
Obtaining and Compiling MPAS-A

How do I get the code?

Before we begin discussing how to run MPAS-Atmosphere in upcoming talks, a reasonable first question is:

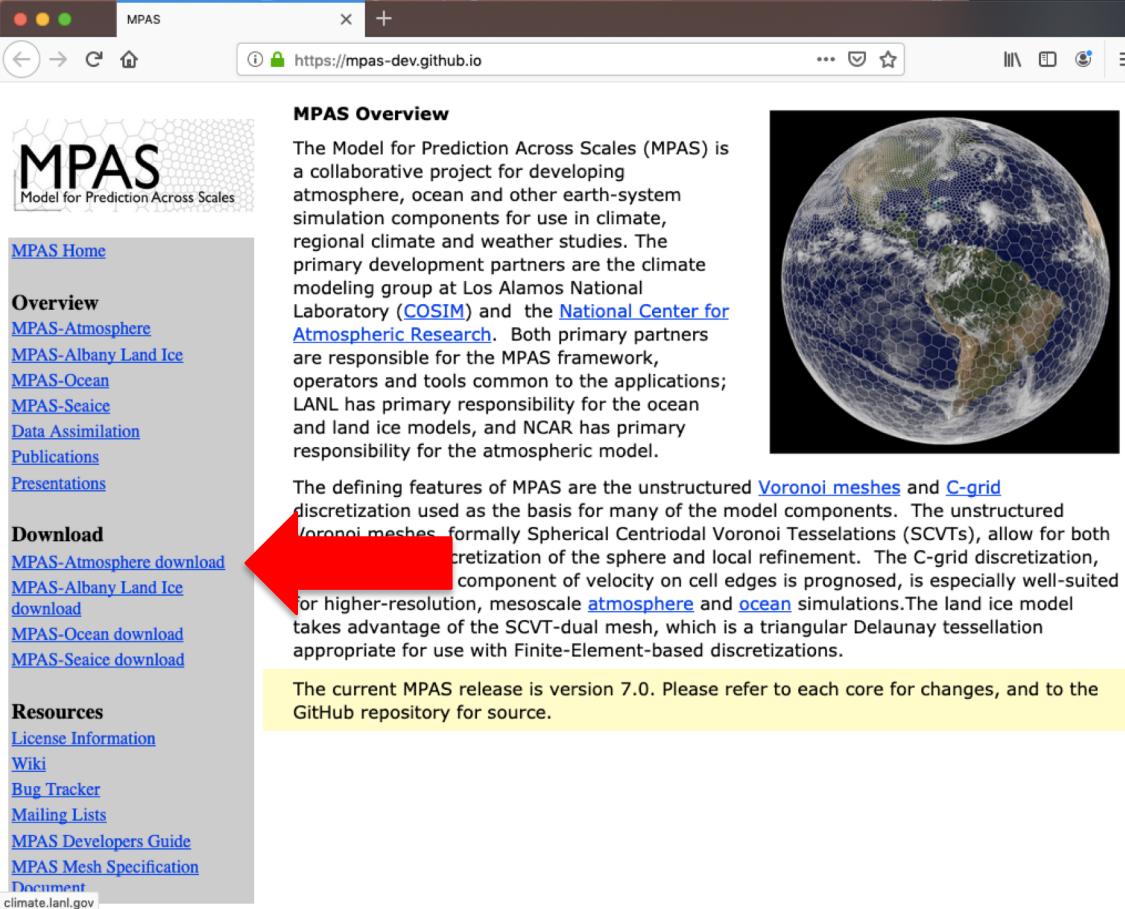
How do I get a copy of the source code?

There are essentially two options:

- 1) The “traditional”, but not necessarily encouraged, method
 - Download a .tar.gz or .zip file
- 2) The preferred method
 - Make a *clone* of the MPAS-Model repository

The “less preferred” method of obtaining code

One can navigate to a download link from the MPAS homepage at <https://mpas-dev.github.io/>



MPAS
Model for Prediction Across Scales

[MPAS Home](#)

Overview

- [MPAS-Atmosphere](#)
- [MPAS-Albany Land Ice](#)
- [MPAS-Ocean](#)
- [MPAS-Seoice](#)
- [Data Assimilation](#)
- [Publications](#)
- [Presentations](#)

Download

- [MPAS-Atmosphere download](#)
- [MPAS-Albany Land Ice download](#)
- [MPAS-Ocean download](#)
- [MPAS-Seoice download](#)

Resources

- [License Information](#)
- [Wiki](#)
- [Bug Tracker](#)
- [Mailing Lists](#)
- [MPAS Developers Guide](#)
- [MPAS Mesh Specification Document](#)

