PhotRedCatalog

An implementation of the Munich Photometric Redshift code

Jan Snigula

PhotRedConfig
PhotRedCatalog

- a wrapper around
  SEDFILTER (PhotRedConfig)
  and
  PHOTOZ (PhotRedCatalog)

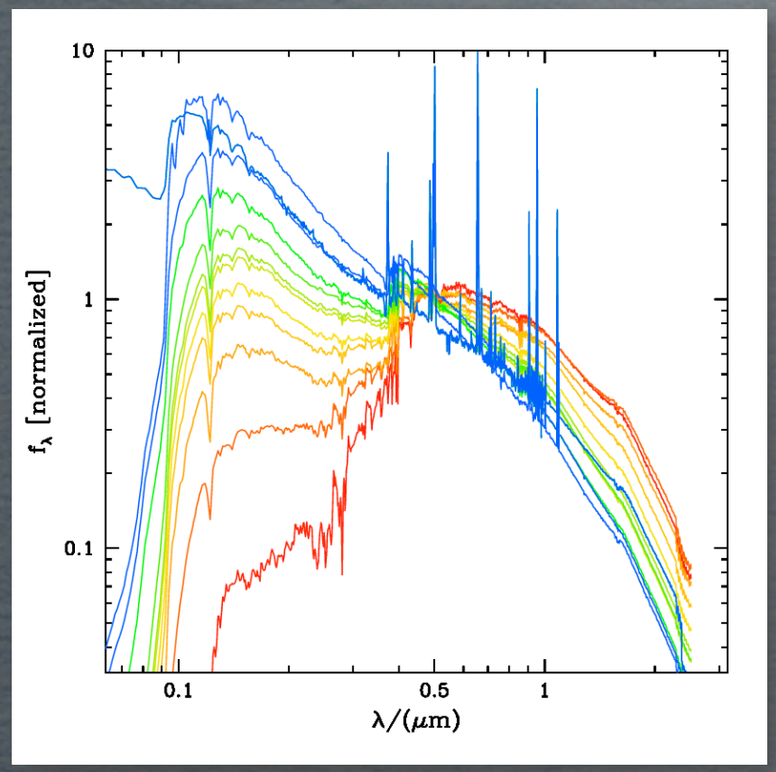
- Reads and creates SourceLists

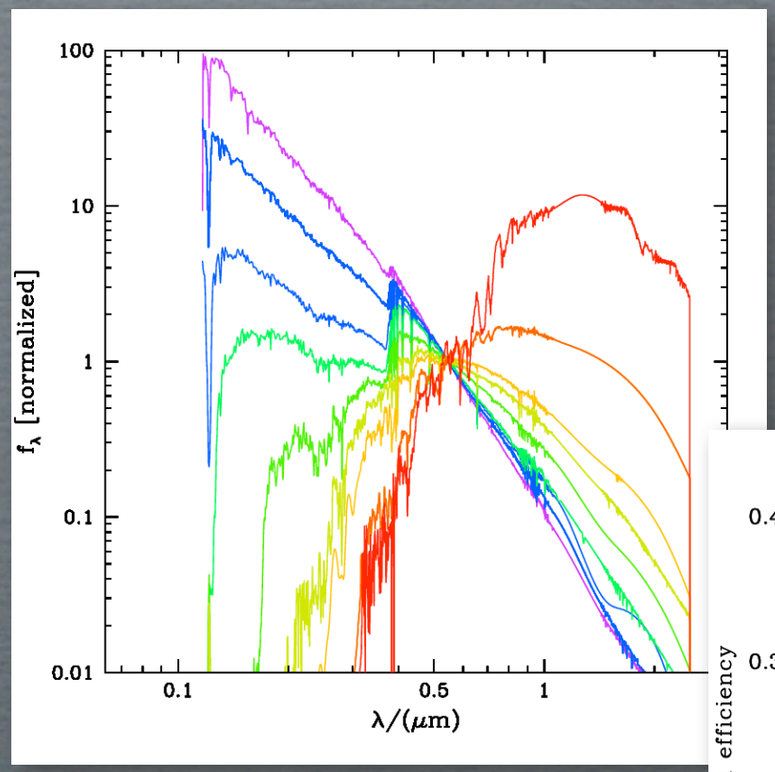
- All relevant information stored in the database

