The SunSPICE Ephemeris Package for Solar Missions

William Thompson
May 7, 2018

1 Introduction

SPICE is a software package developed by the Jet Propulsion Laboratory for handling spacecraft ephemeris and pointing information, with particular emphasis on interplanetary missions. The letters in the name stand for the different kinds of data which the SPICE package handles, which are:

S Spacecraft ephemerides

P Planetary ephemerides (and other solar system bodies)

I Instrument descriptions

C Pointing "C-matrices"

E Events

SPICE is supported in a number of different programming environments, including C, Fortran, IDL, and MATLAB, along with user-community development efforts for other environments such as Python.

The NASA STEREO mission has made extensive use of SPICE in their mission operations and data analysis, and a significant amount of IDL software has been developed in SolarSoft for manipulating SPICE data files (known as "kernels") for the STEREO mission. Even though these routines were written for STEREO, they've also been used for other missions, such as SOHO and Solar Orbiter. The only caveat with other missions is that it's the user's responsibility to make sure that the appropriate ephemerides are loaded, while the software takes care of this automatically for STEREO.

SunSPICE is a reorganizing of the original STEREO software to open it up to other missions, particularly in regard to automatic kernel management. The basic structure of the software is the same, except with new routine names. For example, instead of calling `GET_STEREO_COORD` to get the position of a spacecraft at a particular time, one would instead call `GET_SUNSPICE_COORD`.

For backwards compatibility, the original STEREO software calls remain. After the SunSPICE package has been adequately distributed throughout the community, the STEREO routines will be rewritten to implement SunSPICE to do the actual work.

2 Software organization

The original SolarSoft implementation of SPICE for STEREO was all in one location, in the `$SSW/stereo/gen` tree (with some data files in SSWDB). SunSPICE, on the other hand, is distributed throughout the SSW tree. The main routines are in the `$SSW/packages/sunspice` tree, along with the generic data files shared by all the missions. Then, the software and data needed to implement SunSPICE for a specific mission are in the "gen" part of the tree for that mission.

```
$SSW---packages---sunspice---data---dos
     |                     |      \-earth
     |                     |-icy---...     (executables)
     |                     \-idl
     |-soho---gen---data---spice---dos
     |            \-idl---spice
     \-stereo---gen---data---spice---ah---ahead
                    |              |    \-behind
                    |              |-depm---ahead
                    |              |      \-behind
                    |              |-dos
                    |              |-epm---ahead
                    |              |     \-behind
                    |              |-other
                    |              \-sclk---ahead---dos
                    |                     \-behind---dos
                    \-idl---spice---legacy

$SDB---soho---gen---spice
     \-stereo---gen---spice---ah---ahead
                                \-behind
```

Figure 1: Directory structure for the SunSPICE components in the SolarSoft ($SSW) and SolarSoft Database ($SDB) trees. Some subdirectories associated with the STEREO pre-launch mission simulations have been omitted.

Thus, the STEREO software and data are in the same place as before, with the original routines in a "legacy" subdirectory. The full SunSPICE-related directory tree at the time of writing is illustrated in Figure 1.

In practical terms, what this means is that in order to use the SunSPICE routines for one of the supported missions—including automatic kernel management—you must specify at least one instrument from that mission within SolarSoft, either through the `$SSW_INSTR` environment variable, or by calling `ssw_path`. In addition, the SunSPICE package must also be loaded.

As new missions, such as Solar Orbiter and Solar Probe Plus, are added to the SunSPICE system, the software for those missions will appear in their respective parts of the SolarSoft tree, and will be maintained by those missions.

3 Spacecraft and target IDs

Spacecraft and solar system objects are generally identified by name. For example, if you wanted to know the position of Earth at a particular time, you would type in a command like the following:

```
COORD = GET_SUNSPICE_COORD(DATE, 'EARTH')
```

Case is not important. If you knew the NAIF ID number, you could pass that in instead. Table 1

Table 1: NAIF ID names and numbers for solar system objects which are loaded by default. Character string names are case-insensitive.

