An Overview of Reference Frames and Coordinate Systems in the SPICE Context

June 2019

Purpose of this Tutorial

- **This tutorial provides an overview of reference frames and coordinate systems**
  - It contains conventions specific to SPICE.
- **Details about the SPICE Frames Subsystem are found in other tutorials and one document:**
  - FK (tutorial)
  - Using Frames (tutorial)
  - Dynamic Frames (advanced tutorial)
  - Frames Required Reading (technical reference)
- **Details about SPICE coordinate systems are found in API module headers for coordinate conversion routines.**

A Challenge

- **Next to "time," the topics of reference frames and coordinate systems present some of the largest challenges to documenting and understanding observation geometry. Contributing factors are …**
  - **differences in definitions, lack of concise definitions, and special cases**
  - **evolution of the frames subsystem within SPICE**
  - **the substantial frames management capabilities within SPICE**
- **NAIF hopes this tutorial will provide some clarity on these subjects within the SPICE context.**
  - **Definitions and terminology used herein may not be consistent with those found elsewhere.**

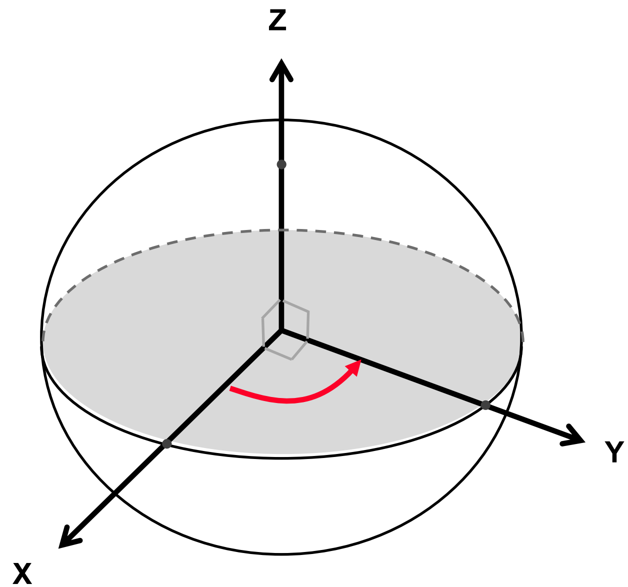
SPICE Definitions

- The definitions below are used within SPICE.
- A reference frame (or simply "frame") is specified by an ordered set of three mutually orthogonal, possibly time dependent, unit-length direction vectors.
  - A reference frame has an associated center.
  - In some documentation external to SPICE, this is called a "coordinate frame."
- A coordinate system specifies a mechanism for locating points within a reference frame.
- When producing or using state (position and velocity) or orientation (pointing) data, one needs to understand both the reference frame and the coordinate system being used.

Reference Frames

Reference Frame Conventions

- **All reference frames used within SPICE are right handed: this means X cross Y = Z**



Reference Frame Center

- **A reference frame's center must be a SPICE ephemeris object whose location is coincident with the origin (0, 0, 0) of the frame.**
  - **The center of any inertial frame is ALWAYS the solar system barycenter.***
  - **The center of a body-fixed frame is the center of the body.**
    - **"Body" means a natural body: sun, planet, satellite, comet, asteroid.**
    - **The location of the "body" center is specified using an SPK file.**
  - **The center of a topocentric, spacecraft or instrument frame is also an object for which the location is specified by an SPK file.**

- **A frame's center may play a role in specification of states.**
  - **The location of the origin cancels out when doing vector subtraction, but the center is used in computing light time to the center of any non-inertial frame being used**

*True even for inertial frames associated with accelerated bodies, such as the MARSIAU frame.

Types of Reference Frames - 1

- **Inertial**
  - Non-rotating with respect to stars
  - Non-accelerating origin
    - » Velocity is typically non-zero, but acceleration is negligible
  - Examples:
    - » J2000
    - » ECLIPJ2000

Types of Reference Frames - 2

- **Non-Inertial**
  - Accelerating, including by rotation
  - Examples
    - » Body-fixed
      - Associated with a natural body (e.g. planets, satellites)
    - » Topocentric
      - Associated with an object on or near the surface of a natural body (e.g. DSN station, rover)
    - » Spacecraft
      - Associated with the main spacecraft structure (the "bus")
    - » Instrument
      - One or more frames are usually associated with each instrument
      - Also applicable to a spacecraft antenna, solar array, etc.
    - » Dynamic
      - A special family of frames unique to SPICE
      - These have time-dependent orientation
        - But this category does not include frames for which the orientation is provided using a C-kernel (CK) or a PC-kernel (PCK)

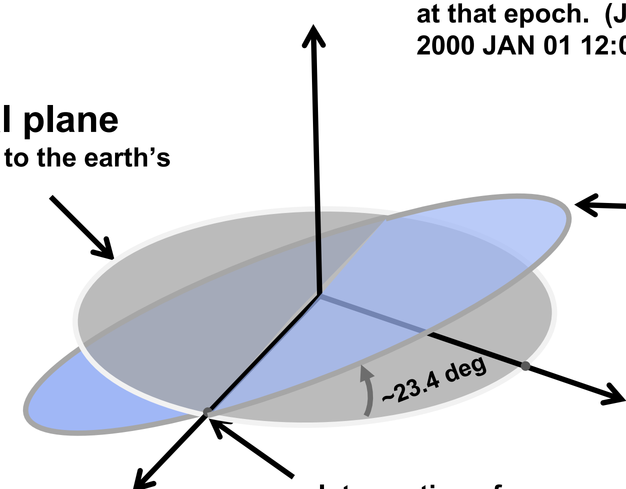
CK ≈ spacecraft orientation; PCK ≈ natural body orientation

The J2000 Inertial Frame

- **The J2000* (aka EME2000) frame definition is based on the earth's equator and equinox, determined from observations of planetary motions, plus other data.**

Z is normal to the mean equator of date at epoch J2000 TDB, which is approximately Earth's spin axis orientation at that epoch. (J2000 TDB is 2000 JAN 01 12:00:00 TDB, or JD 2451545.0 TDB).

