Intro to MesaScript

MESA Summer School 2016
Bill Wolf, UCSB
The Use Case

• You need to do many MESA star runs while changing only a few parameters

• You don’t like sed

• You think run_star_extras.f would be overkill

• Example: isochrones (evolve many models to the same age)
In real life

- Run each model with at least 5 different values of `step_overshoot.f_above_burn_h.core = 0.0, 0.1, 0.2, 0.3, and 0.4`

- Randomly select one of the following initial stellar masses: $M \in \{10, 20, 40, 60, 80\} M_{\odot}$.

- How do the $P - L$ and the $P - R$ relation depend on metallicity? Compute the relations for SMC and LMC compositions. In particular, is the P-R relation as claimed by Gieren et al. indeed insensitive to metallicity? Keep an eye on the effect of metallicity on the size and the topology of the blue loops.

{and the entire premise of today’s labs}
THERE’S GOT TO BE A BETTER WAY!
What is MesaScript?

- A **Domain Specific Language** built on top of Ruby
- Uses inlist-like syntax to mass-produce inlists
- Brings **variables, conditionals, loops, and functions** to inlists
- Sensibly organizes and typechecks inlists
Common Workflow (for me)

• Identify key parameters that need to be varied

• Code up a method that takes those parameters as arguments and creates an inlist

• Executing file (ruby my_script.rb) iterates through parameter values, firing off MESA runs for each combination
Demo: Overshoot Labs
Key Syntax

• **require ‘mesa_reader’:** tells Ruby to actually load mesa_script (makes the **Inlist** class available)

• **Inlist.make_inlist:** Creates a named file based on the code that follows (inside **do...end**)

• NO namelists! MesaScript figures that out.

• Ruby literals, not fortran literals (**true**, not **.true.**)
Assignments

- **Scalars**: `initial_mass 2.333` or `initial_mass(2.333)`
  (parens optional in ruby calls if unambiguous)
  *NOT* `initial_mass = 2.333`
  (simply a declaration)

- **Vectors**:
  `xa_central_lower_limit[1] = 1e-3` or `xa_central_lower_limit(1, 1e-3)` or `xa_central_lower_limit(1, 1e-3)`
  *NOT* `xa_central_lower_limit(1) = 1e-3`
Get Started Quick!

• Rarely start an inlist from scratch

• Instead, start with existing inlist and tweak it

• Convert an inlist to mesascript quickly with included `inlist2mesascript INLIST_FILE MESASCRIPIT_FILE`

• Resulting file, when executed, recreates the original inlist, but clean up and with no comments
FAQ

- Q: “Why didn’t you write this in python?”
  A: “Python isn’t so good at metaprogramming and has inflexible syntax, but I’m looking into it… slowly.”

- Q: “Seriously, I’d like to use this on my cluster, but they don’t have ruby installed, so… Python version?”
  A: “Check out rvm.io. RVM (ruby version manager) is… a ruby version manager (think anaconda for ruby) that doesn’t require root privileges”.
How can I get this?

- `gem install mesa_script`
- that’s it…
- Fuller documentation/installation instructions at [http://wmwolf.github.io/MesaScript/](http://wmwolf.github.io/MesaScript/)
- It’s technically extensible to other namelist-y things, so you could very easily make minor changes to make this work with binary... perhaps even GYRE