# Naval Observatory Vector Astrometry Software (NOVAS) Version 3.1, Introducing a Python Edition

**E. G. Barron, G. H. Kaplan, J. Bangert, J. L. Bartlett, W. Puatua, W. Harris, P. Barrett**

U.S. Naval Observatory

## What is NOVAS?

The Naval Observatory Vector Astrometry Software (NOVAS) is a free source-code library for computing various commonly needed quantities in positional astronomy. NOVAS can supply, in one or two subroutine or function calls, the instantaneous celestial position of any star or planet in a variety of coordinate systems. NOVAS also provides access to all of the "building blocks" that go into such computations—single-purpose subroutines or functions for common astrometric algorithms, such as those for precession, nutation, aberration, parallax, etc. NOVAS calculations are accurate at the sub-milliarcsecond level. The NOVAS package is an easy-to-use facility that can be incorporated into data reduction programs, telescope control systems, and simulations. NOVAS is used in the production of the U.S. parts of *The Astronomical Almanac*.

The NOVAS code is available in Fortran, C, and—with version 3.1—Python. The Fortran edition, first released in the late 1970s, is updated periodically to use new, more accurate models that represent the evolving standards of the international astronomical and geodetic communities. The C edition, introduced in 1996, provides the same functionality and accuracy. The Python edition creates an extension module by providing a wrapper around the C library functions, thereby ensuring the consistency of the calculations.

## Changes from NOVAS 3.0

NOVAS 3.1 includes several additions, changes, performance boosts, and bug fixes over version 3.0 (function names follow the format *Fortran/C*):

- **New function *CELTER/cel2ter*:** the inverse of *TERCEL/ter2cel*, this function transforms a position vector from GCRS (Geocentric Celestial Reference System) to ITRS (International Terrestrial Reference System) coordinates.
- **Reverse polar-motion transformation in *WOBBLE/wobble*:** when given a negative Julian date (Fortran) or a non-negative value for the DIRECTION argument (C, additional argument over the version 3.0 function) the *WOBBLE/wobble* function will perform a reverse polar-motion transformation on the input vector.
- **Handle two J2000.0 inputs in the *PRECES/precession* function**: in version 3.0 the *PRECES/precession* function returns either an undefined vector (Fortran) or a zero vector (C) when both date inputs are J2000.0 (2451545.0 TDB). These functions in version 3.1 return the input vector unmodified given the same date inputs.
- **Array overflow fixes:** several functions in the C version 3.0 library suffer from array overflows when copying strings. This is now fixed.

Improvements specifically to the C ephemeris-handling code include

- **DE421 support:** the *ephem_open* function can now properly open and handle JPL's Development Ephemeris 421 (DE421).
- **Ephemeris file feedback:** *ephem_open* now has a new argument containing the DE number of the open ephemeris file; this allows the user to easily verify which ephemeris is in use.
- **Memory leak fixes:** functions that deal with the ephemeris file now properly free the file buffer and close the file upon encountering an error condition.
- **Proper 64-bit compatibility:** differences in the size of long integers between 64-bit Windows and *nix-based (Mac, Linux) systems result in runtime errors when using NOVAS 3.0 with the JPL ephemerides. This is now fixed.

A full list of changes will be made available with the official release of version 3.1.

## NOVAS Py

A major addition to the NOVAS family shipping with version 3.1 is NOVAS Py, an extension module for the Python programming language. This Python edition borrows its underlying computational code from the C edition and implements nearly all features. However, NOVAS Py does not use an external file with positions to calculate the Celestial Intermediate Origin (CIO) and does not handle the external USNO/AE98 minor planet ephemerides. If there is demand for either or both of these features, support will be implemented in a future version of the module.

## Why Python?

Python (*http://www.python.org*) is a powerful, cross-platform, open-source programming language that is easy to learn and has excellent code readability. Python is interpreted, uses dynamic typing, supports multiple programming paradigms—object-oriented, procedural, functional—automatic memory management, and can be used interactively. It ships with a number of modules that provide tools for a variety of application domains and an immense number of third-party modules are available. Python is already widely used in the astronomy community and is still rapidly gaining popularity, in many cases replacing tools like IDL, Matlab, and Mathematica.

Implementing NOVAS in Python opens up the library to much greater ease-of-use in web applications, quick calculations, and rapid development of large and complex software products.

## NOVAS Py in Three Parts

The release of NOVAS Py will be in three phases. The first phase, to be released alongside the Fortran and C versions 3.1, is a compatibility module. The function calls in the Python edition are almost a one-to-one match to the function calls in the C edition with very minor changes in the order of the input arguments for some functions; these differences are implemented to take advantage of Python's ability to use (optional) keyword arguments.

Phase two of NOVAS Py will add support for large array input and output through use of NumPy (*http://numpy.scipy.org/*), a powerful and widely-used scientific computing package that provides Python with efficient array and matrix handling. Users will be able to perform computations on arrays of stars, dates, observation sites, etc. using a single function call, which significantly improves usability and performance.

The third and final phase will take the existing code from the second phase and transform it into a more object-oriented module. The compatibility code will remain available as a sub-module for users who wish to continue using it.

## How to Obtain NOVAS

NOVAS 3.1 will be released in the first quarter of 2011. It will be available for download on the US Naval Observatory's website at

*http://www.usno.navy.mil/usno/astronomical-applications/software-products/novas*

The Python edition will, at least initially, share the User's Guide with the C edition, while Fortran will have its own documentation. Supplemental documentation for the Python edition will be available in a README distributed with the code. Additionally, the combined Fortran and C User's Guides for version 3.0 are still available as U.S. Naval Observatory Circular 180. Also provided on the NOVAS page is USNO Circular 179, which describes the latest IAU Resolutions on astronomical reference systems, time scales, and Earth rotation models; and USNO Circular 181, which provides information on the nutation models implemented in NOVAS 3.0. See

*http://www.usno.navy.mil/usno/astronomical-applications/publications/usno-special-pubs*

for information on ordering or downloading these documents.

Originally presented at the 217th Meeting of the American Astronomical Society in Seattle, WA, 12 January 2011.