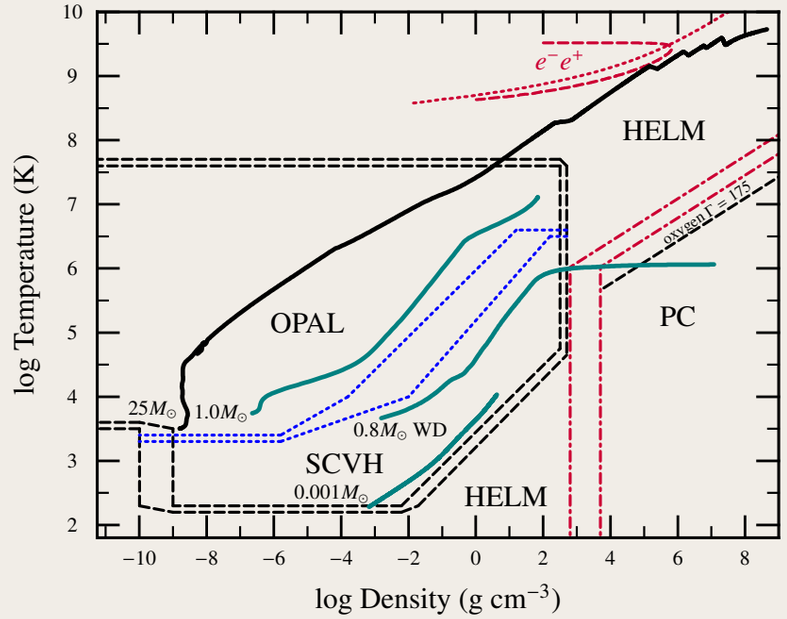
The MESA Stellar Evolution Code

Rich Townsend

University of Wisconsin-Madison



Topics in Computing for Astronomy — Oct 30 2012

*M*odules for *E*xperiments in *S*tellar *A*strophysics

Why a New Stellar Evolution Code?

- **Openness**: anyone can download sources from the website.
- **Modularity**: independent modules for physics and for numerical algorithms; the parts can be used stand-alone.
- **Comprehensive Microphysics**: up-to-date, wide-ranging, flexible, and independently useable microphysics modules.
- **Modern Techniques**: advanced AMR, fully coupled solution for composition and abundances, mass loss and gain, etc.
- **Performance**: runs well on a personal computer and makes effective use of parallelism with multi-core architectures.
- **Wide Applicability**: capable of calculating the evolution of stars in a wide range of environments.

The MESA Council

- Members:
  - Lars Bildsten (UCSB/KITP)
  - Bill Paxton (UCSB/KITP)
  - Frank Timmes (Arizona State)
  - Falk Herwig (University of British Columbia)
  - Ed Brown (Michigan State)
  - Aaron Dotter (STScI; now Oz)
  - Matteo Cantiello (UCSB/KITP)
  - Rich Townsend (University of Wisconsin-Madison)

- Duties
  - Code development
  - Instrument paper(s)
  - Infrastructure maintenance (website, forum, SDK)
  - Summer schools

Online MESA Resources

- Website (main resource):

  http://mesa.sourceforge.net/index.html

- Mailing list (bug reports, results):

  https://lists.sourceforge.net/lists/listinfo/MESA-users

- Forum (bug reports, results, lectures, FAQs):

  http://mesastar.org/

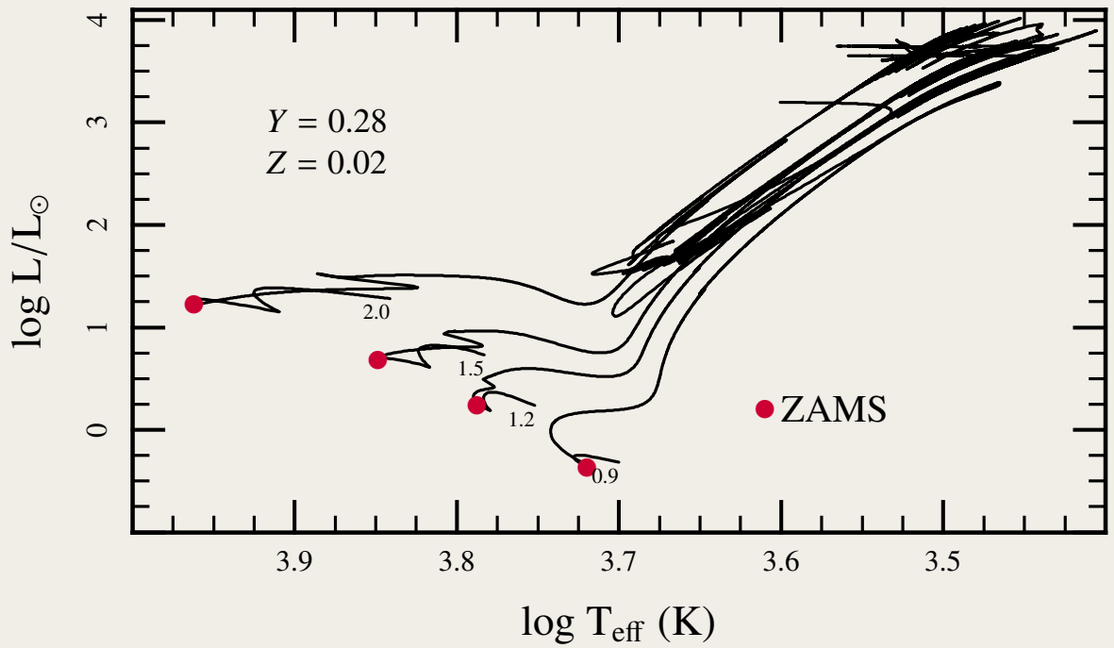
- SDK (software tools):