PhotRedConfig

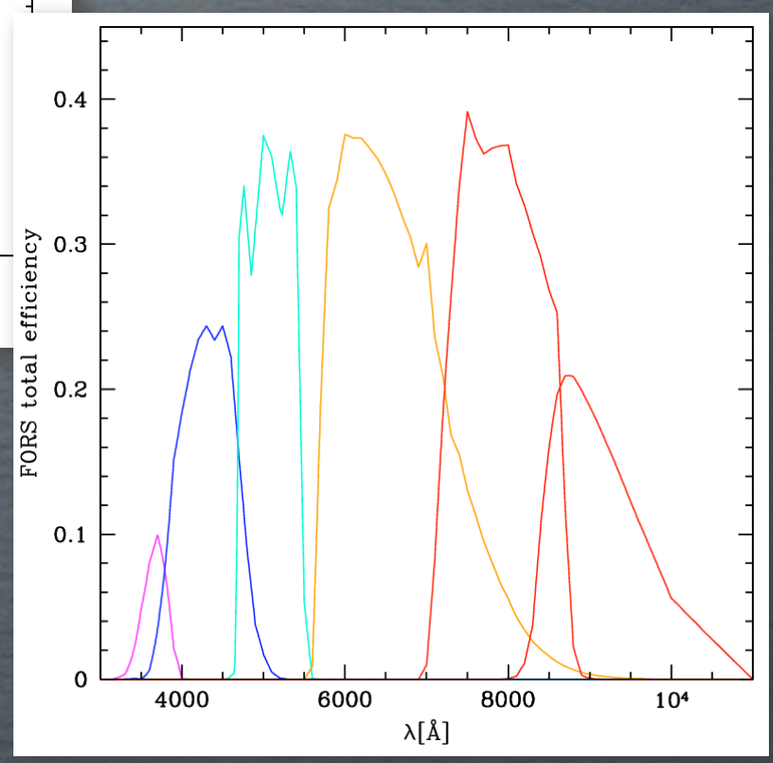
Input files

- Galactic and stellar SEDs (PhotRedSED)



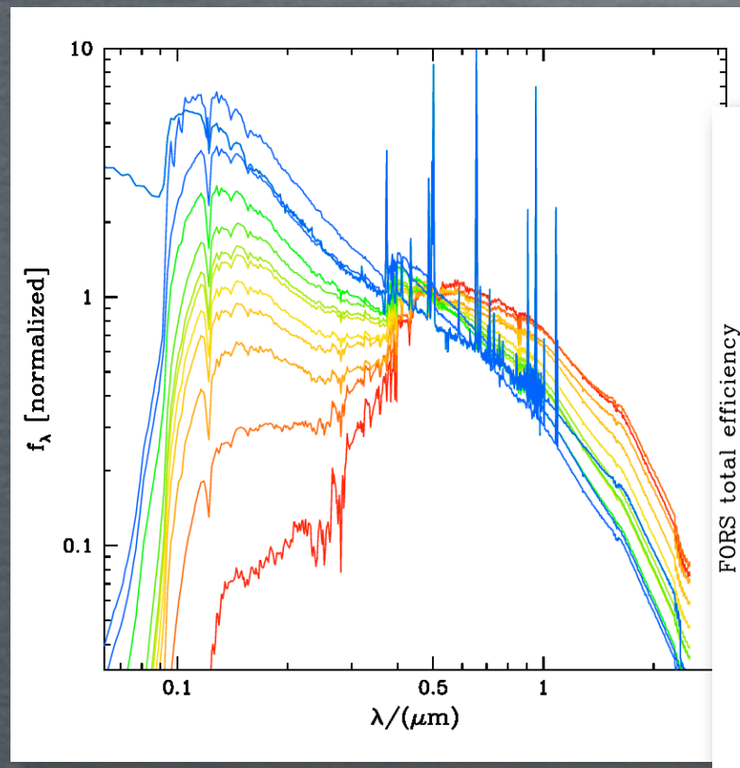


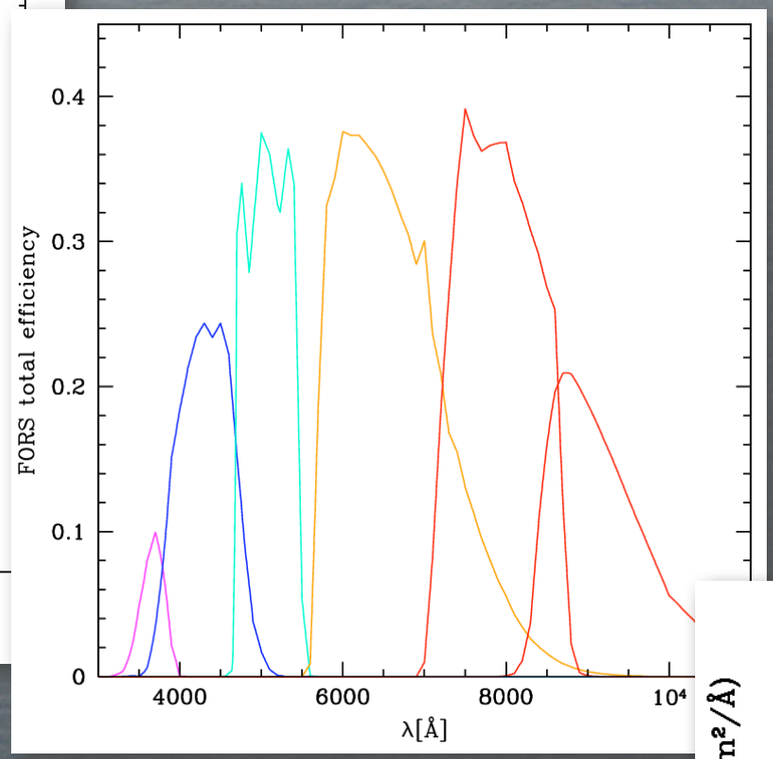
- Filter transmission curves (PhotRedFilter)

