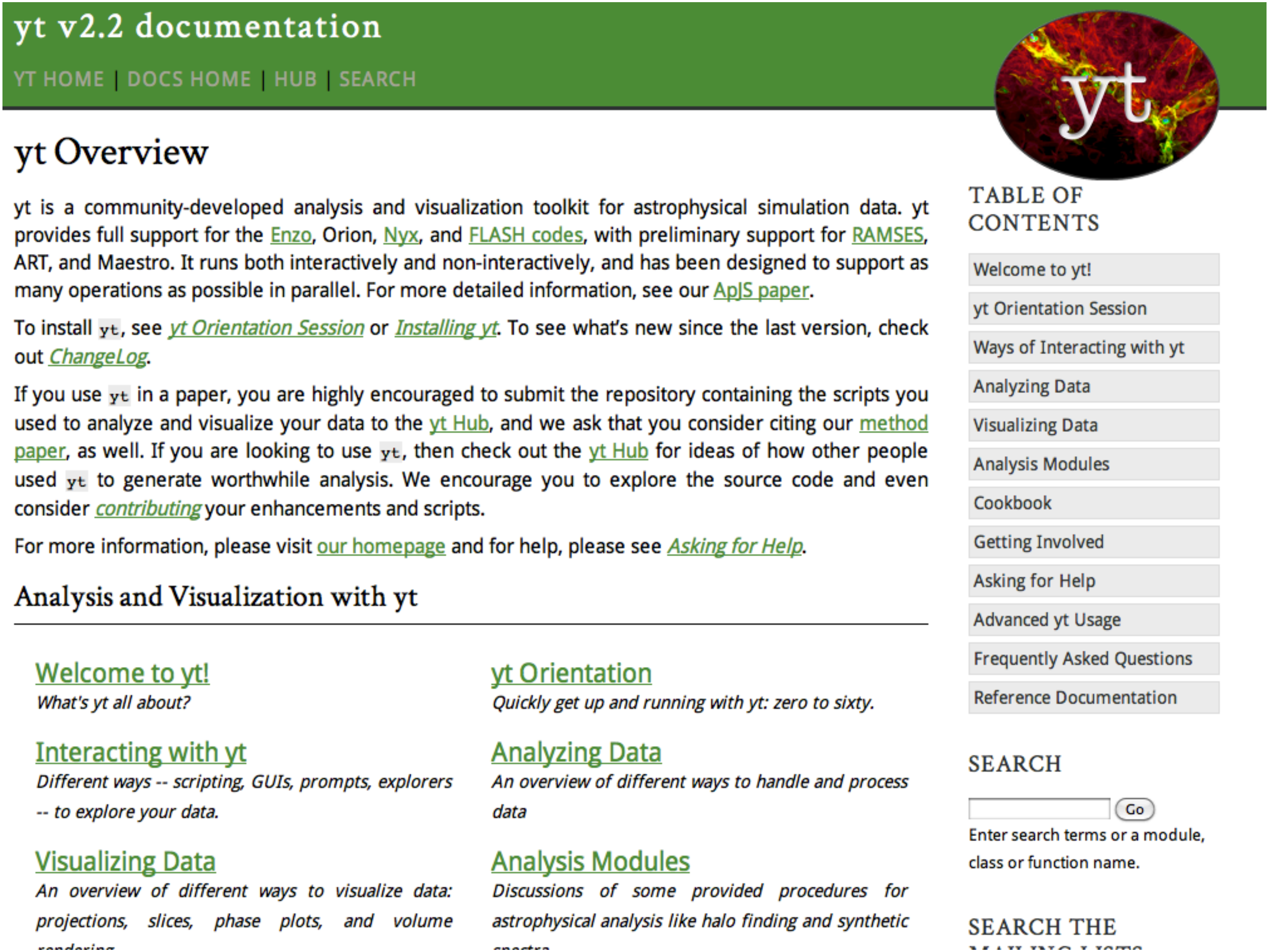
Participants welcome!

yt is composed of a friendly community of users and

FREE SOFTWARE

Open Source, Open Science

yt is developed completely in the open, released under the GPL



yt v2.2 documentation

YT HOME | DOCS HOME | HUB | SEARCH

yt Overview

yt is a community-developed analysis and visualization toolkit for astrophysical simulation data. yt provides full support for the Enzo, Orion, Nyx, and FLASH codes, with preliminary support for RAMSES, ART, and Maestro. It runs both interactively and non-interactively, and has been designed to support as many operations as possible in parallel. For more detailed information, see our ApJS paper.

To install `yt`, see *yt Orientation Session* or *Installing yt*. To see what's new since the last version, check out *ChangeLog*.

If you use `yt` in a paper, you are highly encouraged to submit the repository containing the scripts you used to analyze and visualize your data to the yt Hub, and we ask that you consider citing our method paper, as well. If you are looking to use `yt`, then check out the yt Hub for ideas of how other people used `yt` to generate worthwhile analysis. We encourage you to explore the source code and even consider *contributing* your enhancements and scripts.

For more information, please visit our homepage and for help, please see *Asking for Help*.

Analysis and Visualization with yt

Welcome to yt!

What's yt all about?

yt Orientation

Quickly get up and running with yt: zero to sixty.

Interacting with yt

Different ways -- scripting, GUIs, prompts, explorers -- to explore your data.

Analyzing Data

An overview of different ways to handle and process data

Visualizing Data

An overview of different ways to visualize data: projections, slices, phase plots, and volume rendering

Analysis Modules

Discussions of some provided procedures for astrophysical analysis like halo finding and synthetic spectra

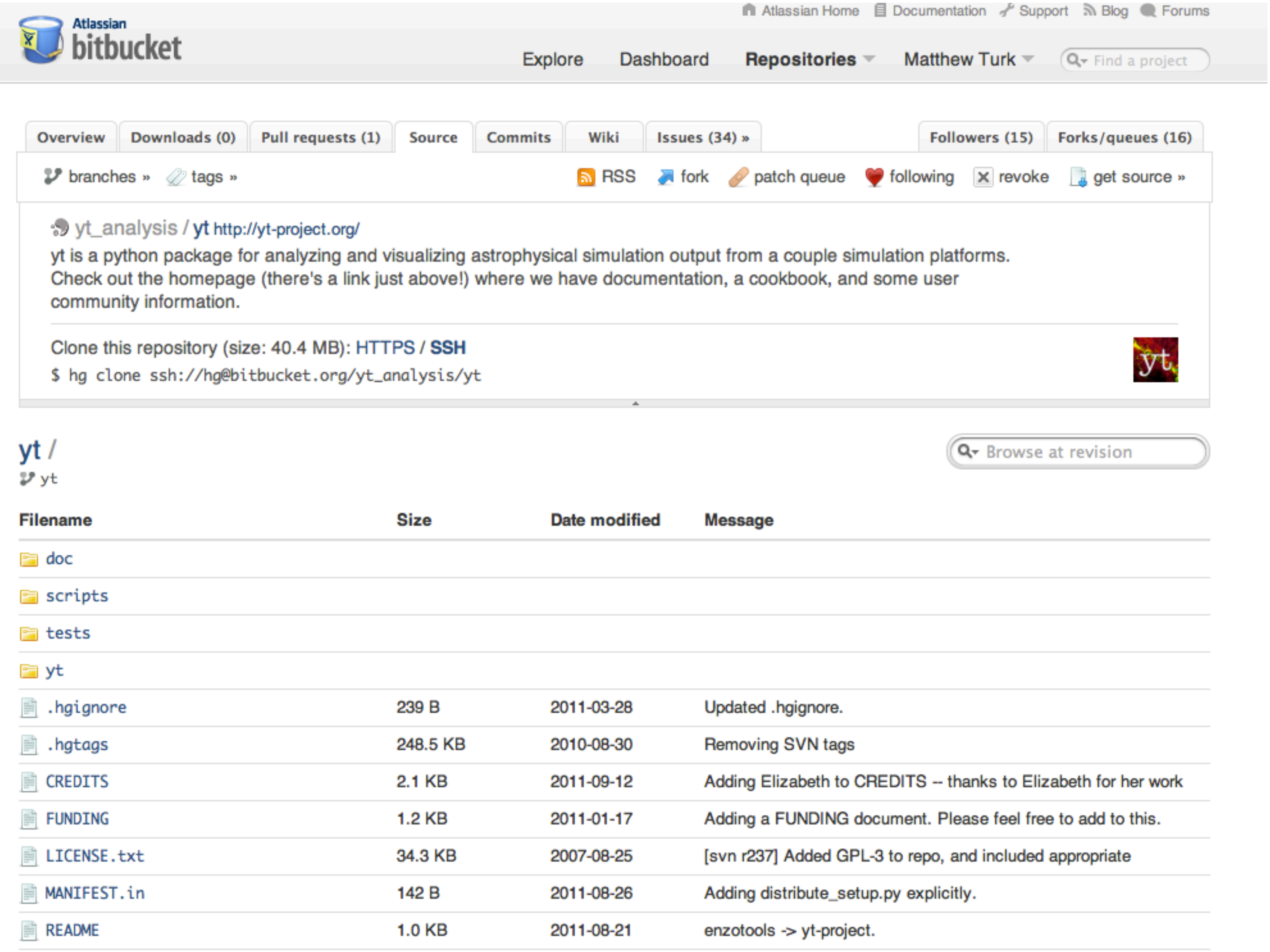
TABLE OF CONTENTS

SEARCH

Go

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

SEARCH THE MAILING LISTS



yt_analysis / yt http://yt-project.org/

yt is a python package for analyzing and visualizing astrophysical simulation output from a couple simulation platforms. Check out the homepage (there's a link just above!) where we have documentation, a cookbook, and some user community information.

Clone this repository (size: 40.4 MB): HTTPS / SSH

```
$ hg clone ssh://hg@bitbucket.org/yt_analysis/yt
```

yt /

yt

Filename	Size	Date modified	Message
doc			
scripts			
tests			
yt			
.hgignore	239 B	2011-03-28	Updated .hgignore.
.hgtags	248.5 KB	2010-08-30	Removing SVN tags
CREDITS	2.1 KB	2011-09-12	Adding Elizabeth to CREDITS -- thanks to Elizabeth for her work
FUNDING	1.2 KB	2011-01-17	Adding a FUNDING document. Please feel free to add to this.
LICENSE.txt	34.3 KB	2007-08-25	[svn r237] Added GPL-3 to repo, and included appropriate
MANIFEST.in	142 B	2011-08-26	Adding distribute_setup.py explicitly.
README	1.0 KB	2011-08-21	enzotools -> yt-project.

install script:

Full dependency stack

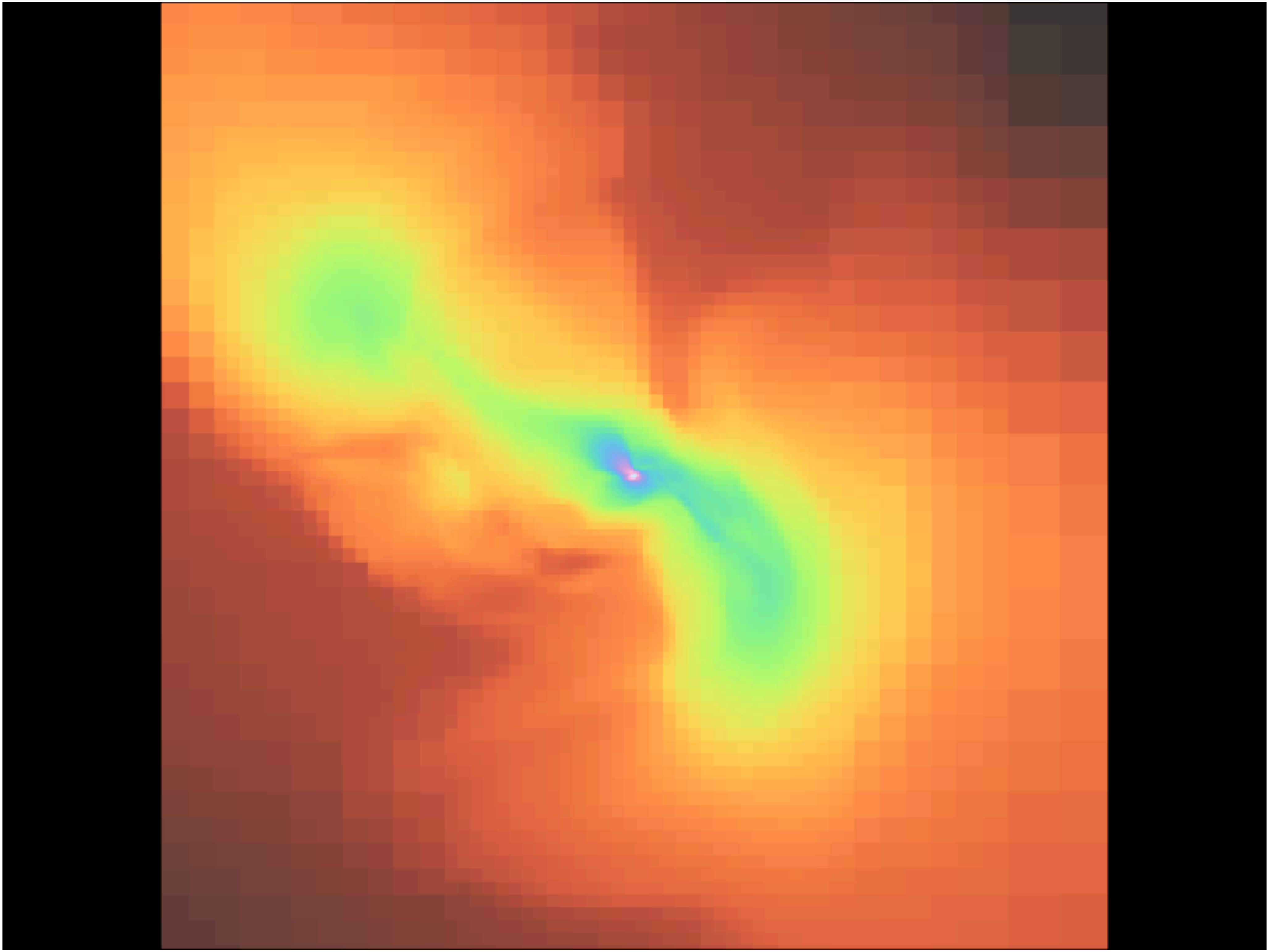
Source code

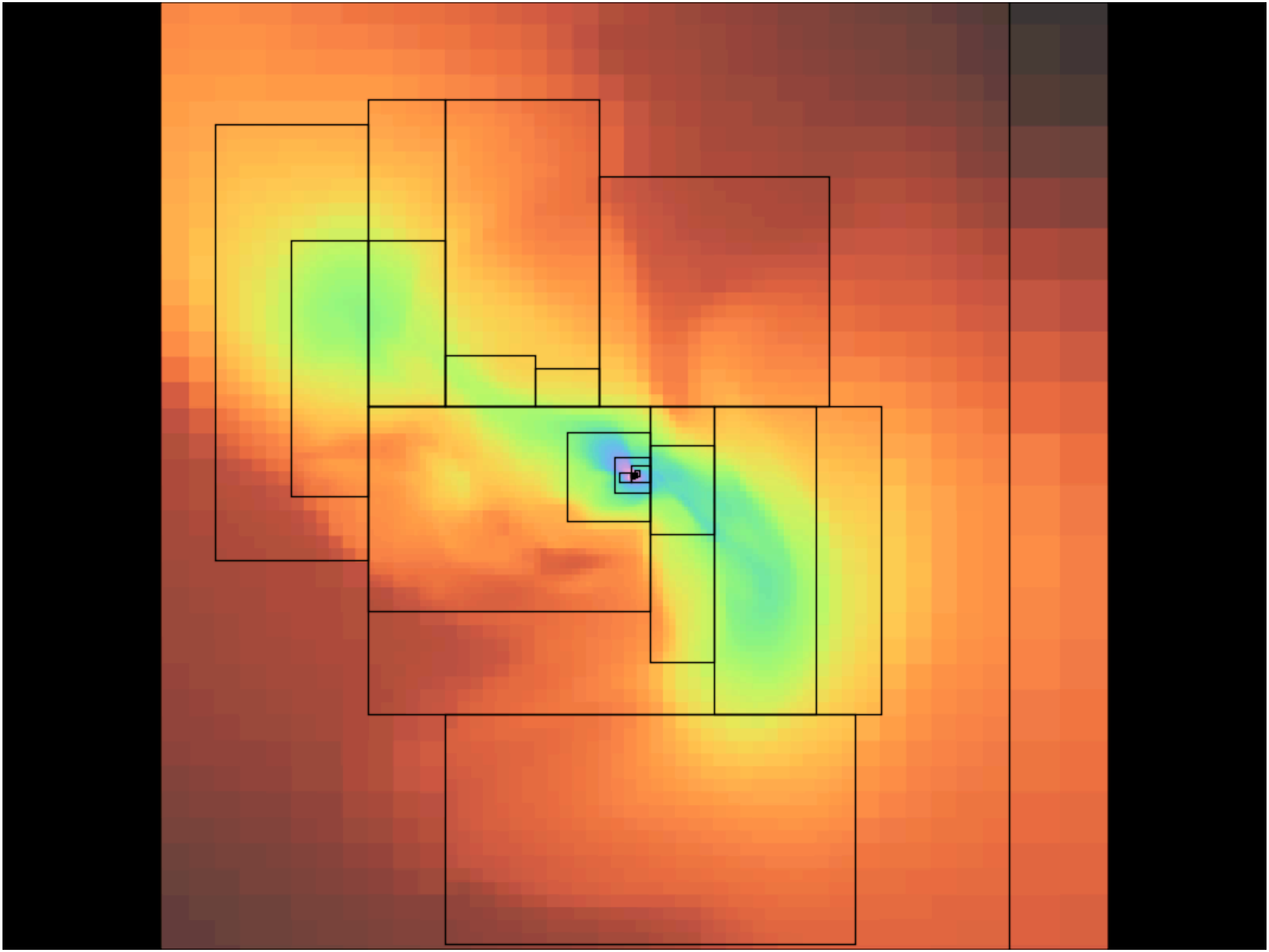
Development environment

GUI

Sample data

yt has been designed to address physical, not computational, entities.





The Universe is full of gas, dark matter and stars. yt makes it easy to access that material.

Transparent IO, masking of overlapping data, load-on-demand, geometric and non-geometric selection, field generation, and common interfaces to different datatypes.

Enzo, Orion, CASTRO, FLASH

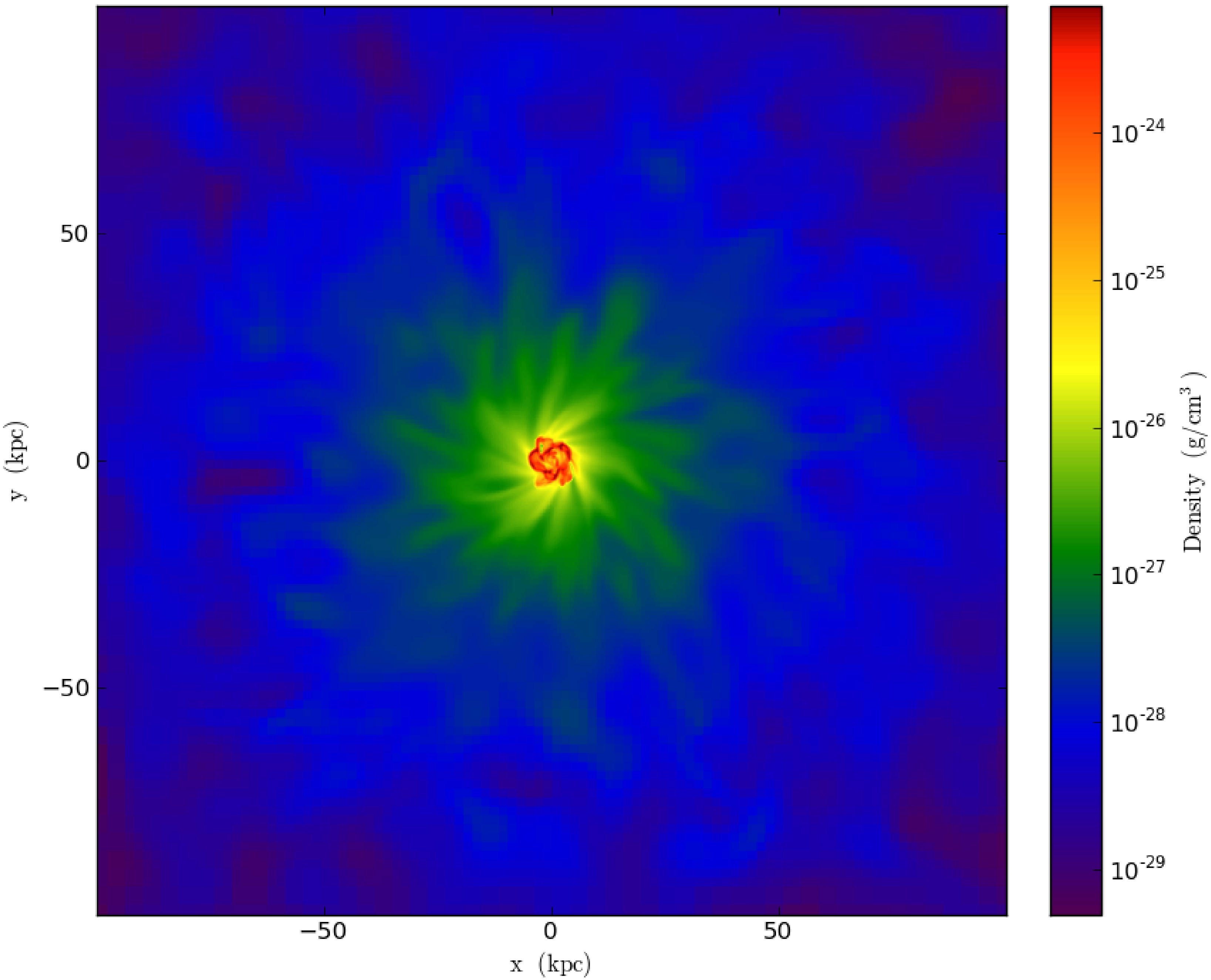
Chombo, Tiger, Athena, ART,
RAMSES

yt is designed to be the *lingua franca*
of astrophysical codes.