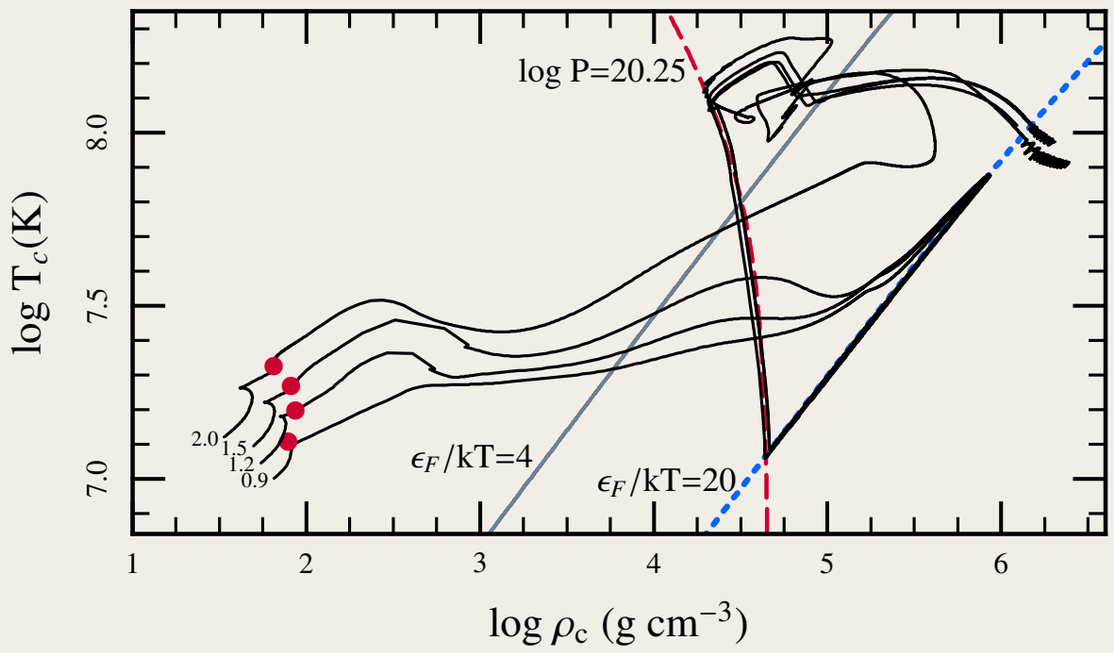
  http://www.astro.wisc.edu/~townsend/static.php?ref=mesasdk

- Instrument paper:

