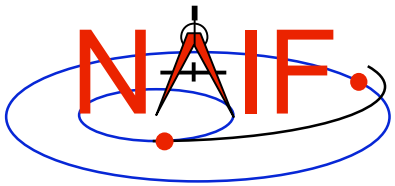
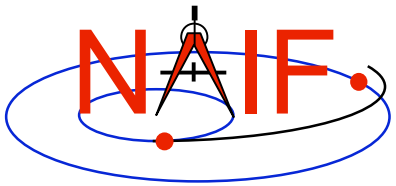


Navigation and Ancillary Information Facility

Frames Kernel
FK

January 2009



Introduction

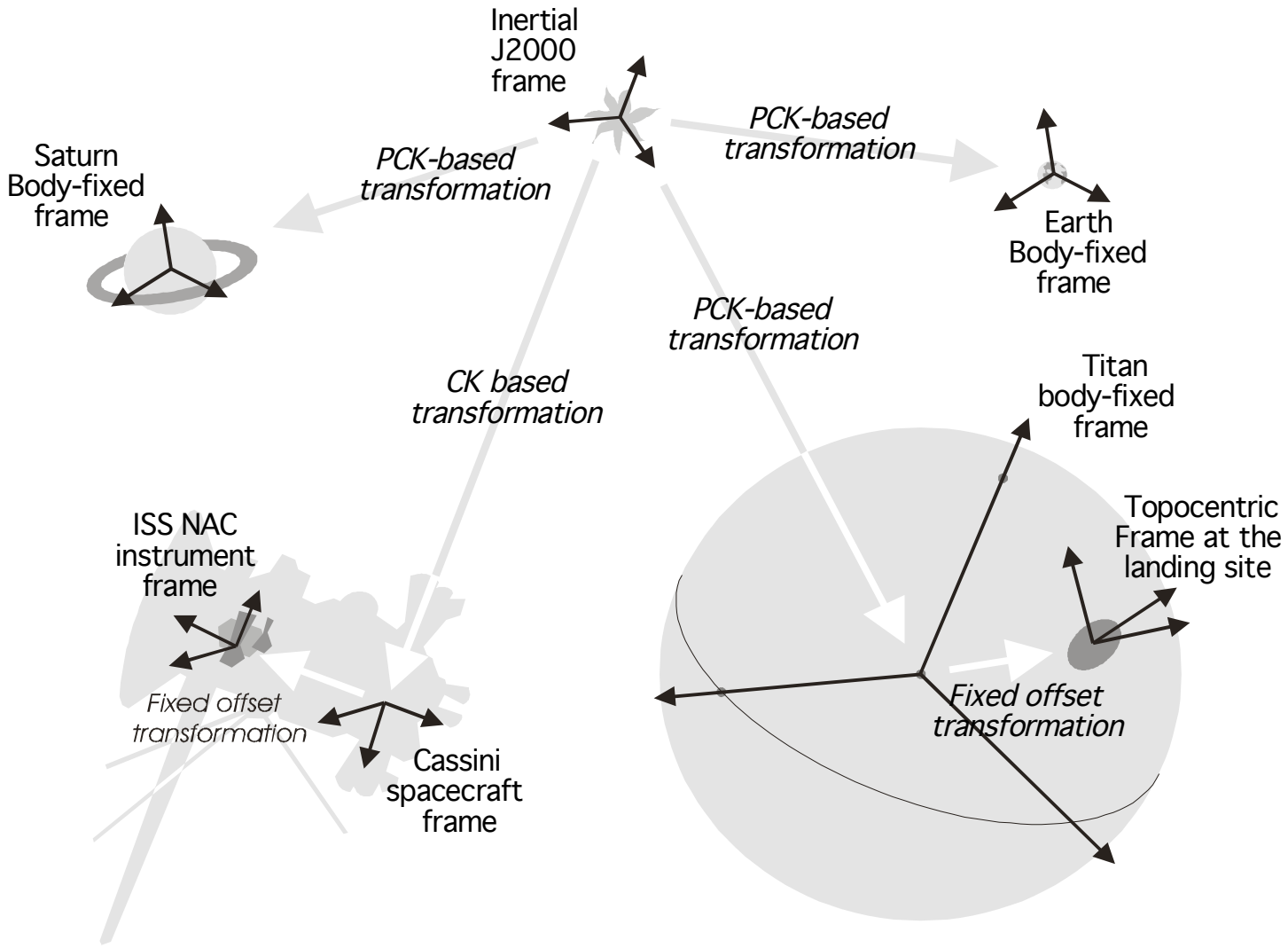
What does the FRAMES subsystem do?

