```
/Users/dantopa/Dropbox/fortra...reduction/lima/mod HELIOS.f90
                                                             Page 1 of 4
Saved: 8/10/13 8:32:37 AM
                                             Printed For: Daniel M. Topa
    1 module HELIOS_data
    2
     ! use constants and parameters
    3
        use linear_regression_results
    4
        use polynomial fit
    5
        use HELIOS_files
    6
        implicit none
    7
    8
        type, public
    9
                                                 :: HELIOS
   10
          ! amplitudes describing the Helios surface
   11
         type ( surface fit )
                                                 :: poly fit
   12
                                                :: HELIOS_tables
         type ( HELIOS_output )
   13
   14
          contains
   15
   16
            ! functions
   17
   18
            ! subroutines
   19
            procedure, public
   20
                                                 :: put
                                                                         =>
   21
   22
        end type
                                                   HELIOS
   23
        private
                                                 :: put_sub
   24
   25
        contains
   26
   27
          28
   29
          subroutine put_sub ( self, amplitudes, errors, sf_quality, pts_tem
   30
                              array scaled, array scaled characteristics, a
   31
   32
            use constants and parameters
   33
           use quality_parameters
   34
            use polynomial fit
   35
            implicit none
   36
   37
```

```
/Users/dantopa/Dropbox/fortra...reduction/lima/mod HELIOS.f90
                                                                 Page 2 of 4
Saved: 8/10/13 8:32:37 AM
                                                Printed For: Daniel M. Topa
            class ( HELIOS ), target
                                                               :: self
   38
            real ( wp ), pointer
                                                               :: surface ampli
   39
                                                               :: surface error
            real ( wp ), pointer
   40
            real ( wp ), pointer
                                                               :: ptr_pts_temp
   41
                                                               :: ptr pts densi
            real ( wp ), pointer
   42
   43
            real ( wp ), optional
                                                               :: amplitudes
   44
            real ( wp ), optional
                                                               :: errors
   45
            real ( wp ), optional
                                                               :: array_scaled
   46
            real ( wp ), optional
                                                               :: array normali
   47
            real ( wp ), optional
                                                               :: pts_temp
   48
            real ( wp ), optional
                                                               :: pts density
   49
   50
            type ( surface fit quality ),
                                                  optional
                                                               :: sf_quality
   51
                                                               :: pts_temp_lm,
            type ( lr results ),
                                                  optional
   52
            type ( array characteristics ),
                                                               :: array scaled
                                                  optional
   53
   54
            ! HELIOS output
   55
            ! maps
   56
            if ( present ( pts_temp ) ) then
   57
              ptr pts temp => self % HELIOS tables % pts temp
   58
   59
              ptr pts temp = pts temp
              ptr_pts_temp => null ( )
   60
            end if
   61
   62
            if ( present ( pts density ) ) then
   63
              ptr pts density => self % HELIOS tables % pts density
   64
              ptr pts density = pts density
   65
              ptr pts density => null ( )
   66
            end if
   67
   68
            if ( present ( pts temp lr ) ) then
   69
              self % HELIOS tables % pts temp lr = pts temp lr
   70
            end if
   71
   72
            if ( present ( pts density lr ) ) then
   73
              self % HELIOS_tables % pts_density_lr = pts_density_lr
   74
```

```
/Users/dantopa/Dropbox/fortra...reduction/lima/mod HELIOS.f90
                                                                 Page 3 of 4
Saved: 8/10/13 8:32:37 AM
                                                Printed For: Daniel M. Topa
            end if
   75
   76
            ! array tables
   77
            if ( present ( array_scaled ) ) then
   78
              self % HELIOS tables % array scaled = array scaled
   79
            end if
   80
   81
            if ( present ( array scaled characteristics ) ) then
   82
              self % HELIOS_tables % array_scaled_characteristics = array_sc
   83
            end if
   84
   85
            if ( present ( array normalized ) ) then
   86
              self % HELIOS_tables % array_normalized = array_normalized
   87
            end if
   88
   89
            if ( present ( array_normalized_characteristics ) ) then
   90
              self % HELIOS tables % array normalized characteristics = amra
   91
            end if
   92
   93
            ! surface fit
   94
            if ( present ( amplitudes ) ) then
   95
              surface_amplitudes => self % poly_fit % amplitudes
   96
              surface amplitudes = amplitudes
   97
              surface amplitudes => null ( )
   98
            end if
   99
   100
            if ( present ( errors ) ) then
   101
              surface_errors => self % poly_fit % errors
   102
              surface errors = errors
  103
              surface errors => null ( )
  104
            end if
   105
  106
            if ( present ( sf_quality ) ) then
  107
              self % poly_fit % sf_quality = sf_quality
   108
            end if
  109
  110
          end subroutine put_sub
   111
```

/Users/dantopa/Dropbox/fortrareduction/lima/mod HELIOS.f90 Page 4 of 4 Saved: 8/10/13 8:32:37 AM Printed For: Daniel M. Topa									
112 113	end module	HELIOS_data							