  http://adsabs.harvard.edu/abs/2011ApJS..192....3P

MESA in Action: Low-Mass Stellar Evolution

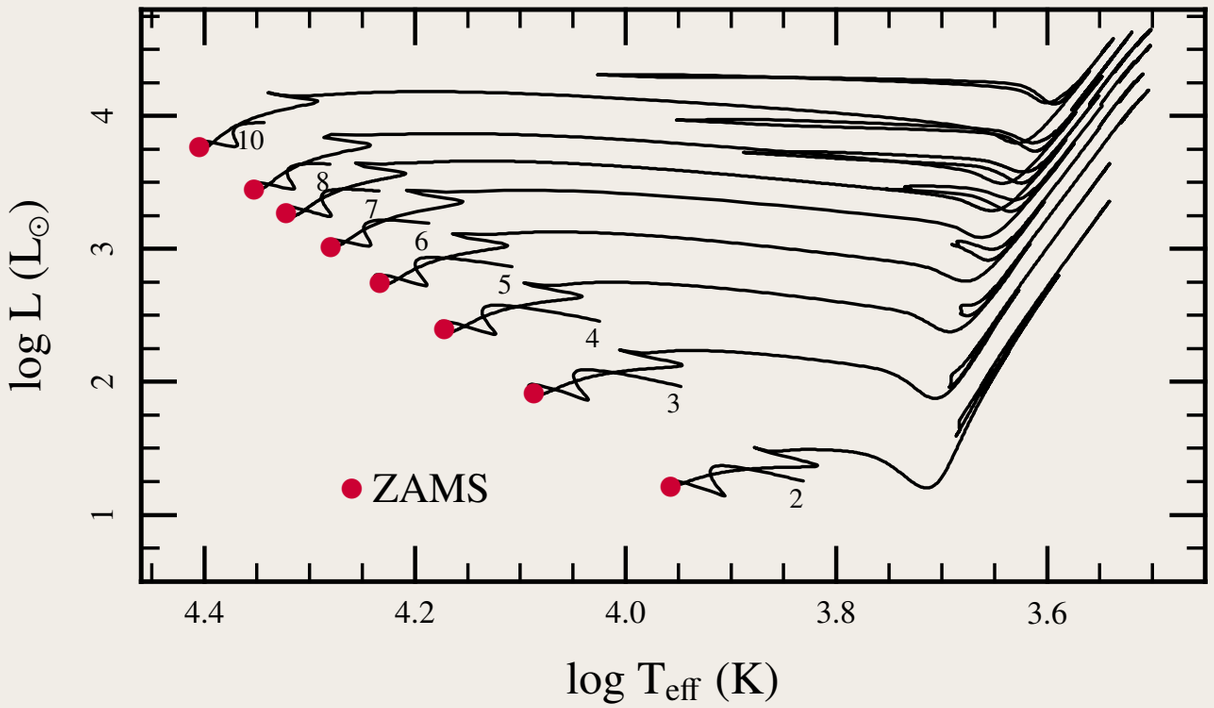


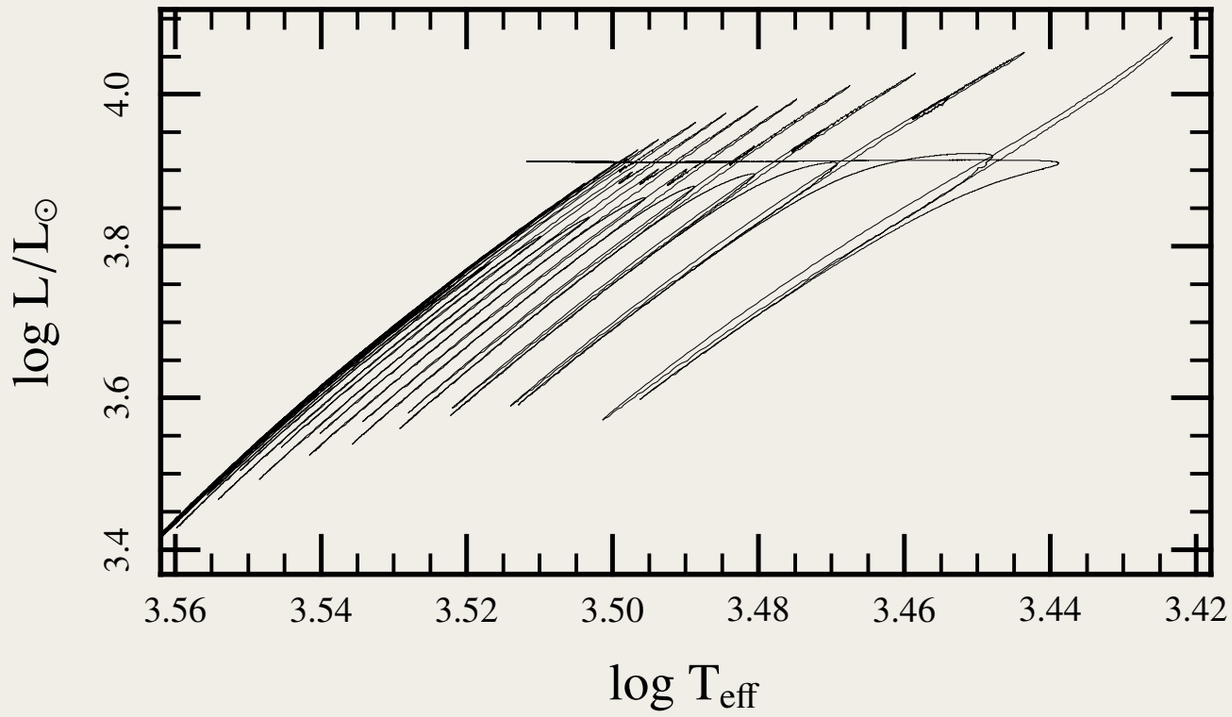


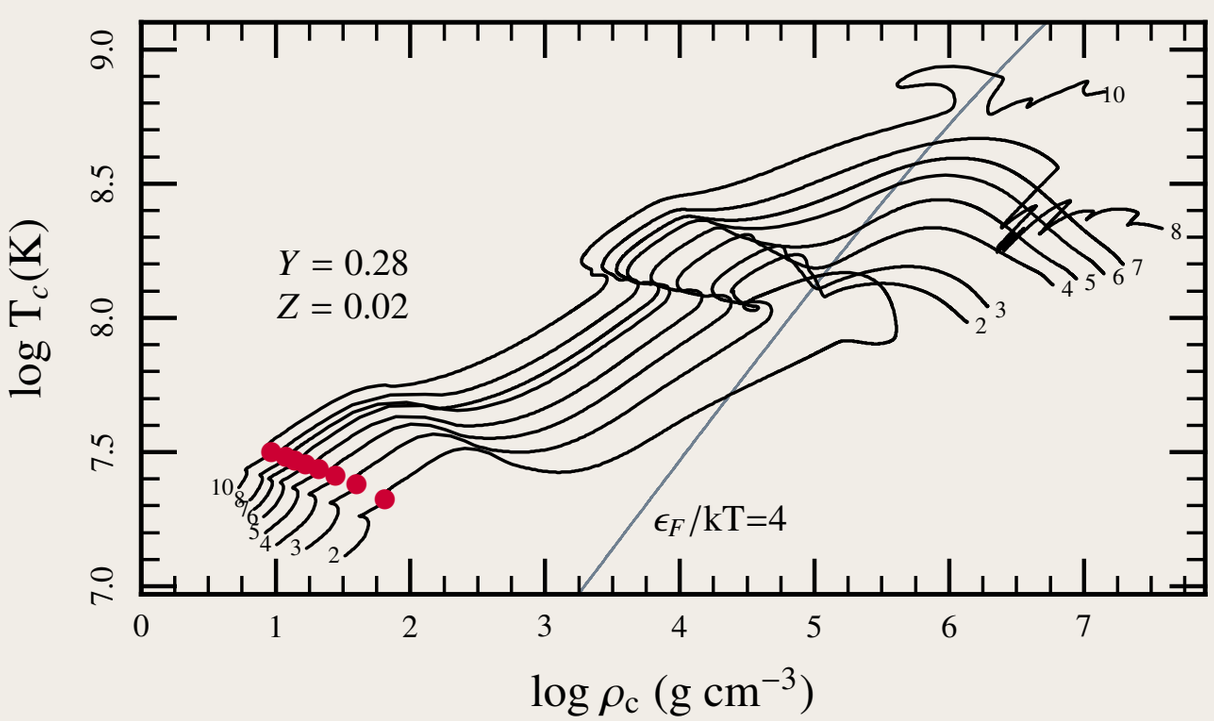
- Code goes through He core flash without problems
- No need for ad-hoc ‘transition’ between RGB and HB (i.e., no *EZ-Web* fudge)
- Thermal pulses on AGB also handled
- Mass-loss allows evolution through to WD remnant