Equatorial plane
Plane normal to the earth's spin axis, Z



Ecliptic plane
Plane defined by movement of the earth around the sun

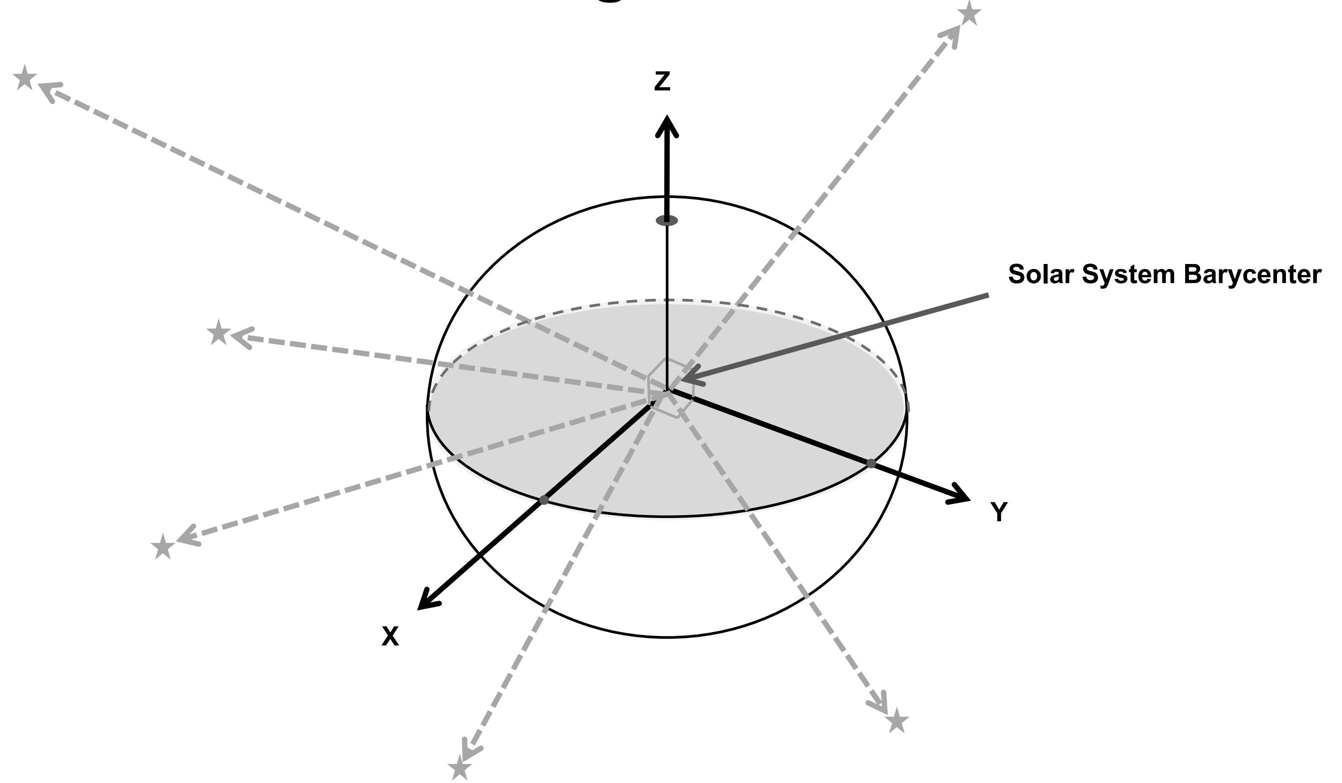
Y = Z cross X

Intersection of equatorial and ecliptic planes, called vernal equinox

*Caution: The name "J2000" is also used to refer to the zero epoch of the ephemeris time system (ET, also known as TDB).

The ICRF Inertial Frame

- **The ICRF* frame is defined by the adopted locations of 295 extragalactic radio sources.**



ICRF = International Celestial Reference Frame

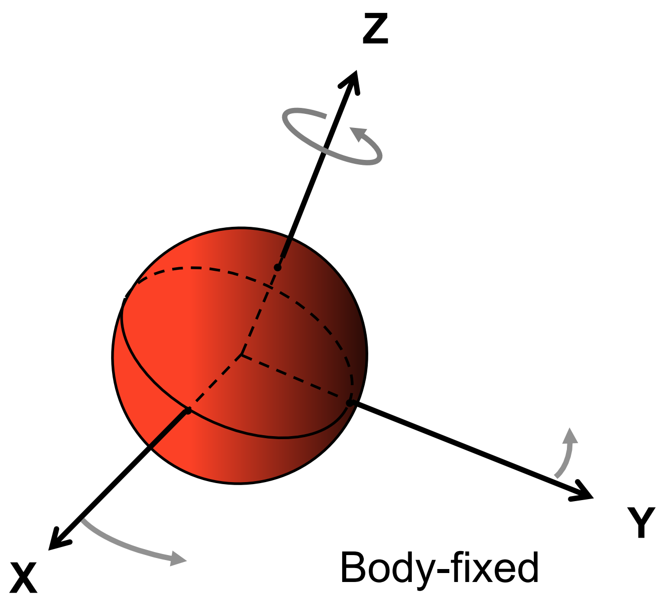
The ICRF is managed by the International Earth Rotation Service (IERS)