MERCURY BARYCENTER	1	SATURN BARYCENTER	6	MERCURY	199
VENUS BARYCENTER	2	URANUS BARYCENTER	7	VENUS	299
EARTH BARYCENTER	3	NEPTUNE BARYCENTER	8	MOON	301
MARS BARYCENTER	4	PLUTO BARYCENTER	9	EARTH	399
JUPITER BARYCENTER	5	SUN	10	MARS	499

SOLAR SYSTEM BARYCENTER (or SSB)	0
EARTH MOON BARYCENTER (or EMB)	3

shows the names and ID numbers for the solar system objects which are loaded automatically. Note that for the outer solar system objects, you have to add "BARYCENTER" to the name, e.g.

```
COORD = GET_SUNSPICE_COORD(DATE, 'JUPITER BARYCENTER')
```

If you want the position of the planet itself (which would include the influences of the moons on the planet's motions) you would need to load another SPICE kernel specific to that planetary system, via the low-level routine `CSPICE_FURNSH`.

When it comes to the spacecraft names, SPICE can be a bit picky about how the name is expressed. To get around this limitation, SunSPICE passes all names through the routine `PARSE_SUNSPICE_NAME`, where a larger number of possible names and abbreviations for the spacecraft can be used. For example, the STEREO Ahead spacecraft can be specified as "STEREO AHEAD", "STEREO-A", "AHEAD", or "STA", among other variations. See the documentation for `PARSE_SUNSPICE_NAME` for more information about what names can be passed in. Any unrecognized names (e.g. "Earth") are simply passed through `PARSE_SUNSPICE_NAME` unchanged.

4 SunSPICE routines

In this section we describe the routines which make up the multimission part of SunSPICE which resides in the `$SSW/gen/idl/spice` tree. Users familiar with the legacy STEREO software will find that these routines work in the same way, e.g. `GET_SUNSPICE_COORD` has the same calling sequence as the older `GET_STEREO_COORD`.

4.1 Basic routines

These are the basic routines which are designed to be called by the typical user.

GET_SUNSPICE_COORD: Returns the x, y, z (and v_x, v_y, v_z) coordinates of a spacecraft or solar system body in one of the supported coordinate systems described in Appendix A. The default coordinate system is Heliocentric Inertial (HCI).

CONVERT_SUNSPICE_COORD: Converts x, y, z positions (and v_x, v_y, v_z velocities) between coordinate systems.

GET_SUNSPICE_LONLAT: Returns the radial distance and angular longitude and latitude of a spacecraft or solar system body.

CONVERT_SUNSPICE_LONLAT: Converts longitude and latitude (and optionally radial distance) between coordinate systems.

GET_SUNSPICE_SEP_ANGLE: Returns the three-dimensional heliocentric separation angle between two bodies. (Note that this can be slightly different than the angular extent along the ecliptic plane.)

GET_SUNSPICE_CARR_ROT: Returns the decimal Carrington rotation number associated with disk center as seen from a spacecraft or solar system body.

GET_SUNSPICE_CARR_DATE: Returns the date corresponding to a decimal Carrington rotation number for a spacecraft or solar system body (inverse of `GET_SUNSPICE_CARR_ROT`).

GET_SUNSPICE_P0_ANGLE: Returns the solar P0 angle (between solar and celestial north) as seen from a spacecraft or other solar system body.

GET_SUNSPICE_HPC_POINT: Returns the spacecraft pointing in helioprojective-cartesian coordinates (θ_x, θ_y) as described in Thompson (2006), as well as the roll angle (ψ) as used in FITS files.

GET_SUNSPICE_ROLL: Returns the roll angle of a spacecraft in various coordinate systems, and optionally the yaw and pitch angles.

4.2 Lower-level routines

The following routines are not usually called directly by the user, but are called by the higher-level routines, or used for debugging.

CONVERT_SUNSPICE_GEO2MAG: Called from `CONVERT_SUNSPICE_COORD` to convert between the GEO and MAG coordinate systems.

CONVERT_SUNSPICE_GSE2GSM: Called from `CONVERT_SUNSPICE_COORD` to convert between the GSE and GSM coordinate systems.

CONVERT_SUNSPICE_GSE2SM: Called from `CONVERT_SUNSPICE_COORD` to convert between the GSE and SM coordinate systems.

GET_SUNSPICE_CMAT: Returns the 3×3 (or 6×6) transformation matrix to convert between instrument and standard coordinate frames. The default coordinate system is Radial-Tangential-Normal (RTN).

GET_SUNSPICE_IGRF_POLE: Returns Earth's magnetic pole position by date to support the calculation of magnetic coordinates.

