

## *NERS* User Guide

### Introduction

NERS library interface provides 8 public routines. Routine **ners\_init** is called before any other *NERS* functions. It specifies *NERS* control file and the time range of intended EOP use. Time in *NERS* library is the interval elapsed since 2000.01.01\_00:00:00.0 in TAI time scale. Units of time are seconds. Instead of time, events are often get tags of UTC function. In order to convert UTC tag to TAI time, *NERS* provides function **ners\_get\_utcmtai** that returns function UTC minus TAI on the specified timetag UTC. Routine **ners\_init** initializes internal data structure of *NERS* object that is defined in **ners.i** or **ners.h**include block. This object is then passed to any other *NERS* function. *NERS* provides either the vector Earth orientation parameters on the specified moment of time or the table of the Earth orientation series for the specified range of time and the specified time step. At the end, function **ners\_quit** releases memory acquired by **ners\_init**.

The Earth orientation can be described either as a 3×3 rotation matrix that transforms a Cartesian vector from the rotating terrestrial coordinate system to the inertial non-rotating celestial coordinate system or as parameters on which this matrix depends. For practical needs the rotation matrix and its time derivatives are sufficient to perform astronomical reduction for Earth's rotation. However, *NERS* also provides the parameters that describes the Earth's rotation on which the rotation matrix depends as well. These parameters are empirical corrections to a deterministic model. Since the Earth's rotation depends on motion of the hydrosphere and atmosphere, it cannot be described with a deterministic model with the accuracy comparable with accuracy of observations and should be continuously monitored using space geodesy observations. For historical reasons several alternative Earth orientation parameters were used. For instance the rate of change of the angular variable along the axis 3, i.e. the axial motion can be described as Euler angle 3, UT1 rate, or the length of day. *NERS* provides many alternative Earth orientation parameters. These parameters are not independent.

### Use cases

- Transform a Cartesian vector of station coordinates in the terrestrial coordinate system to the inertial celestial coordinate system at the specified moment of time. All you need is to call three NERS routines: **ners\_init**, **ners\_get\_eop**, **ners\_quit**. First, you run **ners\_init** and specify the time range. If you need to compute the rotation matrix only for one epoch, the range may be just  $\pm 10$  seconds. If you are going to compute the transformation matrix for a number of epochs, just specify TAI time of the earliest and the latest epoch. The argument time is 64-bit float number of seconds elapsed since 2000.01.01\_00:00:00.0 TAI. Second, you run **ners\_get\_eop** and specify the time. The type of the EOP is "mat". A product of this matrix with the Cartesian vector of station coordinates in the terrestrial coordinate system will transform it to the inertial celestial coordinate system. If you need the transformation matrix for a number of epochs, you call **ners\_get\_eop** in a cycle. Finally, you call **ners\_quit** to release dynamic memory allocated by *NERS*.

See [example/ners\\_fortran\\_example\\_01.f](#) and [example/ners\\_c\\_example\\_01.c](#).

- Get TAI time for a given UTC timetag. It is rather common to have value of the non-differentiable function of time UTC(t) for events instead of time TAI. In order to process such data you need to convert UTC timetag to TAI moment of time. All you need is to call three NERS routines: **ners\_init**, **ners\_get\_utcmtai**, **ners\_quit**. First, you run **ners\_init** and specify the time range. If you are going to compute convert UTC to TAI for a number of epochs, just specify TAI time of the earliest and the latest epoch. Second, you run **ners\_get\_utcmtai** and specify the UTC timetag elapsed since epoch 2000.01.01\_00:00:00.0 UTC. The routine returns function UTC minus TAI. Subtracting this function from UTC you get TAI time:  $TAI = UTC - UTC\_M\_TAI$ . If you need transform UTC time tag to TAI for a number of epochs, you call **ners\_get\_utcmtai** in a cycle. Finally, you call **ners\_quit** to release dynamic memory allocated by *NERS*.