J2000 versus ICRF

- **The realization of ICRF was made to coincide almost exactly with the J2000 frame.**
  - **The difference is very small–a rotation of less than 0.1 arc second.**
  - **These two frames are considered the same in SPICE.**
    - » **In reality, any SPICE data said to be referenced to the J2000 frame are actually referenced to the ICRF frame.**
    - » **For historical and backwards compatibility reasons, only the name "J2000" is recognized by SPICE software as a frame name–not "ICRF."**
  - **No transformation is required to convert SPICE state vectors or orientation data from the J2000 frame to the ICRF.**

Body-fixed Frames

- **Body-fixed frames are tied to a named body and rotate with it**
  - Specifications for the most common body-fixed frames, those for the sun, the planets, many satellites, and a few asteroids and comets, are hard-coded in SPICE software
    - » Frame name style is "IAU_body name"
      - Examples: IAU_MARS, IAU_SATURN
    - » To see all such names, see:
      - Frames Required Reading document, or
      - Latest generic PCK file
  - The rotation state (the orientation at time *T*) is usually determined using a SPICE text PCK containing data published by the IAU
    - » The earth and moon are special cases!
      - IAU_EARTH and IAU_MOON both exist but generally should NOT be used
      - See the SPICE tutorial named "lunar-earth_pck-fk" for the best frames to be used for those bodies
  - On very rare occasions a CK is used to provide a body's rotation state



A Caution for Mars

- **The body-fixed frame for Mars is named IAU_MARS**
  - This follows the SPICE naming standard for such frames

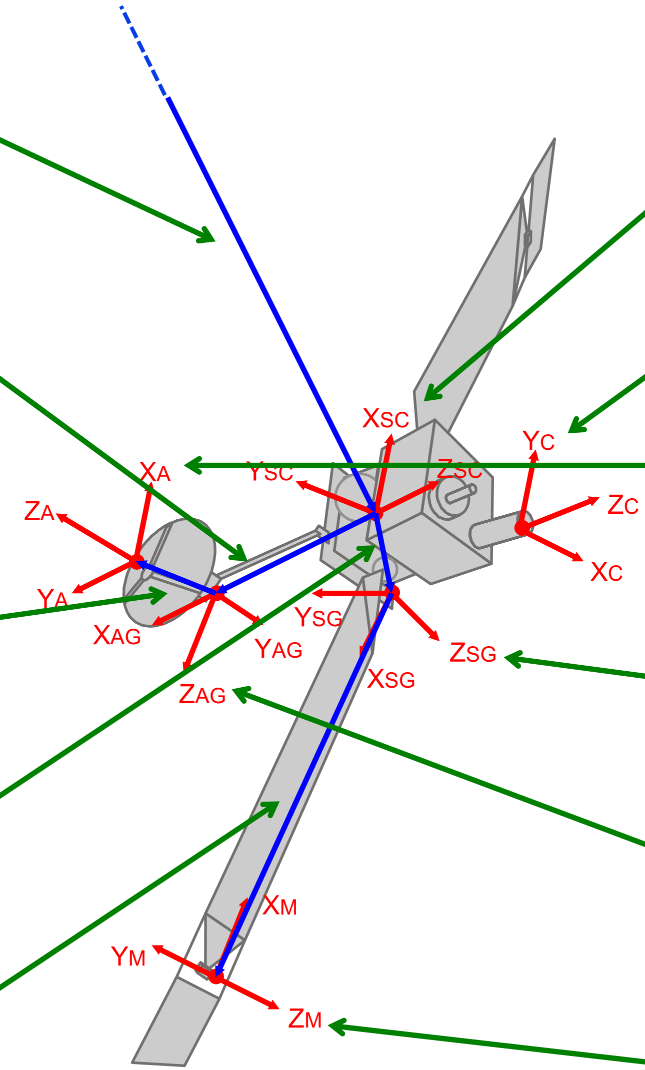
- **However, there also exists in SPICE an <u>inertial</u> frame associated with Mars, named "MARSIAU"**
  - This frame was defined 20 years ago based on old Mars rotation constants, for use by the Mars Observer and Mars Global Surveyor projects
  - This frame has NO relationship to the similarly sounding IAU_MARS frame, other than that they both relate to Mars

Spacecraft and Instrument Frames

- Defined for spacecraft, and items attached to a spacecraft, such as antennas, solar arrays, scan platforms, instruments and moving parts of an instrument (e.g. a scanning mirror)
- For those frames that are time varying (“moving”), the frame name is usually defined in an FK and the frame orientation is usually provided by a CK
- For those frames that are not moving (what we call “fixed offset”) both the frame name and the actual data defining the fixed orientation of the frame are provided in an FK