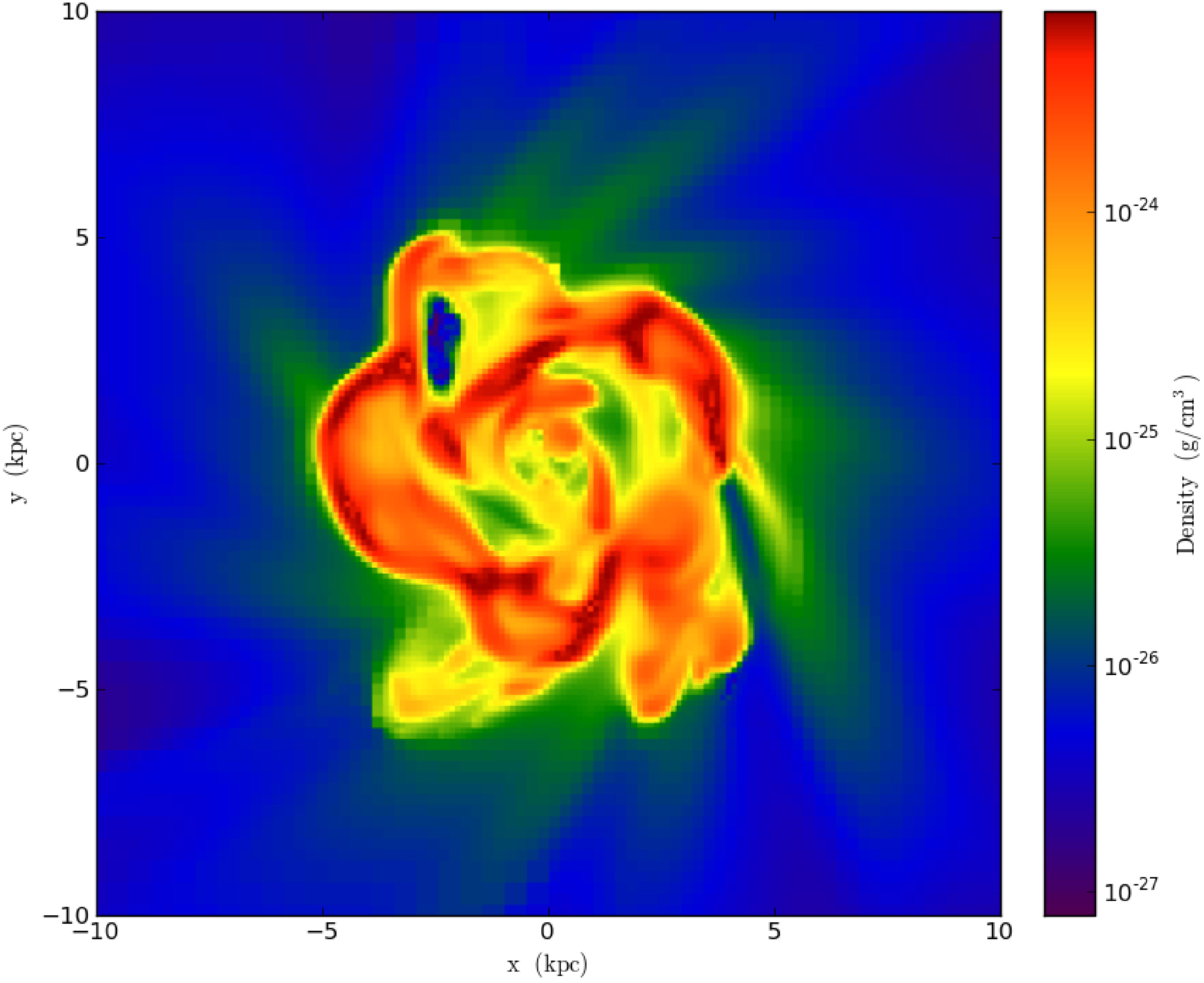
Objects

Orthogonal Rays	1D
Non-orthogonal Rays	
Slices	2D
Oblique Slices	
Projections	
Spheres	3D
Rectangular Prisms	
Disks/Cylinders	
Inclined Boxes	
Clumps	
Extracted Regions	
Boolean combinations	

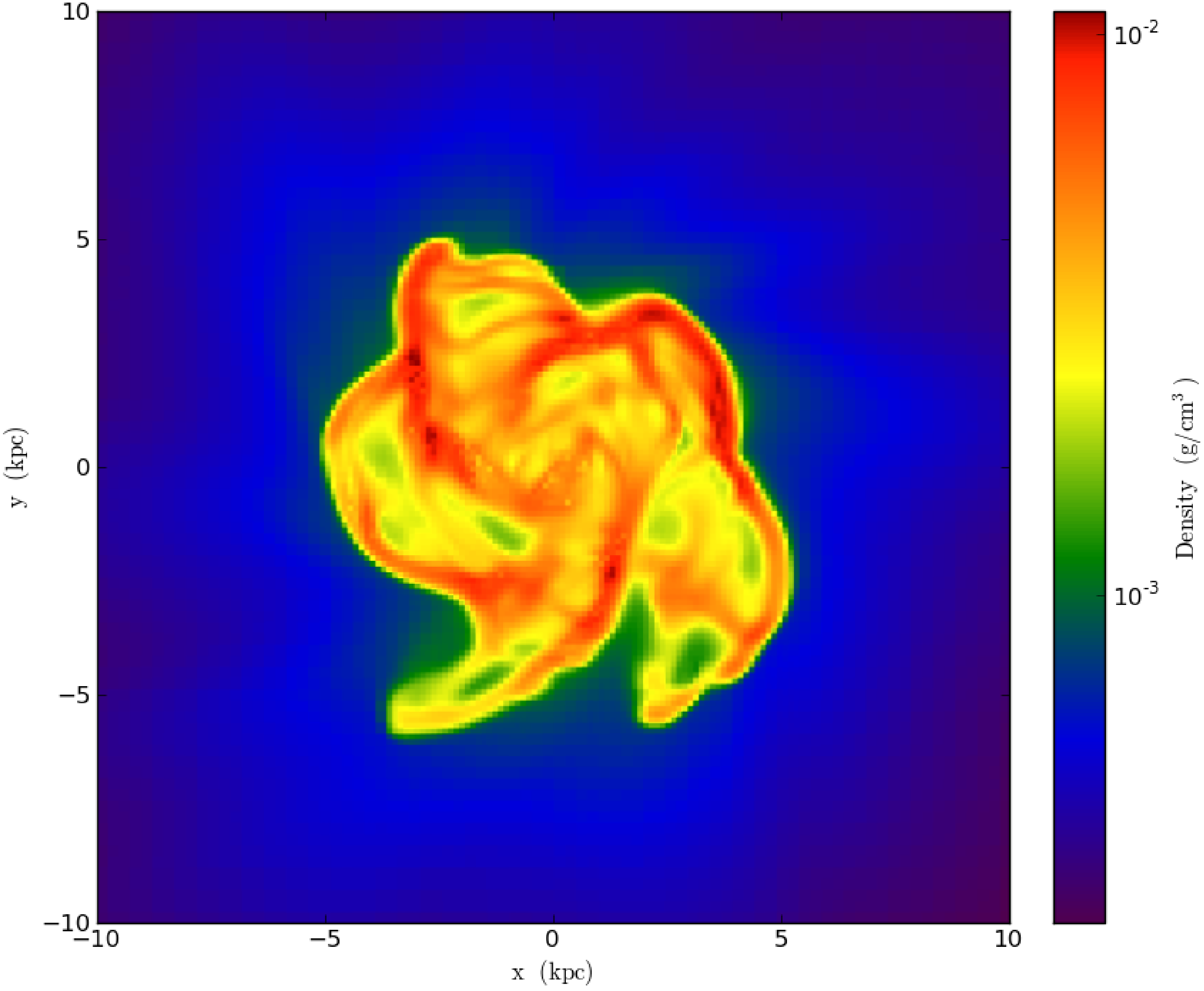
```
from yt.mods import *
pf = load(’galaxy0030/galaxy0030’)
p = SlicePlot(pf, 2, ’Density’, ’c’, (200,’kpc’))
p.save(’Galaxy’)
```


50
0
−50
y (kpc)
−50
0
50
x (kpc)
10^{-24}
10^{-25}
10^{-26}
10^{-27}
10^{-28}
10^{-29}
Density (g/cm^3)

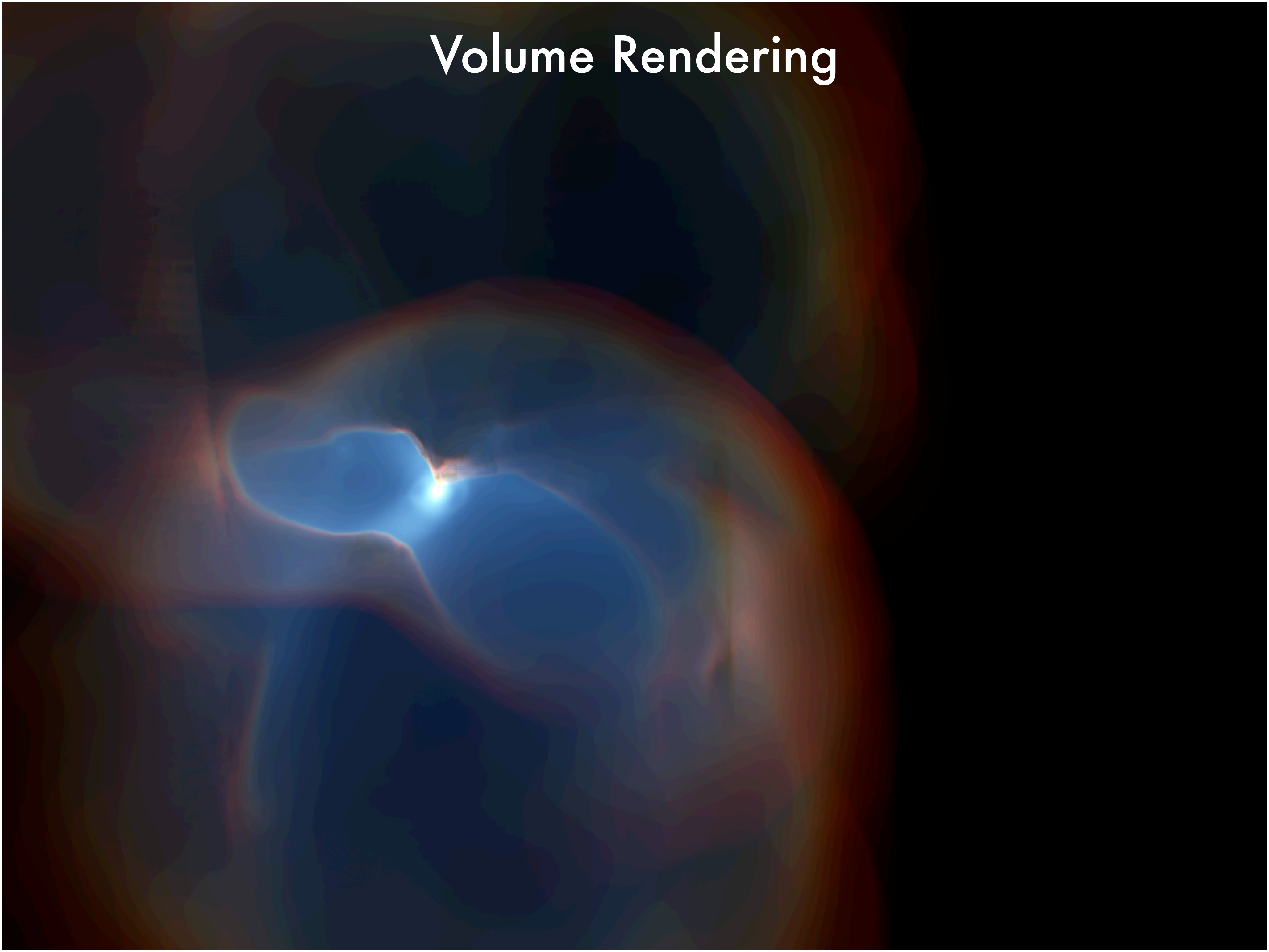
```
from yt.mods import *
pf = load(’galaxy0030/galaxy0030’)
p = SlicePlot(pf, 2, ’Density’, ’c’, (200,’kpc’))
p.set_width(20, ’kpc’)
p.save("GalaxyZoom")
```


10
5
0
−5
−10
y (kpc)
−10
−5
0
5
10
x (kpc)
10^{-24}
10^{-25}
10^{-26}
10^{-27}
Density (g/cm^3)