# NERS User Guide

See [example/ners\\_fortran\\_example\\_02.f](#) and [example/ners\\_c\\_example\\_02.c](#).

- Get the last time epoch of the short-term EOP forecast. You need call three NERS routines: **ners\_init**, **ners\_inq**, **ners\_quit**. First, you run **ners\_init** and specify the time range. For this case the range may be just  $\pm 10$  seconds of the current moment. Second, you run **ners\_inq** with request parameter "range". The routine returns three parameters: start time for EOP assimilation, stop time of the short-term EOP forecast, and stop time of the long-term prediction. Time is in seconds elapsed since 2000.01.01\_00:00:00.0 TAI. Finally, you call **ners\_quit** to release dynamic memory allocated by *NERS*.

See [example/ners\\_fortran\\_example\\_04.f](#) and [example/ners\\_c\\_example\\_04.c](#).

## Description of NERS public functions

- **ners\_init** — initializes internal NERS data structure, parses the configuration file and reads the leap second file. It also specifies the interval of time for the Earth orientation parameters that will be computed at the next call to NERS. This interval should not exceed 10 days.

FORTTRAN: NERS\_INIT ( CONFIG\_FILE, NERS, TIME\_TAI\_START, TIME\_TAI\_STOP, IUER )

C:    cners\_init ( char \* config\_file, struct ners\_struct \* ners,  
                  double time\_tai\_beg, double time\_tai\_end, int \* iuer )

Input parameters:

config\_file ( CHARACTER\*(\*) ) -- name of the NERS configuration file  
                  NERS\_\_CONFIG defined in ners\_local.i  
                  specifies the default configuration file  
ners ( NERS\_\_TYPE ) -- NERS internal data structure  
time\_tai\_start ( REAL\*8 ) -- Start time of the time range  
                  Units: seconds since 2000.01.01\_00:00:00.0 TAI  
Value -1.0D0 means the current moment.  
time\_tai\_stop ( REAL\*8 ) -- Stop time of the time range.  
                  Units: seconds since 2000.01.01\_00:00:00.0 TAI  
Value -1.0D0 means the current moment.

Input/Output parameter:

iuier ( INTEGER\*4 ) -- Error parameter.  
                  On input: -1 -- to print the error message if  
                            an error is detected;  
                            otherwise, not to print the message:  
                  On output: 0 -- normal completion  
                            >0 -- error code during NERS  
                            initialization

- **ners\_get\_eop** — returns array of the Earth Orientation Parameters on the specified moment of time. The type of Earth orientation parameter is defined in variable cpar. If it is called the first time, it communicates with the server, downloads the NERS message and prepares the interpolating polynomials for the specified time range, computes intermediate angles and expand them into B-spline basis over the specified time period. Then it computes the requested Earth orientation parameters. When it is called the next time, it reuses the B-spline coefficients computed previously, unless more than 20 minutes elapsed from the computation of the coefficients.

## NERS User Guide

FORTTRAN: NERS\_GET\_EOP ( NERS, TIME\_TAI, CPAR, M\_PAR, L\_PAR, PARS, IUER )

C: cners\_get\_eop ( struct ners\_struct \* ners, double time\_tai\_obs, char \* cpar,  
int mpar, int \* lpar, double eops[], int \* iuer )

Input parameters:

ners ( NERS\_\_TYPE ) -- NERS internal data structure

time\_tai ( REAL\*8 ) -- Time. Units: seconds since 2000.01.01\_00:00:00.0 TAI

cpar ( CHARACTER\*(\*) ) -- Earth orientation parameter name:

mat -- 3x3 matrix of the transformation from the terrestrial  
coordinate system to the celestial coordinate system

matr -- First time derivative of the 3x3 matrix of the  
transformation from the terrestrial coordinate system  
to the celestial coordinate system

matrr -- Second time derivative of the 3x3 matrix of the  
transformation from the terrestrial coordinate system  
to the celestial coordinate system

matall -- 3x3x3 array of the matrix of the transformation from  
the terrestrial coordinate system to the celestial  
coordinate system, its first and second time derivative.