Some Examples of Spacecraft and “Instrument” Frames

Navigation and Ancillary Information Facility



Position Vectors

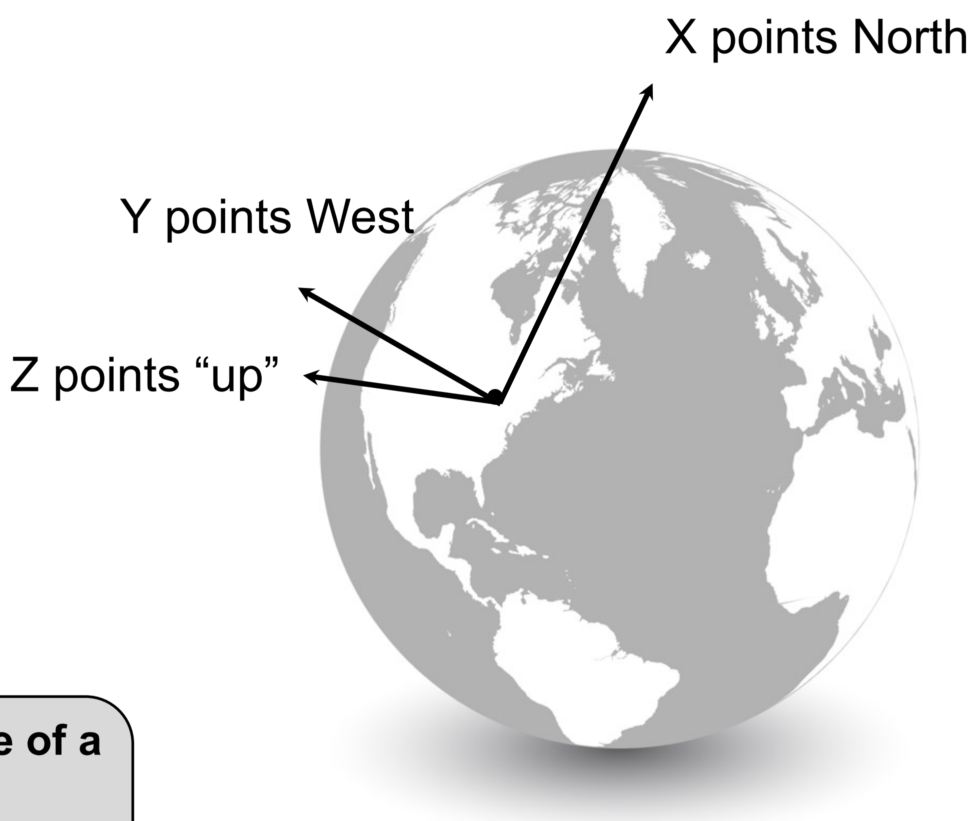
- Spacecraft position relative to planet center (“spacecraft” SPK file)
- High gain antenna gimbal position relative to spacecraft (“structures” SPK file)
- High gain antenna phase center location relative to high gain antenna gimbal (“structures” SPK file)
- Solar array gimbal position relative to spacecraft center (“structures” SPK file)
- Magnetometer position relative to solar array gimbal (“structures” SPK file)

Frame Orientations

- Spacecraft frame orientation relative to inertial frame (“spacecraft” CK file)
- Camera frame orientation relative to spacecraft frame (“mission” FK file)
- High gain antenna frame orientation relative to high gain antenna gimbal frame (“mission” FK file)
- Solar array gimbal frame orientation relative to spacecraft frame (“solar array” CK file)
- High gain antenna gimbal frame orientation relative to spacecraft frame (“antenna” CK file)
- Magnetometer frame orientation relative to solar array gimbal frame (“mission” FK file)

Topocentric Frames

- **Topocentric frames are located at or near to a surface**
- **One axis is normal to a reference spheroid, or parallel to the gravity gradient***
- **Examples: frames defined for telecommunications stations, or for landers or rovers**

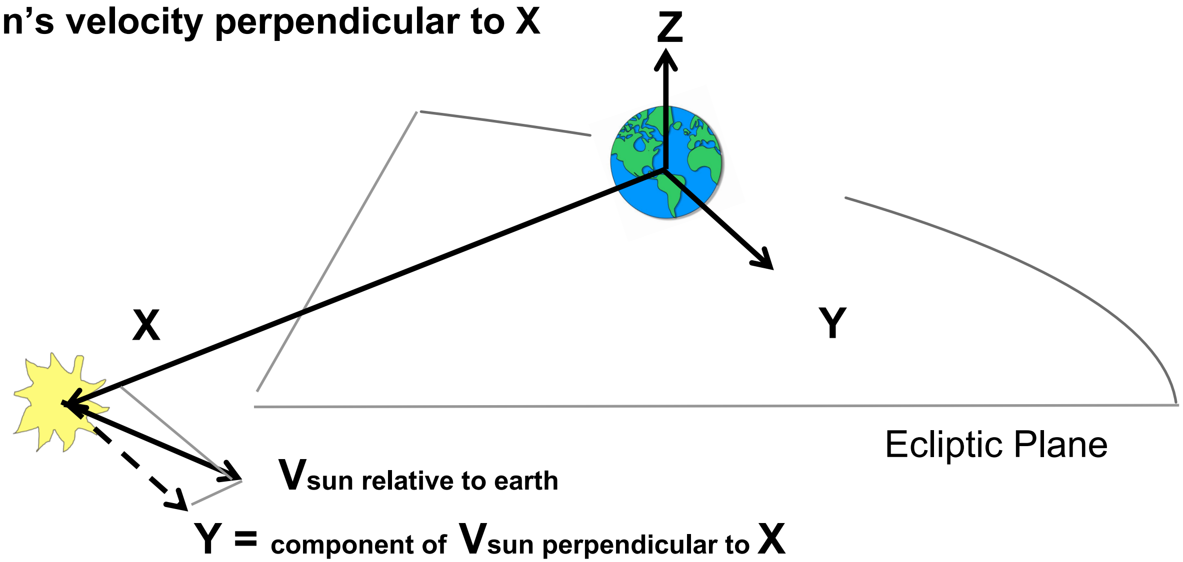


The graphic illustrates one example of a topocentric frame. There is not a standard definition–for example, the z-axis could point down, the x-axis North, and the y-axis East.