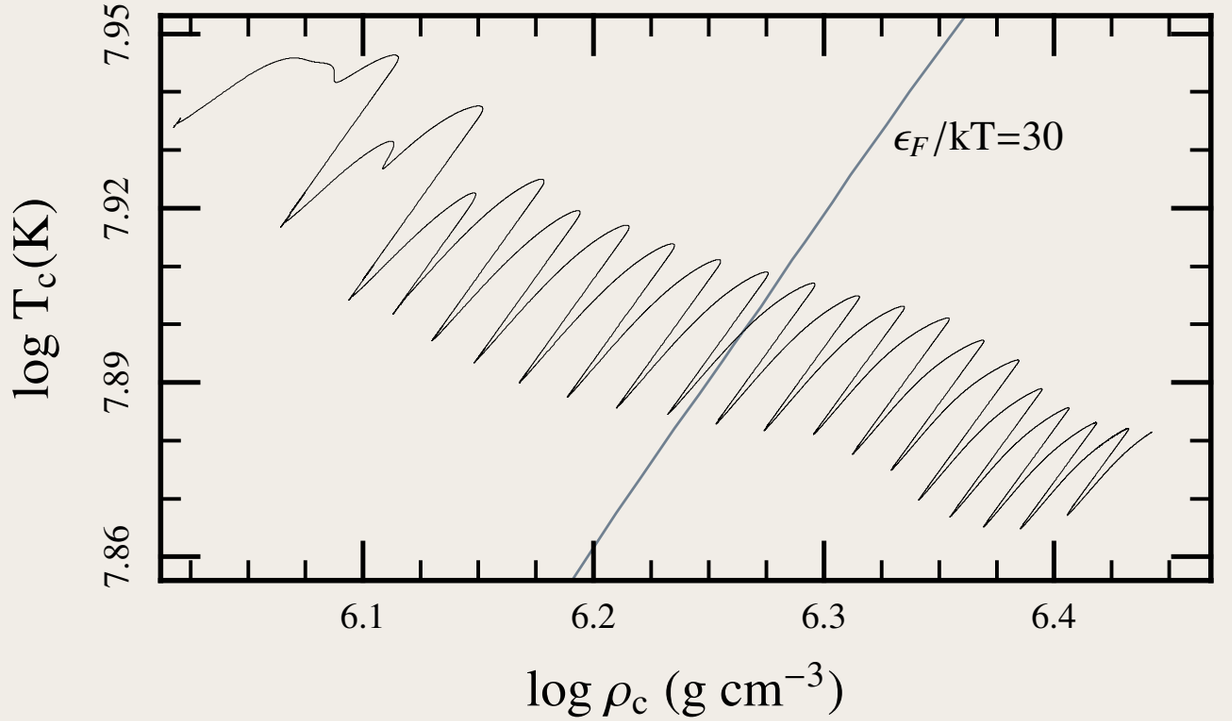
From MESA Instrument Paper

MESA in Action: Intermediate-Mass Stars