GET_SUNSPICE_KERNEL: Returns the name of the kernel containing the ephemeris or attitude data for a given date and spacecraft.

GET_SUNSPICE_RANGE: Returns the range of coverage of a binary SPICE kernel.

LIST_SUNSPICE_KERNELS: Prints out or returns the filename of each loaded SPICE kernel.

LOAD_SUNSPICE_EARTH: Loads the SPICE Earth PCK kernels needed to support ITRF93 coordinates. These are currently used mainly for generating beacon station ephemerides for the STEREO mission. This routine depends on keeping the kernels in the directory `$SSW/gen/data/spice/earth` up-to-date with those on the NAIF server.

LOAD_SUNSPICE_GEN: Loads the generic SPICE kernels used by all the missions.

PARSE_SUNSPICE_NAME: As discussed in Section 3, this routine parses the names of supported spacecraft into forms compatible with the SPICE software.

REGISTER_SUNSPICE_DLM: Registers the SPICE/Icy DLM. Should be called at start-up.

TEST_SUNSPICE_DLM: Tests to see if the SPICE/Icy DLM is available.

UNLOAD_SUNSPICE_EARTH: Unloads the SPICE Earth PCK kernels previously loaded by `LOAD_SUNSPICE_EARTH`.

UNLOAD_SUNSPICE_GEN: Unloads the generic SPICE kernels previously loaded by `LOAD_SUNSPICE_GEN`.

4.3 Front-ends to mission routines

The following routines serve as front-ends to mission-specific routines which perform the actual work.

LOAD_SUNSPICE: Front-end to mission-specific software which loads the SPICE kernels to support a given mission.

LOAD_SUNSPICE_ATT: Front-end to mission-specific software which provides "on demand" loading of spacecraft attitude files in SPICE format.

UNLOAD_SUNSPICE: Front-end to mission-specific software to unload the SPICE kernels for a given mission, both for orbit and attitude.

GET_SUNSPICE_CONIC: Front-end to mission-specific software which returns the heliocentric conic parameters describing the orbit of a spacecraft at the end of the available predictive ephemerides. These parameters allow the orbit to be projected beyond the end of the SPICE kernels, assuming a simple Keplerian orbit. Currently, the only mission for which this is considered relevant is STEREO.

5 Mission-specific software and data

5.1 Mission-specifc software

Most of the actual calculations within SunSPICE are done by the general purpose routines described in Section 4. Generally speaking, only a small number of routines are needed within a mission's branch in SolarSoft, mainly to manage the loading and unloading of mission kernels. It's instructive to look at the routines in the STEREO tree, as that mission uses the full capabilities of SunSPICE. The directory `$SSW/stereo/gen/idl/spice` contains the following routines which are needed to support the STEREO mission within SunSPICE.

LOAD_SUNSPICE_STEREO: Called from `LOAD_SUNSPICE` to load the kernels for the STEREO mission.

LOAD_SUNSPICE_ATT_STEREO: Called from `LOAD_SUNSPICE_ATT` to manage the STEREO attitude kernels which are loaded, based on the dates being addressed.

UNLOAD_SUNSPICE_STEREO: Called from `UNLOAD_SUNSPICE` to unload all the STEREO kernel files which have been loaded.

GET_SUNSPICE_CONIC_STEREO: Called from `GET_SUNSPICE_CONIC` to return the conic parameters for one of the two STEREO spacecraft to estimate the spacecraft trajectory past the end of the SPICE kernels.

On the other hand, only two routines are needed for the SOHO mission, `LOAD_SUNSPICE_SOHO` and `UNLOAD_SUNSPICE_SOHO`, since there are no SPICE attitude kernels for SOHO, and conic projections are not relevant. Note that the naming convention is that the mission-specific routines are formed by appending the mission name (or "nickname") to the name of the front-end SunSPICE routine that calls it.

Other routines can also appear in the mission tree to satisfy specific requirements for that mission. For example, the STEREO tree also contains the routine `STEREO_COORD_INFO` which prints out a summary of the ephemeris data for the mission for any given date.

