The yt Project: Visualizing and Analyzing Volumetric Data Across Domains

Expanding the yt Community

The yt Project aims to produce an integrated science environment based on the Python programming language for collaborative, asking and answering questions about volumetric data. Currently, yt is mostly used by astrophysicists who run and analyze simulations of astrophysical phenomena but we see the potential for easy-to-use, loosely-coupled components that can be easily integrated with other Python tools.

In a Nutshell, yt ...

...has had 250+ commits made by 19 contributors representing 104 days of coding
...is mostly written in Python with an average of 4-5 source code commits per day
...has a well-established, mature codebase maintained by a very large development team
...includes an estimated 6 years of effort (2008-2015) spanning 182 commits, ending with the most recent commit about 1 month ago

Current status of the yt Project

The yt codebase is made up of over 250,000 lines of Python source code. The full proposal can be found at https://arxiv.org/abs/1810.06566

Path Traversal and Non-local Analysis

Currently, the yt field system requires all input fields to be defined at the location of a data element or its neighbors. We will add an API and associated infrastructure for traversing a path through a volume while simultaneously accumulating data. This will make it easier for users to perform operations on a large dataset, such as computing the density of a field or performing a cumulative sum over a field.

Symbolic Fields

We are rewriting the yt field system to simplify the definition of a field in terms of a field dependency graph rather than using common subexpression elimination, as well as performance optimization via code generation or offloading computation to a JIT compiler like numba or distributed computing engines like Dask.

Community Developments

yt Extension Ecosystem

As part of our effort to make yt more modular, we are creating a system for community-maintained extensions to yt. The goal is that these extensions will enable domain-specific analysis that requires extensive domain knowledge to maintain. This will allow the yt team to focus on core algorithms and functionality while domain experts can seamlessly release and maintain functionality in a way that benefits the community and the larger scientific community.

John Wise

So cool! My 1st year grad student, Corey Brummel-Smith, just showed the group a 3D printed model of the triggered star formation region he’s studying. See the density projection in the BS. Thanks @yt_atmos for the iso-surfaces and @GeorgiaTech Innovation Lab for free printing!

Nathan Goldbaum, Matthew Turk, Sam Walkow NCSA, University of Illinois ngoldbau@illinois.edu yt-project.org Data-exp-lab.github.io

The yt units module provides a system for tracking unit consistency and converting data between different units and units systems. This substantially improved the yt user experience by making clear the units of data returned by yt and has eliminated an entire class of bugs in user scripts and in yt's modules.

Since the release of yt 3.0, the yt.units module has provided a system for tracking unit consistency and converting data between different units and units systems. This substantially improved the yt user experience by making clear the units of data returned by yt and has eliminated an entire class of bugs in user scripts and in yt's modules.

This work is supported by NSF SI2-SSI OAC-1663914 (goo.gl/6w25zy)