

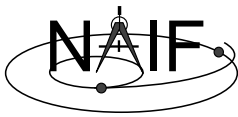


---

Navigation and Ancillary Information Facility

# Introduction to the SPICE Toolkit

October 2007



## Topics

---

Navigation and Ancillary Information Facility

- **SPICE Toolkit**
  - SPICELIB
  - CSPICE
  - Icy
  - Mice
- **Installed Directory Structure**
- **Toolkit Documentation**
- **Toolkit Utility Programs**
- **Toolkit Application Programs**
- **Supported Platforms**



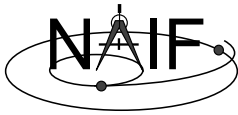
# SPICE Toolkit

Navigation and Ancillary Information Facility

- **The SPICE Toolkit is available in Fortran, C, IDL (Interactive Data Language), and MATLAB**
- **Toolkits contain:**
  - **Software**
    - » **Subroutine libraries, with source code**
      - SPICELIB (Fortran)
      - CSPICE (C)
      - Icy (C)
      - Mice (C and MATLAB script)
    - » **Executable programs**
      - application and utility programs
      - cookbook examples
    - » **Installation/build scripts**
  - **Documentation**
    - » Available in ASCII and HTML
  - **Data**
    - » **Sample kernel files**
      - Supplied **ONLY** for use with cookbook programs, not valid for general use

Introduction to the SPICE Toolkit

3



# SPICE Toolkit

Navigation and Ancillary Information Facility

- **The components listed on the previous page comprise the generic Toolkit**
  - Toolkits delivered to missions or other special customers may be augmented with mission- or customer-specific products
- **The Fortran, C, IDL, and MATLAB Toolkits are delivered as standalone products**
  - The IDL and MATLAB products include the CSPICE Toolkit
- **The Fortran, CSPICE, IDL, and MATLAB Toolkits use identical kernel files on all platforms.**

Introduction to the SPICE Toolkit

4

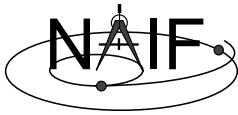


# SPICE Toolkit

Navigation and Ancillary Information Facility

## • Toolkit Version

- Generic SPICE Toolkits have an associated Version number
  - » Example: “N0061” (also written as “N61”)
- The version number applies to the Fortran, C, IDL and MATLAB implementations for all supported platforms.
- When does NAIF release new SPICE Toolkit versions?
  - » Not according to a fixed schedule
  - » Primarily driven by addition of significant, new capabilities to Toolkit
    - Icy, for example
  - » Occasionally Toolkit updates are released to fix bugs, improve documentation, or satisfy an urgent request from a flight project



## Toolkit Library Functionality

Navigation and Ancillary Information Facility

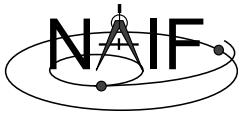
- The toolkit libraries contain a fairly broad set of capabilities related to the computations needed for “space geometry” and time conversions.
  - Broad categories are enumerated on the next several pages
- Caution: not all functionality is yet present in all four language versions of the Toolkit library.
  - The Fortran (SPICE) and C (CSPICE) Toolkits provide virtually identical functionality.
  - The IDL (Icy) toolkit duplicates most functionality from the C Toolkit wrapper routines.
  - The MATLAB (Mice) Toolkit provides interfaces only to those routines NAIF considers the most often needed by users.



# Toolkit Library Functionality

Navigation and Ancillary Information Facility

- **Kernel read access**
  - “Load” kernels
  - Get state or position vectors (SPK)
  - Get orientation of planets, natural satellites, etc. (PCK)
  - Get body shape parameters or physical constants (PCK)
  - Get orientation of spacecraft or spacecraft instruments or structures (CK, FK)
  - Get instrument parameters (e.g., FOV) (IK)
  - Query binary EK files (EK-ESQ)
- **Kernel write access**
  - SPK writers
  - CK writers
  - EK writers (sequence component, ESQ)
  - PCK writers (only for binary PCK files)



# Toolkit Library Functionality

Navigation and Ancillary Information Facility

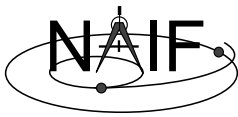
- **Additional ephemeris functions**
  - Classical osculating elements
  - Two-body Keplerian propagation
  - NORAD two line elements sets (TLE) propagation
  - Light time and Stellar aberration computation
- **Frame transformation**
  - Obtain 3x3 matrices for frame transformations of positions
  - Obtain 6x6 matrices for frame transformations of states
- **Time conversion**
  - Conversion between standard systems: TDB, TT (TDT), UTC
  - Conversion between SCLK and other systems
  - Parsing and formatting