climate.lanl.gov

MPAS Overview

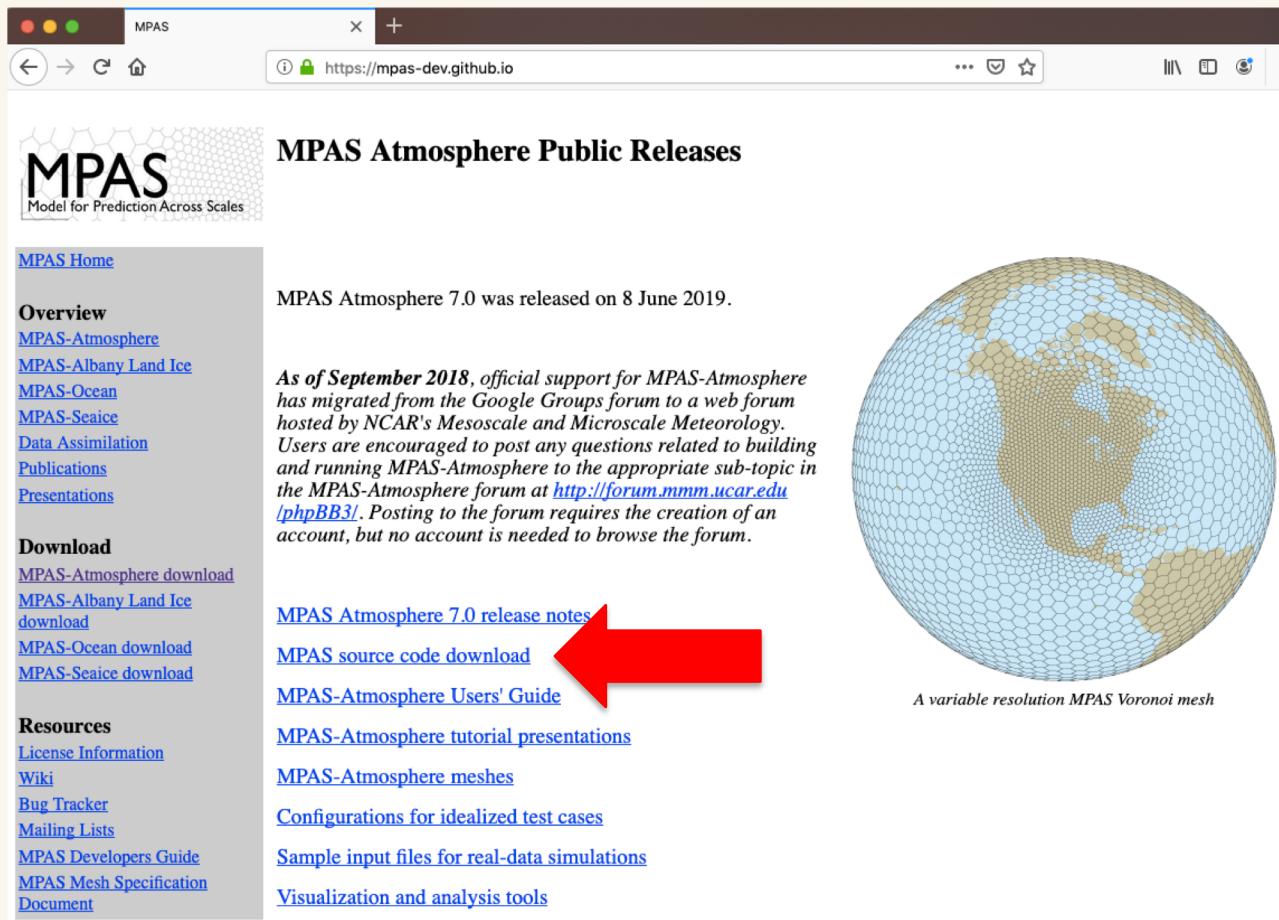
The Model for Prediction Across Scales (MPAS) is a collaborative project for developing atmosphere, ocean and other earth-system simulation components for use in climate, regional climate and weather studies. The primary development partners are the climate modeling group at Los Alamos National Laboratory ([COSIM](#)) and the [National Center for Atmospheric Research](#). Both primary partners are responsible for the MPAS framework, operators and tools common to the applications; LANL has primary responsibility for the ocean and land ice models, and NCAR has primary responsibility for the atmospheric model.

The defining features of MPAS are the unstructured [Voronoi meshes](#) and [C-grid](#) discretization used as the basis for many of the model components. The unstructured Voronoi meshes, formally Spherical Centriodal Voronoi Tessellations (SCVTs), allow for both discretization of the sphere and local refinement. The C-grid discretization, component of velocity on cell edges is prognosed, is especially well-suited for higher-resolution, mesoscale [atmosphere](#) and [ocean](#) simulations. The land ice model takes advantage of the SCVT-dual mesh, which is a triangular Delaunay tessellation appropriate for use with Finite-Element-based discretizations.

The current MPAS release is version 7.0. Please refer to each core for changes, and to the GitHub repository for source.

The “less preferred” method of obtaining code

One can navigate to a download link from the MPAS homepage at <https://mpas-dev.github.io/>



A screenshot of a web browser displaying the MPAS Atmosphere Public Releases page. The URL in the address bar is <https://mpas-dev.github.io/>. The page title is "MPAS Atmosphere Public Releases". On the left, there is a sidebar with links for Overview, Download, and Resources. The main content area contains a brief history of releases, a large image of a globe with a Voronoi mesh, and a list of download links. A red arrow points to the "MPAS source code download" link.

MPAS Atmosphere Public Releases

MPAS Home

Overview

- [MPAS-Atmosphere](#)
- [MPAS-Albany Land Ice](#)
- [MPAS-Ocean](#)
- [MPAS-Seoice](#)
- [Data Assimilation](#)
- [Publications](#)
- [Presentations](#)

Download

- [MPAS-Atmosphere download](#)
- [MPAS-Albany Land Ice download](#)
- [MPAS-Ocean download](#)
- [MPAS-Seoice download](#)

Resources

- [License Information](#)
- [Wiki](#)
- [Bug Tracker](#)
- [Mailing Lists](#)
- [MPAS Developers Guide](#)
- [MPAS Mesh Specification Document](#)

MPAS Atmosphere 7.0 was released on 8 June 2019.

As of September 2018, official support for MPAS-Atmosphere has migrated from the Google Groups forum to a web forum hosted by NCAR's Mesoscale and Microscale Meteorology. Users are encouraged to post any questions related to building and running MPAS-Atmosphere to the appropriate sub-topic in the MPAS-Atmosphere forum at <http://forum.mmm.ucar.edu/phpBB3/>. Posting to the forum requires the creation of an account, but no account is needed to browse the forum.

[MPAS Atmosphere 7.0 release notes](#)

[MPAS source code download](#) 

[MPAS-Atmosphere Users' Guide](#)

[MPAS-Atmosphere tutorial presentations](#)

[MPAS-Atmosphere meshes](#)

[Configurations for idealized test cases](#)