*SPICE tools always have the “up” or “down” axis being normal to the spheroid. But one could use external data to determine the local gravity gradient and construct a frame based on that.

Dynamic Frames

- In a dynamic frame the orientation changes with time
  - Families: Two-vector, Euler, and Of-date (refer to Dynamic Frames tutorial)
  - This category excludes frames for which the orientation is determined by a PCK or CK
  - Example of a two-vector dynamic frame: Geocentric Solar Ecliptic (GSE)
    - » X = earth – sun vector
    - » Y = component of the sun’s velocity perpendicular to X
    - » Z = X cross Y



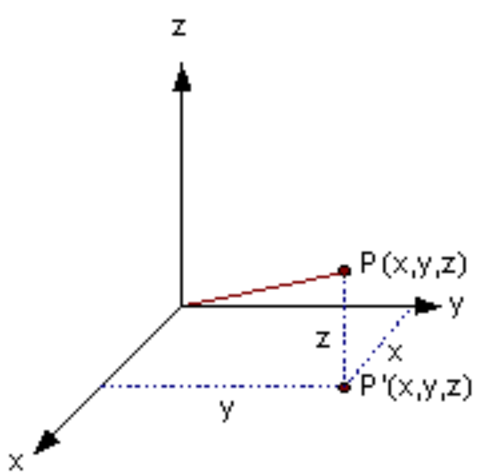
- Eventually NAIF will offer a generic dynamic frames kernel

Coordinate Systems

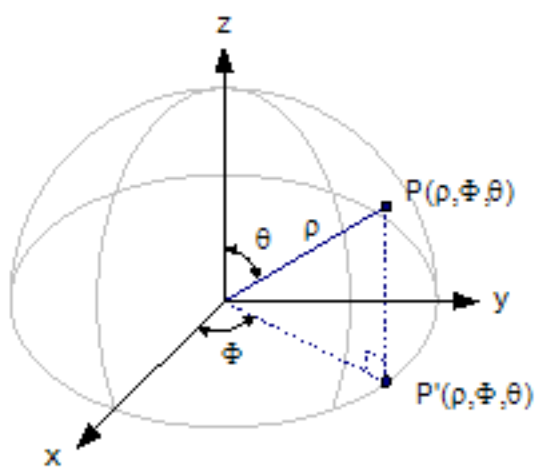
SPICE Coordinate Systems

- **A coordinate system specifies the method used to locate a point within a particular reference frame.**

Two examples of coordinate systems used to locate point "P"



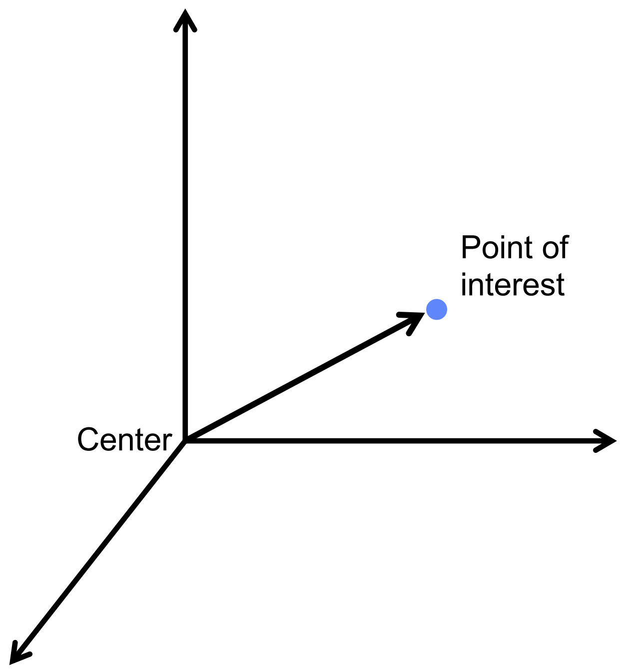
Rectangular or Cartesian coordinates:
X, Y, Z