# Toolkit Library Functionality

Navigation and Ancillary Information Facility

- **Math**
  - Vector/Matrix operations
  - Rotations, Euler angles, quaternions
  - Coordinate conversion (systems: latitudinal, cylindrical, rectangular, RA and DEC, spherical, geodetic, planetographic)
  - Geometry: ellipsoids, ellipses, planes
  - High-level functions: illumination angles, sub-observer point, sub-solar point, surface intercept point.
- **Constants**
  - Julian date of epoch J2000, SPD(seconds per day), PI, etc.
- **Strings**
  - Parsing: find tokens, words
  - Numeric conversion
  - Pattern matching
  - Replace marker, substring
  - Suffix, prefix
  - Case conversion
  - Find first/last non-blank character, first/last printing character



# Toolkit Library Functionality

Navigation and Ancillary Information Facility

- **Arrays**
  - Sorting, finding order vector, reordering
  - Searching: linear, binary
  - Insertion and deletion
- **Name/code conversion**
  - Bodies
  - Frames
- **I/O support**
  - Logical unit management
  - Open, read, write text files
  - Kernel pool API
- **Exception handling**
  - Control exception handling behavior: mode, message set, output device
  - Construct error messages

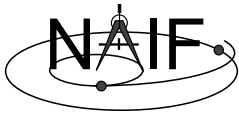


# Toolkit Library Functionality

Navigation and Ancillary Information Facility

---

- **Advanced data types**
  - Cells, Sets
  - Windows (sometimes called schedules)
  - Symbol Tables
  - Planes, Ellipses



## Fortran SPICE Toolkit

Navigation and Ancillary Information Facility

---

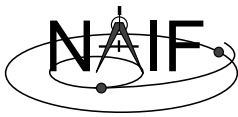
- **The Fortran SPICE Toolkit:**
  - Developed first: in use since February 1990
  - Contains code written in ANSI Standard Fortran 77
    - » A few widely supported non-ANSI extensions are used, for example DO WHILE, DO...END DO
  - Compiles under a wide variety of Fortran compilers



# CSPICE Toolkit

Navigation and Ancillary Information Facility

- **CSPICE is designed to duplicate the functionality of the Fortran Toolkit**
  - All CSPICE source code is in ANSI C
    - » The Fortran SPICE Toolkit code is converted to ANSI C using the automatic translation program f2c
    - » High-level functions have been hand-coded in C and documented in C style in order to provide a natural C-style API. These functions are called “wrappers”
    - » Most wrappers encapsulate calls to C functions generated by f2c
      - The simpler wrappers do their work in-line to boost performance
    - » f2c'd functions may be called directly, but this is strongly discouraged since f2c'd functions emulate Fortran functionality:
      - Call by reference
      - Fortran-style array indexing
      - Fortran-style strings



# CSPICE Toolkit

Navigation and Ancillary Information Facility

- **CSPICE runs under a wide variety of ANSI C compilers**
- **CSPICE functions may be called from within C++ source code**
  - CSPICE prototypes are protected from name mangling
- **Current CSPICE Limitations**
  - Not all “Required Reading” reference documents have been converted to C style, with C examples
    - » Eventually all will be converted
  - CSPICE wrappers exist for only a subset of the functionality provided by SPICELIB
    - » Includes all the most commonly used modules
    - » More will be added as time permits
  - In some very limited cases, code generated by f2c fails to emulate Fortran accurately. Should not be a problem.
    - » List-directed I/O has some problems (not consequential for CSPICE)
    - » Treatment of white space in text output is slightly different in CSPICE
    - » Logical unit-to-file name translation does not handle file name “synonyms” properly under Linux: once opened with a specified name, a file must be referred to using the same name throughout a program run.

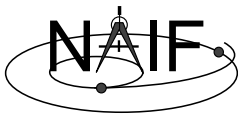


# Icy Toolkit

Navigation and Ancillary Information Facility

---

- **Icy is a SPICE Toolkit for IDL users.**
  - First delivered as part of the N0057 Toolkit in March, 2004.
- **Icy provides an IDL-callable “wrapper” interface for many CSPICE wrapper routines**
  - Example:
    - » CSPICE: `spkezt_c ( targ, et, ref, abcorr, obs, state, &ltime );`
    - » Icy: `cspice_spkezt, targ, et, ref, abcorr, obs, state, ltime`
  - More IDL-callable wrappers added as time permits
- **By necessity the Icy Toolkit includes the complete CSPICE Toolkit.**
  - Additional Icy software components are:
    - » IDL interface wrappers (implemented in ANSI C)
    - » Icy cookbook programs (implemented in IDL)