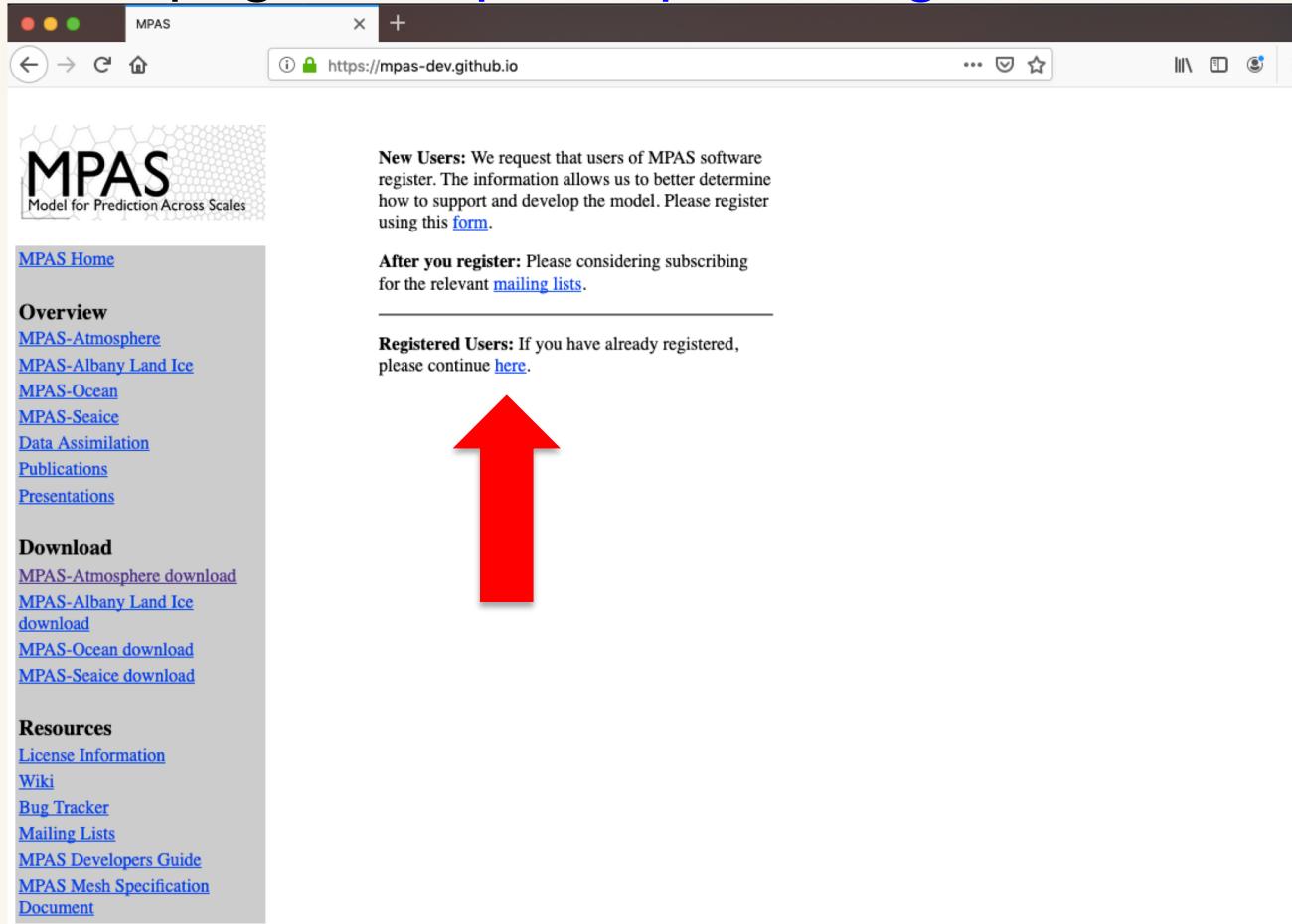
[Sample input files for real-data simulations](#)

[Visualization and analysis tools](#)

A variable resolution MPAS Voronoi mesh

The “less preferred” method of obtaining code

One can navigate to a download link from the MPAS homepage at <https://mpas-dev.github.io/>



A screenshot of a web browser window displaying the MPAS homepage. The URL in the address bar is <https://mpas-dev.github.io/>. The page content includes a logo and text for new users, a section for registered users, and a sidebar with navigation links for Overview, Download, and Resources. A large red arrow points upwards from the bottom of the page towards the download links.

New Users: We request that users of MPAS software register. The information allows us to better determine how to support and develop the model. Please register using this [form](#).

After you register: Please consider subscribing for the relevant [mailing lists](#).

Registered Users: If you have already registered, please continue [here](#).

Download

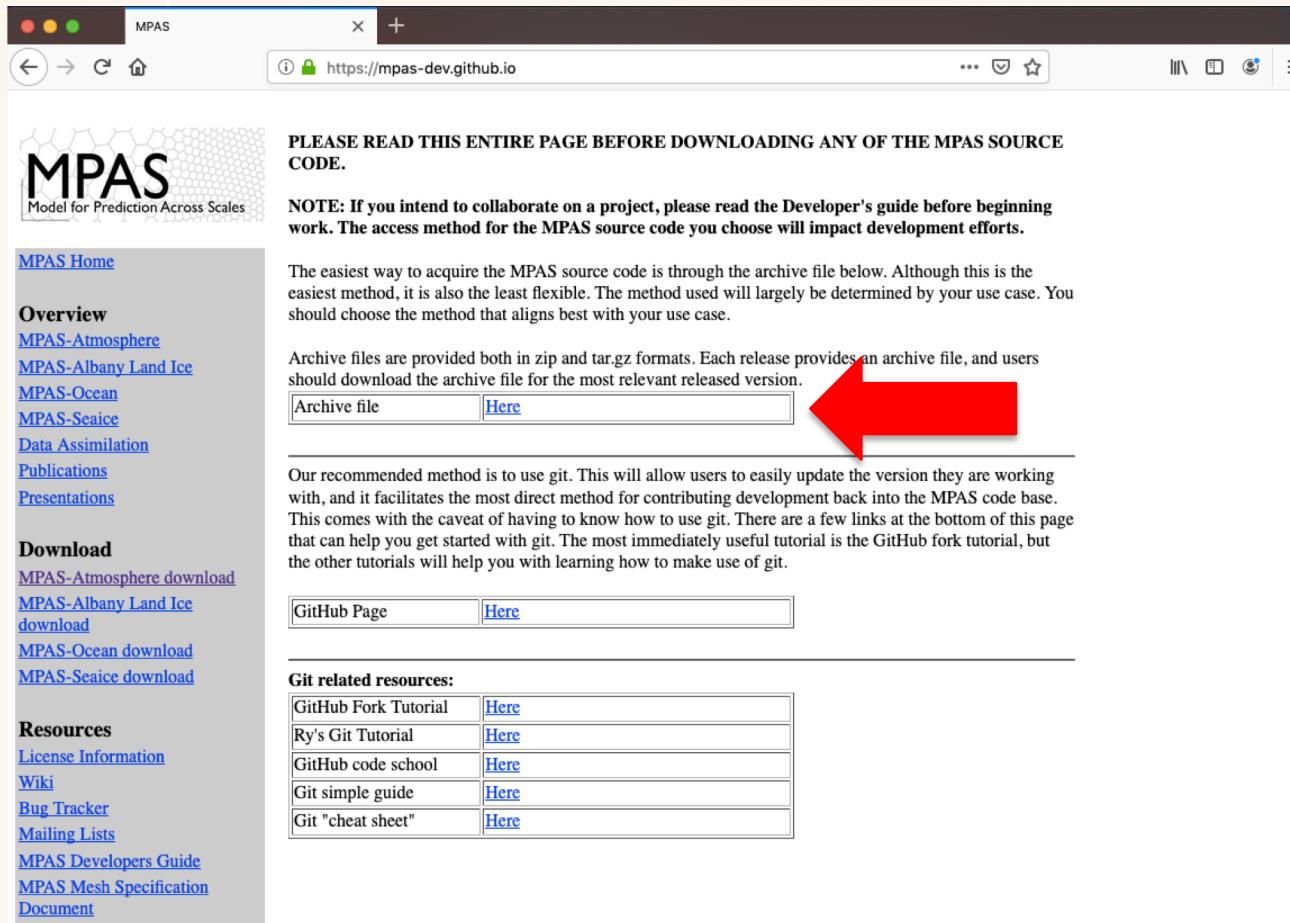
[MPAS-Atmosphere download](#)
[MPAS-Albany Land Ice download](#)
[MPAS-Ocean download](#)
[MPAS-Seaiice download](#)