5.2 Mission-specific data files

It is the responsibility of each mission to organize its own data files for SunSPICE, and reflect that organization in the SunSPICE routines in their part of SolarSoft (Section 5.1. As an example, Figure 1 shows how the data are organized for the STEREO mission, which has the following subtrees:

ah	Attitude history files
depm	Definitive ephemerides
epm	Predictive ephemerides
sclk	Spacecraft clock files

Other missions may organize their data in a different fashion. Note the subdirectories labelled "dos". While most SPICE kernels are binary, and cross-platform compatible, some kernels are text files. It's a limitation of SPICE that it can only read text files in the proper format for that computer. Unix-based computers (including Mac OSX) end each line with a linefeed character (`^J`), while Windows/DOS computers use carriage-return, line-feed (`^M^J`). The "dos" subdirectories are the way that the STEREO project chose to separate the Windows/DOS versions of the text files from their Unix counterparts.

```
\begindata

        FRAME_SOHOHGRTN              =  1810460
        FRAME_1810460_NAME           = 'SOHOHGRTN'
        FRAME_1810460_CLASS          =  5
        FRAME_1810460_CLASS_ID       =  1810460
        FRAME_1810460_CENTER         =  10
        FRAME_1810460_RELATIVE       = 'J2000'
        FRAME_1810460_DEF_STYLE      = 'PARAMETERIZED'
        FRAME_1810460_FAMILY         = 'TWO-VECTOR'
        FRAME_1810460_PRI_AXIS       = 'X'
        FRAME_1810460_PRI_VECTOR_DEF = 'OBSERVER_TARGET_POSITION'
        FRAME_1810460_PRI_OBSERVER   = 'SUN'
        FRAME_1810460_PRI_TARGET     = 'SOHO'
        FRAME_1810460_PRI_ABCORR     = 'NONE'
        FRAME_1810460_PRI_FRAME      = 'IAU_SUN'
        FRAME_1810460_SEC_AXIS       = 'Z'
        FRAME_1810460_SEC_VECTOR_DEF = 'CONSTANT'
        FRAME_1810460_SEC_FRAME      = 'IAU_SUN'
        FRAME_1810460_SEC_SPEC       = 'RECTANGULAR'
        FRAME_1810460_SEC_VECTOR      = ( 0, 0, 1 )

\begintext
```

Figure 2: Frame definition for HGRTN coordinates for the SOHO spacecraft.

One SPICE kernel that each mission should have is a frames definition file to implement the HGRTN and HERTN alignment frames for that mission, which is used to support the RTN, HGRTN, HERTN, and HPC coordinate systems (see Appendix A). For example, the directory `$SSW/soho/gen/data/spice` contains the file `soho_rtn.tf`. Figure 2 shows the lines from this file which define the HGRTN coordinate frame for SOHO. The frame name (SOHOHGRTN) and ID number (1810460) must be unique, with the ID number being between 1400000 and 2000000. See `soho_rtn.tf` and the corresponding `stereo_rtn.tf` for STEREO for more examples of mission-specific telemetry frame definitions.

5.3 Adding new missions to SunSPICE

When new missions are added to SunSPICE, along with writing the necessary mission-specific routines (Section 5.1) some of the basic routines in Section 4 will also need to be modified to incorporate the new routines in the overall SolarSoft package.

First of all, the routine `PARSE_SUNSPICE_NAME` must be updated to recognize the mission name, along with any desired abbreviations or "nicknames", and translate it into the proper NAIF ID code.

In some cases, all that needs to be done is to add the call to the appropriate mission-specific routine (e.g. `LOAD_SUNSPICE_SPP` for Solar Probe Plus) to the case statement in the front-end routine. When applicable, this is done in the following routines:

```
LOAD_SUNSPICE
LOAD_SUNSPICE_ATT
UNLOAD_SUNSPICE
GET_SUNSPICE_CONIC
```

Some routines need to know the name of the HGRTN or HERTN coordinate frame for the mission. Again, this is a simple addition to some case statements in the following routines:

```
CONVERT_SUNSPICE_COORD
GET_SUNSPICE_CMAT
GET_SUNSPICE_COORD
```

These routines also contain code to implement the two STEREO-specific coordinate systems "SCI" and "STPLN" described in Appendix A, while `GET_SUNSPICE_HPC_POINT` and `GET_SUNSPICE_ROLL` were modified to support the STEREO-specific keyword `/POST_CONJUNCTION`. Because not all spacecraft use the same definitions for the body axes, `GET_SUNSPICE_HPC_POINT` and `GET_SUNSPICE_ROLL` may also need to be updated to reflect new spacecraft.