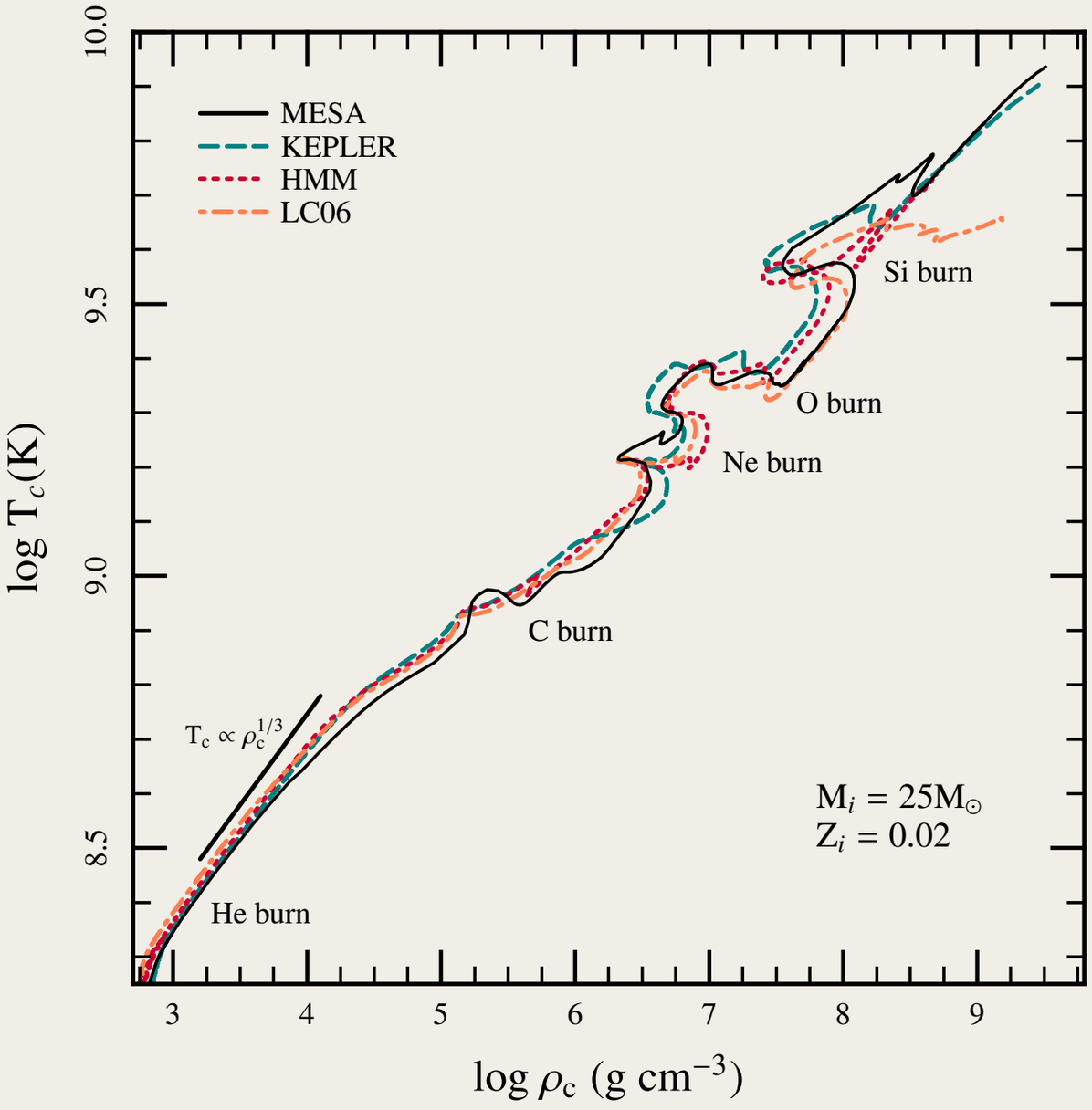


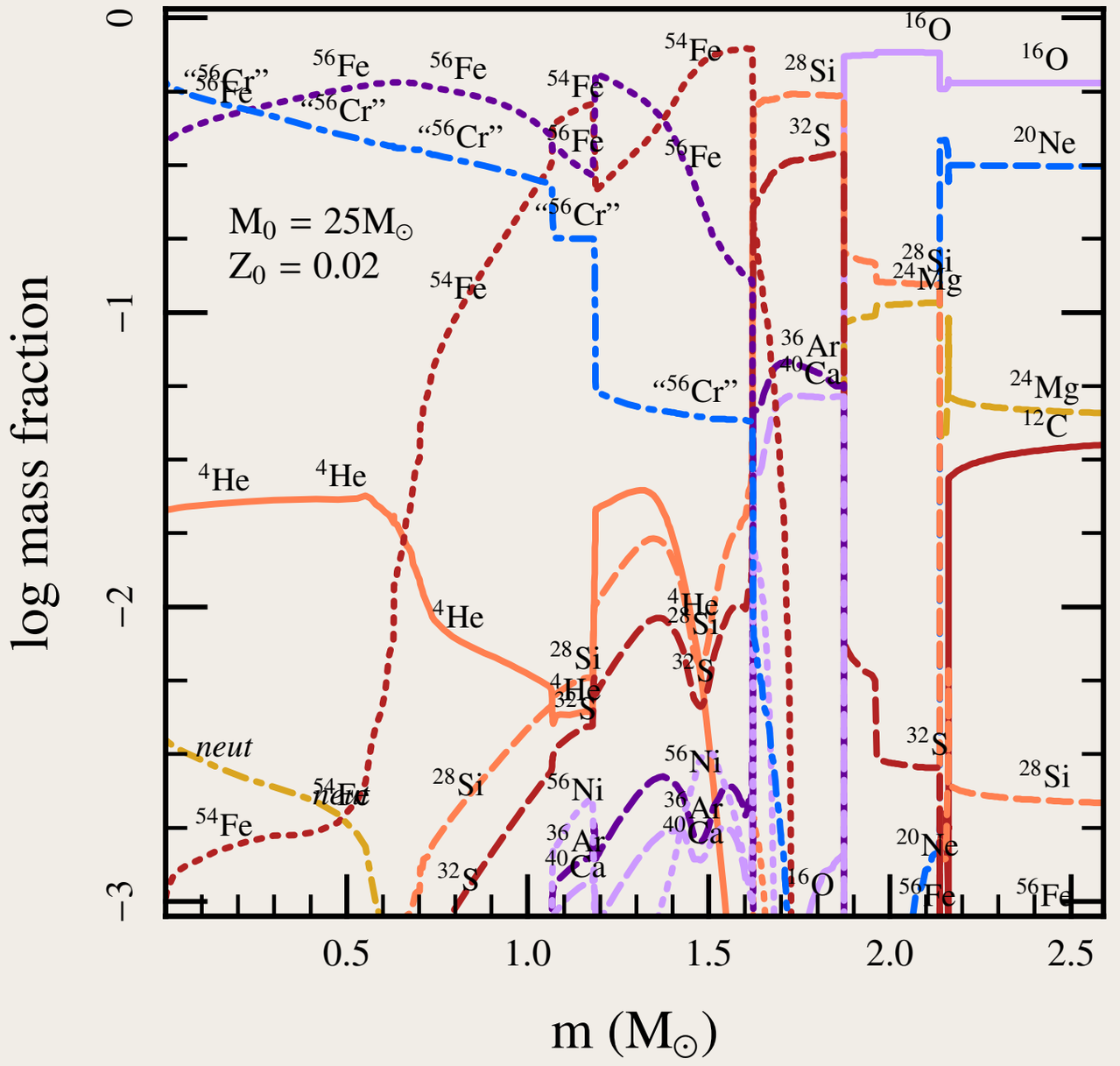




