TARDIS

Fast modular supernova spectral synthesis

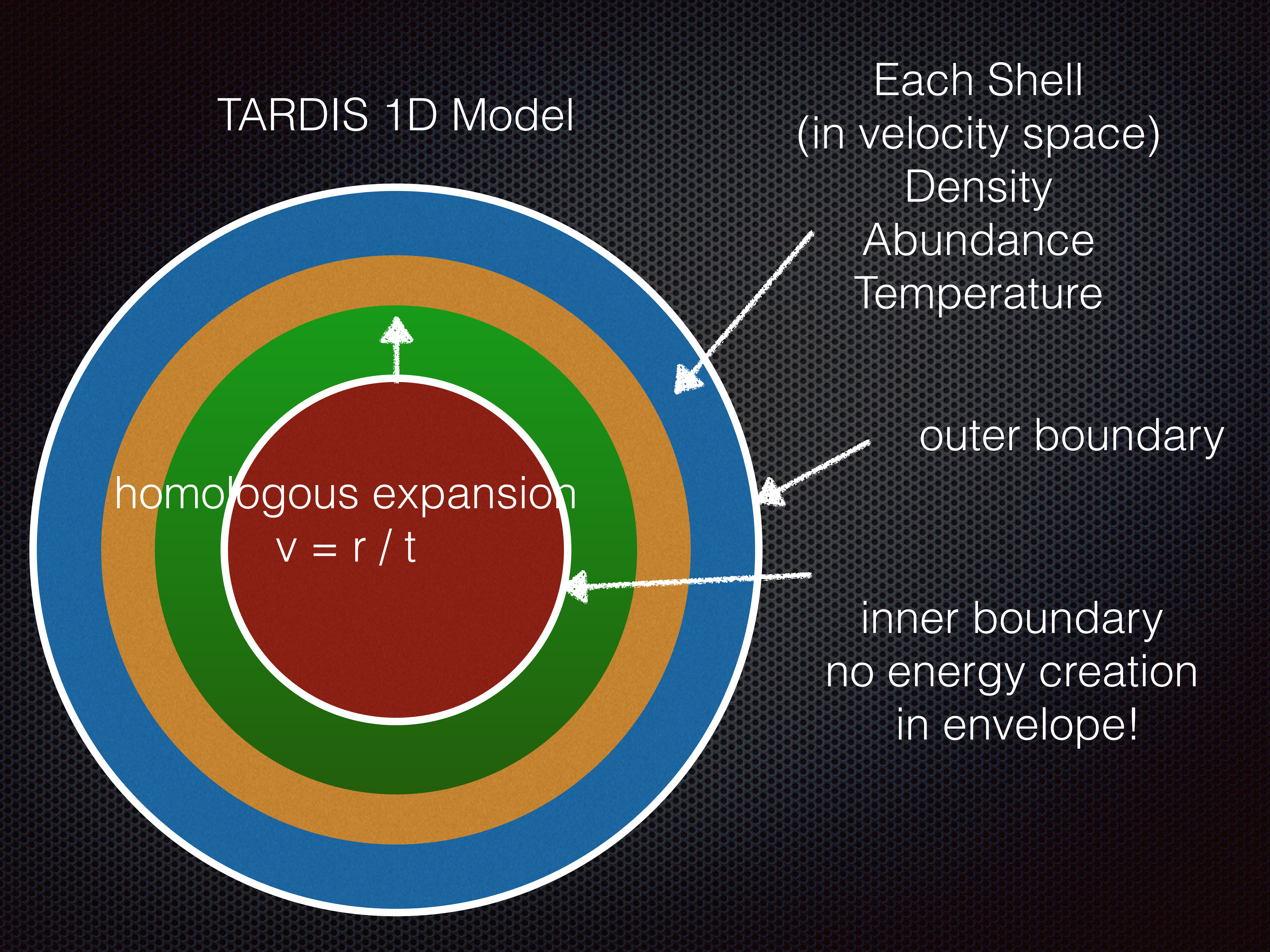
Wolfgang Kerzendorf
on behalf of the TARDIS collaboration

TARDIS 1.0

- one dimensional
- time independent
- photospheric phase of transients (no energy deposition in envelope)
- a couple of different plasma/interaction physics
- NLTE level populations

How does TARDIS work?