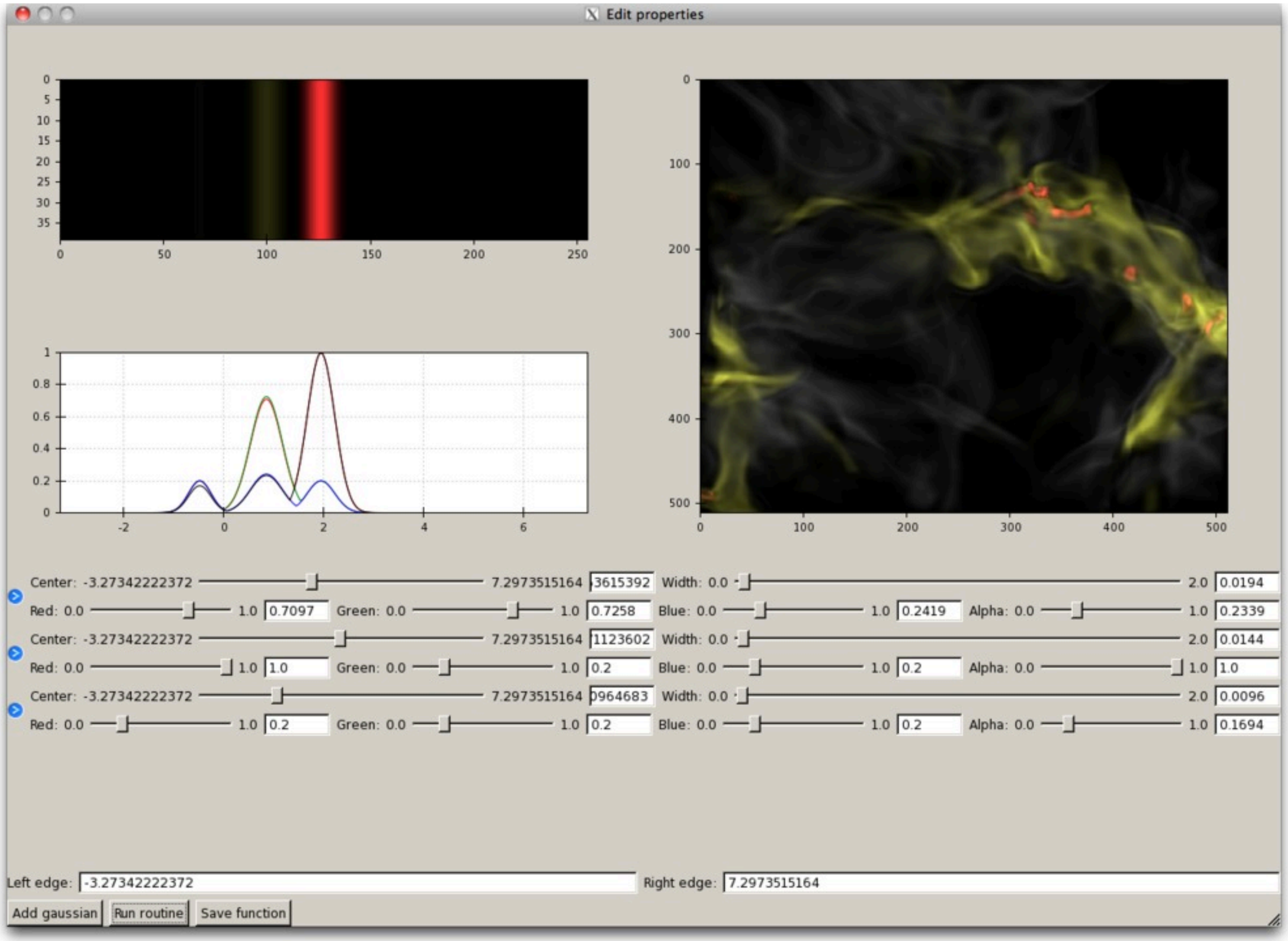
```
from yt.mods import *
pf = load(’galaxy0030/galaxy0030’)
p = ProjectionPlot(pf, 2, ’Density’, ’c’, (20,’kpc’))
p.save(’Galaxy’)
```


10
5
0
−5
−10
−10
−5
0
5
10
x (kpc)
y (kpc)
10^{-2}
10^{-3}
Density (g/cm^3)

Volume Rendering

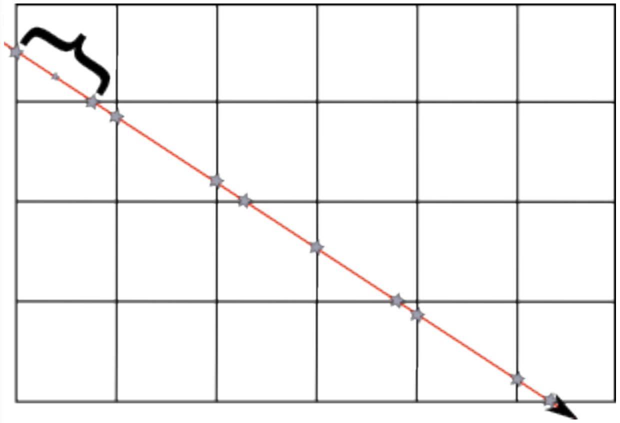


Designed around integrating through a volume: visualization is a side effect.

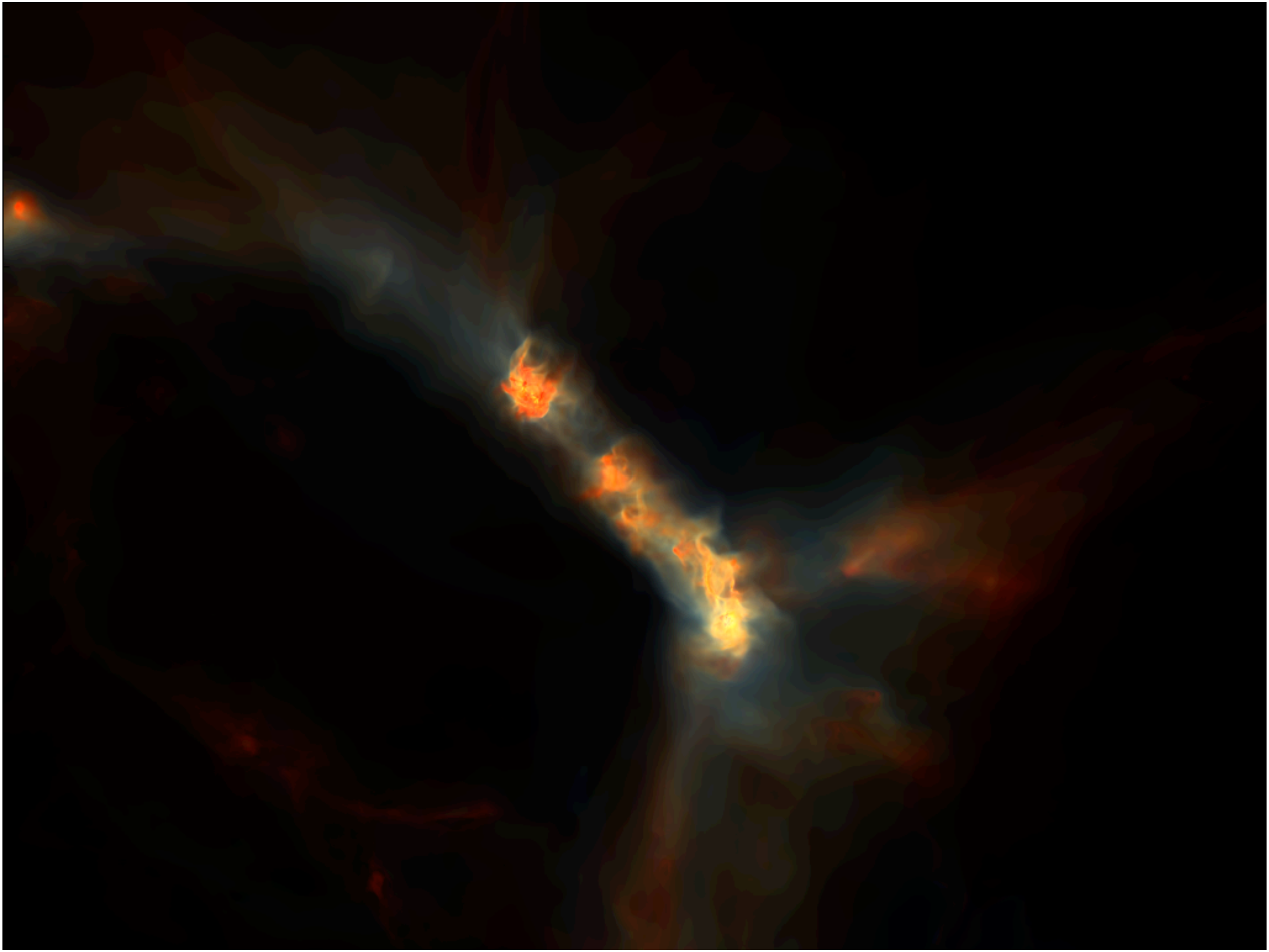


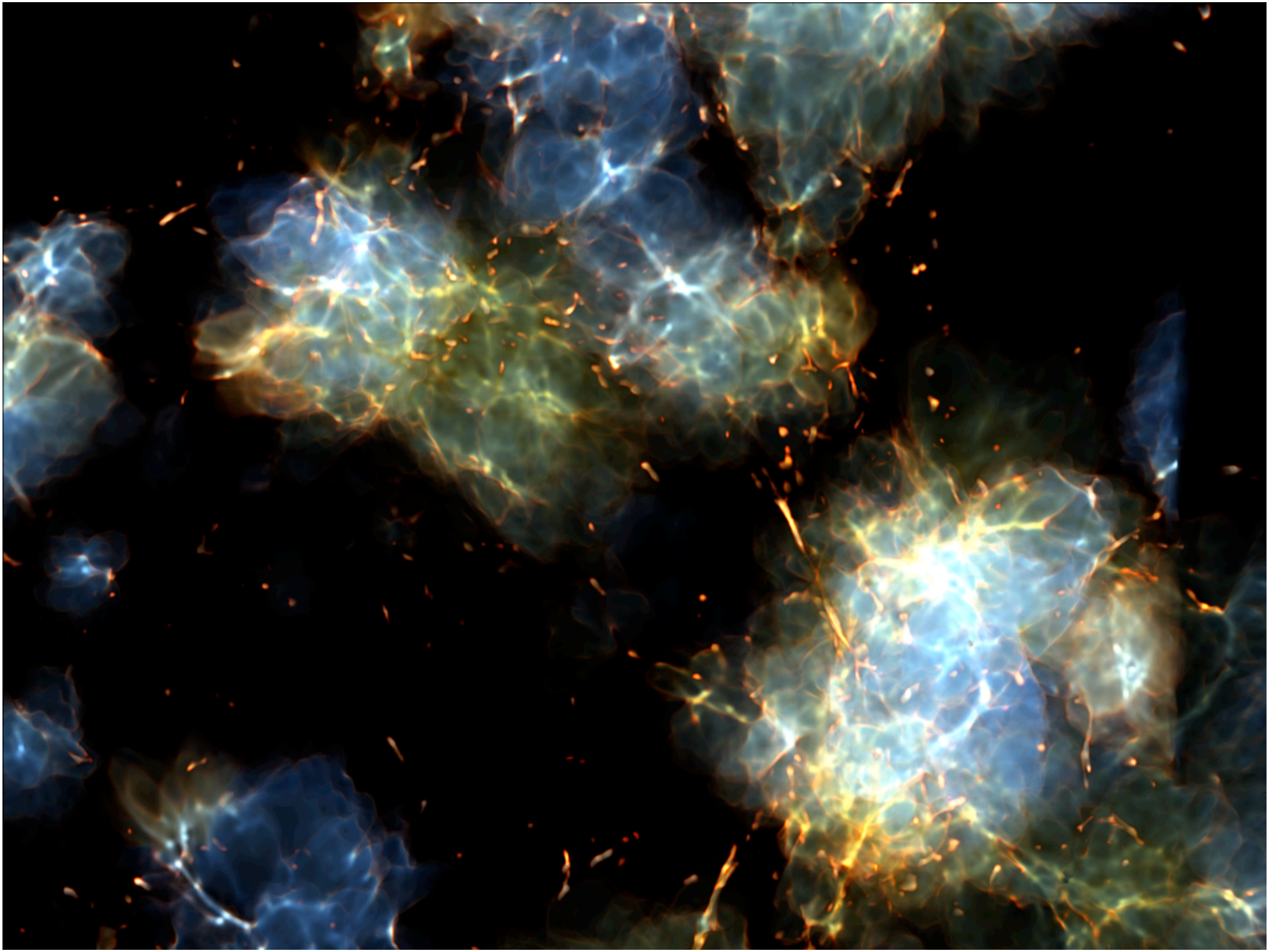
Center: -3.27342222372 7.2973515164 3615392 Width: 0.0 2.0 0.0194
Red: 0.0 1.0 0.7097 Green: 0.0 1.0 0.7258 Blue: 0.0 1.0 0.2419 Alpha: 0.0 1.0 0.2339
Center: -3.27342222372 7.2973515164 1123602 Width: 0.0 2.0 0.0144
Red: 0.0 1.0 1.0 Green: 0.0 1.0 0.2 Blue: 0.0 1.0 0.2 Alpha: 0.0 1.0 1.0
Center: -3.27342222372 7.2973515164 0964683 Width: 0.0 2.0 0.0096
Red: 0.0 1.0 0.2 Green: 0.0 1.0 0.2 Blue: 0.0 1.0 0.2 Alpha: 0.0 1.0 0.1694