- It establishes relationships between reference frames used in geometry computations -- it "chains frames together."
- It connects frames with sources of their orientation specifications.
- Based on these relationships and orientation source information, it allows SPICE software to compute transformations between neighboring frames in the "chain," and to combine these transformations in the right order, thus providing an ability to compute orientation of any frame in the chain with respect to any other frame in the chain at any time. (*)

(*) If the complete set of underlying SPICE data needed to compute the transformation is available.

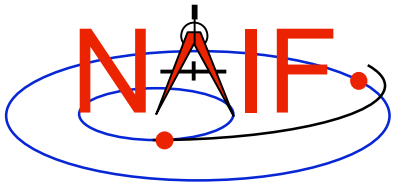
Sample Frame Tree and Chains

Navigation and Ancillary Information Facility



Frame Classes

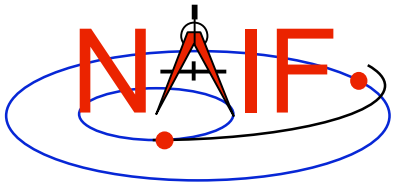
Frame class	***Examples***
Inertial	• Earth Equator/Equinox of Epoch (J2000, …) • Planet Equator/Equinox of Epoch (MARSIAU, ...) • Ecliptic of Epoch (ECLIPJ2000, ...)
Body-fixed	• Solar system body IAU frames (IAU_SATURN, …) • High accuracy Earth frames (ITRF93, …) • High accuracy Moon frames (MOON_PA, MOON_ME)
CK-based	• Spacecraft (CASSINI_SC_BUS, …) • Moving parts of an instrument (MPL_RA_JOINT1, ...)
Fixed Offset	• Instrument mounting alignment (CASSINI_ISS_NAC, …) • Topocentric (DSS-14_TOPO, …)
Dynamic	• See the Dynamic Frames tutorial



Frames Class Specifications

Navigation and Ancillary Information Facility

Frame class	*Frame Defined in*	*Orientation provided in*
Inertial	Toolkit	Toolkit
Bodyfixed	Toolkit or FK	PCK
CK based	FK	CK
Fixed offset	FK	FK
Dynamic	FK	Toolkit, or computed using FK, SPK, CK, and/or PCK



FRAMES Subsystem Interfaces

Navigation and Ancillary Information Facility

SXFORM/PXFORM returns state or position transformation matrix

```
CALL SXFORM ( 'FROM_FRAME_NAME', 'TO_FRAME_NAME', ET, MAT6x6 )
CALL PXFORM ( 'FROM_FRAME_NAME', 'TO_FRAME_NAME', ET, MAT3X3 )
```

SPKEZR/SPKPOS returns state or position vector in specified frame

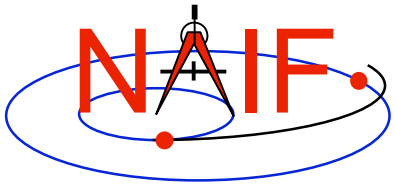
```
CALL SPKEZR ( BOD, ET, 'FRAME_NAME', CORR, OBS, STATE,  LT )
CALL SPKPOS ( BOD, ET, 'FRAME_NAME', CORR, OBS, POSITN, LT )
```

The above are FORTRAN examples, using SPICELIB modules.
The same interfaces exist for C, using CSPICE modules, and for Icy and Mice.

What are the Names of Frames?