Resources

[License Information](#)
[Wiki](#)
[Bug Tracker](#)
[Mailing Lists](#)
[MPAS Developers Guide](#)
[MPAS Mesh Specification Document](#)

The “less preferred” method of obtaining code

One can navigate to a download link from the MPAS homepage at <https://mpas-dev.github.io/>



PLEASE READ THIS ENTIRE PAGE BEFORE DOWNLOADING ANY OF THE MPAS SOURCE CODE.

NOTE: If you intend to collaborate on a project, please read the Developer's guide before beginning work. The access method for the MPAS source code you choose will impact development efforts.

The easiest way to acquire the MPAS source code is through the archive file below. Although this is the easiest method, it is also the least flexible. The method used will largely be determined by your use case. You should choose the method that aligns best with your use case.

Archive files are provided both in zip and tar.gz formats. Each release provides an archive file, and users should download the archive file for the most relevant released version.

Archive file [Here](#)

Our recommended method is to use git. This will allow users to easily update the version they are working with, and it facilitates the most direct method for contributing development back into the MPAS code base. This comes with the caveat of having to know how to use git. There are a few links at the bottom of this page that can help you get started with git. The most immediately useful tutorial is the GitHub fork tutorial, but the other tutorials will help you with learning how to make use of git.

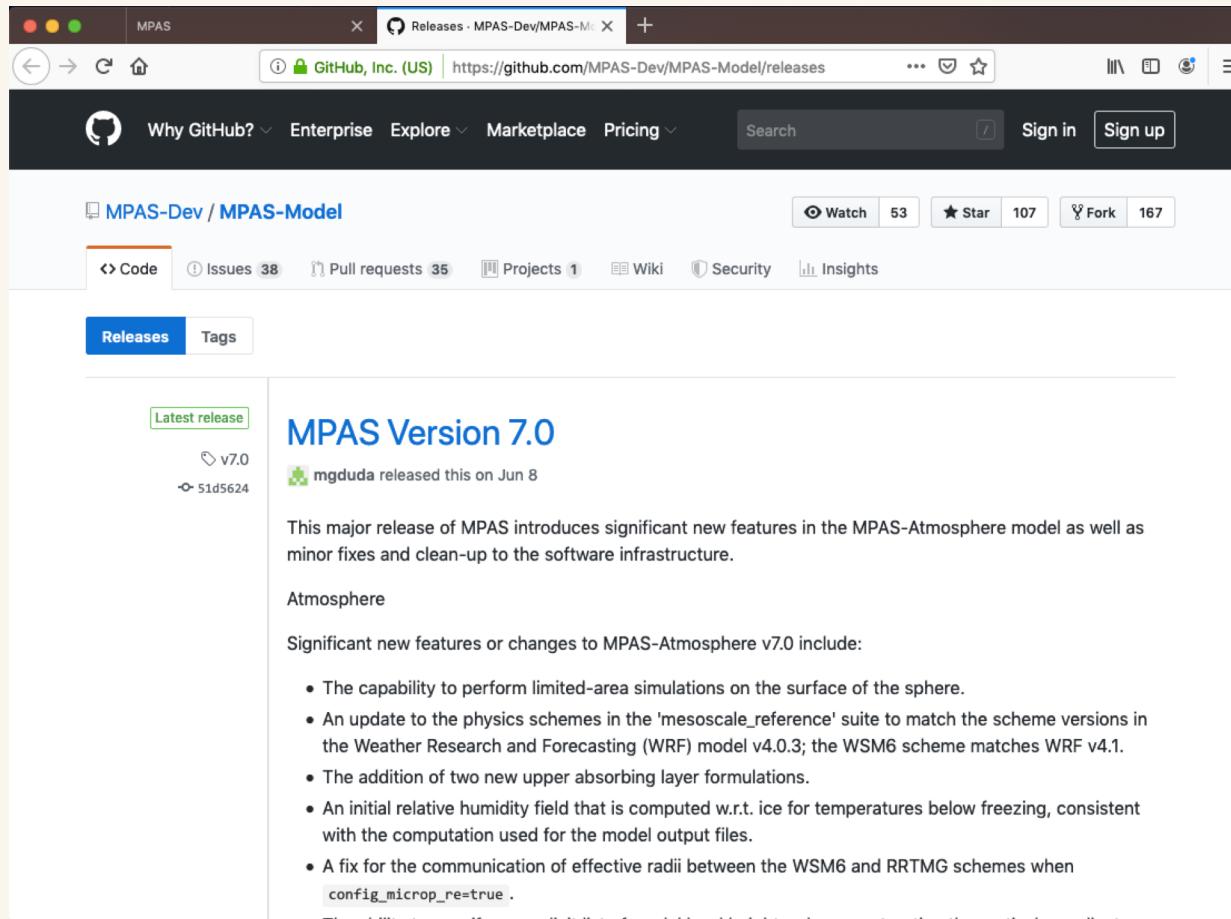
GitHub Page [Here](#)