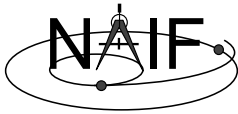
# Icy Toolkit

Navigation and Ancillary Information Facility

---

- **Icy Documentation**
  - Icy Reference Guide
    - » Principal documentation showing how to call Icy wrappers
    - » Each Icy wrapper has an HTML page containing usage examples serving as the SPICE “module header”
  - Icy Required Reading
    - » Provides background information essential for programming with Icy

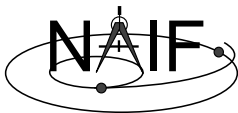




# Mice Toolkit

Navigation and Ancillary Information Facility

- **Mice is a SPICE Toolkit for MATLAB users.**
  - Currently under limited distribution as a Beta-test product
  - Expected official distribution late in CY2007
- **Mice provides a MATLAB-callable “wrapper” interface for many CSPICE wrapper routines**
  - Example:
    - » CSPICE: `spkezr_c ( targ, et, ref, abcorr, obs, state, &ltime );`
    - » Mice: `[state, ltime] = cspice_spkezr( targ, et, ref, abcorr, obs)`
  - More MATLAB-callable wrappers added as time permits
- **By necessity all Mice Toolkit packages include the complete CSPICE Toolkit.**
  - Additional Mice software components are:
    - » MATLAB interface wrappers (implemented in MATLAB wrapper scripts calling the ANSI C based interface library)
    - » Mice cookbook programs (implemented in MATLAB script)



# Mice Toolkit

Navigation and Ancillary Information Facility

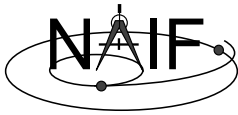
- **Mice Documentation**
  - **Mice Reference Guide**
    - » Principal documentation showing how to call Mice wrappers
    - » Each Mice wrapper script has a documentation header containing usage examples, serving as SPICE “module header”, available from the `help` command. This documentation also exists as a HTML page.
  - **Mice Required Reading**
    - » Provides background information essential for programming with Mice



# Installed Directory Structure

Navigation and Ancillary Information Facility

- **Directory structures for the Toolkits are almost identical. However...**
  - Unlike for SPICE, the CSPICE, Icy and Mice packages have a directory for include files
  - The names for application source code directories in CSPICE, Icy and Mice differ, slightly, from those in SPICE
  - Icy and Mice include additional directories for
    - » Icy/Mice source code
    - » Icy/Mice cookbook programs
- **The top level directory name for each Toolkit is:**
  - “toolkit” for Fortran Toolkits
  - “cspice” for C Toolkits
  - “icy” for IDL Toolkits
  - “mice” for MATLAB Toolkits



# Installed Directory Structure

Navigation and Ancillary Information Facility

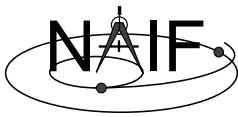
- **The next level is comprised of:**
  - **data**
    - » Cookbook example kernels (use ONLY for training with cookbook programs)
  - **doc**
    - » Text documents — \*.req, \*.ug, spicelib.idx/cspice.idx, whats.new, dscriptn.txt, version.txt.
    - » Subdirectory containing HTML documentation, called “html”.
      - The “html” subdirectory contains a single file — the top level HTML documentation index called “index.html” — and a number of subdirectories, one for each of the various groups of documents in HTML format (API Reference Guide pages, User’s Guide pages, etc.)
  - **etc**
    - » In generic Toolkits this directory is empty.
  - **exe**
    - » Executables for brief, chronos, ckbrief, commnt, inspekt, mkspk, msopck, spacit, spkdiff, spkmerge, tobin, toxfr, version.
    - » Executables for the several cookbook example programs.



# Installed Directory Structure

## Navigation and Ancillary Information Facility

- **include** (applies to CSPICE, Icy, and Mice)
  - » **API header files.**
    - File to include in callers of CSPICE is SpiceUsr.h
- **lib**
  - » **Toolkit libraries:**
    - **For Fortran SPICE Toolkits**
      - spicelib.a (public modules; use these)
      - support.a (private modules; don't use these)
    - **For CSPICE Toolkits**
      - cspice.a (public modules; use these)
      - csupport.a (private modules; don't use these)
    - **For Icy Toolkits:**
      - icy.so (shared object library)
      - icy.dlm (dynamically loadable module)
      - cspice.a
      - csupport.a
    - **For Mice Toolkits:**
      - mice.mex\* (shared object library)
      - cspice.a
      - csupport.a
- **src**
  - » **Source code directories for executables and libraries**
    - Files have type \*.f, \*.for, \*.inc, \*.pgm, \*.c, \*.h, \*.x, \*.pro, \*.m
    - \*.h files appearing here are not part of the user API