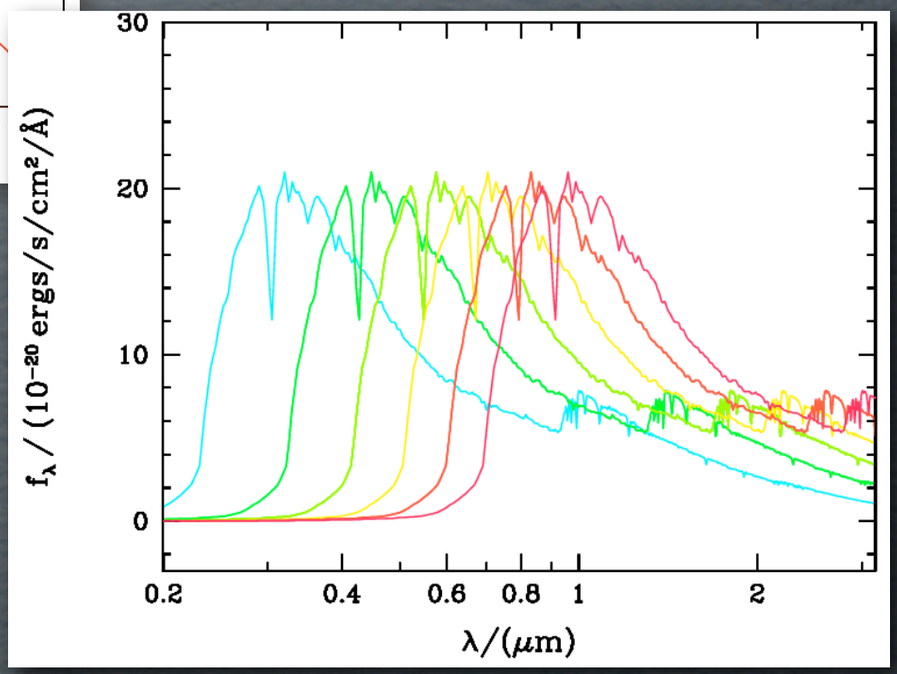


PhotRedConfig

- SEDs are multiplied with the filter curves







PhotRedConfig

PhotRedSED

```
# manucci_soc.sed
#    lambda              flux
     ...
     5.39000e+03      1.02960e+00
     5.39500e+03      1.00520e+00
     5.40000e+03      9.91200e-01
     5.40500e+03      9.76800e-01
     5.41000e+03      9.89200e-01
     5.41500e+03      1.01550e+00
     5.42000e+03      1.04220e+00
     5.42500e+03      1.04000e+00
     5.43000e+03      1.02520e+00
     5.43500e+03      1.02310e+00
     5.44000e+03      1.03310e+00
     ...
```

wavelength and SED flux

PhotRedConfig

PhotRedFilter

```
# 843.filter
#   wavelength transmission
    ...
    5180    0.8363
    5185    0.84207
    5190    0.8474
    5195    0.8524
    5200    0.85657
    5205    0.86059
    5210    0.86385
    5215    0.86622
    5220    0.86808
    5225    0.8693
    5230    0.86966
    ...
```