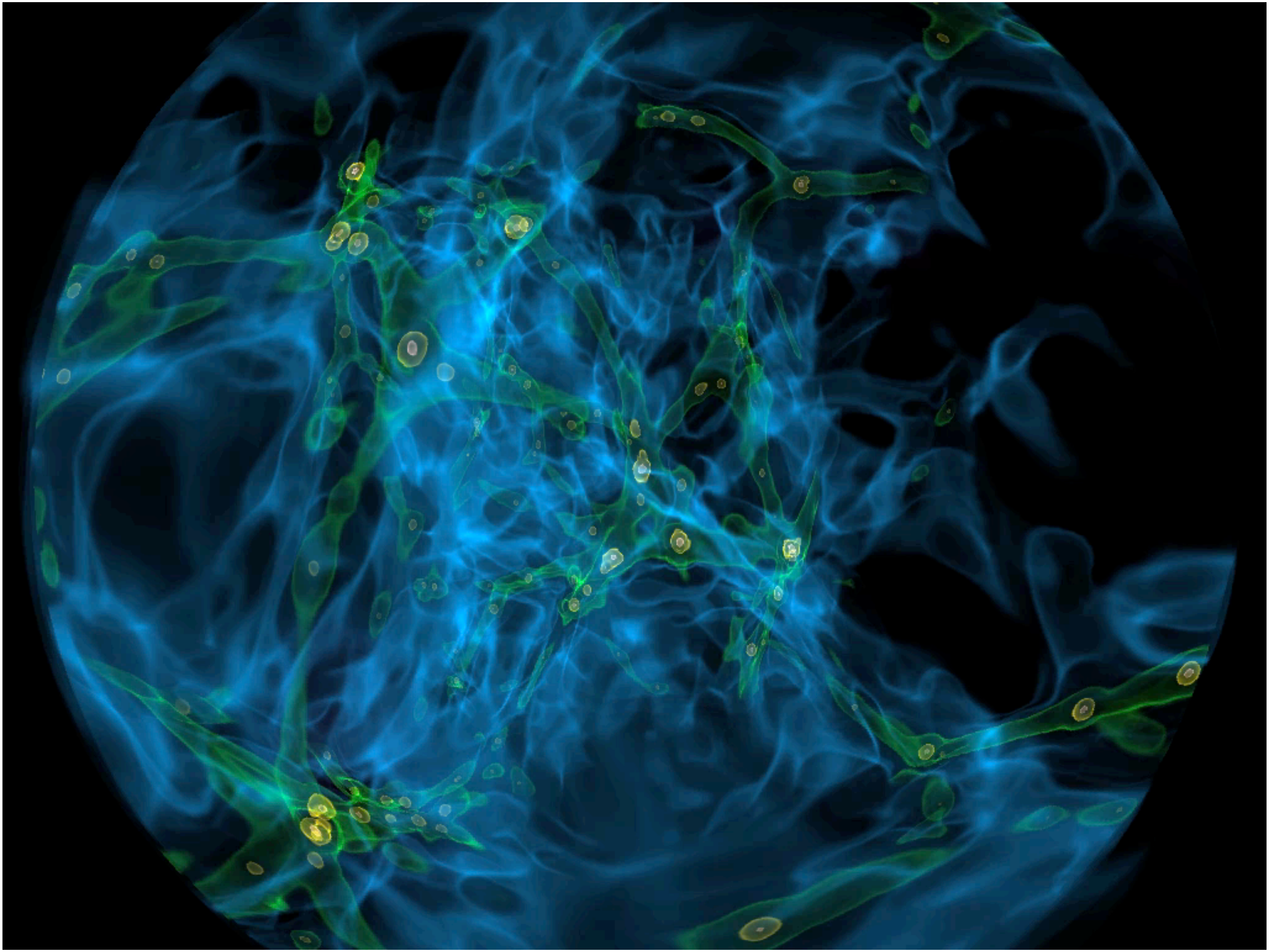
Left edge: -3.27342222372
Right edge: 7.2973515164

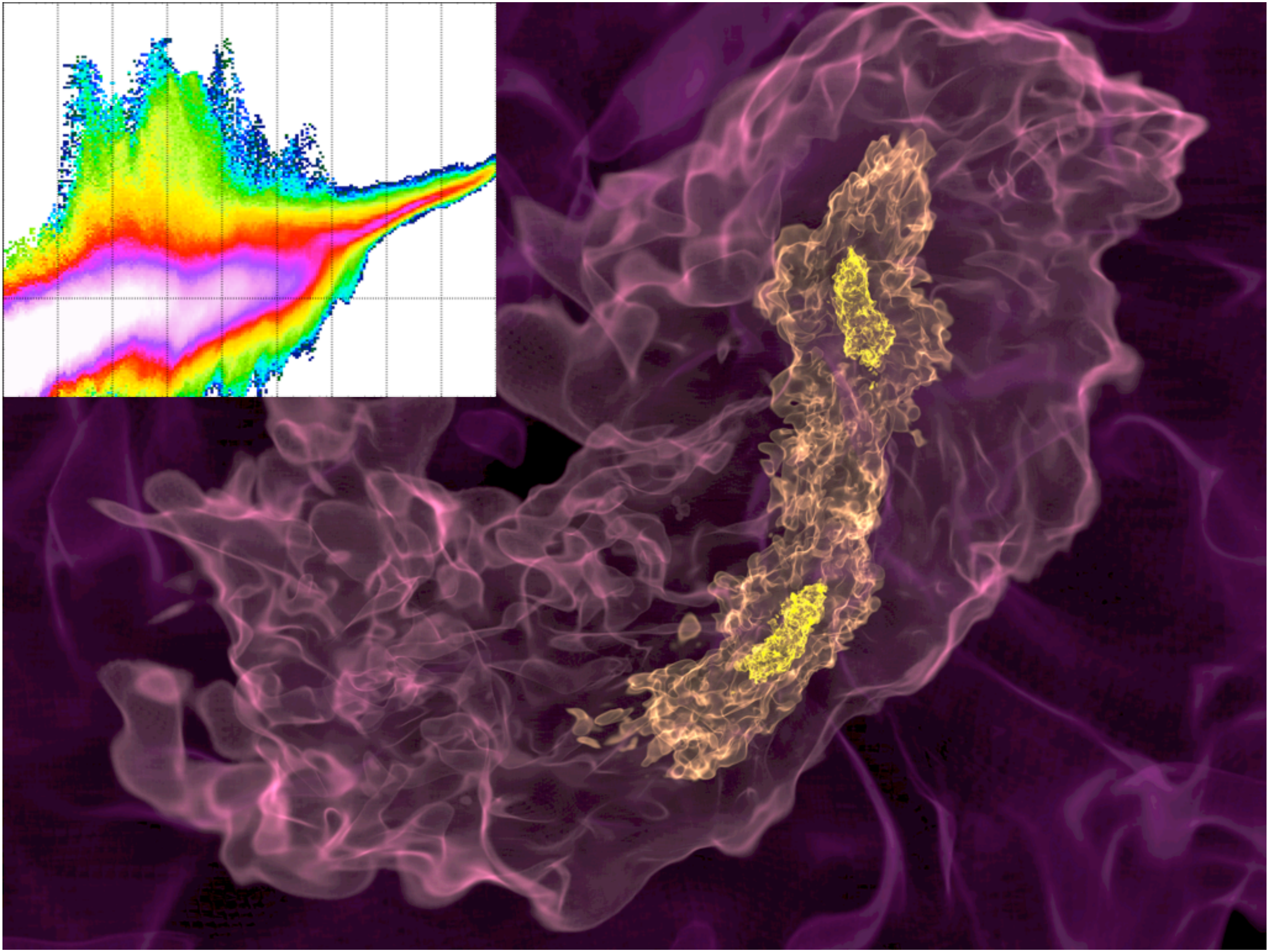


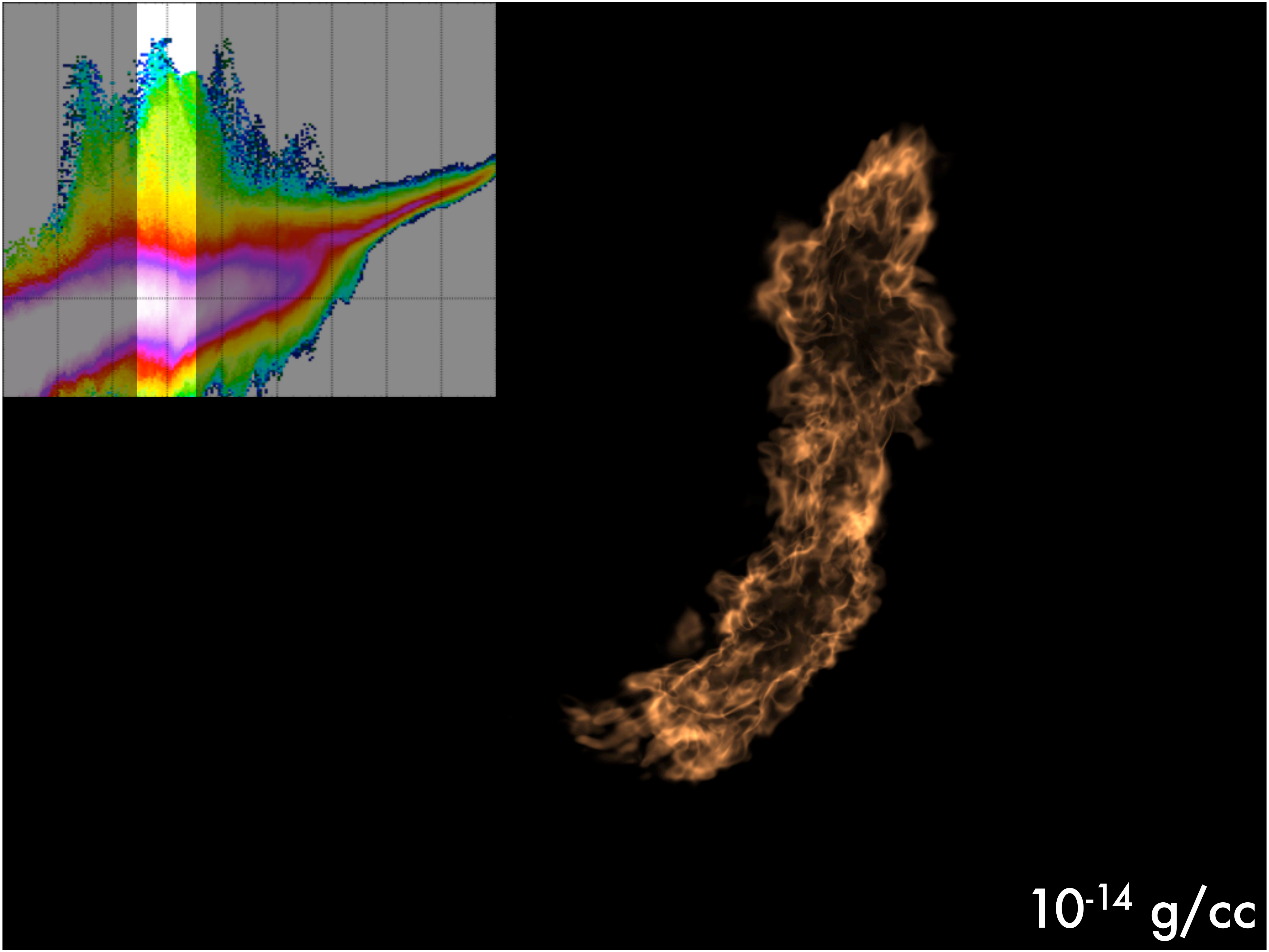
$$\frac{dI_\nu}{ds} = j_\nu - \alpha_\nu I_\nu$$