References

Fränz, M. and Harper, D. (2002). Heliospheric coordinate systems. *Planetary and Space Science*, **50**, 217–239.

Hapgood, M. A. (1992). Space physics coordinate transformation: A user guide. *Planetary and Space Science*, **40**, 711–717.

Russell, C. T. (1971). Geophysical coordinate transformations. *Cosmic Electrodynamics*, **2**, 184–196.

Thompson, W. T. (2006). Coordinate systems for solar image data. *Astron. Astrophys.*, **449**, 791–803.

Appendices

A Coordinate systems

The coordinate systems used by the SunSPICE software are based on Russell (1971), Hapgood (1992), and Fränz and Harper (2002), with special emphasis on the last. The various coordinate systems are summarized below. Two axes are chosen to define each coordinate system, and the missing axis satisfies the right-hand rule.

GEI: Geocentric Equatorial Inertial. The X axis points toward the first point of Aries (i.e. the vernal equinox), and the Z axis is aligned with the geographic north pole. When expressed in terms of longitude and latitude, this is the well known celestial coordinate system of right ascension and declination. This is realized with the J2000 ecliptic.

GEO: Geographic. X is the intersection of the Greenwich meridian and the geographic equator, and Z is the geographic North Pole. This is usually expressed as geographic longitude and latitude, where the Earth's oblateness is taken into account. The SunSPICE software supports both geocentric (default) and geographic forms for the GEO and GEI coordinate systems, though only geocentric is currently supported for coordinate conversions.

GSE: Geocentric Solar Ecliptic. X is the Earth-Sun line, and Z is aligned with the north pole for the ecliptic of date.

GAE: Geocentric Aries Ecliptic. X axis points toward the first point of Aries, and Z is aligned with the ecliptic north pole, based on either the J2000 ecliptic (default), or the ecliptic of date.

MAG: Geomagnetic. Z is Earth's north dipole axis, and Y is the intersection between the geographic equator, and the geographic meridan 90° east of the meridian containing the dipole axis. In other words, the geomagnetic meridian containing X also passes through the south geographic pole. The dipole axis is defined as the centered dipole approximation to the International Geomagnetic Reference Frame (IGRF) model, as described in Fränz and Harper (2002).

GSM: Geocentric Solar Magnetospheric. X is the Earth-Sun line, and Z is the projection of the north dipole axis.

SM: Solar Magnetic. Z is Earth's north dipole axis, and X is the projection of the Earth-Sun line. Although not evident in the name, this is a geocentric coordinate system. Users should be aware that the acronym SM has sometimes in the past been used for what is now referred to as GSM.

HCI: Heliocentric Inertial. Z is the solar north rotational axis, and X is the solar ascending node on the J2000 ecliptic.

HAE: Heliocentric Aries Ecliptic. X is in the direction of the first point of Aries, and Z is the ecliptic North Pole. This can be realized with either the J2000 ecliptic (default), or the ecliptic of date.

HEE: Heliocentric Earth Ecliptic. X is the Sun-Earth line, and Z is the north pole for the ecliptic of date.

HEEQ: Heliocentric Earth Equatorial. Z is the solar rotation axis, and X is in the plane containing the Z axis and Earth, at the intersection of the solar central meridian, and the heliographic equator. When converted to longitude and latitude, this is known as Stonyhurst heliographic coordinates. In FITS files, this coordinate system is abbreviated as "HEQ", so that variation is also recognized by the software.

Carrington heliographic. Z is the solar rotational axis. The X axis is at the intersection of the Carrington prime meridian and the heliographic equator. The coordinate system rotates with a sidereal period of 25.38 days. This coordinate system is usually expressed in terms of longitude and latitude.

HGRTN/RTN: Radial-Tangential-Normal. X axis points from Sun center to the spacecraft, and the Y axis is the cross product of the solar rotational axis and X, and lies in the solar equatorial plane (towards the West limb). Spacecraft-specific coordinate frames must be loaded to utilize these coordinate systems; currently implemented for SOHO and STEREO. When the Sun is used as the origin, the designation is HGRTN—with the spacecraft as origin, it's simply RTN.

GRTN: Geocentric Radial-Tangential-Normal. Special case of RTN where Earth is used in place of the spacecraft, and implemented through the GEORTN reference frame. This is used as the default when requesting RTN coordinates as seen from a spacecraft which doesn't have it's own RTN frame definition, e.g. in low Earth orbit.