Navigation and Ancillary Information Facility

- Refer to “NAIF IDs” Tutorial for an introduction to reference frame names and IDs
- Refer to FRAMES.REQ for the list of NAIF “built in” (hard coded) inertial and body-fixed frames
- Refer to a project’s Frames Kernel (FK) file for a list of frames defined for the spacecraft, its subsystems and instruments
- Refer to an earth stations FK for a list of frames defined for the DSN and other stations
- Refer to the moon FKs for descriptions of the body-fixed frames defined for the moon

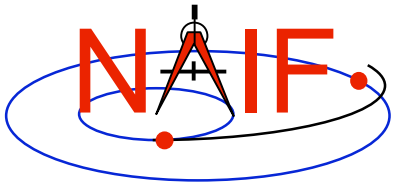


Frames Kernel File

Navigation and Ancillary Information Facility

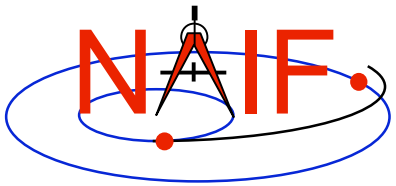
- Uses the SPICE text kernel file standards
- Loaded using the FURNSH routine
- Usually contains comprehensive information about the defined frames in the text section(s) of the file
- Contains frame definition information consisting of a set of keywords in the data sections of the file. For example, the frame for the scanner assembly of the ASPERA instrument on the Mars Express spacecraft is defined as:

```
\begindata
   FRAME_MEX_ASPERA_SAF            = -41111
   FRAME_-41111_NAME               = 'MEX_ASPERA_SAF'
   FRAME_-41111_CLASS              =  3
   FRAME_-41111_CLASS_ID           = -41111
   FRAME_-41111_CENTER             = -41
   CK_-41111_SCLK                  = -41
   CK_-41111_SPK                   = -41
\begintext
```



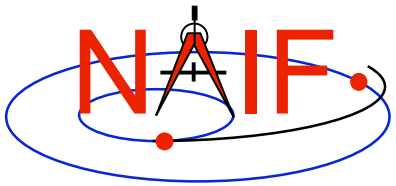
Frame Definition Details - 1

- Frame ID is an integer number used by the SPICE system as a “handle” in buffering and retrieving various parameters associated with a frame. In an FK it “glues” together the keywords defining the frame.
- FRAME…NAME assigns a name to the Frame ID code.
- FRAME…CLASS specifies the method by which the frame is related to some other frame — inertial, PCK, CK, TK, dynamic.



Frame Definition Details - 2

- **FRAME…CLASS_ID is the number that connects a frame with the orientation data for it.**
  - **For body-fixed frames CLASS_ID is the ID of the natural body. It is used as input to PCK routines called by the Frame subsystem to compute orientation of the frame.**
    - » **Frame ID and CLASS_ID are not the same for the body-fixed frames defined in the Toolkit but they can be the same for frames defined in FK files.**
  - **For CK-based frames CLASS_ID is the CK structure ID. It is used as input to CK routines called by the Frame subsystem to compute orientation of the frame.**
    - » **Normally CLASS_ID of a CK-based frame is the same as the frame ID.**
  - **For fixed offset and dynamic frames CLASS_ID is the ID that is used to retrieve the frame definition keywords.**
    - » **CLASS_ID of a fixed offset or dynamic frame is the same as the frame ID**
- **FRAME…CENTER specifies the ephemeris object at which the frame origin is located.**
- **CK…SCLK identifies the spacecraft clock associated with the CK structure ID. Used by the frames subsystem to convert ET to SCLK used to look up CK data.**
- **CK…SPK identifies the ephemeris object associated with the CK structure ID (not currently used; included for future use).**



CK-Based Frames "Must Know"