WAVELENGTH AND FILTER TRANSMISSION

PhotRedConfig

An example from users view

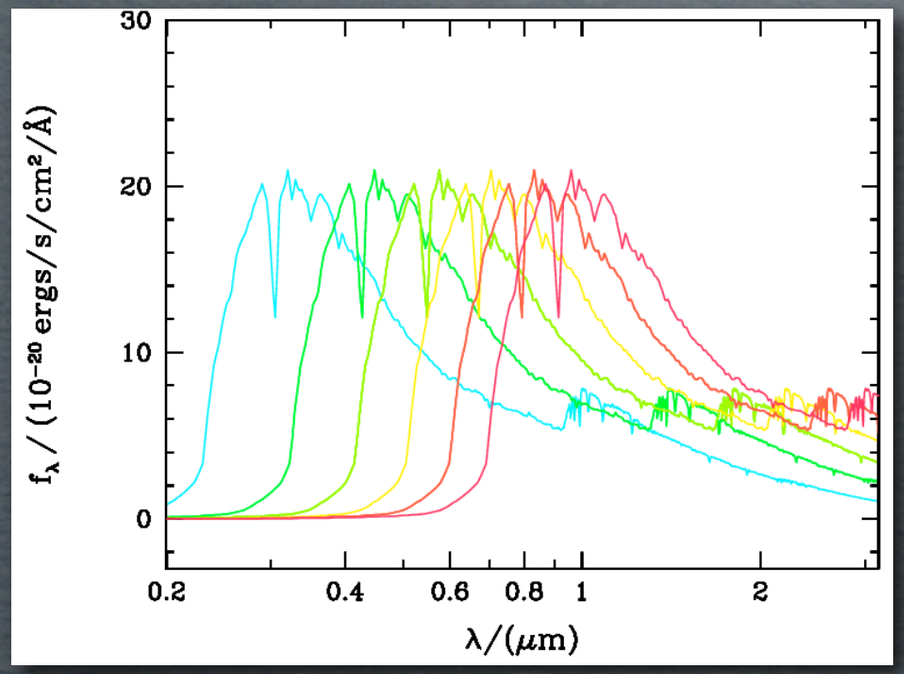
```
awe> from Experimental.PhotRedCatalog import *
# We assume the SEDs and Filters were already ingested
# Create the PhotRedConfig object
# Select the filters
awe> pfu = (PhotRedFilter.filename == 'fors_bess_u_eso_tot.filter' )[0]
...
# Select the SEDs
awe> ps01 = (PhotRedSED.filename == 'manucci_soc.sed' )[0]
awe> ps02 = (PhotRedSED.filename == 'manucci_sac.sed' )[0]
...
# Create the PhotRedConfig
awe> pc = PhotRedConfig()
awe> pc.SEDs=[ps01,ps02,ps03,ps04,ps05,ps06,ps07,ps08,ps09,ps10,
     ps11,ps12,ps13,ps14,ps15,ps16,ps17,ps18,ps19,ps20,
     ps21,ps22,ps23,ps24,ps25,ps26,ps27,ps28,ps29]
awe> pc.filters=[pfu,pfb,pfv,pfr,pfi,pfj,pfk,pfz]
awe> pc.starlib=(PhotRedStarlib.filename=='starlib_pickles_v.lis')[0]
awe> pc.name='fors_demo'
awe> pc.make()
```

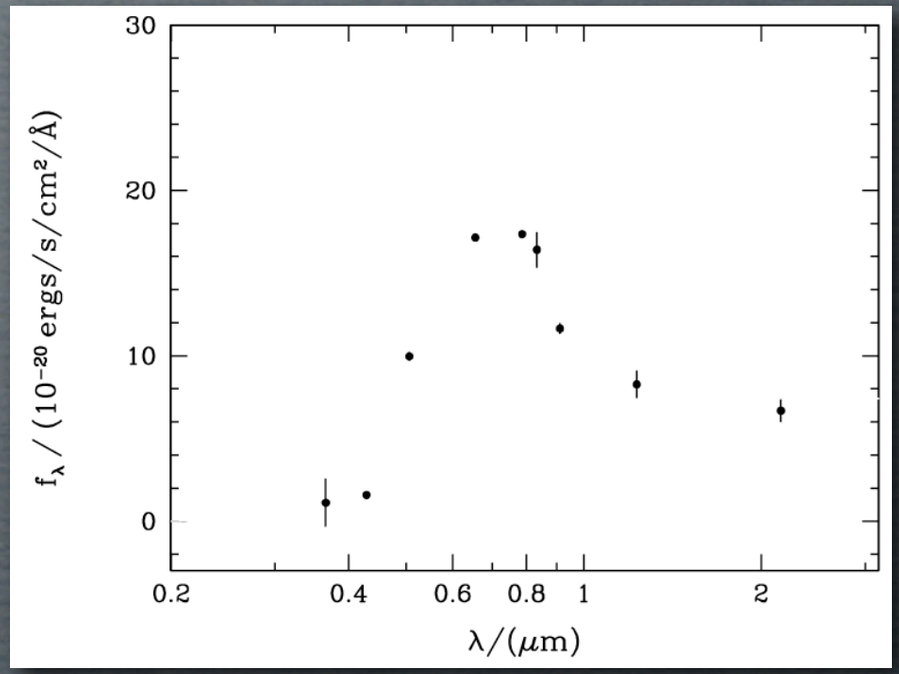
PhotRedConfig

- Can be created using up to 15 filters
- Not all filters have to be used when running PhotRedCatalog
- A set of SEDs will be supplied