# Toolkit Documentation

## Navigation and Ancillary Information Facility

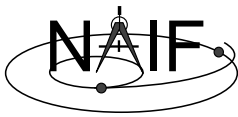
- **All Toolkits include documentation in plain text and HTML formats**
  - Plain text documents are located under the “doc” directory
  - HTML documents are located under the “doc/html” directory with the “doc/html/index.html” file as the top level index
- **All Toolkits include the following kinds of documents**
  - **Module headers**
    - » **Act as primary functional specification:** I/O, exceptions, particulars defining behavior of module
    - » **Contain code examples**
    - » **A standard format is used for each routine or entry point**
    - » **Plain text Module Headers:**
      - **Fortran:** the top comment block in the source code files under “src/spicelib”
      - **C:** the top comment block in the source code files under “src/cspice”
      - **IDL:** Icy Module Headers are not available in plain text format
      - **MATLAB** accessible via “help *function\_name*” command
    - » **HTML Module Headers are accessible using the “API Reference Guide” link from the top level index.**



# Toolkit Documentation

Navigation and Ancillary Information Facility

- **Kinds of documents in the Toolkit (continued)**
  - **Required Reading**
    - » References for principal subsystems
    - » Provide many low-level details
    - » Provide code examples
    - » Plain text versions are located under “doc” and have extension “.req”
    - » HTML versions are accessible using the “Required Reading Documents” link from the top level index.
    - » Not all of Required Readings were adapted for all languages
      - Some of the Required Reading documents provided with CSPICE still cover Fortran SPICE
      - No Required Readings were adapted for Icy or Mice (Icy and Mice toolkits contain CSPICE Required Readings)
  - **User’s Guides**
    - » Interface specifications for the Toolkit utility programs and applications
    - » Plain text versions are located under “doc” and have extension “.ug”
    - » HTML versions are accessible using the “User’s Guide Documents” link from the top level index.



# Toolkit Documentation

Navigation and Ancillary Information Facility

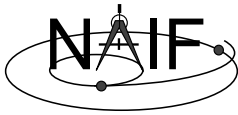
- **Other documents**
  - **Permuted Index**
    - » Maps phrases describing functionality to corresponding module names and file names
    - » Shows names of all entry points in Fortran toolkit APIs
    - » Plain text version is located under “doc” and has extension “.idx”:
      - Fortran: spicelib.idx
      - C: cspice.idx
      - IDL, MATLAB: not available
    - » HTML version is accessible using the “Permuted Index” link from the top level index.
  - **Toolkit Description**
    - » Describes the directory structure and contents of an installed Toolkit
    - » Customized based on set of delivered products and platform
    - » Plain text version is “doc/dscriptn.txt”
    - » HTML version is accessible using the “Toolkit Contents” link from the top level index.



# Toolkit Documentation

Navigation and Ancillary Information Facility

- **Other documents (continued)**
  - **Introduction to SPICE**
    - » **Brief introduction to the Toolkit and SPICE system**
    - » **Not available in plain text**
    - » **HTML version is accessible using the “Introduction to the SPICE System” link from the top level index.**
  - **What’s New in SPICE**
    - » **Describes new features and bug fixes**
    - » **Plain text version is “doc/whats.new”**
    - » **HTML version is accessible using the “What’s New in SPICE” link from the top level index.**
  - **Toolkit Version Description**
    - » **Indicates Toolkit version**
    - » **Plain text version is “doc/version.txt”**
    - » **Not available in HTML**



# Toolkit Utility Programs

Navigation and Ancillary Information Facility

- **SPICE Toolkit utility programs are available to:**
  - **port binary SPICE kernels between incompatible systems**
    - » **tobin, toxfr, spacit**
  - **add comments to binary kernels**
    - » **commnt**
  - **read comments from binary kernels**
    - » **commnt, spacit**
    - » **inspekt (only for EK/ESQ files)**
  - **summarize coverage of binary kernels**
    - » **brief, ckbrief, spacit**
  - **merge or subset SPK files**
    - » **spkmerge**
  - **indicate current Toolkit version**
    - » **version**

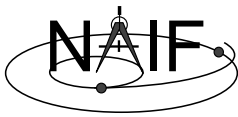


# Toolkit Application Programs

---

Navigation and Ancillary Information Facility

- **SPICE Toolkit application programs perform various tasks:**
  - create a new SPK file from a text file of state vectors
    - » mkspk
  - compare (diff) two SPKs
    - » spkdiff
  - create a new CK from a text file of attitude data
    - » msopck
  - carry out a wide assortment of time conversions
    - » chronos
  - query Event Kernels (EKs)
    - » inspekt