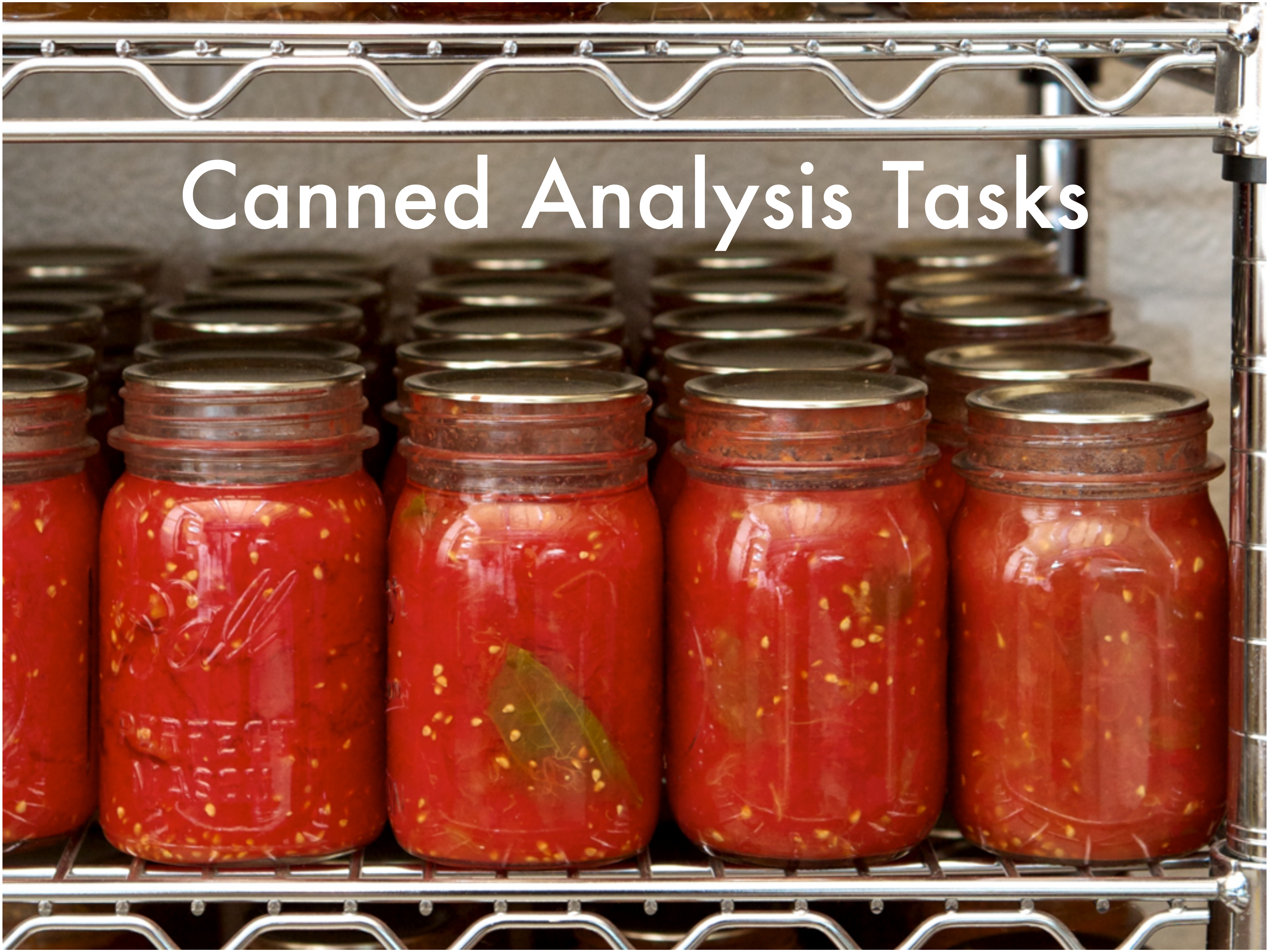






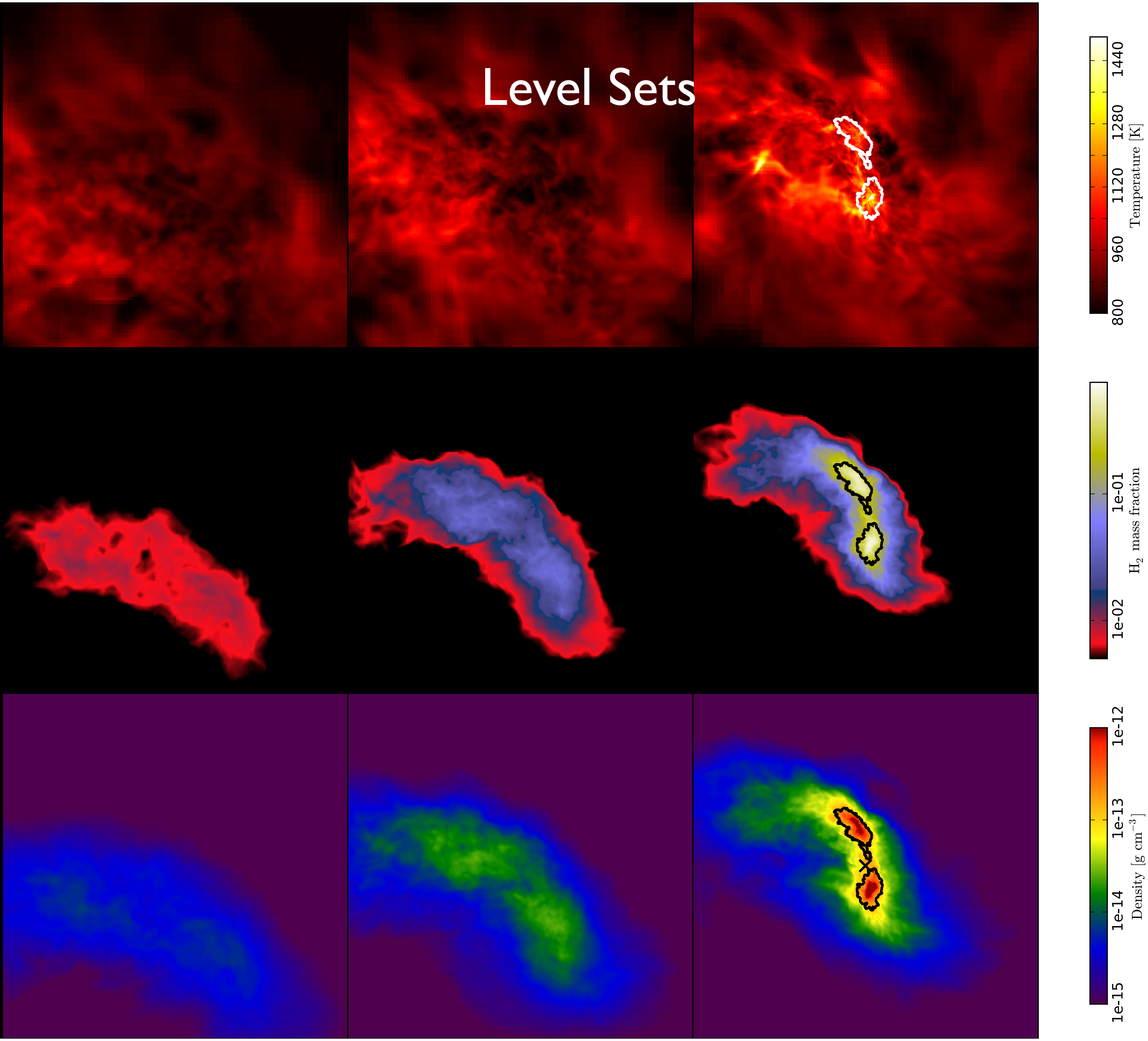

10^{-14} g/cc

Canned Analysis Tasks

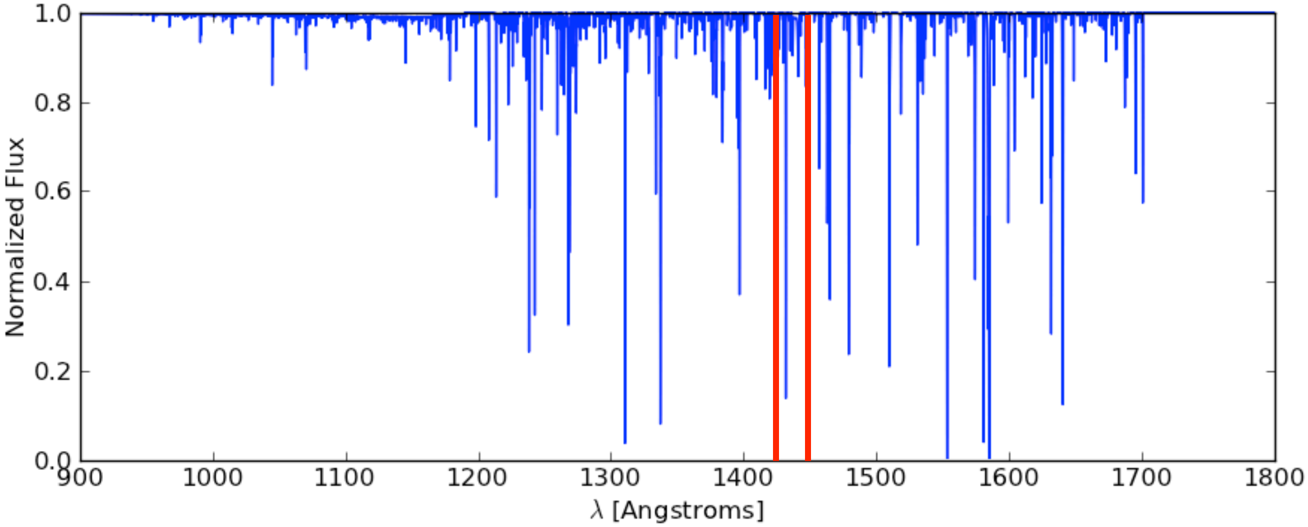


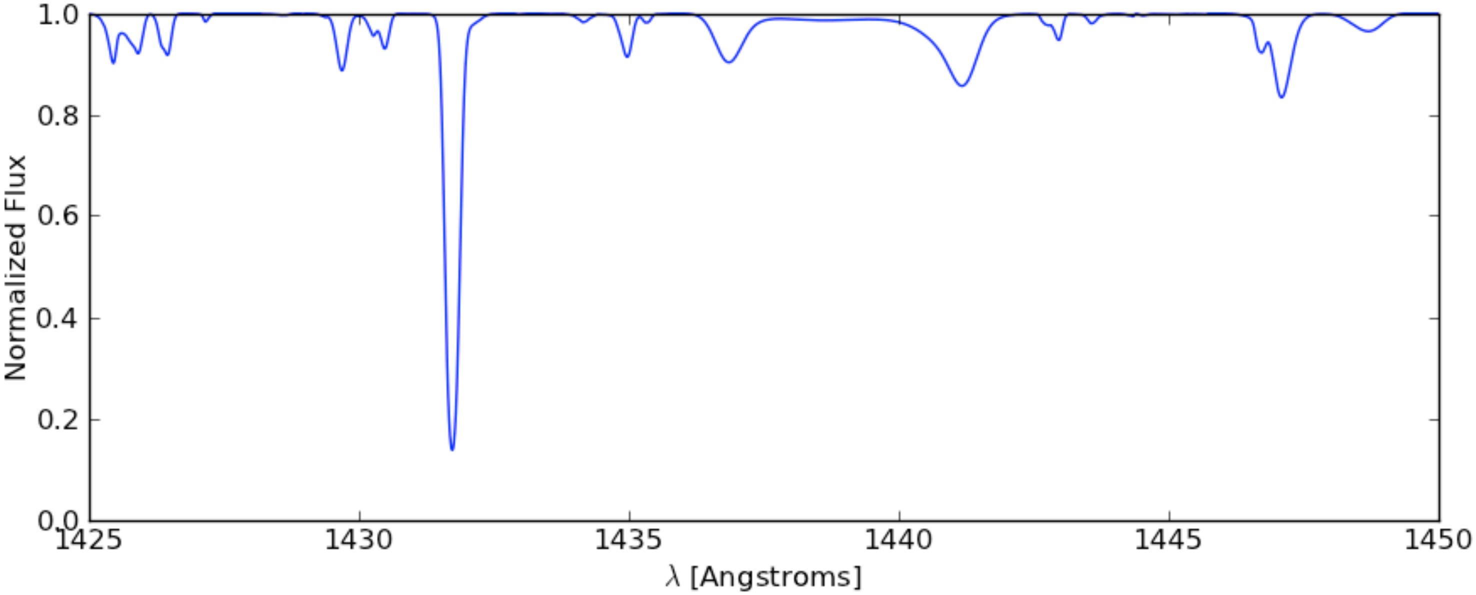
Absorption Spectrum
Coordinate Transformations
Halo Finding
Mass Functions
Merger Trees
Halo Profiling
Level Sets
Light Cones
Light Rays
Time Series
Star Analysis
Two-Point Analysis