From MESA Instrument Paper

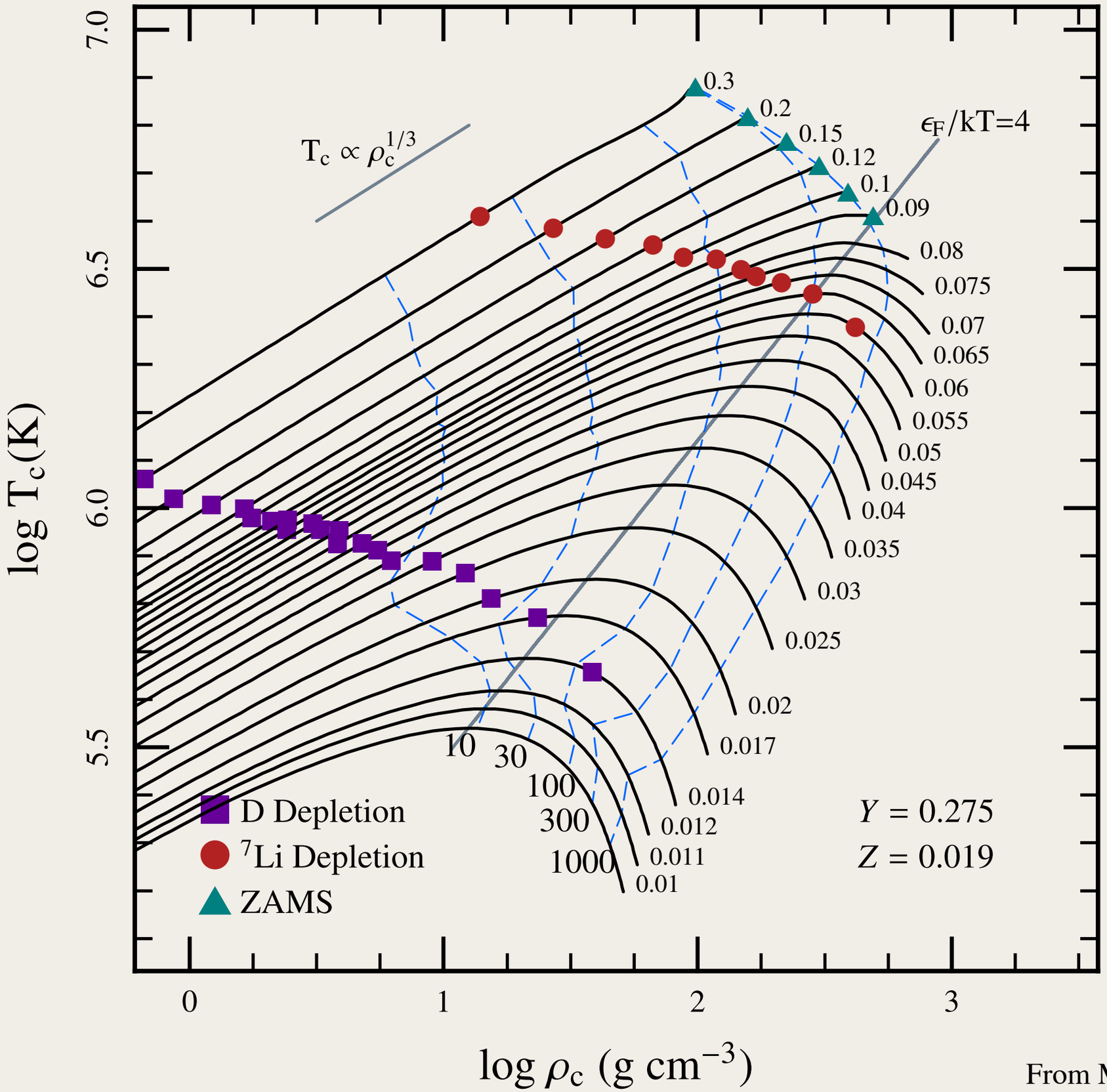
MESA in Action: Massive Stars to Core Collapse