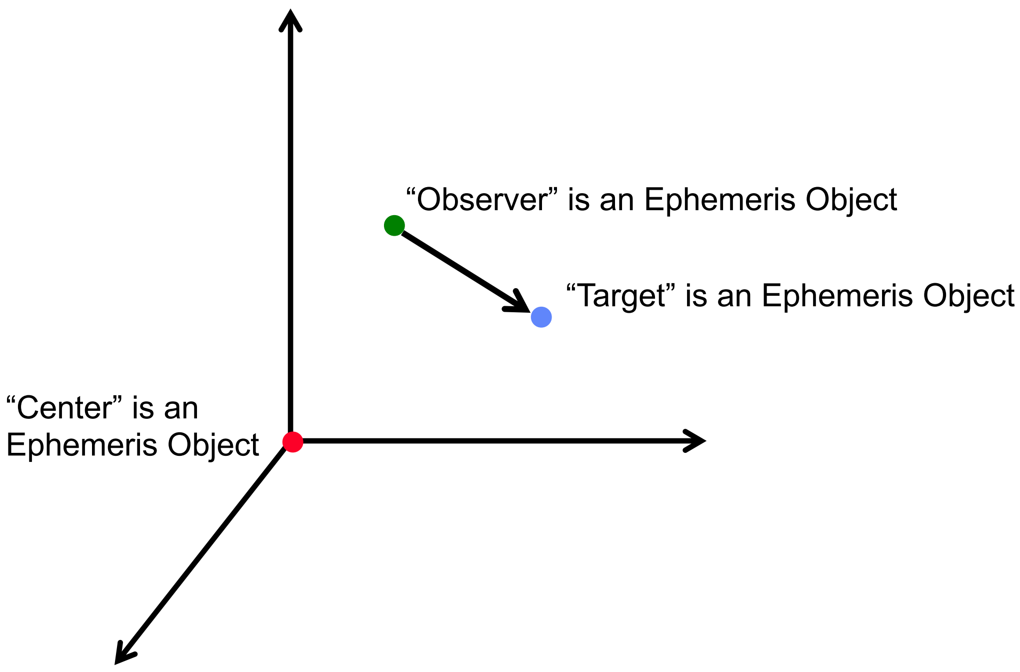
Spherical coordinates:
ϕ, θ, ρ

Specifying Positions

Common Style



SPICE Style



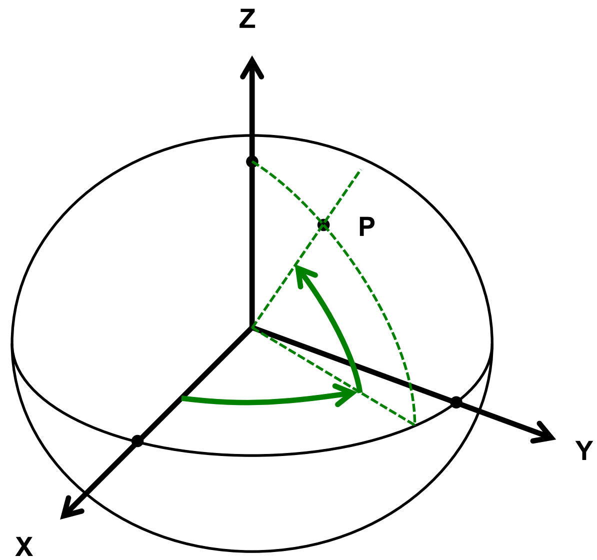
Many Coordinate Systems Used

- **In the Planetary Science discipline there are a number of coordinate systems in use, just as there are quite a few reference frames in use.**

- **Some of these coordinate systems have well accepted standard definitions, while others are anything but standard.**
  - **This means data producers and especially data users need to pay close attention to what they are doing!**

Planetocentric Coordinate System

- **For planets and their satellites the +Z axis (+90 latitude) always points to the north side of the invariable plane (the plane whose normal vector is the angular momentum vector of the solar system)**
  - Planetocentric longitude increases positively eastward (-180 to +180)
  - Planetocentric latitude increases positively northward (-90 to +90)
- **Dwarf planets*, asteroids and comets spin in the right hand sense about their "positive pole."**
  - What the IAU now calls the "positive pole" is still referred to as the "north pole" in SPICE documentation.
  - The "positive pole" may point above or below the invariable plane of the solar system (see above).
  - This revision by the IAU Working Group (2006) inverts what had been the direction of the north pole for Pluto, Charon and Ida.
- **Toolkit planetocentric APIs:**
  - LATREC, RECLAT, DRDLAT, DLATDR, XFMSTA

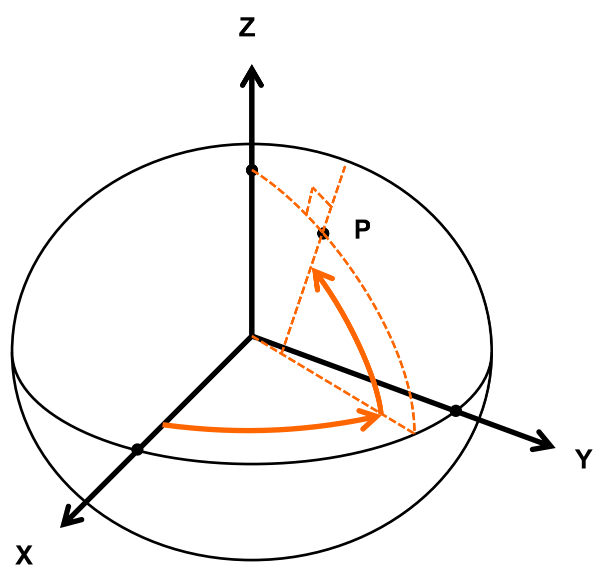


*The dwarf planets are: Ceres, Eris, Haumea, Makemake, Pluto

Planetodetic Coordinate System

Navigation and Ancillary Information Facility

- **Planetodetic longitude is the same as planetocentric longitude**
  - Increases positively eastward (-180 to +180)
- **Planetodetic latitude**
  - Tied to a reference ellipsoid
  - For a point, P, on a reference ellipsoid, angle measured from X-Y plane to the surface normal at the point of interest. For other points, equals latitude at the nearest point on the reference ellipsoid
  - Increases positively northward (-90 to +90)
- **Toolkit planetodetic APIs are:**
  - GEOREC, RECGEO, DRDGEO, DGEODR, XFMSTA