PhotRedCatalog (photoz)

- Takes PhotRedConfig and SourceLists
- Matches the SourceLists
- Redshifts the SEDs
- Least-Squares fit of the magnitudes derived from the SEDs against the observed data





PHOTOZ

Best fitting z and SED are determined by minimizing:

$$\chi^2(z, SED) = \frac{1}{N_{filt}} \sum_{i=1}^{N_{filt}} \frac{[f_i - \alpha f_i(z, SED)]^2}{\sigma_i^2 + [0.05\alpha f_i(z, SED)]^2}$$

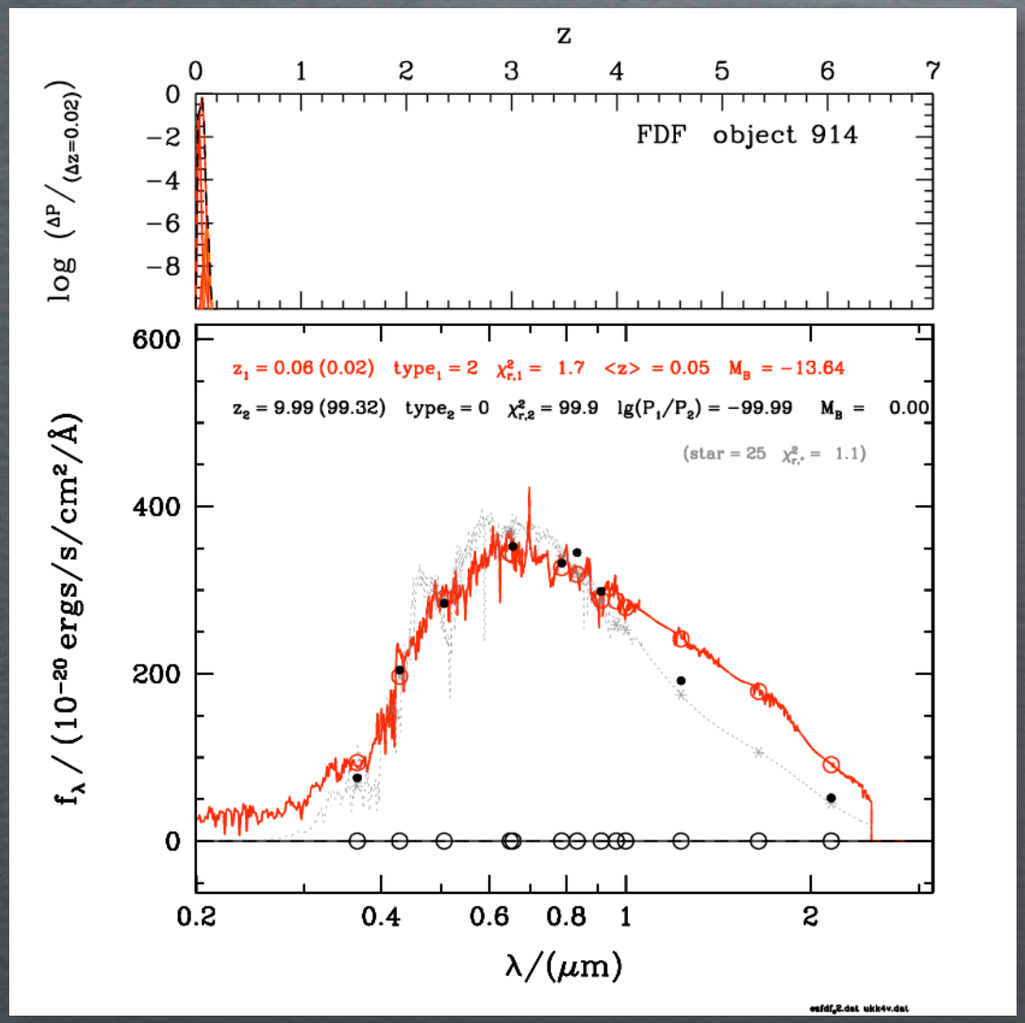
The probability of a source being a given redshift is:

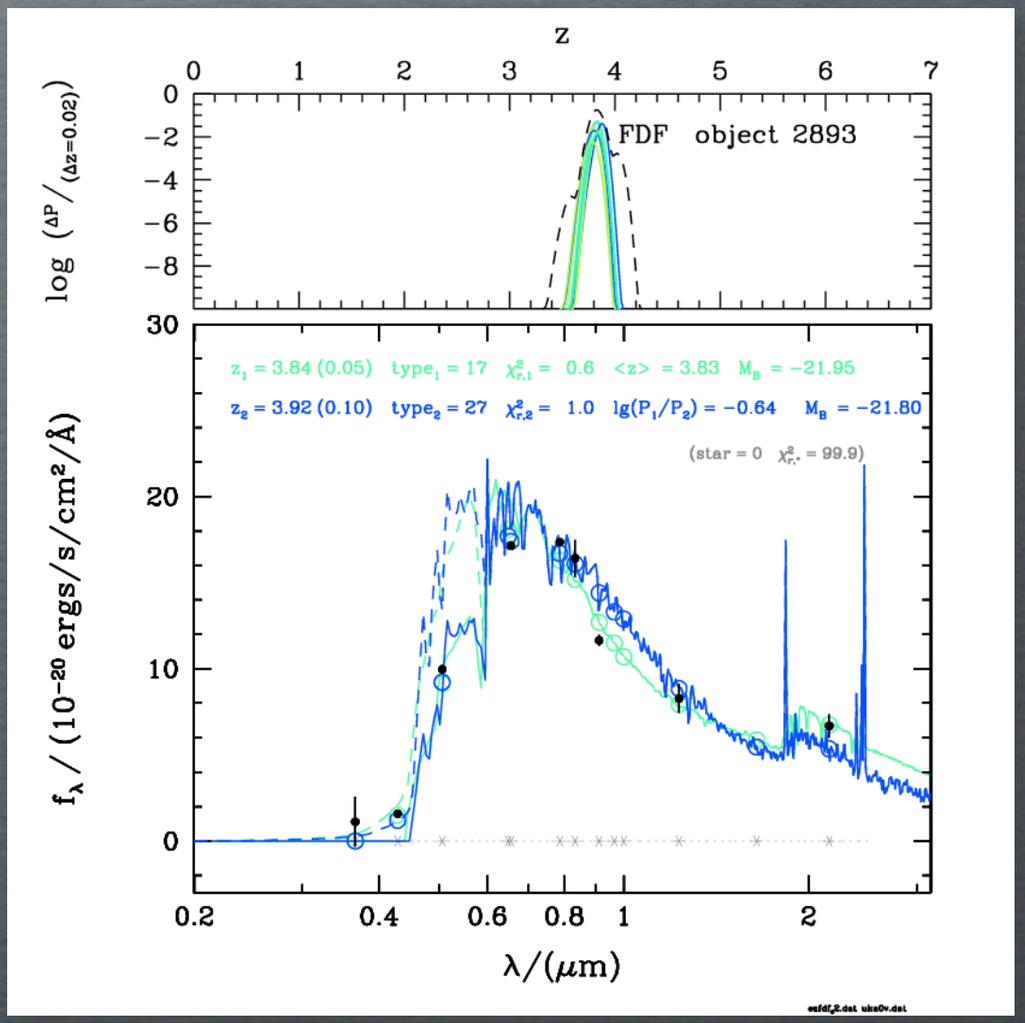
$$P_T = P_\chi \cdot P_L \cdot P_z = e^{-\frac{1}{2}\chi^2} \cdot e^{-k_\beta\left(\frac{M-M_\star}{\sigma}\right)^\beta} \cdot e^{-k_\gamma\left(\frac{z}{z_{lim}}\right)^\gamma}$$

PhotRedCatalog

Output

- z of best fitting galactic SED, and bestfitting stellar SED





PhotRedCatalog

datpz1 SourceList

```
RA,DEC,Xpos,Ypos,A,B,POSANG  - Standard photred parameters
obj                          - Object ID
best_z,err_z                 - redshift and error of best fit
mod                          - Model id
rchi2                        - reduced chi^2
z2                           - second best fitting z
lg_Pz2z1                     - Ratio of the probabilities of z2/z
<z>                          - weighted mean of the z distribution

fU,fB,fV,fR,fI,fJ,fH,fK,fF1,fF2,fF3,fF4,fF5,fF6,fF7
M_B,M_R,M_I,M_K              - fluxes of object and derived
absolute magnitudes

DMOD                         - derived distance modulus

f_dat/f_mod                  - Ratio between observed/model flux

best_model                   - Best fitting model
```