[1:3,1:3,1] -- transformation matrix from the terrestrial  
coordinate system to the celestial  
coordinate system,

[1:3,1:3,2] -- first time derivative of the transformation  
matrix above.

[1:3,1:3,3] -- second time derivative of the transformation  
matrix above.

utcm tai -- Value of function UTC minus TAI. Units: s.

e1 -- Euler angle around axis one. Units: rad.

e2 -- Euler angle around axis two. Units: rad.

e3 -- Euler angle around axis three. Units: rad.

e1r -- First time derivative of Euler angle around axis one. Units: rad/s.

e2r -- First time derivative of Euler angle around axis two. Units: rad/s.

e3r -- First time derivative of Euler angle around axis three. Units: rad/s.

e1rr -- Second time derivative of Euler angle around axis one. Units: rad/s.

e2rr -- Second time derivative of Euler angle around axis two. Units: rad/s.

e3rr -- Second time derivative of Euler angle around axis three. Units: rad/s.

ut1mtai -- Angle UT1 minus TAI. Units: s.

ut1rat -- First time derivative of angle UT1 minus TAI. Units s/s.

ut1rr -- Second time derivative of angle UT1 minus TAI. Units s/s^2.

lod -- Length of day. Units: s.

lodr -- Rate of change of the length of day. Units: s/day.

xpol -- X pole coordinate. Unit: arcsec.

ypol -- Y pole coordinate. Unit: arcsec.

xpolr -- First time derivative of X pole coordinate. Unit: arcsec/day.

## NERS User Guide

ypolr -- First time derivative of Y pole coordinate. Unit: arcsec/day.  
xpolrr -- Second time derivative of X pole coordinate. Unit: arcsec/day\*\*2.  
ypolrr -- Second time derivative of Y pole coordinate. Unit: arcsec/day\*\*2.  
eop3 -- Array of three EOP parameters:  
    1: X pole coordinate. Unit: arcsec.  
    2: Y pole coordinate. Unit: arcsec.  
    3: UT1 minus TAI. Unit: s.  
eop3r -- Array of six EOP parameters:  
    1: X pole coordinate. Unit: arcsec.  
    2: Y pole coordinate. Unit: arcsec.  
    3: UT1 minus TAI. Unit: s.  
    4: First time derivative of X pole coordinate. Unit: arcsec/day.  
    5: Second time derivative of Y pole coordinate. Unit: arcsec/day.  
    6: First time derivative of UT1 minus TAI coordinate. Unit: s/day.  
  
dpsi -- Nutation angle in longitude. Units: rad  
deps -- Nutation angle in obliquity. Units: rad  
dpsir -- First time derivative of nutation angle in longitude. Units: rad/s.  
depsr -- First time derivative of nutation angle in obliquity. Units: rad/s.  
nut -- Array of two EOP parameters:  
    1: Nutation angle in longitude. Units: rad  
    2: Nutation angle in obliquity. Units: rad  
nutr -- Array of four EOP parameters:  
    1: Nutation angle in longitude. Units: rad  
    2: Nutation angle in obliquity. Units: rad  
    3: First time derivative of nutation angle in longitude. Units: rad/s.  
    4: First time derivative of nutation angle in obliquity. Units: rad/s.  
eops -- Array of eight EOP parameters:  
    1: X pole coordinate. Unit: arcsec.  
    2: Y pole coordinate. Unit: arcsec.  
    3: UT1 minus TAI. Unit: s.  
    4: First time derivative of X pole coordinate. Unit: arcsec/day.  
    5: Second time derivative of Y pole coordinate. Unit: arcsec/day.  
    6: First time derivative of UT1 minus TAI coordinate. Unit: s/day.  
    7: Nutation angle in longitude. Units: arcsec  
    8: Nutation angle in obliquity. Units: arcsec  
  