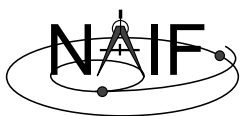


# Supported Platforms

---

Navigation and Ancillary Information Facility

- **NAIF ports the SPICE Toolkit to several popular platforms**
  - Each platform is characterized by
    - » Hardware type
    - » Operating System
    - » Compiler
    - » Selected compilation options
- **NAIF provides SPICE Toolkit packages for each supported platform**
  - If you cannot find a package built for the platform of interest to you, contact NAIF
    - » Don't try to port a Toolkit built for another platform



## Supported Platforms - Fortran

Navigation and Ancillary Information Facility

Product Name	Operating System	Compiler
Mac/PowerPC, OS-X, Absoft FORTRAN	OS X 10.3.x	Absoft Pro Fortran 9.0
Mac/Intel, OS-X, Intel FORTRAN	OS X 10.4.x	Intel Fortran 9.1
Mac/PowerPC, OS-X, g77	OS X 10.3.x	g77, GNU Fortran 3.4.2
PC, CYGWIN, g77	Windows/Cygwin	g77, GNU Fortran 3.2
PC, Linux, g77	Red Hat Linux (RHE4)	g77, GNU Fortran 3.4.5
PC, Windows, Compaq Visual (Digital) FORTRAN	Windows NT/2K/XP	Compaq Digital Fortran 6.0
PC, Windows, Intel FORTRAN	Windows XP	Intel Visual Fortran 9.1
PC, Windows, Lahey FORTRAN95	Windows NT/2K/XP	Lahey FORTRAN 95 5.6
Sun, Solaris, SUN FORTRAN	Solaris 9	Sun WorkShop 6 update 2 FORTRAN 77 5.3



## Supported Platforms - C

Navigation and Ancillary Information Facility

Product Name	Operating System	Compiler
Mac/PowerPC, OS-X, Apple C	OS X 10.3.x	gcc, GNU C 3.3
Mac/Intel, OS-X, Apple C	OS X 10.4.x	gcc, GNU C 4.0.1
PC, CYGWIN, gCC	Windows/Cygwin	gcc, GNU C 3.2
PC, Linux, gCC	Red Hat Linux (RHE4)	gcc, GNU C 3.4.5
PC, Linux, gCC/64bit	Red Hat Linux (RHE4)	gcc, GNU C 3.4.5
PC, Windows, Microsoft Visual C	Windows NT/2K/XP	Microsoft Visual Studio .NET 7.0 C
Sun, Solaris, SUN C	Solaris 9	Sun WorkShop 6 update 2 C 5.3
Sun, Solaris, gCC	Solaris 9	gcc, GNU C 3.3.2
Sun, Solaris, gCC/64bit	Solaris 9	gcc, GNU C 3.3.2

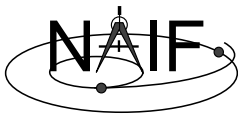


## Supported Platforms - IDL

Navigation and Ancillary Information Facility

Product Name	Operating System	Compiler
Mac/PowerPC, OS-X, Apple C/IDL 6.3	OS X 10.3.x	gcc, GNU C 3.3
Mac/Intel, OS-X, Apple C/IDL 6.3	OS X 10.4.x	gcc, GNU C 4.0.1
PC, Linux, gcc/IDL 6.3	Red Hat Linux (RHE4)	gcc, GNU C 3.4.5
PC, Windows, Microsoft Visual C/IDL 6.3	Windows XP	Microsoft Visual Studio .NET 7.0 C
Sun, Solaris, SUN C/IDL 6.3	Solaris 9	Sun WorkShop 6 update 2 C 5.3
Sun, Solaris, gcc/IDL 6.3	Solaris 9	gcc, GNU C 3.3.2

**Icy was built and tested using IDL version 6.3**



## Supported Platforms - MATLAB

Navigation and Ancillary Information Facility

Product Name	Operating System	Compiler
Mac/PowerPC, OS-X, Apple C	OS X 10.3.x	gcc, GNU C 3.3
PC, Linux, gCC	Red Hat Linux (RHE4)	gcc, GNU C 3.4.5
PC, Windows, Microsoft Visual C	Windows XP	Microsoft Visual Studio .NET 7.0 C
Mac/Intel, OS-X, Apple C	OS X 10.4.x	gcc, GNU C 4.0.1

**Mice was built and tested using MATLAB version 7.2 (2006a)**