HPC: Helioprojective-cartesian. RTN coordinates are closely related to helioprojective-cartesian coordinates (Thompson, 2006)) via the following relationships:

$$
\begin{aligned}
X_{\mathrm{HPC}} &= Y_{\mathrm{RTN}} \\
Y_{\mathrm{HPC}} &= Z_{\mathrm{RTN}} \\
Z_{\mathrm{HPC}} &= X_{\mathrm{RTN}}
\end{aligned}
$$

The routines associated with pointing also support "HPC" as a coordinate specification.

HERTN: Helioecliptic Radial-Tangential-Normal. Equivalent to HGRTN, except that it is defined relative to ecliptic north rather than solar north. X axis points from Sun center to the spacecraft, and the Y axis is the cross product of the ecliptic north axis and X, and lies in the ecliptic plane (towards the West limb).

Some missions have coordinate systems which are specific to that mission. The following coordinate systems have been defined for the STEREO mission:

SCI: STEREO Science Pointing. X axis points at the center of the Sun, while the $+Z$ axis points away from Earth such that the X-Z plane contains the observatory, Sun, and Earth. The origin is at the observatory center of mass. Defines the nominal pointing reference frame used by the flight operations team during science operations.

STPLN: STEREO Mission Plane. X axis points at the center of the Sun, while the Y axis lies in the plane containing the Sun and both STEREO spacecraft, with $+Y$ towards the West limb. This is the natural frame for stereoscopy with the two spacecraft.

B Reference frames

SPICE incorporates two basic kinds of reference frames: inertial and body-fixed frames. Examples of inertial frames include:

J2000: Celestial reference frame, with Z aligned with Earth's rotational axis, and X aligned with the direction of the vernal equinox, at the dynamical epoch of J2000.

ECLIPJ2000: Ecliptic coordinates based upon the J2000 frame.

among others. Note that these inertial frames are based on the Earth's rotational axis and ecliptic for a specific date, and do not take precession into account. This is discussed in more detail in Appendix C.

Body-fixed frames include geographic and Carrington heliographic, as well as spacecraft and instrument-based frames. The reference frames for geographic and Carrington heliographic coordinates are IAU_EARTH and IAU_SUN respectively.

Versions N0058 and later of the SPICE toolkit also allows one to define additional non-inertial reference frames based on certain kinds of parameters. This facility has been used to implement many of the coordinate systems described in Section A, in the frame definition file `heliospheric.tf`.

C Precession

Internally, SPICE works within the J2000 inertial reference frame. However, Earth's rotational axis slowly precesses around the pole of the ecliptic at a rate of about 50″ per year, with a period of about 26,000 years. Also, the plane of the ecliptic itself precesses at a rate of about 47″ per century. There may be situations where these effects need to be taken into account.

Fränz and Harper (2002), discuss the effects of precession on the standard heliospheric coordinate systems, and distinguish between three different versions of the Geocentric Equatorial Inertial system:

$\mathrm{GEI_{J2000}}$	Earth mean equator at J2000.
$\mathrm{GEI_D}$	Earth mean equator of date, i.e. corrected for precession.
$\mathrm{GEI_T}$	Earth true equator of date, i.e. corrected for precession and nutation.

SPICE returns $\mathrm{GEI_{J2000}}$. On the other hand, geographic coordinates (GEO) are defined relative to the equator of date, which is implemented in SunSPICE through the IAU_EARTH frame by default, which corrects for precession but not other effects. The optional `/ITRF93` keyword informs the software to call `LOAD_SUNSPICE_EARTH` to load in more accurate and up-to-date Earth rotation data.

Similarly, Fränz and Harper distinguish between two versions of Heliocentric Aries Ecliptic:

$\mathrm{HAE_{J2000}}$	Earth mean ecliptic at J2000.
$\mathrm{HAE_D}$	Earth mean ecliptic of date.

Again, SPICE natively returns HAE_{J2000}. However, the other ecliptic systems, GSE and HEE, are explicitly defined by Fränz and Harper in terms of the mean ecliptic of date, while SPICE wants to work within the J2000 reference frame. The frame definition file `heliospheric.tf` defines the ECLIPDATE frame to implement the ecliptic of date.