From MESA Instrument Paper

MESA in Action: Brown Dwarfs & Giant Planets

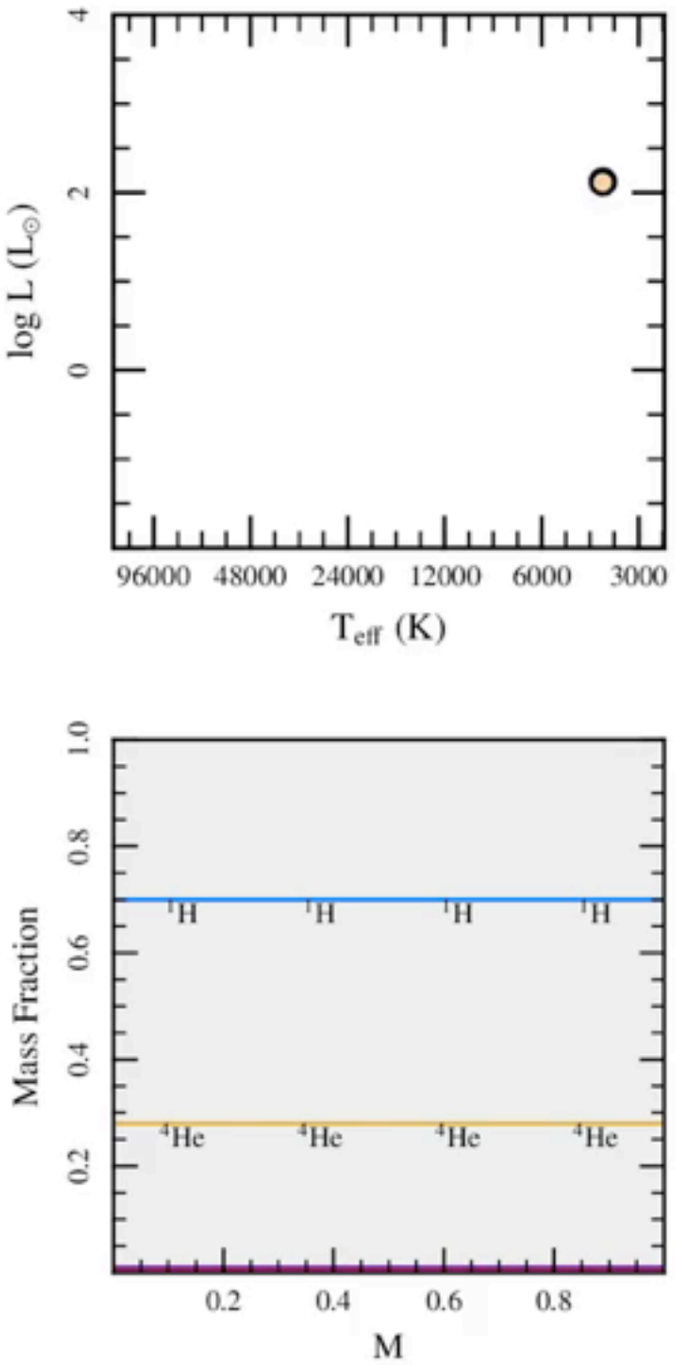


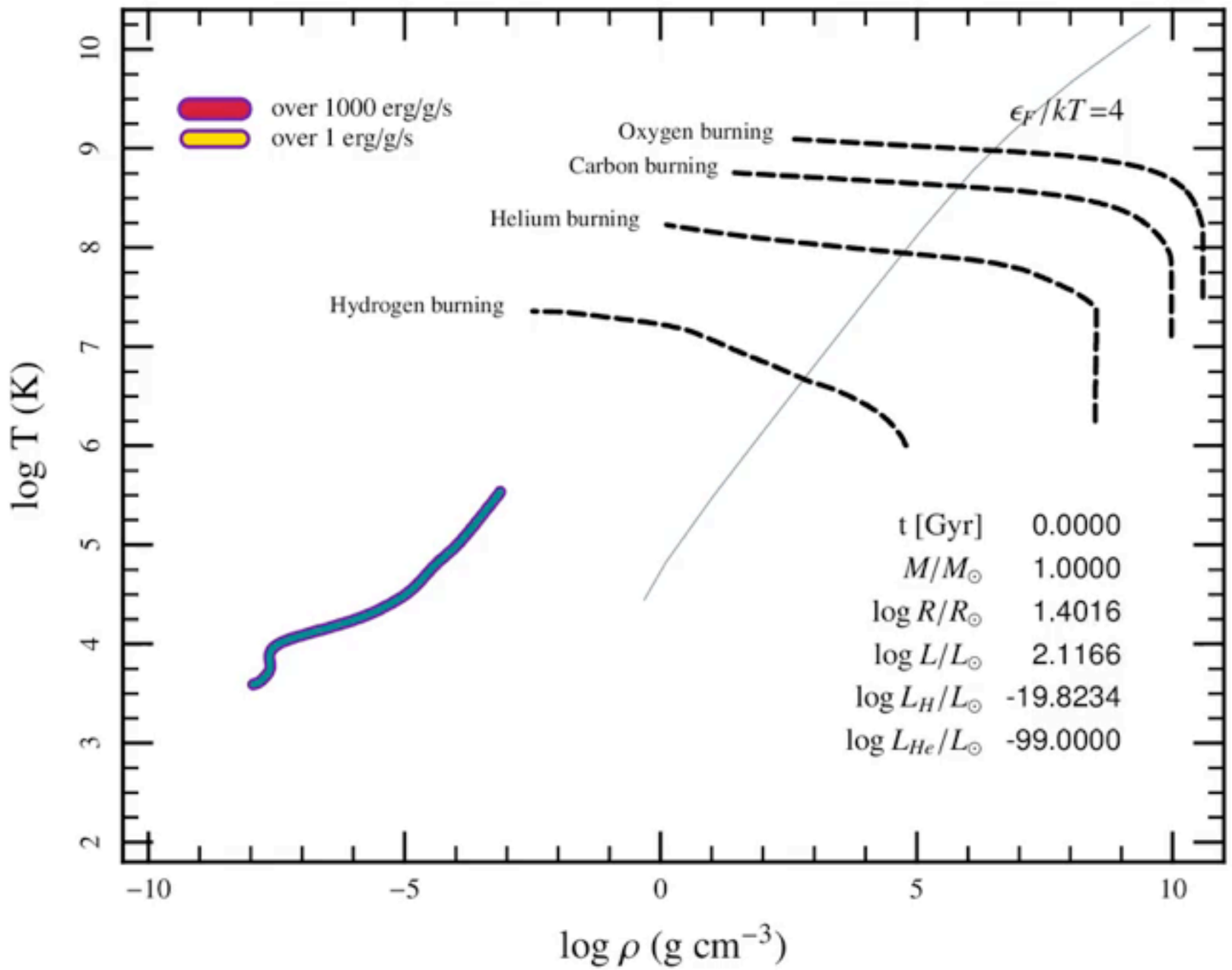
From MESA Instrument Paper

MESA in Action: Evolution of a Solar-Mass Star

Credit: Josiah Schwab

MESA in Action: Evolution of a Solar-Mass Star





Credit: Josiah Schwab

Installing MESA

- Obtain a current copy of the MESA tree:

```
svn co -r 4631 http://mesa.svn.sourceforge.net/svnroot/mesa/trunk mesa-4631
```

- Set environment variables:

```
export OMP_NUM_THREADS=8
export MESA_DIR=<blah>/mesa-4298
```

- Build MESA:

```
cd $MESA_DIR
./install
```

- Caveats:
  - Revision number is always changing; see http://mesa.sourceforge.net/getting_started.html
  - Takes up 4GB of disk space or more
  - Change OMP_NUM_THREADS to number of processor cores on your machine
  - Change syntax of environment-variable setting if you are using csh

The MESA Software Development Kit (SDK)

- What's it for?
  - Hassle-free compilation of MESA
  - Works on Linux and Mac OS X (Intel-based)

- What's in it?
  - gcc/gfortran 4.7 compilers (good support for Fortran 2003)
  - BLAS/LAPACK libraries (linear algebra)
  - PGPLOT library (graphics)
  - HDF5 library (file storage)

- Where do I get it from?

  http://www.astro.wisc.edu/~townsend/static.php?ref=mesasdk

- How do I install?
  - Linux: unpack tar archive (anywhere; don't need to be root user)
  - OS X: drag package into Applications folder

Running an Example: Solar-Mass Evolution

- Change into the example directory:

```
cd $MESA_ROOT/star/test_suite/1M_pre_ms_to_wd/
```

- Build the code:

```
./mk
```

- Run the code:

```
./rn
```

- Output produced in `LOGS` subdirectory:
  - `history.data` — global properties of all models in run
  - `profileN.data` (`N`=1,2,...) — internal structure of selected models
  - `profiles.index` — mapping between model number and `N`

Modifying the Example: Enabling Plotting

- Edit the `inlist_1.0` file:

```
!pgstar_flag = .true.
```

```
pgstar_flag = .true.
```

- Run the code:

```
./rn
```

Understanding `inlist` Files