Git related resources:

GitHub Fork Tutorial	Here
Ry's Git Tutorial	Here
GitHub code school	Here
Git simple guide	Here
Git "cheat sheet"	Here

The “less preferred” method of obtaining code

One can navigate to a download link from the MPAS homepage at <https://mpas-dev.github.io/>



A screenshot of a web browser displaying the GitHub releases page for the MPAS-Model repository. The URL in the address bar is <https://github.com/MPAS-Dev/MPAS-Model/releases>. The page shows the 'Releases' tab selected. A green button labeled 'Latest release' is visible. Below it, the title 'MPAS Version 7.0' is displayed, along with the release date 'Jun 8' and the user 'mgduda'. A brief description states: 'This major release of MPAS introduces significant new features in the MPAS-Atmosphere model as well as minor fixes and clean-up to the software infrastructure.' Under the 'Atmosphere' heading, a list of changes includes:

- The capability to perform limited-area simulations on the surface of the sphere.
- An update to the physics schemes in the 'mesoscale_reference' suite to match the scheme versions in the Weather Research and Forecasting (WRF) model v4.0.3; the WSM6 scheme matches WRF v4.1.
- The addition of two new upper absorbing layer formulations.
- An initial relative humidity field that is computed w.r.t. ice for temperatures below freezing, consistent with the computation used for the model output files.
- A fix for the communication of effective radii between the WSM6 and RRTMG schemes when `config_microp_re=true`.

The “less preferred” method of obtaining code

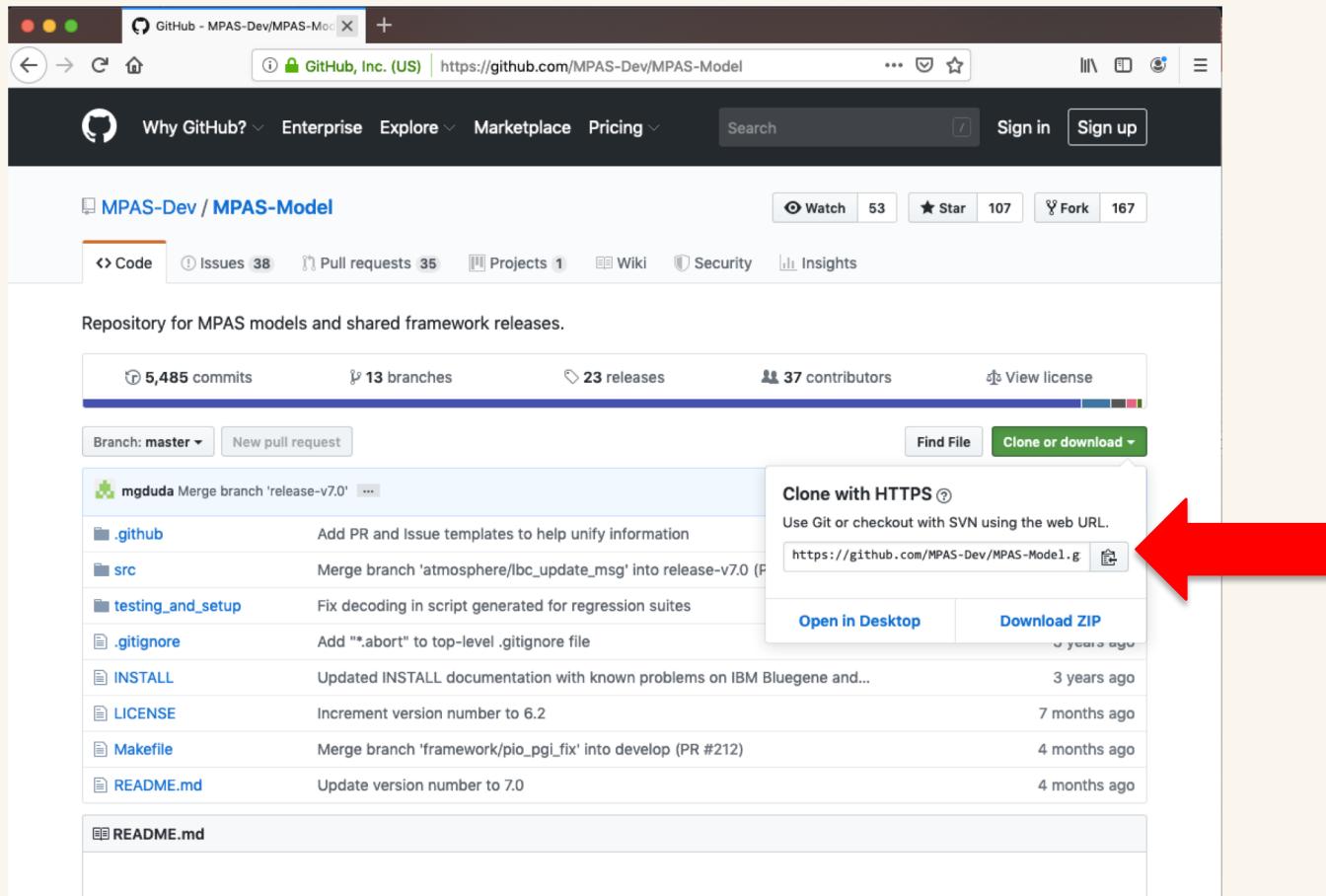
Downloading a `.tar.gz` file of a particular release of the MPAS code certainly works, but it has several disadvantages:

1. You'll only obtain a specific release of the code
2. It's more difficult to see what local modifications have been made to the code
3. There's no easy route to updating to a newer release while preserving your local code modifications
4. There's no direct way to see the history of changes to parts of the code

The preferred method of obtaining code

A much better option is to *clone* the MPAS-Model repository

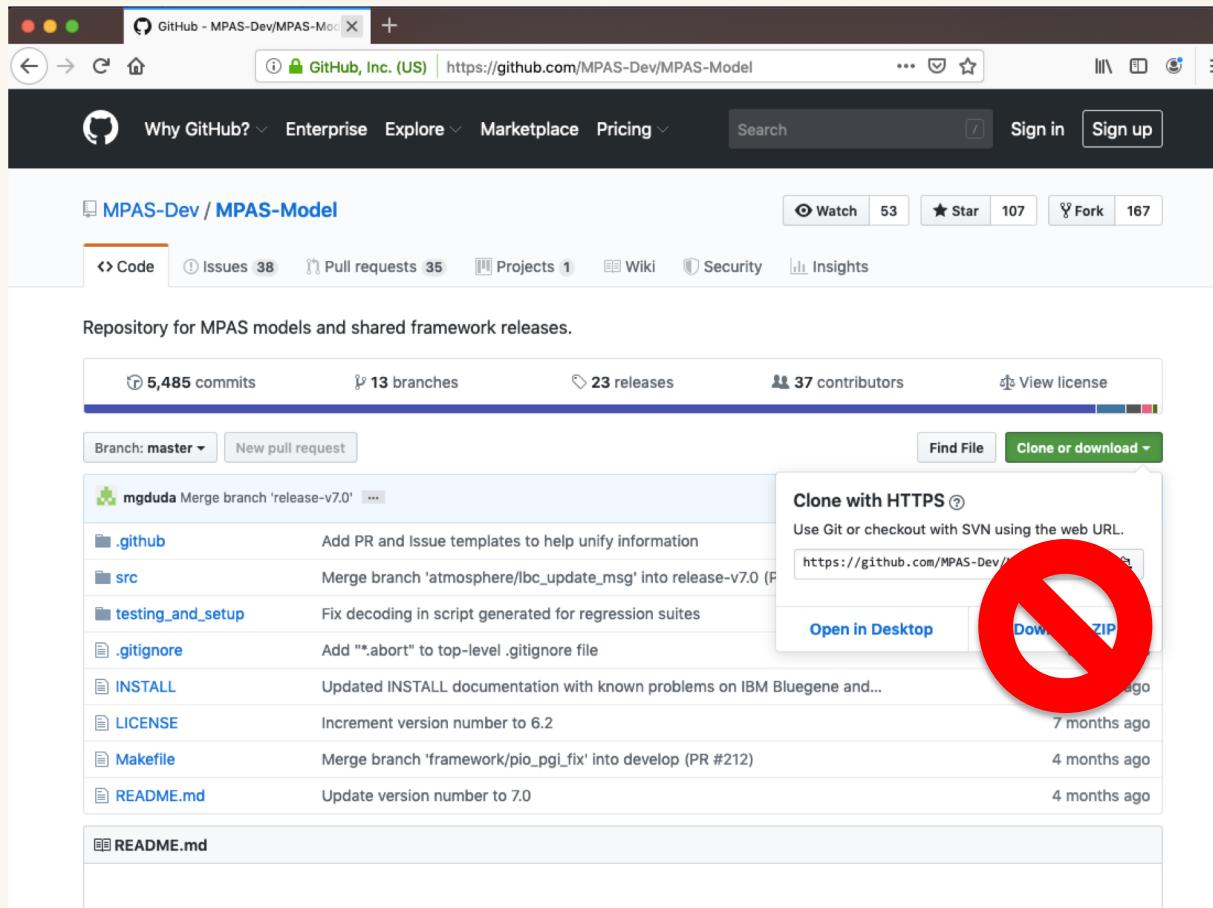
- The repository URL can be found from the MPAS GitHub page at <https://github.com/MPAS-Dev/MPAS-Model>



The preferred method of obtaining code

A much better option is to *clone* the MPAS-Model repository

- The repository URL can be found from the MPAS GitHub page at <https://github.com/MPAS-Dev/MPAS-Model>



The preferred method of obtaining code

From the command-line, the following should be sufficient:

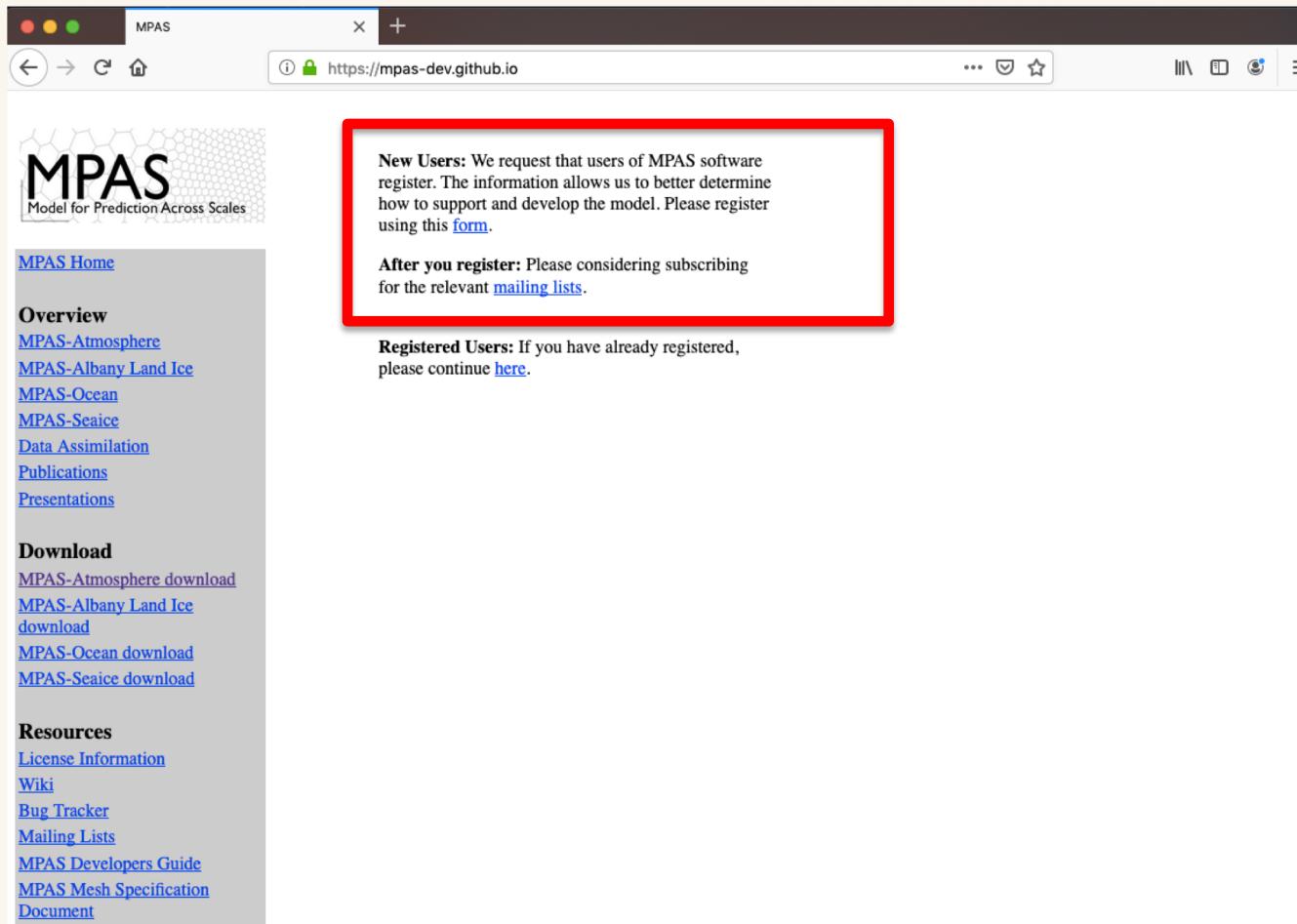
```
git clone https://github.com/MPAS-Dev/MPAS-Model.git
```

Cloning the repository should take about 10 seconds or less...

```
$ git clone https://github.com/MPAS-Dev/MPAS-Model.git
Cloning into 'MPAS-Model'...
remote: Enumerating objects: 71, done.
remote: Counting objects: 100% (71/71), done.
remote: Compressing objects: 100% (47/47), done.
remote: Total 46608 (delta 38), reused 42 (delta 23), pack-reused
46537
Receiving objects: 100% (46608/46608), 19.65 MiB | 2.57 MiB/s,
done.
Resolving deltas: 100% (35848/35848), done.
```

After obtaining the code for the first time

You may also like to register as an MPAS user and join the MPAS-Atmosphere Users mailing list



The screenshot shows a web browser window with the MPAS logo in the top left corner. The URL in the address bar is <https://mpas-dev.github.io>. The main content area contains the following text:

New Users: We request that users of MPAS software register. The information allows us to better determine how to support and develop the model. Please register using this [form](#).

After you register: Please consider subscribing for the relevant [mailing lists](#).

Registered Users: If you have already registered, please continue [here](#).

The left sidebar contains navigation links for Overview, Download, and Resources.

Preliminary requirements

In order to compile MPAS and its required libraries, working C and Fortran compilers are necessary

- The Fortran compiler should be recent enough to support the ISO_C_BINDING module from the Fortran 2003 standard and procedure pointer components of derived types
- Most versions of common compilers from the last couple of years should be fine

Building MPAS requires *at least* the following libraries:

- Any implementation of MPI-2, e.g., MPICH, MVAPICH, OpenMPI
 - Ensure that `mpif90` and `mpicc` commands are in your path
- Parallel-netCDF (<http://trac.mcs.anl.gov/projects/parallel-netcdf/>)
 - Set `PNETCDF` environment variable to base installation directory
- PIO (<https://github.com/NCAR/ParallelIO/>)
 - Set `PIO` environment variable to base installation directory

Preliminary requirements: the easiest route

Assuming Fortran and C compilers are available, and a working MPI installation is also available, installing Parallel-netCDF and PIO should take less than 10 minutes:

Parallel-netCDF 1.8.1

```
$ setenv CC gcc
$ setenv FC gfortran
$ setenv F77 gfortran
$ setenv MPICC mpicc
$ setenv MPIF90 mpif90
$ setenv MPIF77 mpif90
$ setenv PNETCDF /home/duda/pnetcdf
$ cd parallel-netcdf-1.8.1
$ ./configure \
  --prefix=$PNETCDF \
  --disable-cxx
$ make
$ make install
```

PIO 1.7.1

(Assuming environment variables from Parallel-NetCDF installation)

```
$ setenv MPIFC mpif90
$ setenv PNETCDF_PATH $PNETCDF
$ setenv PIO /home/duda/pio
$ cd piol_7_1/pio
$ ./configure \
  --prefix=$PIO \
  --disable-netcdf \
  --disable-mpio
$ make
$ make install
```

Which versions of PIO should I try?

The PIO library is undergoing rapid development, and many different versions of the library are available; *which versions are supported and recommended?*

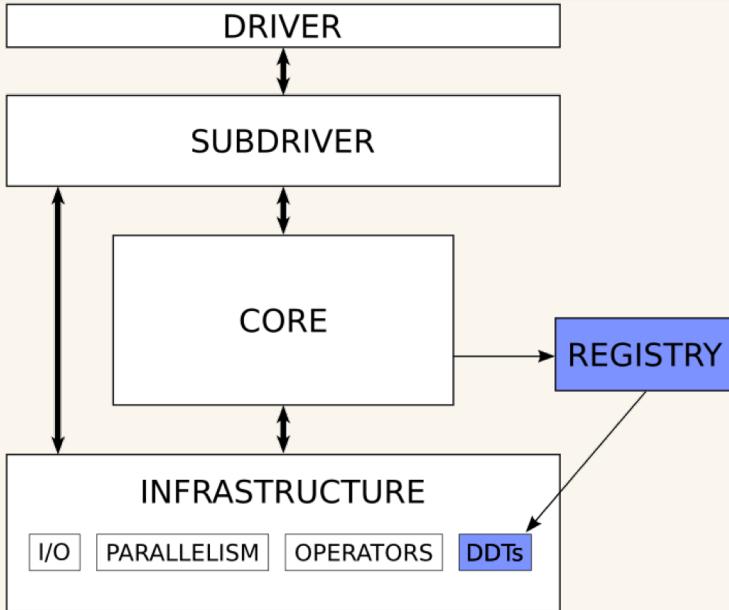
1) For ease of installation, try PIO 1.7.1