h1 -- Contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad.  
h2 -- Contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad.  
h3 -- Contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad.  
h1r -- First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad/s.  
h2r -- First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad/s.  
h3r -- First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad/s.  
h1rr -- Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad/s\*\*2.  
h2rr -- Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad/s\*\*2.  
h3rr -- Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad/s\*\*2.  
heo -- Array of three components of the contribution of empirical harmonic variations in the EOPs:  
    1: Contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad.  
    2: Contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad.

## NERS User Guide

3: Contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad.

heor -- Array of three components of the first time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3.

1: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 1.

2: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 2.

3: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3.

heorr -- Array of three components of the first time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3.

1: Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 1.

2: Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 2.

3: Second time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3.

m\_par ( INTEGER\*4 ) -- Maximum number of the elements in the output array.

Output parameters:

l\_par ( INTEGER\*4 ) -- Actual number of the EOPs

pars ( REAL\*8 ) -- Array of EOPs. Dimension: m\_par.

Input/Output parameter:

iuerr ( INTEGER\*4 ) -- Error parameter.

On input: -1 -- to print the error message if an error is detected;  
otherwise, not to print the message:

On output: 0 -- normal completion  
>0 -- error code during NERS initialization

- **ners\_quit** — releases memory for internal data structures and un-initialize them.

FORTRAN: NERS\_QUIT ( CODE, NERS )

C: cners\_quit ( int \* quit\_code, struct ners\_struct \* ners )

Input parameter:

code ( INTEGER\*4 ) -- Level of re-initialization:

NERS\_\_EXP -- release memory with coefficients of EOP expansion for the interval of time specified in previous call of routine ners\_init.

NERS\_\_FCS -- release memory with the NERS forecast message

NERS\_\_ALL -- release memory for all internal data structures

Input/Output parameter:

ners ( NERS\_\_TYPE ) -- NERS internal data structure

- **ners\_inq** — inquires either the time interval of EOP availability or the time of the EOP forecast generation.

FORTRAN: NERS\_INQ ( NERS, REQ, M\_PAR, L\_PAR, PARS, IUER )

C: cners\_inq ( int \* quit\_code, struct ners\_struct \* ners )

## NERS User Guide

Input parameter:

ners ( NERS\_\_TYPE ) -- NERS internal data structure

req ( CHARACTER\*(\*) ) -- request type. Supported requests:

range -- request of the EOP range. Returns three values:  
1: start time of EOP availability. Units: seconds since  
2000.01.01\_00:00:00.0 TAI.  
2: stop time of the EOP or the EOP forecast availability.  
Units: seconds since 2000.01.01\_00:00:00.0 TAI.  
3: stop time of the long-term EOP prediction  
fcs\_gen\_time -- request of the EOP forecast message generation time.  
Return one value:  
1: time of the EOP forecast generation. Units: seconds since  
2000.01.01\_00:00:00.0 TAI.

m\_par ( INTEGER\*4 ) -- Maximum number of the elements in the output array.

Output parameters:

l\_par ( INTEGER\*4 ) -- Actual number of the EOPs

pars ( REAL\*8 ) -- Array of EOPs. Dimension: m\_par.

Input/Output parameter:

iuer ( INTEGER\*4 ) -- Error parameter.

On input: -1 -- to print the error message if  
an error is detected;

otherwise, not to print the message:

On output: 0 -- normal completion

>0 -- error code during NERS

- **ners\_get\_series** — returns a 2D array of the Earth Orientation Parameters time series for the specified time range with the specified time step. The type of Earth orientation parameter(s) is defined in variable cpar.