- `inlist` format is defined by Fortran standard

- Overall file structure:

```
&star_job
   …
/ ! end of star_job namelist
```

Overall run parameters

```
&controls
   …
/ ! end of controls namelist
```

Detailed control parameters

```
&pgstar
   …
/ ! end of pgstar namelist
```

Plot parameters

Understanding `inlist` Files (cont.)

```
&star_job

      mesa_dir = '../../..'

      read_extra_star_job_inlist1 = .true.
      extra_star_job_inlist1_name = 'inlist_sub'


/ ! end of star_job namelist

&controls

      initial_mass = 1.0
      initial_z = 0.02d0

/ ! end of controls namelist

…
```

Path to $MESA_DIR dir

Read subsidiary inlist

Mass & metallicity

Reading MESA Output Files

- Grab the `read_mesa.pro` IDL procedure :

  http://mesastar.org/tools-utilities/idl

- Read data into IDL structures:

```
IDL> s = read_mesa('history.data')
IDL> help, s, /str
IDL> plot, s.log_Teff, s.log_L, xrange=[4.5,3.5], yrange=[0,4]
```

```
IDL> s = read_mesa('profile1.data')
IDL> help, s, /str
IDL> plot, s.logrho, s.logT
```

Other `test_suite` Examples

- **`sample_zams`** — build multiple ZAMS models
- **`7M_prems_to_AGB`** – 7 $M_\odot$, pre-main sequence to AGB
- **`example_astero`** — asteroseismology using `adipls` code (supplied)
- **`binary_rlo`** — Roche-lobe overflow in binary system
- **`1.5M_with_diffusion`** – 1.5 $M_\odot$, elemental diffusion
- **`massive_rotating`** – 15 $M_\odot$, initial rotation at 50% critical
- **`ns_c`** – 1 $M_\odot$ neutron star, accreting/burning carbon

Rolling Your Own

- Copy one of the test_suite directories:

```
cp -a $MESA_DIR/star/test_suite/1M_pre_ms_to_wd my_project
```

- Delete the `MESA_DIR` definition from `my_project/make/makefile` (this will cause MESA to pick up the definition from the MESA_DIR environment variable)
- Delete the `mesa_dir` definition from the inlist (this will cause MESA to pick up the definition from the MESA_DIR environment variable)
- Edit the `inlist(s)`
- Edit source files in `my_project/src`
- Clean, make and run:

```
cd my_project; ./clean; ./mk; ./rn
```