Level Sets

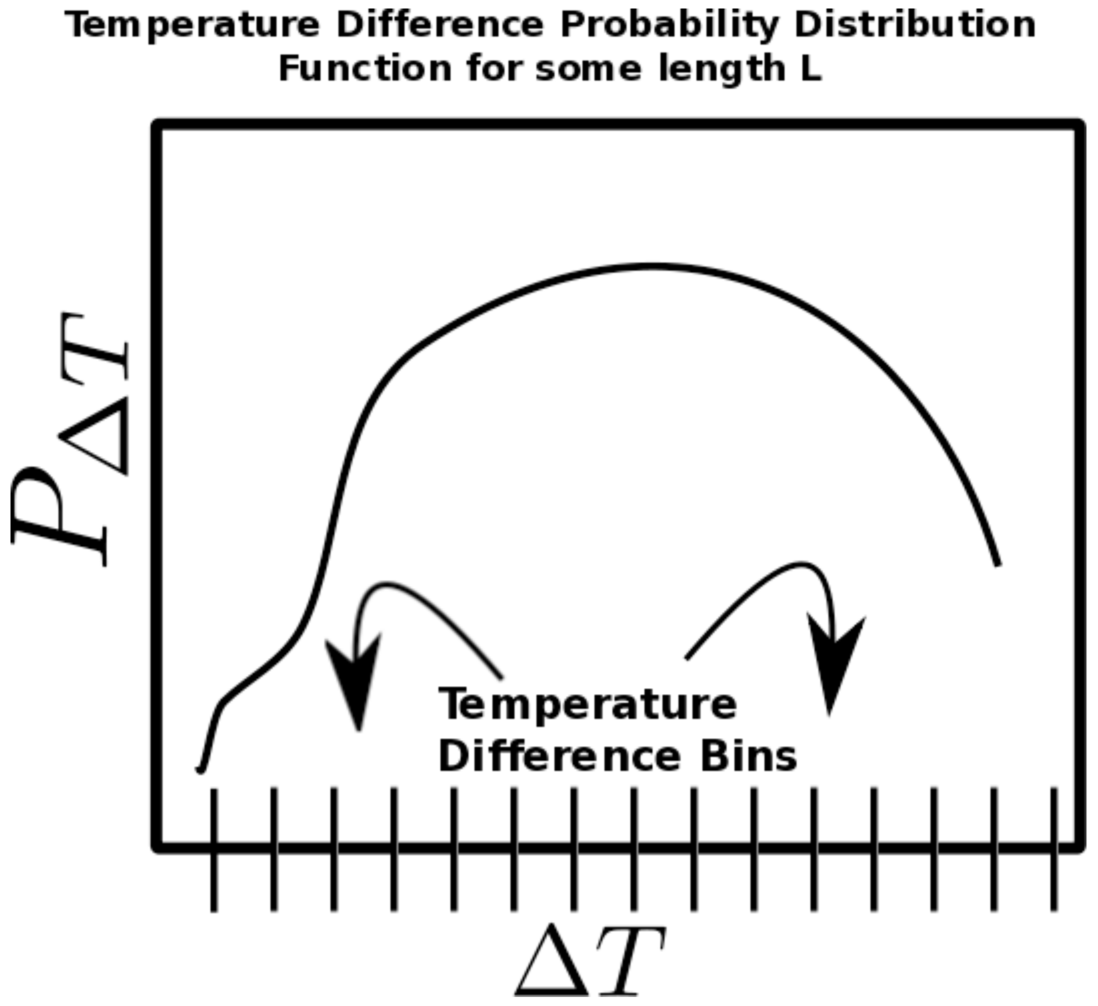

800
960
1120
1280
1440
Temperature [K]
1e-02
1e-01
H_2 mass fraction
1e-15
1e-14
1e-13
1e-12
Density [g cm^{-3}]

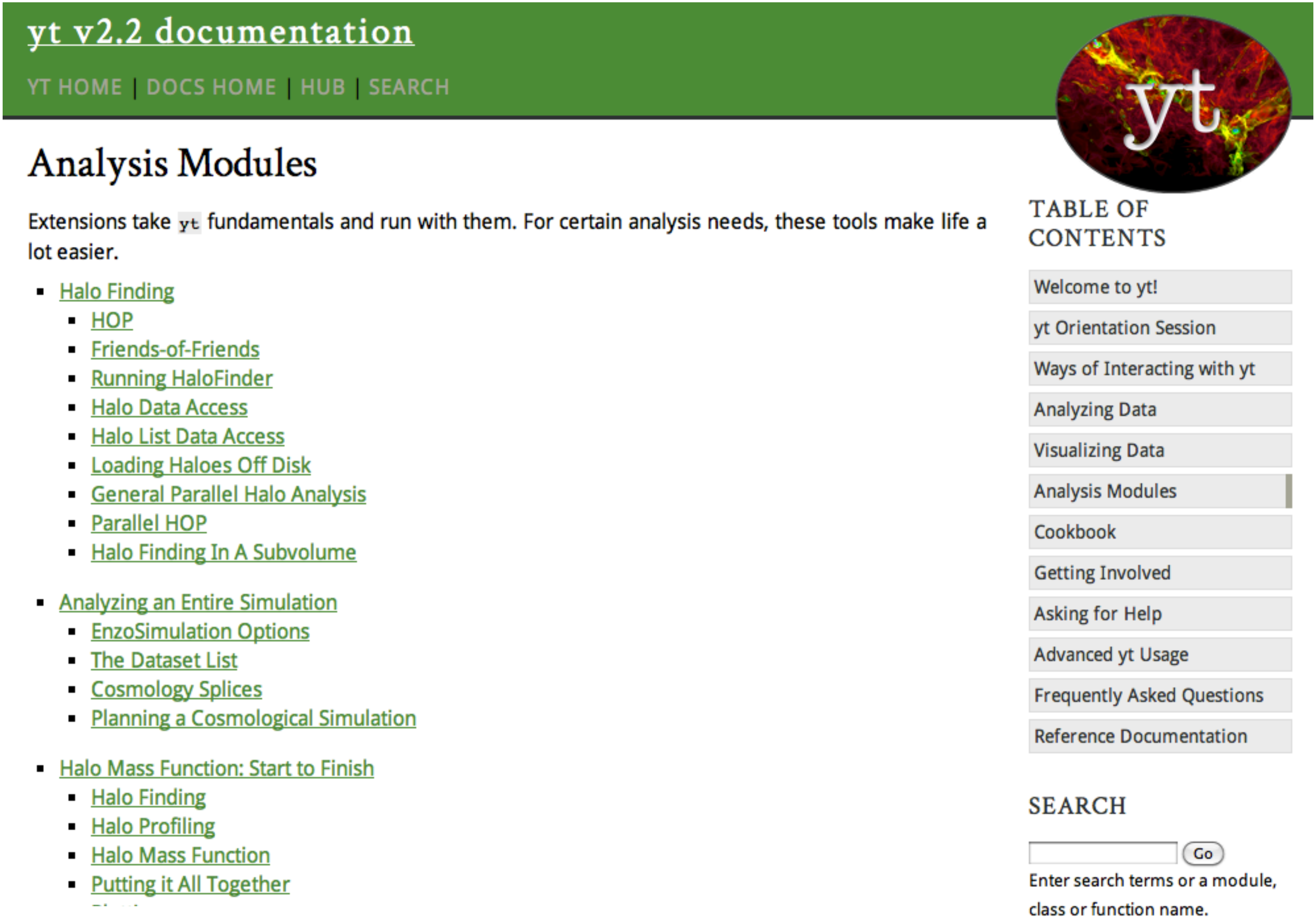
Synthetic Spectra





Two-Point Functions


Temperature Difference Probability Distribution
Function for some length L
$P_{\Delta T}$
Temperature
Difference Bins
ΔT



yt v2.2 documentation

YT HOME | DOCS HOME | HUB | SEARCH

Analysis Modules

Extensions take `yt` fundamentals and run with them. For certain analysis needs, these tools make life a lot easier.

- Halo Finding
  - HOP
  - Friends-of-Friends
  - Running HaloFinder
  - Halo Data Access
  - Halo List Data Access
  - Loading Haloes Off Disk
  - General Parallel Halo Analysis
  - Parallel HOP
  - Halo Finding In A Subvolume
- Analyzing an Entire Simulation
  - EnzoSimulation Options
  - The Dataset List
  - Cosmology Splices
  - Planning a Cosmological Simulation
- Halo Mass Function: Start to Finish
  - Halo Finding
  - Halo Profiling
  - Halo Mass Function
  - Putting it All Together

TABLE OF CONTENTS

SEARCH

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

Parallelism

Multi-level parallelism: dynamic workgroups, communicators, subgroups and task queues

Parallelism

Embarassingly Parallel	Spatial Decomposition
Decomposed by load or IO characteristics	Helper functions to decompose the domain

Parallelism

Embarassingly Parallel	Spatial Decomposition
Quantities Profiles Slices Projections Volume Rendering	Halo Finding

Developing as a Team

Forky development:

very low barrier to entry;
everything comes in the box.

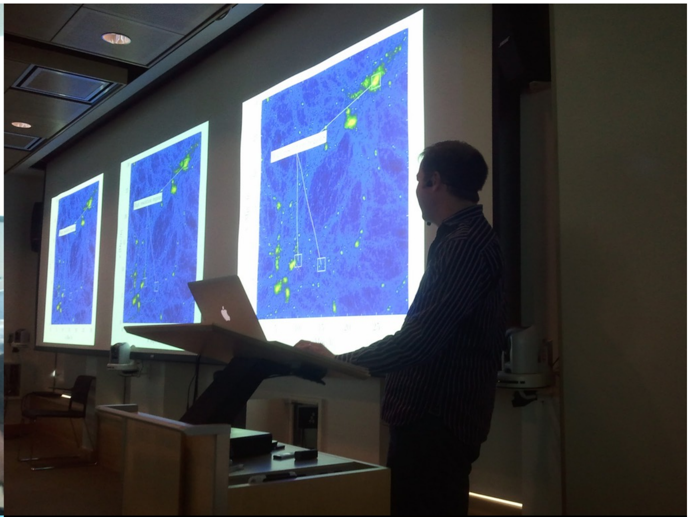
Testing:

answer as well as integration tests get run every 30 minutes.