FORTRAN: NERS\_GET\_SERIES ( NERS, TIME\_TAI\_START, TIME\_TAI\_STOP, TIME\_STEP, CPAR,  
DIM1, DIM2, NS, TIM, SER, IUER )

C: cners\_get\_series ( struct ners\_struct \* ners, double time\_tai\_beg, double time\_tai\_end,  
double tim\_step, char \* cpar, int m\_par, int m\_ser, int \* ns,  
double tim[], double ser[], int \* iuer )

Input parameters:

ners ( NERS\_\_TYPE ) -- NERS internal data structure

time\_tai\_start ( REAL\*8 ) -- Start time of EOP series. Units: seconds since 2000.01.01\_00:00:00.0 TAI

time\_tai\_stop ( REAL\*8 ) -- Stop time of EOP series. Units: seconds since 2000.01.01\_00:00:00.0 TAI

time\_step ( REAL\*8 ) -- Time step of the EOP series. Units: seconds since 2000.01.01\_00:00:00.0 TAI

cpar ( CHARACTER\*(\*) ) -- Earth orientation parameter name.

euler -- Euler angles of the Earth rotation

## NERS User Guide

- 1: Euler angle around axis one. Units: rad.
- 2: Euler angle around axis two. Units: rad.
- 3: Euler angle around axis three. Units: rad.

euler\_r -- First time derivative of Euler angles of the Earth rotation

- 1: First time derivative of Euler angle around axis one. Units: rad/s.
- 2: First time derivative of Euler angle around axis two. Units: rad/s.
- 3: First time derivative of Euler angle around axis three. Units: rad/s.

polu -- Array of three EOP parameters:

- 1: X pole coordinate. Unit: arcsec.
- 2: Y pole coordinate. Unit: arcsec.
- 3: UT1 minus TAI. Unit: s.

polur -- Array of six EOP parameters:

- 1: X pole coordinate. Unit: arcsec.
- 2: Y pole coordinate. Unit: arcsec.
- 3: UT1 minus TAI. Unit: s.
- 4: First time derivative of X pole coordinate. Unit: arcsec/day.
- 5: Second time derivative of Y pole coordinate. Unit: arcsec/day.
- 6: First time derivative of UT1 minus TAI coordinate. Unit: s/day.

eops -- Array of eight EOP parameters:

- 1: X pole coordinate. Unit: arcsec.
- 2: Y pole coordinate. Unit: arcsec.
- 3: UT1 minus TAI. Unit: s.
- 4: First time derivative of X pole coordinate. Unit: arcsec/day.
- 5: Second time derivative of Y pole coordinate. Unit: arcsec/day.
- 6: First time derivative of UT1 minus TAI coordinate. Unit: s/day.
- 7: Nutation angle in longitude. Units: arcsec
- 8: Nutation angle in obliquity. Units: arcsec

heo -- Array of three components of the contribution of empirical harmonic variations in the EOPs:

- 1: Contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad.
- 2: Contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad.
- 3: Contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad.

heor -- Array of three components of the first time derivative of the contribution of empirical harmonic variations in the EOPs:

- 1: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 1. Units: rad/s.
- 2: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 2. Units: rad/s.
- 3: First time derivative of the contribution of empirical harmonic variations in the EOPs with respect to axis 3. Units: rad/s.

dim1 ( INTEGER\*4 ) -- First dimension of the EOP series array. The dimension runs over time.

dim2 ( INTEGER\*4 ) -- Second dimension of the EOP series array. The dimension runs over parameters.

Output parameters:

ns ( INTEGER\*4 ) -- the number of returned EOP values

tim ( REAL\*8 ) -- Array of EOP time epochs. Dimension: ns. Units: seconds since 2000.01.01\_00:00:00.

ser ( REAL\*8 ) -- Two dimensional Array of EOPs. Dimensions: dim1,dim2. The number of filled elements along the first dimension is ns. The number of elements along the second dimension is dim2.

## NERS User Guide

filled elements along the second dimension depends on value of parameter cpar.