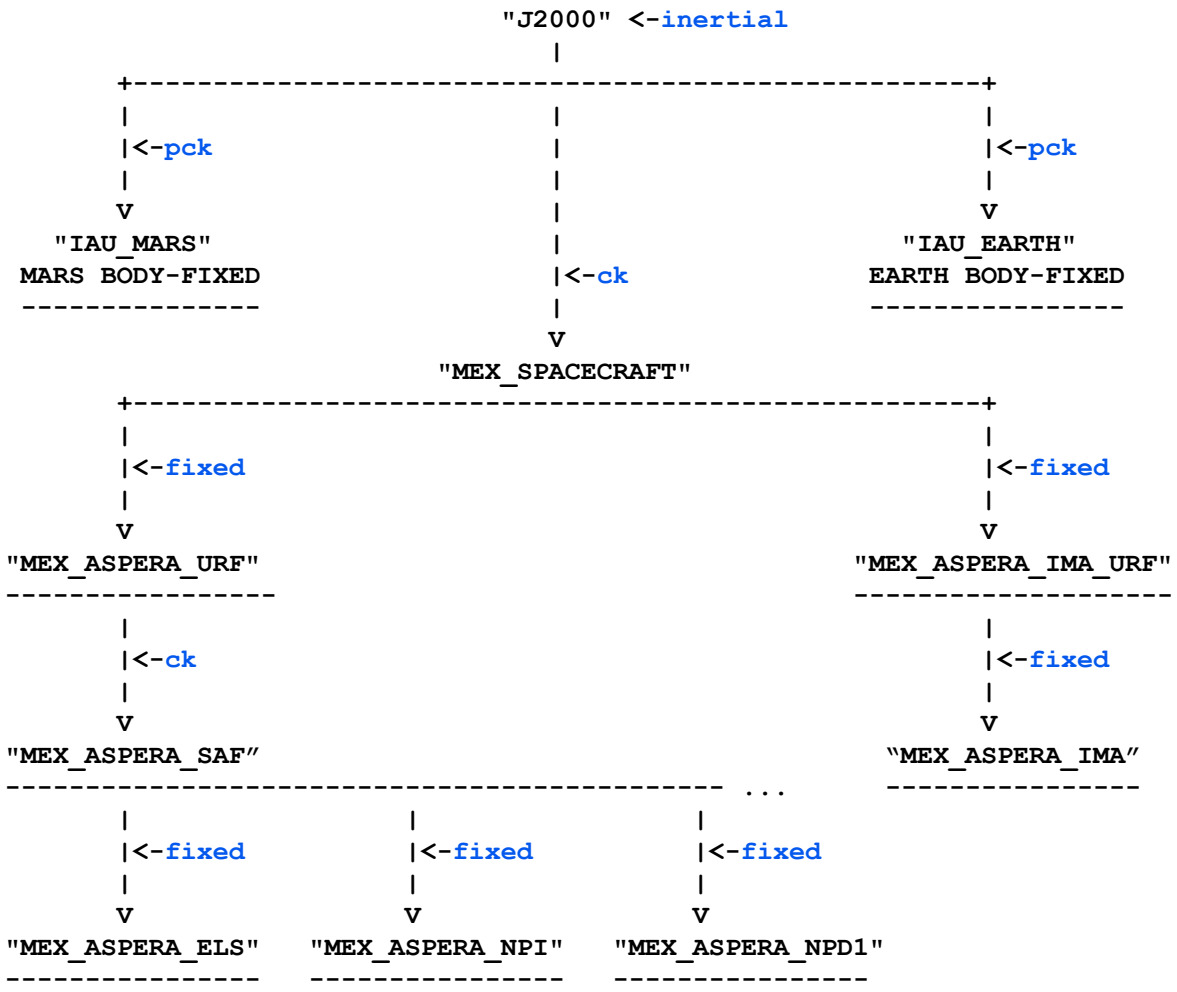
Navigation and Ancillary Information Facility

These are VERY IMPORTANT points you must understand!

- **The frames routines (SPKEZR, SPKPOS, SXFORM, PXFORM) all read CK files using tolerance = 0**
  - **For discrete CKs the orientation of a CK-based frame will be computed only if the time provided to a Frames routine exactly matches one of the times stored in the CK file; otherwise an error will be signaled.**
  - **For continuous CKs the orientation of a CK-based frame will be computed only if the time provided to a Frames routine falls within one of the interpolation intervals defined by the CK file; otherwise an error will be signaled.**
- **Using SPKEZR or SXFORM requires CKs with angular rates**
  - **Since these routines return a state vector (6x1) or state transformation matrix (6x6), angular rates must be present in the CK in order to compute vectors and matrices; if rates are not present, an error will be signaled.**
  - **SPKPOS and PXFORM, which return a position vector (3x1) and a position transformation matrix (3x3) respectively, can be used instead because they require only orientation data to be present in the CK.**
- **Ephemeris time input to Frames routines is converted to SCLK to access CKs**
  - **SCLK and LSK kernels must be loaded to support this conversion.**
  - **SCLK ID is specified in one of the CK frame definition keywords; if not, it's assumed to be the Frame ID divided by a 1000.**

Frame Tree Example: ASPERA Instrument on Mars Express

Navigation and Ancillary Information Facility



Blue text indicates frame class