- Can be installed using only standard ‘configure’ and ‘make’ tools
- Supports NetCDF-3 and Parallel-netCDF I/O
- Download: https://github.com/NCAR/ParallelIO/releases/tag/pio1_7_1

2) If netCDF-4 I/O is needed or desirable, try the latest PIO release

- Requires recent versions of ‘cmake’, plus standard ‘make’
- Supports netCDF-3, netCDF-4 (in parallel via PHDF5), and Parallel-netCDF I/O
- Download: <https://github.com/NCAR/ParallelIO/>
- See iolib_installation.sh at
<http://www2.mmm.ucar.edu/people/duda/files/mpas/sources/>

Model Organization



MPAS-Model/
 ↳ src/

```

  ↳ core_atmosphere/
  ↳ core_init_atmosphere/
  ↳ core_landice/
  ↳ core_ocean/
  ↳ driver/
  ↳ external/
  ↳ framework/
  ↳ operators/
  
```

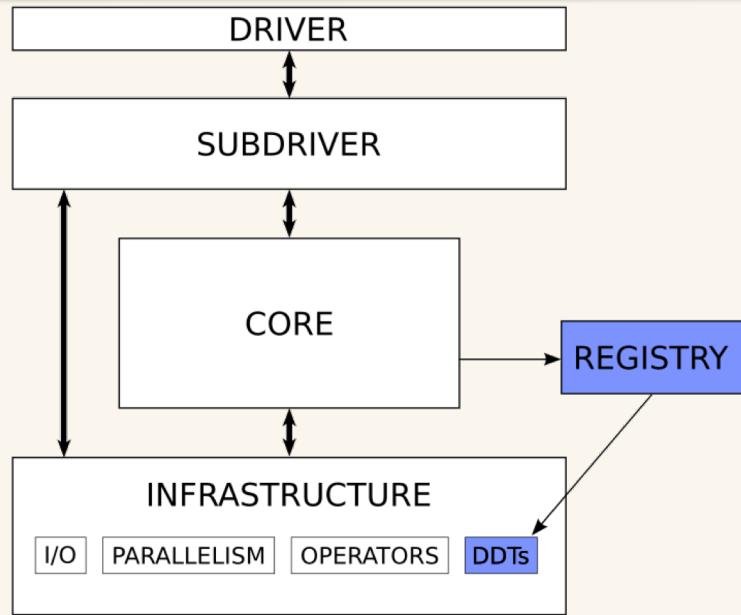
MPAS-Atmosphere Tutorial

9 – 11 September 2019, Boulder

Checking out the MPAS code provides all MPAS models, not just MPAS-Atmosphere

- All models share a common set of infrastructure modules
- Each MPAS model is implemented as a “core” that lives in its own directory
- User must select which “core” to compile
- Each “core” is associated with a source code subdirectory under src/ and has a Registry file (similar to WRF)

Model Organization



Running MPAS-Atmosphere involves two “cores”:

- The **init_atmosphere** core is responsible for
 - Interpolating static fields to the mesh (similar to geogrid.exe)
 - Generating a vertical grid (similar to real.exe)
 - Horizontally and vertically interpolating meteorological data to the 3-d grid (similar to metgrid.exe and real.exe)
 - *Where do we get meteorological data? From ungrb.exe!*
- The **atmosphere** core is the model itself, the equivalent of wrf.exe

Compiling MPAS

There is no “configuration” step for MPAS, unlike, e.g., for the WRF model

- All build flags are either set in the top-level Makefile or on the command-line

General MPAS build command:

```
$ make target CORE=core <options>
```

target can be either

clean

or

xlf

gfortran

ifort

pgi

llvm

... plus a few others...

For MPAS-Atmosphere, **core** may be

atmosphere

init_atmosphere

<options> can be zero or more of

DEBUG=true

AUTOCLEAN=true

PRECISION=single

OPENMP=true

USE_PIO2=true

Compiling MPAS

There is no “configuration” step for MPAS, unlike, e.g., for the WRF model

- All build flags are either set in the top-level Makefile or on the command-line

General MPAS build command:

```
$ make target CORE=core <options>
```

target can be either

clean

or

xlf

gfortran

ifort

pgi

bluegene

... plus a few others...

For MPAS-Atmosphere, core may be

atmosphere

init_atmosphere

<options>

DEBUG=true

AUTOCLI=true

PRECISION=double

OPENMP=true

USE_PIO2=true

If using the latest PIO2 library, be sure to set this option!

Compiling MPAS

Typical build of both the `init_atmosphere` and `atmosphere` cores involves:

```
$ make gfortran CORE=init_atmosphere (build init_atmosphere_model)
```

```
$ make clean CORE=atmosphere (clean any infrastructure files used by both  
init_atmosphere and atmosphere)
```

```
$ make gfortran CORE=atmosphere (build atmosphere_model)
```

By default, MPAS cores are built with double-precision reals

MPAS-Atmosphere can be built in single precision

- Add `PRECISION=single` to build commands for single-precision executables
- execution time ~35% less compared with double-precision
- output files approximately half as large
- Beginning with MPAS v3.0, it is possible to run the model in double precision while writing history files in single precision!

Cheyenne users: you're in luck!

If you're working on Cheyenne, there are just seven commands to obtain and build everything you need (assuming the default module setup):

```
module unload netcdf
module load pio
git clone https://github.com/MPAS-Dev/MPAS-Model.git
cd MPAS-Model
make ifort CORE=init_atmosphere PRECISION=single USE_PIO2=true
make clean CORE=atmosphere
make ifort CORE=atmosphere PRECISION=single USE_PIO2=true
```