Input/Output parameter:

iuier ( INTEGER\*4 ) -- Error parameter.

On input: -1 -- to print the error message if  
an error is detected;

otherwise, not to print the message:

On output: 0 -- normal completion

>0 -- error code during NERS

- **ners\_get\_utcmtai** — returns value of UTC minus TAI on the specified UTC time tag.

FORTRAN: NERS\_GET\_UTCMtai ( NERS, UTC, UTC\_M\_TAI, IUER )

C: cners\_get\_utcmtai ( struct ners\_struct \* ners, double \* utc\_obs,  
double \* utc, int \* iuer )

Input parameters:

ners ( NERS\_\_TYPE ) -- NERS internal data structure

utc ( REAL\*8 ) -- UTC time tag. Units: seconds elapsed since 2000.01.01\_00:00:00.0 UTC.

Output:

utc\_m\_tai ( REAL\*8 ) -- UTC minus TAI function. Units: seconds.

Input/Output parameter:

iuier ( INTEGER\*4 ) -- Error parameter.

On input: -1 -- to print the error message if  
an error is detected;

otherwise, not to print the message:

On output: 0 -- normal completion

>0 -- error code during NERS

- **ners\_azelha\_comp** — Computes azimuth, elevation, hour angle, and their time derivatives at moment of time TIM\_TAI for station with coordinates in the terrestrial coordinate system COO\_TRS that observes a source with coordinates in the barycentric celestial coordinate system, right ascension and declination RA, DEC.

FORTRAN: NERS\_AZELHA\_COMP ( NERS, TIM\_TAI, COO\_TRS, RA, DEC,  
REFR\_MODE, AZ, EL, HA,  
AZ\_RATE, EL\_RATE, HA\_RATE, IUER )

C: cners\_azelha\_comp ( struct ners\_struct \* ners, double time\_tai\_beg, double coo\_trs[],  
double ra, double dec, char \* refr, double \* az, double \* el,  
double \* ha, double \* az\_rate, double \* el\_rate, double \* ha\_rate,  
int \* iuer ) ;



# NERS User Guide

## Input parameters:

**TIM\_TAI** ( REAL\*8 ) -- Time for which the Earth orientation parameter(s) is to be computed elapsed since 2000.01.01\_00:00:00.0 TAI. Unit: sec. If TIM\_TAI .LE. -1.0D14, then the azimuth and elevation at the current moment of time will be computed.

**COO\_TRS** ( REAL\*8 ) -- Vector of station coordinates in the rotating crust-fixed coordinate system. Units: m.

**RA** ( REAL\*8 ) -- Source right ascension in the inertial barycenter coordinate system. Units: rad.

**DEC** ( REAL\*8 ) -- Source declination in the inertial barycenter coordinate system. Units: rad.

**REFR\_MODE** ( CHARACTER ) -- Refraction mode. Supported values:

- none -- refractivity is not accounted.
- optic -- formula Bennett for optic range is used ( Bennett, G.G. (1982). "The Calculation of Astronomical Refraction in Marine Navigation". Journal of Navigation, 35(2), 255-259.
- radio --  $3.13D-4/tg(el)$  expression suitable for radio waves is used.

both formulae have a floor of 3 deg, i.e. refraction at elevations below 3 deg is considered to be equal to refraction at 3 deg.

## Output parameters:

**AZ** ( REAL\*8 ) -- Azimuth in rad.

**EL** ( REAL\*8 ) -- Elevation in rad.

**HA** ( REAL\*8 ) -- Hour angle in rad.

**AZ\_RATE** ( REAL\*8 ) -- Time derivative of azimuth in rad/s.

**EL\_RATE** ( REAL\*8 ) -- Time derivative of elevation in rad/s.

**HA\_RATE** ( REAL\*8 ) -- Time derivative of hour angle in rad/s.

## Input/Output parameter:

**NERS** ( NERS\_\_TYPE ) -- The data structure that keeps internal parameters related to the Network Earth Rotation Service.