PhotRedCatalog

datstar SourceList

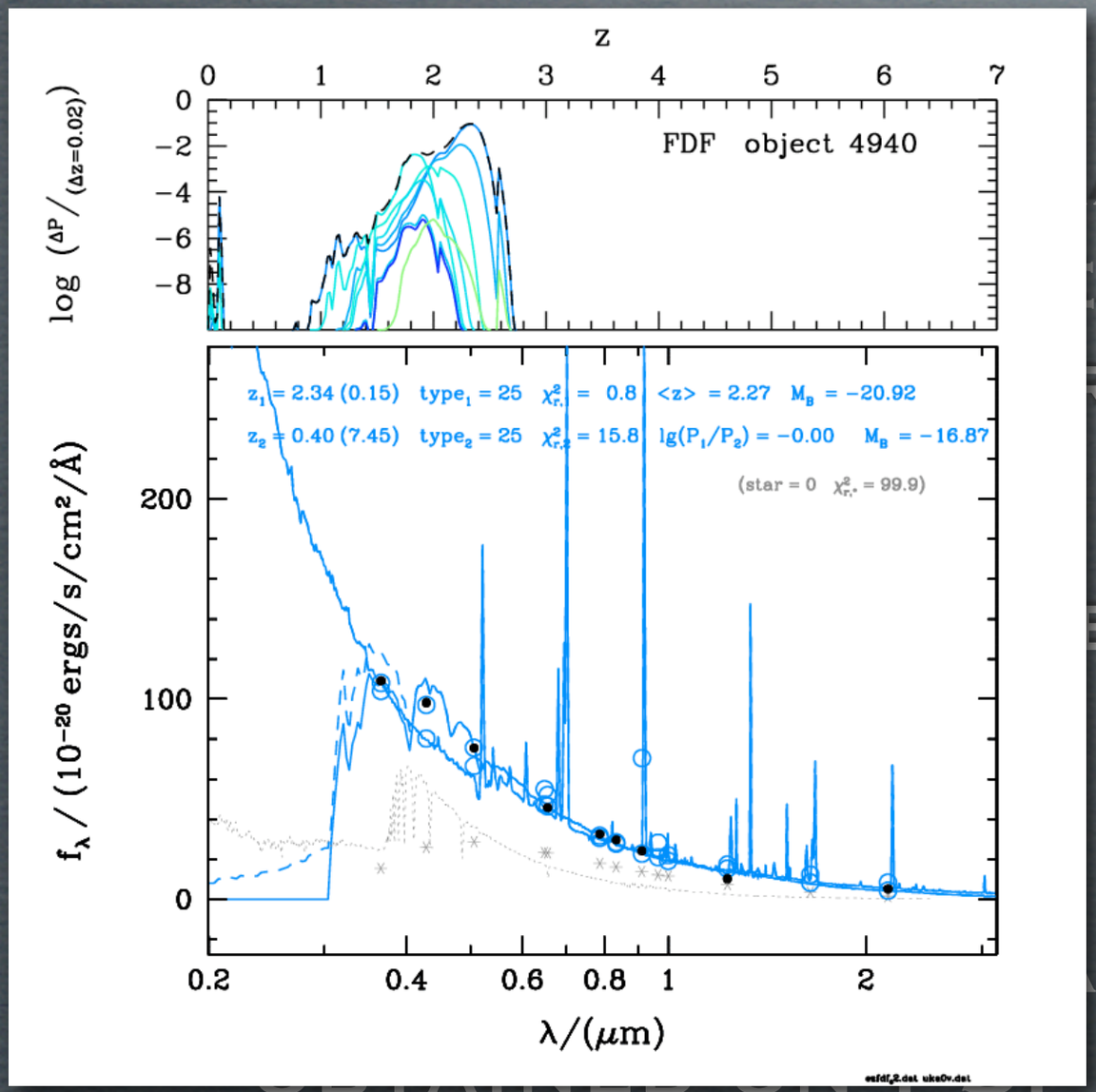
```
RA,DEC,Xpos,Ypos,A,B,POSANG  - Standard photred parameters
obj                          - Object ID
best_star                    - Model id of bestfitting star
rchi2                        - reduced chi²

fU,fB,fV,fR,fI,fJ,fH,fK,
fF1,fF2,fF3,fF4,fF5,fF6,fF7  - object fluxes

f_dat/f_mod                  - Ratio between observed/model flux
```