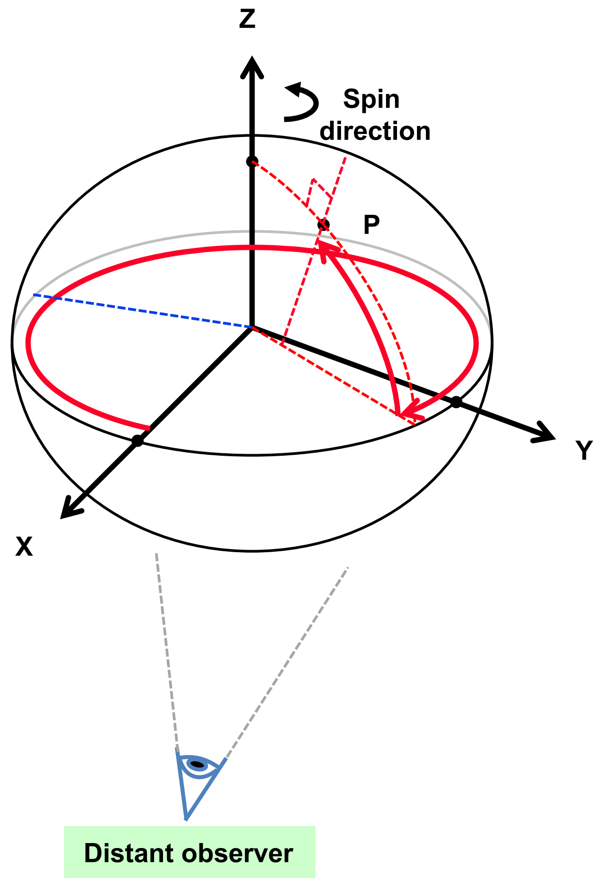


Planetographic Coordinate System

Navigation and Ancillary Information Facility

- **For planet and satellite planetographic coordinate systems:**
  - Planetographic longitude is usually defined such that the sub-observer longitude increases with time as seen by a distant, fixed observer (0 to 360)
  - The earth, moon and sun are exceptions; planetographic longitude is positive east by default (0 to 360)
  - Planetographic latitude is planetodetic latitude (-90 to +90)
  - Toolkit planetographic APIs are:
    - » PGRREC, RECPGR, DRDPGR, DPGRDR, XFMSTA

- **For dwarf planets, asteroids and comets:**
  - There are multiple, inconsistent standards! (USNO, IAU, PDS)
  - NAIF strongly suggests you use only planetocentric or planetodetic coordinates for these objects



*The dwarf planets are: Ceres, Eris, Haumea, Makemake, Pluto

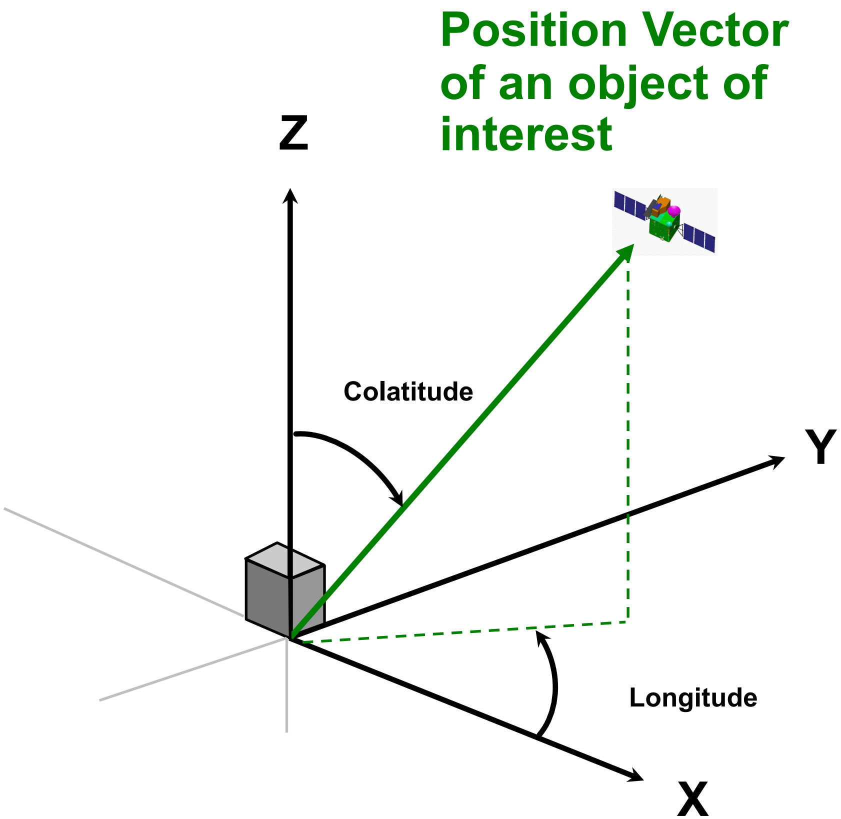
Spherical Coordinates

Navigation and Ancillary Information Facility

- **Longitude:**
  - angle from +X axis to projection of position vector on X-Y plane
  - increases in counter-clockwise direction
  - see the API header for restrictions on ranges

- **Colatitude:**
  - Angle between +Z axis and position vector (0 to 180)
  - Other names used elsewhere are zenith angle, inclination angle and polar angle.