**IUER** ( INTEGER\*4 ) -- Error parameter.

On input: -1 -- to print the error message if an error is detected;  
otherwise, not to print the message:

## NERS User Guide

On output: 0 -- normal completion  
>0 -- error code during NERS

- **ners\_radec\_comp** — Computes right ascension, declination and their time derivatives in the barycentric coordinate system at moment of time TIM\_TAI for station with coordinates in the terrestrial coordinate system COO\_TRS that observes a source with given azimuth and elevation.

FORTTRAN: NERS\_RADEC\_COMP ( NERS, TIM\_TAI, COO\_TRS, AZ, EL, REFR\_MODE,  
RA, DEC, RA\_RATE, DEC\_RATE, IUER )

C: cners\_radec\_comp ( struct ners\_struct \* ners, double tim\_tai,  
double coo\_trs[], double az, double el,  
char \* refr\_mode, double \* ra, double \* dec,  
double \* ra\_rate, double \* dec\_rate, int \* iuer )

Input parameters:

TIM\_TAI ( REAL\*8 ) -- Time for which the Earth orientation  
parameter(s) is to be computed elapsed  
since 2000.01.01\_00:00:00.0 TAI. Unit:  
sec. If TIM\_TAI .LE. -1.0D14, then the  
right ascension and declination at the  
current moment of time will be computed.

COO\_TRS ( REAL\*8 ) -- Vector of station coordinates in the  
rotating crust-fixed coordinate system.  
Units: m.

AZ ( REAL\*8 ) -- Azimuth in rad.

EL ( REAL\*8 ) -- Elevation in rad.

REFR\_MODE ( CHARACTER ) -- Refraction mode. Supported values:

none -- refractivity is not accounted.  
optic -- formula Bennett for optic range  
is used ( Bennett, G.G. (1982).  
"The Calculation of Astronomical  
Refraction in Marine Navigation".  
Journal of Navigation, 35(2),  
255-259.  
radio --  $3.13D-4/tg(el)$  expression  
suitable for radio waves is used.  
both formulae have a floor of 3 deg, i.e.  
refraction at elevations below 3 deg is  
considered to be equal to refraction at  
3 deg.

Output parameters:

RA ( REAL\*8 ) -- Source right ascension in the inertial  
barycentric coordinate system. Units: rad.

DEC ( REAL\*8 ) -- Source declination in the inertial  
barycentric coordinate system. Units: rad.

RA\_RATE ( REAL\*8 ) -- Rate of change of right ascension in the

## NERS User Guide

inertial barycentric coordinate system.

Units: rad/s.

DEC\_RATE ( REAL\*8 ) -- Rate of change of declination in the  
inertial barycentric coordinate system.

Units: rad/s.

Input/Output parameter:

NERS ( NERS\_\_TYPE ) -- The data structure that keeps internal  
parameters related to the Network Earth  
Rotation Service.

IUER ( INTEGER\*4, OPT ) -- Universal error handler.

Input: switch IUER=0 -- no error messages  
will be generated even in the case  
of error. IUER=-1 -- in the case of  
error the message will be put on  
stdout.

Output: 0 in the case of successful  
completion and non-zero in the  
case of error.

### Auxilliary programs

Program **ners\_inq** returns information about package *NERS*

Usage: **ners\_inq** query

where query is one of

- version — Version of NERS library.
  - prefix — Directlry where NERS is installed.
  - include — Directory with NERS include files.
  - libdir — Directory that contains NERS library.
  - bindir — Directory that contains NERS binary executatble files.
  - root — Root directroy with NERS distribution.
  - share — Directory that contains NERS data.
  - script — Directory that contains NERS python script.
  - doc — Directory that contains NERS documentation.
  - config — File name with default NERS configuration.
  - standalone — string whether NERS was conifgured in standalone mode (YES or NO)
-