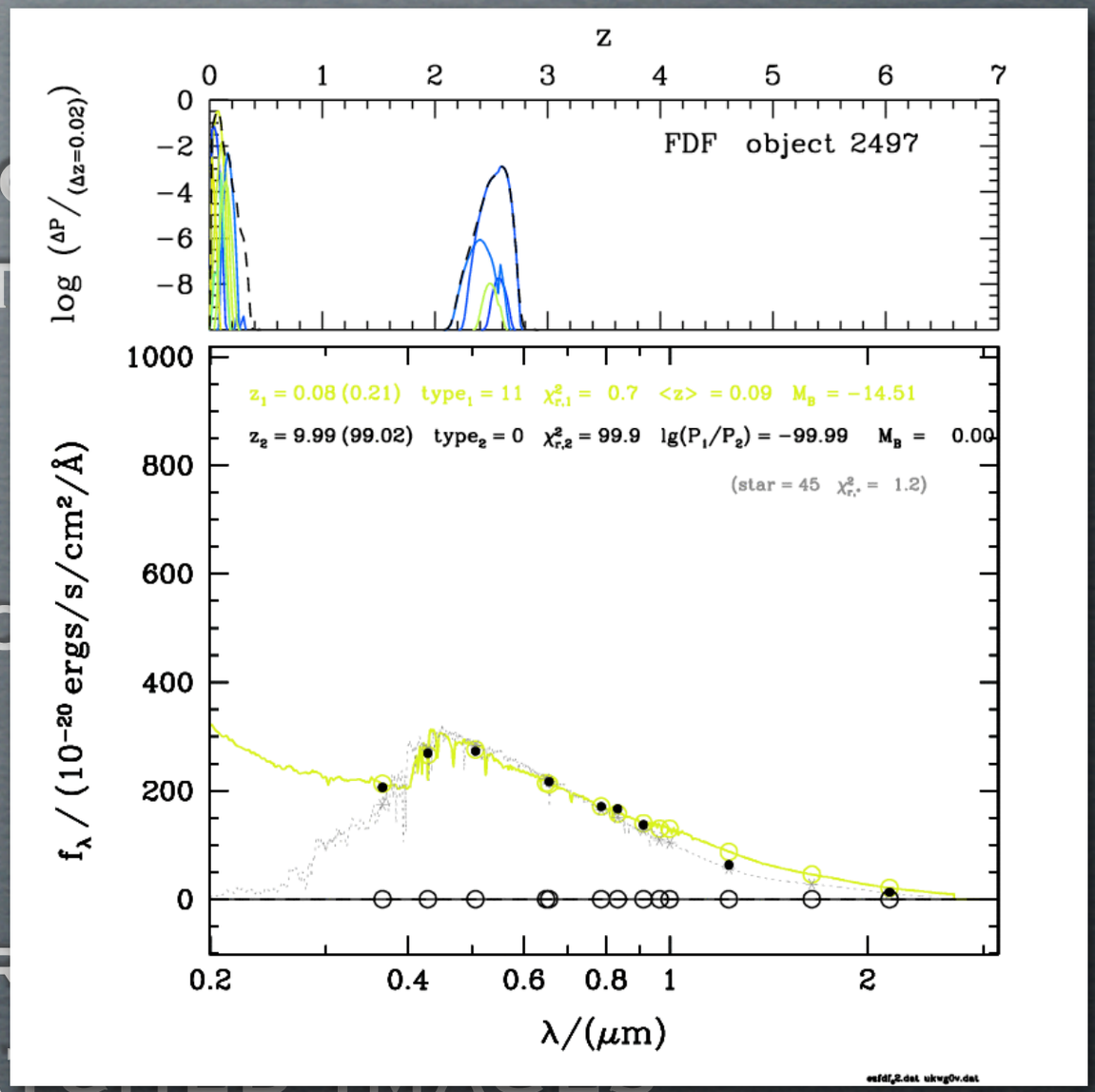
PhotRedCatalog

An example from users view

```
awe> from Experimental.PhotRedCatalog import *
# Select the config
awe> pc=(PhotRedConfig.name=='fors_demo')[0]
# Select the SourceLists
awe> sU=(SourceList.SLID==21)[0]
...
# Create the PhotRedCatalog
awe> pr = PhotRedCatalog()
pr.config=pc
pr.master=sI
pr.sourcelists=[sU,sB,sg,sR,sI,sJ,sK,sz]
pr.extinc=[-0.35,-0.15,-0.11,0.06,0.19,-0.09,-0.120,1.03]
pr.model_error=[0.07,0.03,0.03,0.03,0.05,0.15,0.3,0.1]
pr.mag='MAG_APER'
pr.flux='FLUX_APER'
pr.fluxerr='FLUXERR_APER'
pr.make()
    ...
# Plot photometric redshift for object with AID 32
pr.plot(32)
```

Caveats


z
FDF object 4940
log (ΔP/(Δz=0.02))
z1 = 2.34 (0.15) type1 = 25 χ²r,1 = 0.8 <z> = 2.27 MB = −20.92
z2 = 0.40 (7.45) type2 = 25 χ²r,2 = 15.8 lg(P1/P2) = −0.00 MB = −16.87
(star = 0 χ²r,* = 99.9)
fλ / (10⁻²⁰ ergs/s/cm²/Å)
λ/(μm)


z
FDF object 2497
log (ΔP/(Δz=0.02))
z1 = 0.08 (0.21) type1 = 11 χ²r,1 = 0.7 <z> = 0.09 MB = −14.51
z2 = 9.99 (99.02) type2 = 0 χ²r,2 = 99.9 lg(P1/P2) = −99.99 MB = 0.00
(star = 45 χ²r,* = 1.2)
fλ / (10⁻²⁰ ergs/s/cm²/Å)
λ/(μm)

DEMO