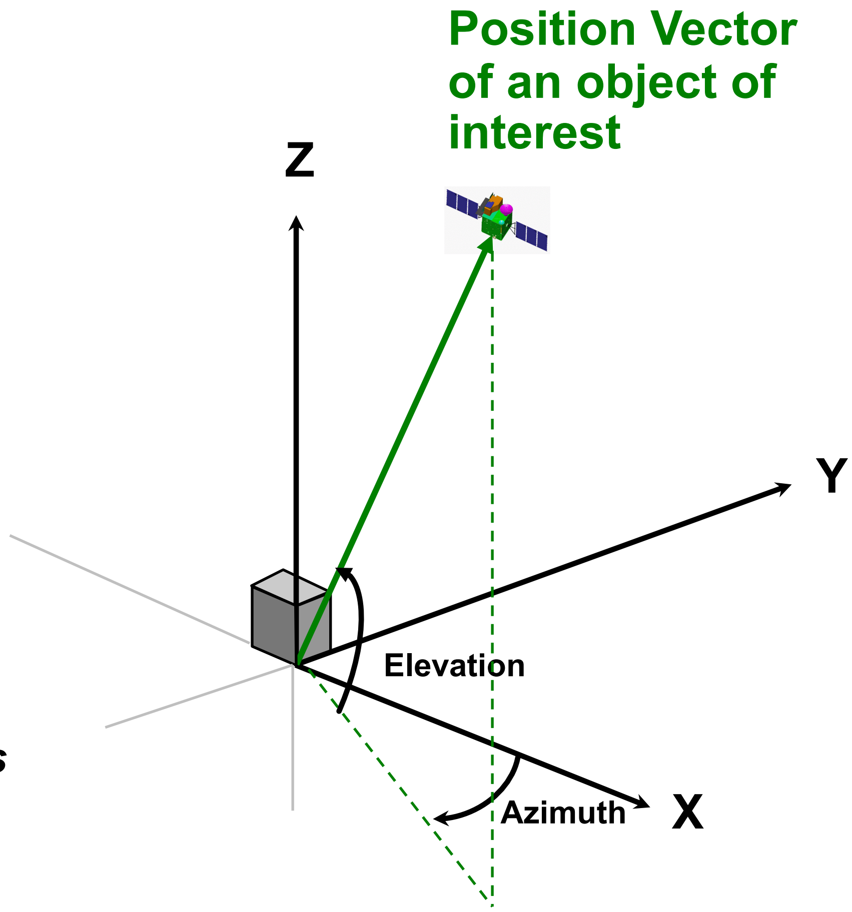
- **Toolkit spherical APIs :**
  - SPHREC, RECSPH, DRDSPH, DSPHDR, XFMSTA



An Example of Azimuth-Elevation Coordinates

- **Azimuth:**
  - Angle from +X axis to projection of position vector on x-y plane
  - Increases in clockwise direction (0 to 360)

- **Elevation:**
  - Angle between position vector and x-y plane (-90 to +90)
  - In this example, +Z is in the "up" direction, which might not be true for you.



SPICE does not currently contain APIs specific to converting between AZ-EL and other coordinate systems due to lack of standard definitions for AZ-EL. See the next page for methods for doing this conversion.

Converting Rectangular to AZ-EL Coordinates

- **Rectangular to AZ-EL**
  - Using RA-DEC as intermediary
    - » Convert rectangular to RA-DEC using RECRAD (where the range for RA is [0, 2*pi]
    - » Then map RA-DEC to whatever is the AZ-EL convention you are using (how does DEC compare with your definition of EL?)
  - Using LAT-LON as intermediary
    - » Convert rectangular to LAT-LON using RECLAT, where the range for LON is [-pi, pi]
    - » Then map LAT-LON to whatever is the AZ-EL convention you are using (e.g. you could negate LAT to achieve positive EL being “up” in a frame having Z pointed “down.”

Summary of SPICE Coordinate Transformation APIs

Coordinate Systems	APIs for Position Transformation	APIs for Velocity Transformation	Notes
Latitudinal to/from Rectangular	LATREC RECLAT	DRDLAT DLATDR	More commonly called Planetocentric. Use these APIs for Azimuth/Elevation as well.
R.A. & Dec. to/from Rectangular	RADREC RECRAD	DRDLAT DLATDR	Same as for latitudinal except for range of LON and RA when converting rectangular to angular. LON: -Pi to +Pi RA: 0 to 2 Pi
Planetographic to/from Rectangular	PGRREC RECPGR	DRDPGR DPGRDR	Best restricted to planets, satellites and the sun. Requires a text PCK to be loaded to determine body spin direction.
Geodetic to/from Rectangular	GEOREC RECGEO	DRDGEO DGEODR	
Cylindrical to/from Rectangular	CYLREC RECCYL	DRDCYL DCYLDR	
Spherical to/from Rectangular	SPHREC RECSPH	DRDSPH DSPHDR	Shape must be a true sphere.
AZ-EL to/from Rectangular	none	none	See earlier chart titled “Converting Rectangular to AZ-EL Coordinates”

See also the next page re XFMSTA

Examples of Velocity Coordinate Transformations

Navigation and Ancillary Information Facility

This example is for rectangular to spherical

Fortran examples

- **Using full state vector transformation API**

```
CALL SPKEZR ( TARG, ET, REF, CORR, OBS, STATE, LT )
CALL XFMSTA ( STATE, 'RECTANGULAR', 'SPHERICAL', ' ' , OUTSTATE )
```

- **Using velocity-only (Jacobian) APIs**
  - Transform velocities from rectangular to spherical coordinates using the SPICE Jacobian matrix routines. The SPICE calls that implement this computation are:

```
CALL SPKEZR  (  TARG,  ET,  REF,  CORR,  OBS,  STATE,  LT  )
CALL DSPHDR (  STATE(1), STATE(2),  STATE(3), JACOBI  )
CALL MXV       (  JACOBI,    STATE(4),  SPHVEL  )
```

  - After these calls, the vector SPHVEL contains the velocity in spherical coordinates: specifically, the derivatives

    (d (r) / dt,   d (colatitude) / dt,   d (longitude) /dt)

  - Caution: coordinate transformations often have singularities, so derivatives may not exist everywhere.
    - » Exceptions are described in the headers of the SPICE Jacobian matrix routines.
    - » SPICE Jacobian matrix routines signal errors if asked to perform an invalid computation.

- **Note: Using XFMSTA for velocity transformations is slower than using the Jacobian API**