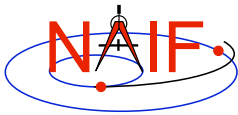


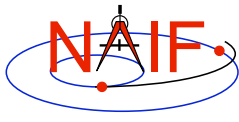
Getting Started Using SPICE

March 2006



Which Pieces Do I Use?

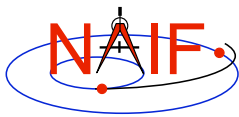
- **There's no single answer**
 - Depends on what task you wish to accomplish
 - Depends on what mission you are working on
- **The next several charts highlight some key stuff**
 - We assume you have already looked at the major SPICE tutorials, or already have some familiarity with SPICE.
 - We assume you have successfully downloaded and installed the SPICE Toolkit.
- **Consider printing these charts and keeping them near your workstation**



Reminder of Key Subsystems

Navigation and Ancillary Information Facility

- **SPK:** Position (and velocity) of things
- **PCK:** Size/shape/orientation of target bodies
- **IK:** Instrument field-of-view geometry
- **CK:** Orientation of spacecraft or s/c structures that move
- **FK:** Definition/specification of non-core reference frames
- **LSK:** UTC (SCET) \Leftrightarrow ET time conversions
- **SCLK and LSK:** SCLK \Leftrightarrow ET time conversions



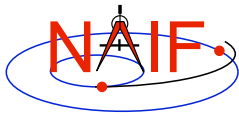
Primary Kernel Interfaces - 1

Navigation and Ancillary Information Facility

Which SPICE interface modules are most commonly called to **use** the data from a given kernel type?

SPK	SPKEZR, SPKPOS	FK	SXFORM, PXFORM, SPKEZR, SPKPOS
PCK	SXFORM, PXFORM, SPKEZR, SPKPOS, BODVRD	LSK	STR2ET, TIMOUT
IK	G*POOL, GETFOV	SCLK	SCS2E, SCE2S, SXFORM, PXFORM, SPKEZR, SPKPOS
CK	SXFORM, PXFORM, SPKEZR, SPKPOS, (CKGPAV, CKGP)	EK/ESQ	EKFIND, EKG*

Notes: FURNISH is used to load (provide access to) all SPICE kernels.
 API names shown are for FORTRAN versions:
 - use lower case and add an “_c” when using C
 - use lower case and prepend “cspice_” when using Icy (IDL)



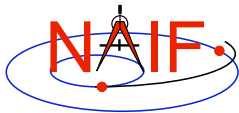
Primary Kernel Interfaces - 2

Navigation and Ancillary Information Facility

For a given module, which kind(s) of kernel(s) will or may be needed?

Module Name	Kernel Type(s) Needed						
	SPK	PCK	IK	CK	FK	LSK	SCLK
SPKEZR, SPKPOS	Y	M		M	M	M	M
SXFORM, PXFORM		M		M	M	M	M
CKGP, CKGPAV		M		Y	M	M	M
GETFOV			Y				
G*POOL			M				
STR2ET, TIMEOUT						Y	
SCS2E, SCE2S						Y	Y
CHRONOS (time conversion app.)	M	M		M	M	Y	M

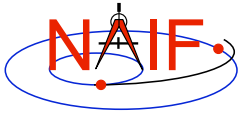
Y = the indicated kernel type is always needed
M = the indicated kernel type may be needed



Kernel "Coverage" Cautions

Navigation and Ancillary Information Facility

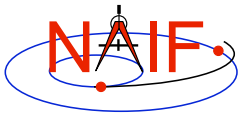
- Your set of kernels must:
 - contain data for all "objects" of interest
 - » Sometimes you must include intermediary objects that provide a connection
 - contain data covering the time span of interest to you
 - » Watch out for data gaps within that time span
 - contain all the kernel types needed by SPICE to answer your question
 - » As the previous charts allude, you may need one or more kernels that are not obvious
 - be managed (loaded) properly if there are overlapping (competing) data within the set of files you are using



How Can I Find Possibly Useful Modules?

Navigation and Ancillary Information Facility

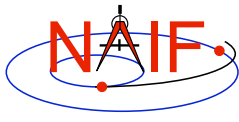
- Review the previous charts
- Look at the appropriate SPICE tutorial(s)
- Look at the “Most Useful SPICELIB Subroutines” document (`../doc/mostused.pdf`)
- Search the permuted index:
 - `spicelib.idx` for the FORTRAN toolkits (`../doc/spicelib.idx`)
 - » (This document also correlates entry point names with source code files)
 - `cspice.idx` for the C and Icy toolkits (`../doc/cspice.idx`)
- Read relevant portions of a SPICE “required reading” reference document (e.g. “`spk.req`”)



How Can I Understand How To Use Those Modules?

Navigation and Ancillary Information Facility

- The primary user-oriented documentation about each module is found in the “header” at the top of each source code file
 - (More documentation is found at the additional entry points for those FORTRAN modules that have multiple entry points.)
- Reference documentation for major subsystems is found in like-named “required reading” documents (e.g. `spk.req`, `ck.req`, etc.)
- The SPICE tutorials contain much helpful information
- See “SPICE Documentation Taxonomy” in the tutorials collection for additional reading suggestions



Does NAIF Provide Any Examples?

Navigation and Ancillary Information Facility

- **Nearly all module headers contain one or more working examples**
- **“Most Useful SPICELIB Subroutines” has code fragments (.../doc/mostused.pdf)**
- **The relevant “required reading” reference documents often contain examples**
- **Three tutorials offer programming examples**
- **Some simple “cookbook” programs are found in the Toolkit (.../src/cookbook/...)**
- **Make use of the SPICE Programming Lessons available from the NAIF server**