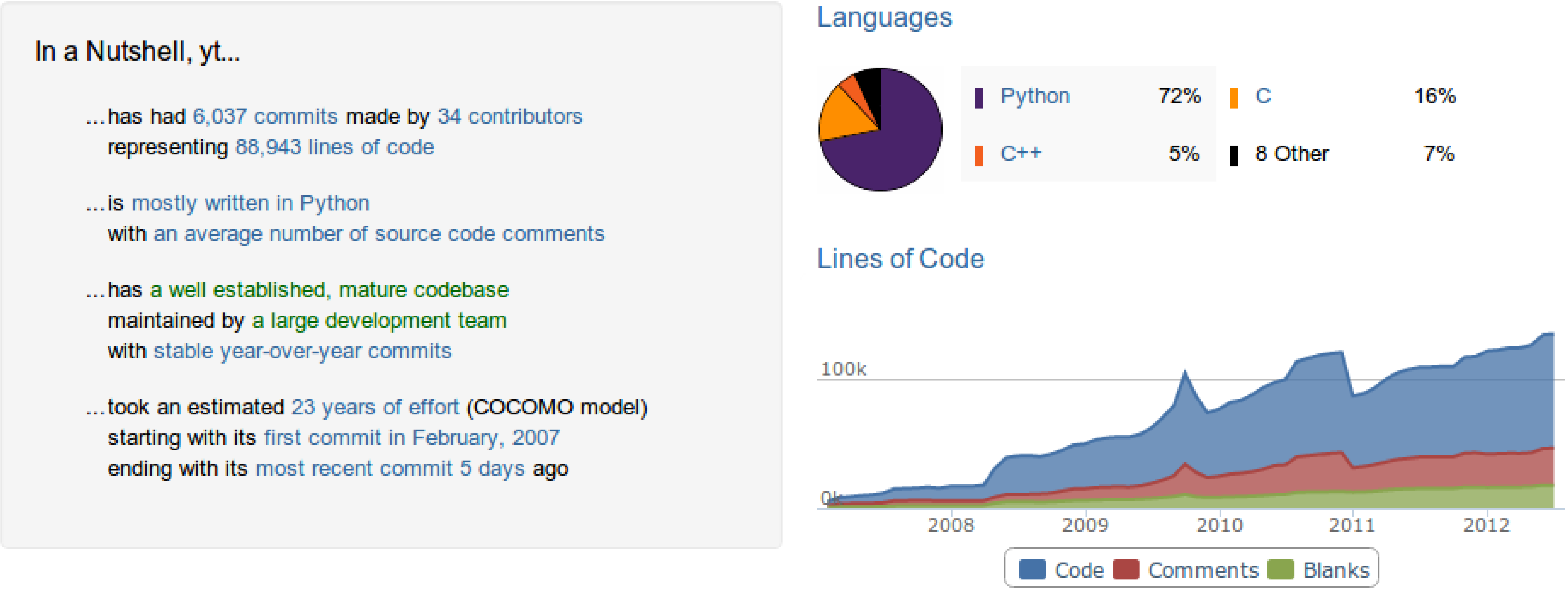
Community

- Mailing lists: community, development
- IRC (`#yt` on `irc.freenode.net`)
- Cookbook, docs
- Curriculum from workshop

The University of Chicago
The Flash Center for Computational Science
yt WORKSHOP
(1.24 - 26. 2012)
Searle CI-Conference
240 a&b



What is volume rendering / offaxis projection?
Numerically integrates the radiative transfer equations
What is volume rendering / offaxis projection?
Numerically integrates the radiative transf
What is volume rendering / offaxis projection?
Numerically integrates the radiative transfer equations



In a Nutshell, yt...

...has had 6,037 commits made by 34 contributors
representing 88,943 lines of code

...is mostly written in Python
with an average number of source code comments

...has a well established, mature codebase
maintained by a large development team
with stable year-over-year commits

...took an estimated 23 years of effort (COCOMO model)
starting with its first commit in February, 2007
ending with its most recent commit 5 days ago

Languages

Python	72%	C	16%
C++	5%	8 Other	7%

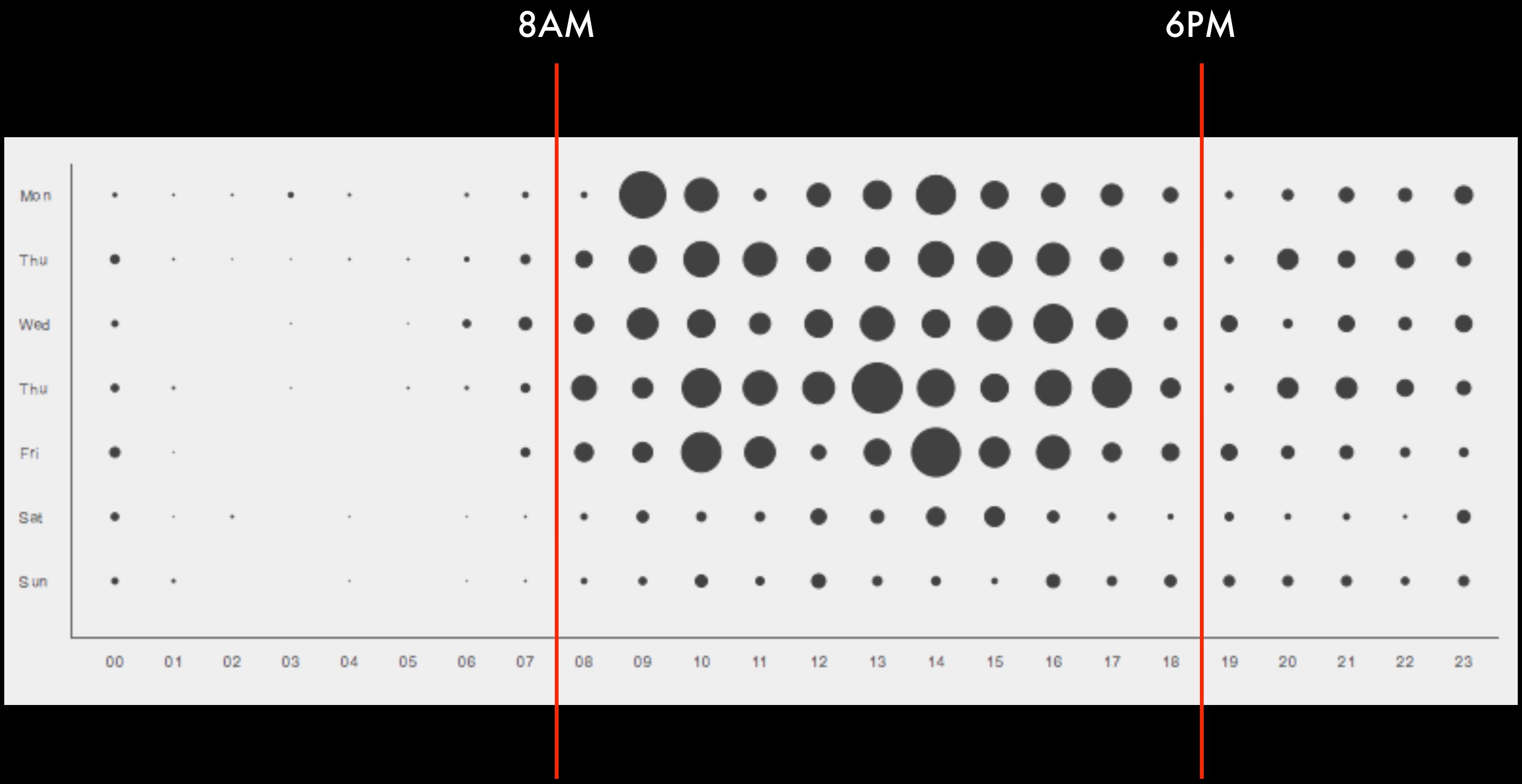
Lines of Code

80,000 lines of code

Python, Cython, C

20 contributors (60+ users)

Contributors from 10+ institutions



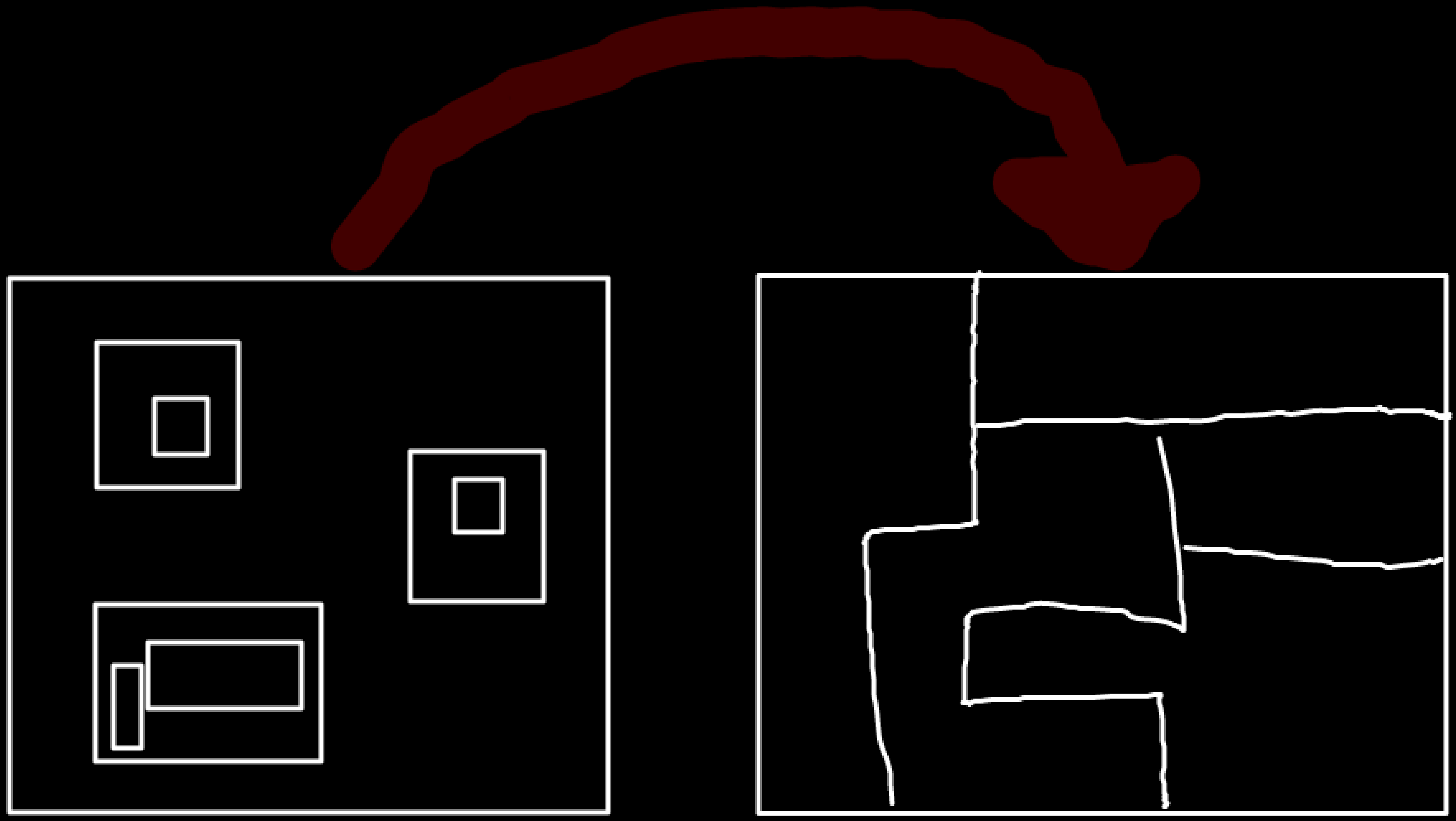
Contributors

Tom Abel
David Collins
Andrew Cunningham
Nathan Goldbaum
Cameron Hummels
Ji-hoon Kim
Steffen Klemer
Kacper Kowalik
Michael Kuhlen
Eve Lee
Chris Malone
Chris Moody
Andrew Myers
Jeff Oishi

Jean-Claude Passy
Thomass Robitaille
Anna Rosen
Anthony Scopatz
Devin Silvia
Sam Skillman
Stephen Skory
Britton Smith
Geoffrey So
Casey Stark
Elizabeth Tasker
Rick Wagner
John Wise
John ZuHone

yt 3.0

No more grids, only chunks



Grids

Chunks

Particle segregation, multiple fluids

Address technical debt

Strategies

- Patch: chunks are collections of grids
- Octree: direct octree parsing
- SPH/N-body: octree indexing

3.0 Milestones

1. Patch-based analysis and 2D viz
2. Patch-based volume rendering
3. Halo Finding
4. Octree analysis and 2D viz
5. Particle analysis and 2D viz

Caveats!

SPH Support

- Particle support only at first
- No first light (ETA: 3-6 weeks)
- Many free parameters, many output formats

Octree Support

- Volume rendering not ready yet
- Subtleties may lurk
- ART half-finished

I volunteer!

(demo)

Thank you.

yt-project.org