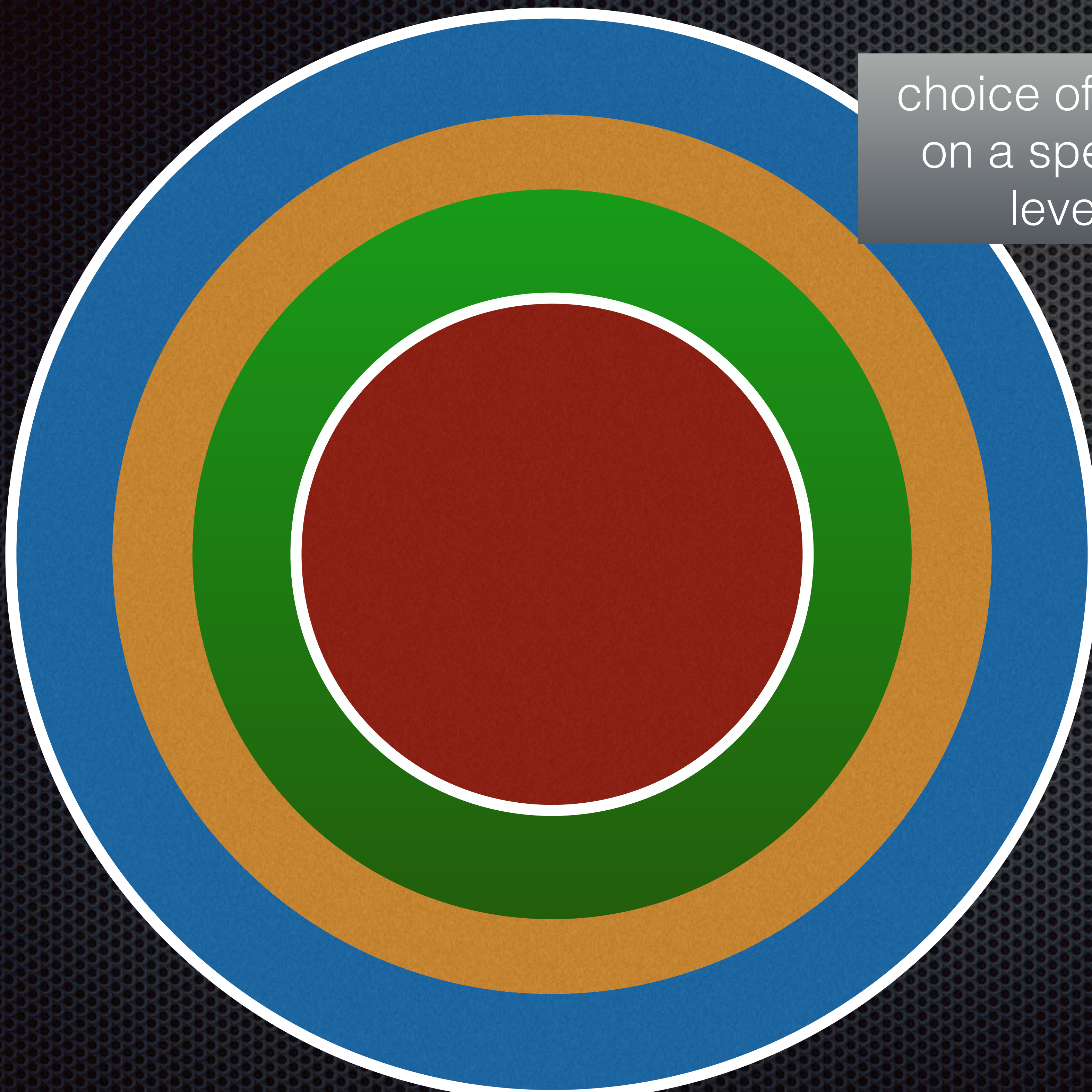
TARDIS Model


TARDIS 1D Model
Each Shell
(in velocity space)
Density
Abundance
Temperature
outer boundary
homologous expansion
v = r / t
inner boundary
no energy creation
in envelope!

TARDIS Simulation

pick a time
pick output luminosity

TARDIS 1D Model



choice of NLTE on a species level

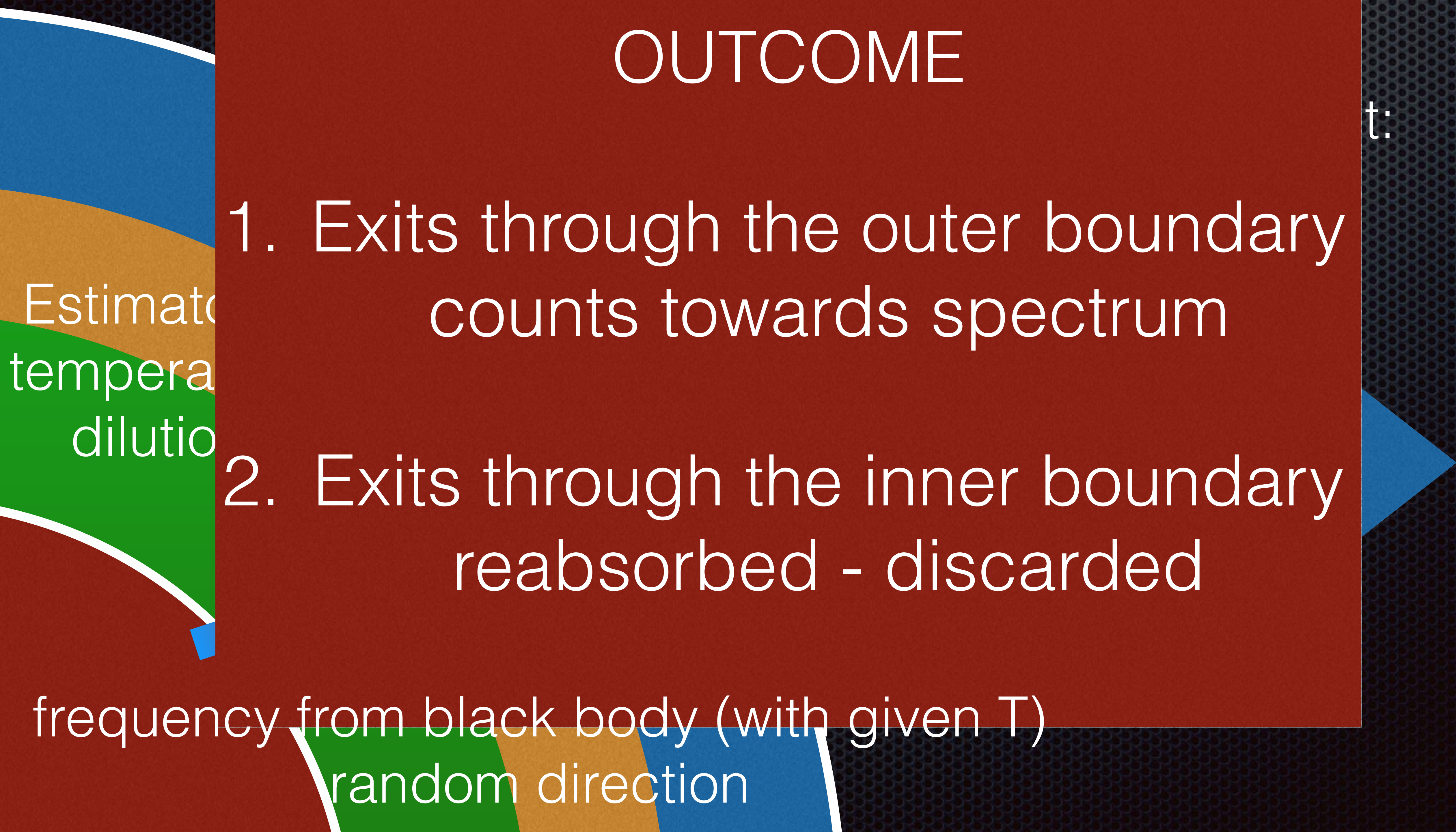
TARDIS solves plasma states

Montecarlo radiative transfer

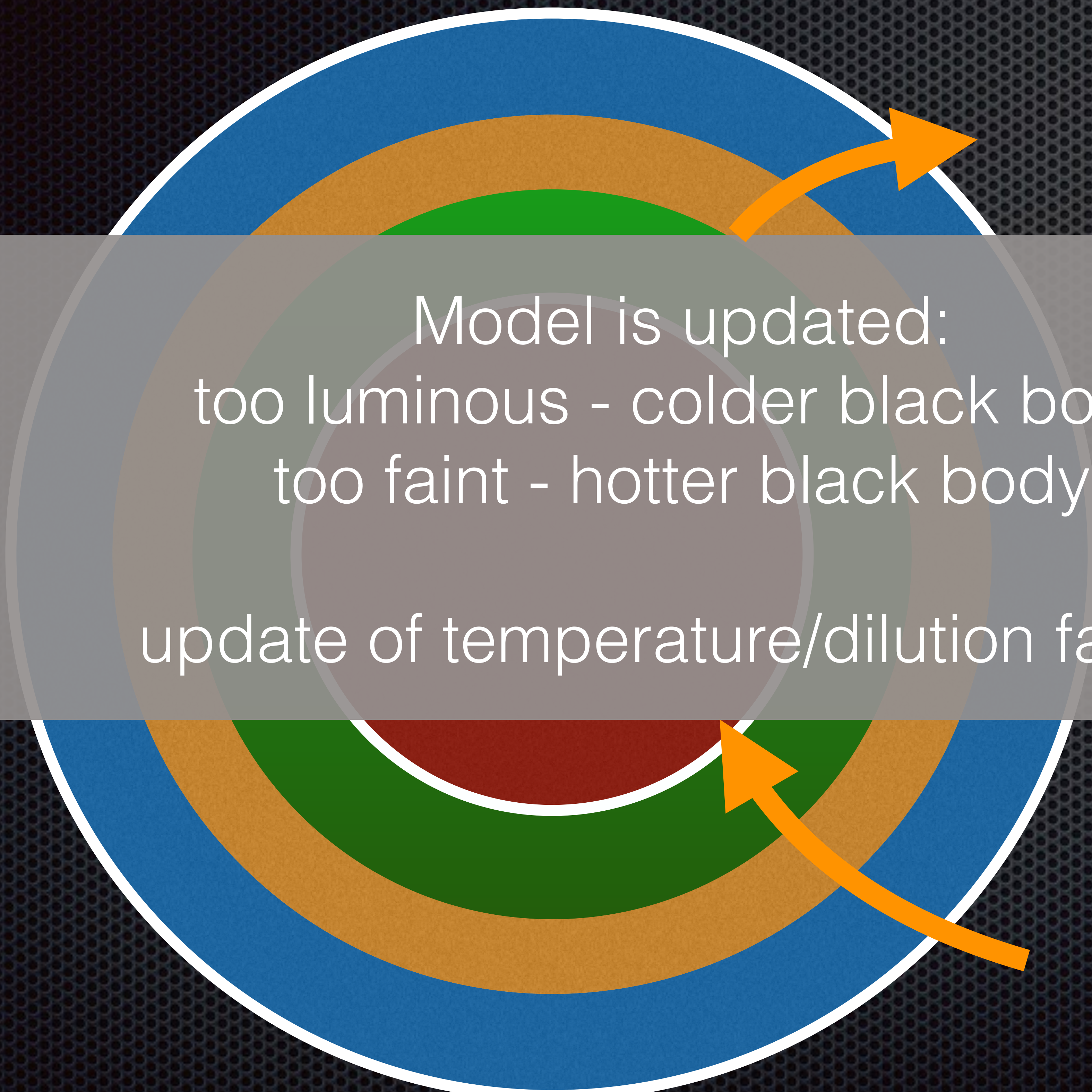
OUTCOME

1. Exits through the outer boundary counts towards spectrum

2. Exits through the inner boundary reabsorbed - discarded



TARDIS 1D Model



Model is updated:
too luminous - colder black body
too faint - hotter black body

update of temperature/dilution factor

pick a time
pick output luminosity

TARDIS solves
plasma states

Montecarlo starts

Spectrum
estimators:
rad. Temperature
radiation diluted

Lessons learned

No fear of dependencies

- numpy>=1.4.0
  scipy>=0.10
  pandas>=0.12
  pytables
  h5py>=2.0
  matplotlib>=1.1
  astropy>=0.4
  PyYAML>=3.0
  numexpr>=2.0.0
  Cython>=0.21

- Anaconda is your friend

YAML input

```yaml
supernova:
    luminosity_requested: 9.34 log_lsun
    time_explosion: 11.12 day
    distance : 1 Mpc


atom_data: ../atomic_data/kurucz_atom_chianti_many.h5


model:
    structure:
        type: file
        filetype: artis
        filename: artis_model.dat
        v_inner_boundary: 11000.0 km/s
        v_outer_boundary: 22000.0 km/s


    abundances:
        type: file
        filetype: artis
        filename: artis_abundances.dat

plasma:
    disable_electron_scattering: no
    type: nebular
    excitation: dilute-lte
    ionization: nebular
    radiative_rates_type: detailed
    line_interaction_type: macroatom
```

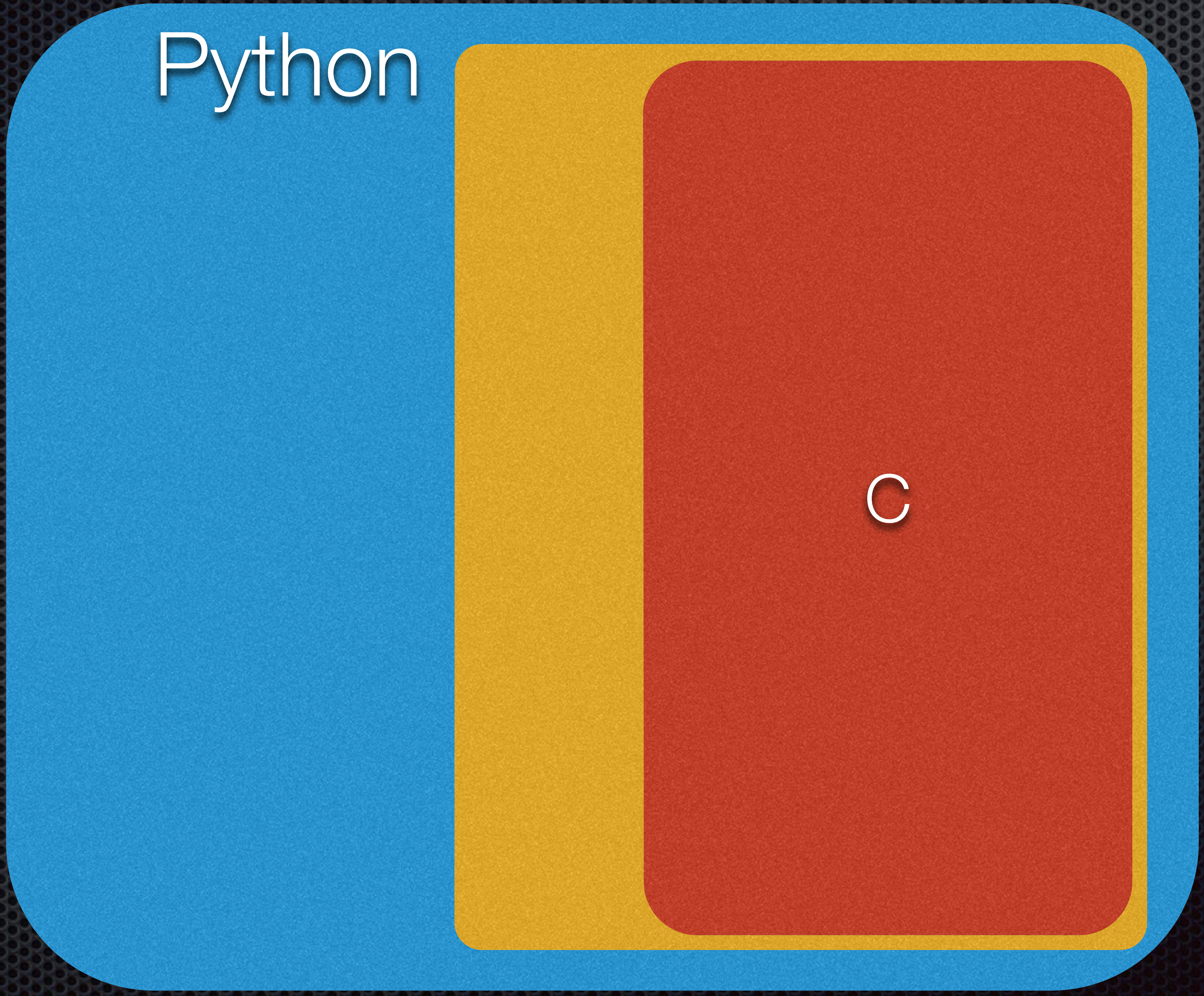
I ♥ astropy.units

- No more confusion
- easy conversion
- … but they are slow
- units go in -> cgs floats

PREMATURE OPTIMIZATION

Come on, do it! Do it now! It feels soooo good.

Structure



FREE!!! Web services

coveralls commented 8 days ago

coverage 55%

Coverage remained the same at 55.04% when pulling `de6b7ce` **on GAT007:GAT007-patch-1** into `e444ffe` **on tardis-sn:master**.

landscape-bot commented 8 days ago

health 85%

Repository health increased by 0.31% when pulling **de6b7ce** **on GAT007:GAT007-patch-1** into **e444ffe** **on tardis-sn:master**.

- 4 new problems were found (including 0 errors and 1 code smell).
- 5 problems were fixed (including 0 errors and 5 code smells).

✔ **All is well** — 2 successful checks

Show all checks

This pull request can be automatically merged.

You can also merge branches on the command line.

Merge pull request

Document!

Radiationfield estimators

During the monte-carlo run we collect two estimators for the radiation field:

$$J_{\text{estimator}} = \sum \epsilon l$$
$$\bar{\nu}_{\text{estimator}} = \sum \epsilon \nu l,$$

where ϵ, ν are comoving energy and comoving frequency of a packet respectively.

To calculate the temperature and dilution factor we first calculate the mean intensity in each cell ($J = \frac{1}{4\pi\,\Delta t\,V} J_{\text{estimator}}$)., [Lucy03].

The weighted mean frequency is used to obtain the radiation temperature. Specifically, the radiation temperature is chosen as the temperature of a black body that has the same weighted mean frequency as has been computed in the simulation. Accordingly,

$$\frac{h\bar{\nu}}{k_B T_R} = \frac{h}{k_B T_R} \frac{\bar{\nu}_{\text{estimator}}}{J_{\text{estimator}}} = 24\zeta(5)\,\frac{15}{\pi^4},$$

package-template

That’s all folks

Here the links:
github.com/tardis-sn/tardis
tardis.readthedocs.org