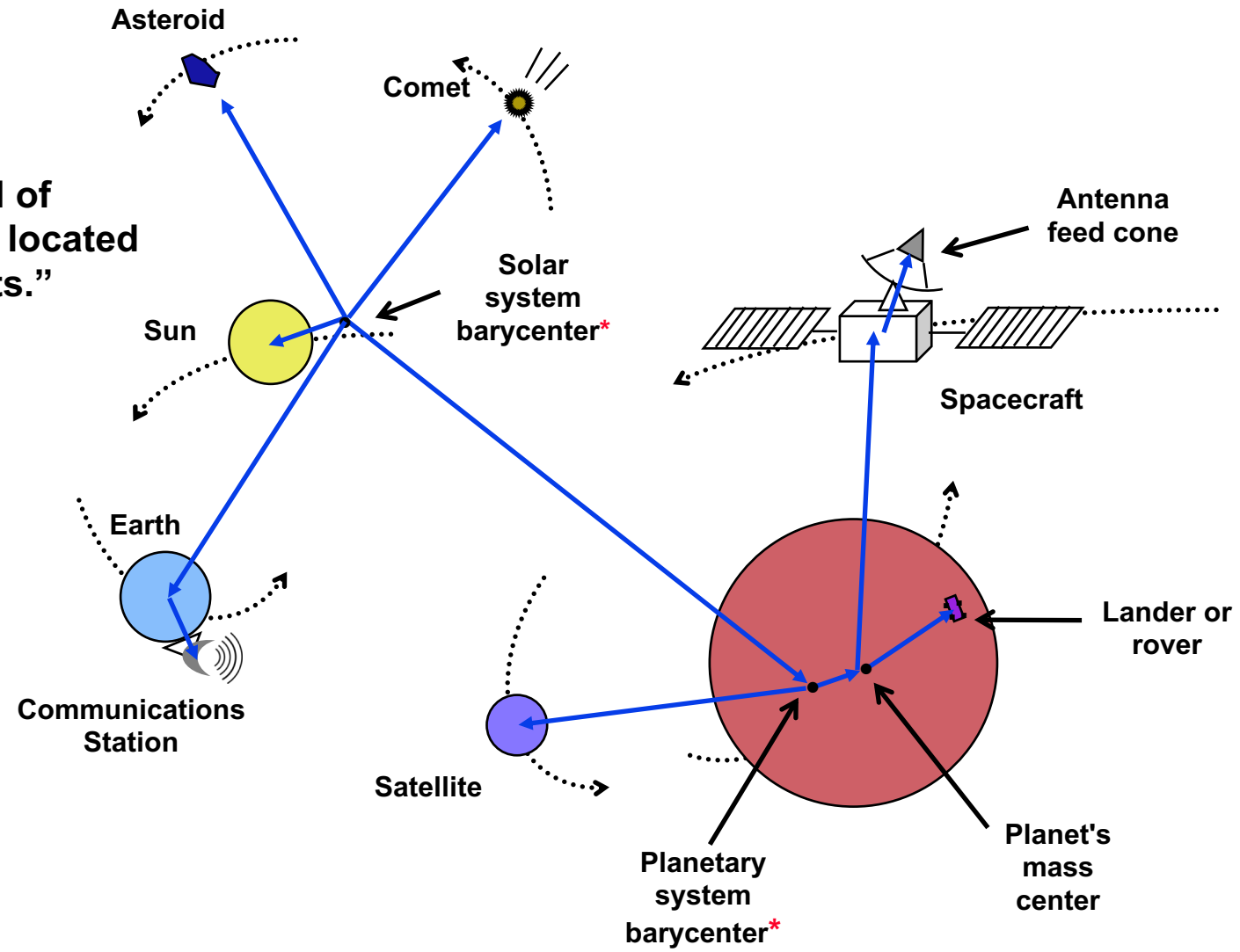
Examples of SPICE Ephemeris Objects

Navigation and Ancillary Information Facility

The head and the tail of every blue arrow are located at "ephemeris objects."



*A barycenter is the center of mass of a system of bodies, such as Saturn plus all of Saturn's satellites, or, the sun and all the planets, satellites, comets and asteroids in the solar system.

SPICE Ephemeris Data

Navigation and Ancillary Information Facility

- **An SPK file contains ephemeris (trajectory) data for "ephemeris objects."**
  - **"Ephemeris" means position and velocity as a function of time**
    - **Position + velocity is often referred to as "state"**

- **"Ephemeris objects" are spacecraft, planets, satellites, comets and asteroids.**
  - **But the following are also ephemeris objects:**
    - **the center of mass of our solar system (solar system barycenter)**
    - **the center of mass of a planet/satellite system (planet barycenter)**
    - **a rover on the surface of a body**
    - **a camera on top of a mast on a lander**
    - **a transmitter cone on a spacecraft**
    - **a deep space communications antenna on the earth**

- **A single SPK file can contain data for multiple ephemeris objects, and often does.**

Imagine Some Ephemeris Data

Navigation and Ancillary Information Facility

```
epoch_1,  x1,  y1,  z1,  vx1,  vy1,  vz1
epoch_2,  x2,  y2,  z2,  vx2,  vy2,  vz2
epoch_3,  x3,  y3,  z3,  vx3,  vy3,  vz3
epoch_4,  x4,  y4,  z4,  vx4,  vy4,  vz4
 ..................  etc. ..................
 ..................  etc. ..................
epoch_n,  xn,  yn,  zn,  vxn,  vyn, vzn
```

Perhaps this is an ASCII table or an Excel spreadsheet containing rows of time-tagged Cartesian state vectors.

"epoch" = time

It may not be written inside the table or spreadsheet, but perhaps an interface agreement somehow tells you:

- what object this ephemeris is for
- what is the name of the reference frame ("coordinate frame") in which the data are given
- what is the center of motion of the object
- what time system is being used for the epochs
- maybe also what are the start and stop times of the file
  - » meaning, what are "epoch_1" and "epoch_n"

Imagine a Simple Ephemeris File

Navigation and Ancillary Information Facility

```
epoch_1,  x1,  y1,  z1,  vx1,  vy1,  vz1
epoch_2,  x2,  y2,  z2,  vx2,  vy2,  vz2
epoch_3,  x3,  y3,  z3,  vx3,  vy3,  vz3
epoch_4,  x4,  y4,  z4,  vx4,  vy4,  vz4
 ..................  etc. ..................
 ..................  etc. ..................
epoch_n,  xn,  yn,  zn,  vxn,  vyn,  vzn
```

We'll represent that simple ephemeris data shown on the previous page as a "block" like this.

Imagine a Simple Ephemeris File

Navigation and Ancillary Information Facility

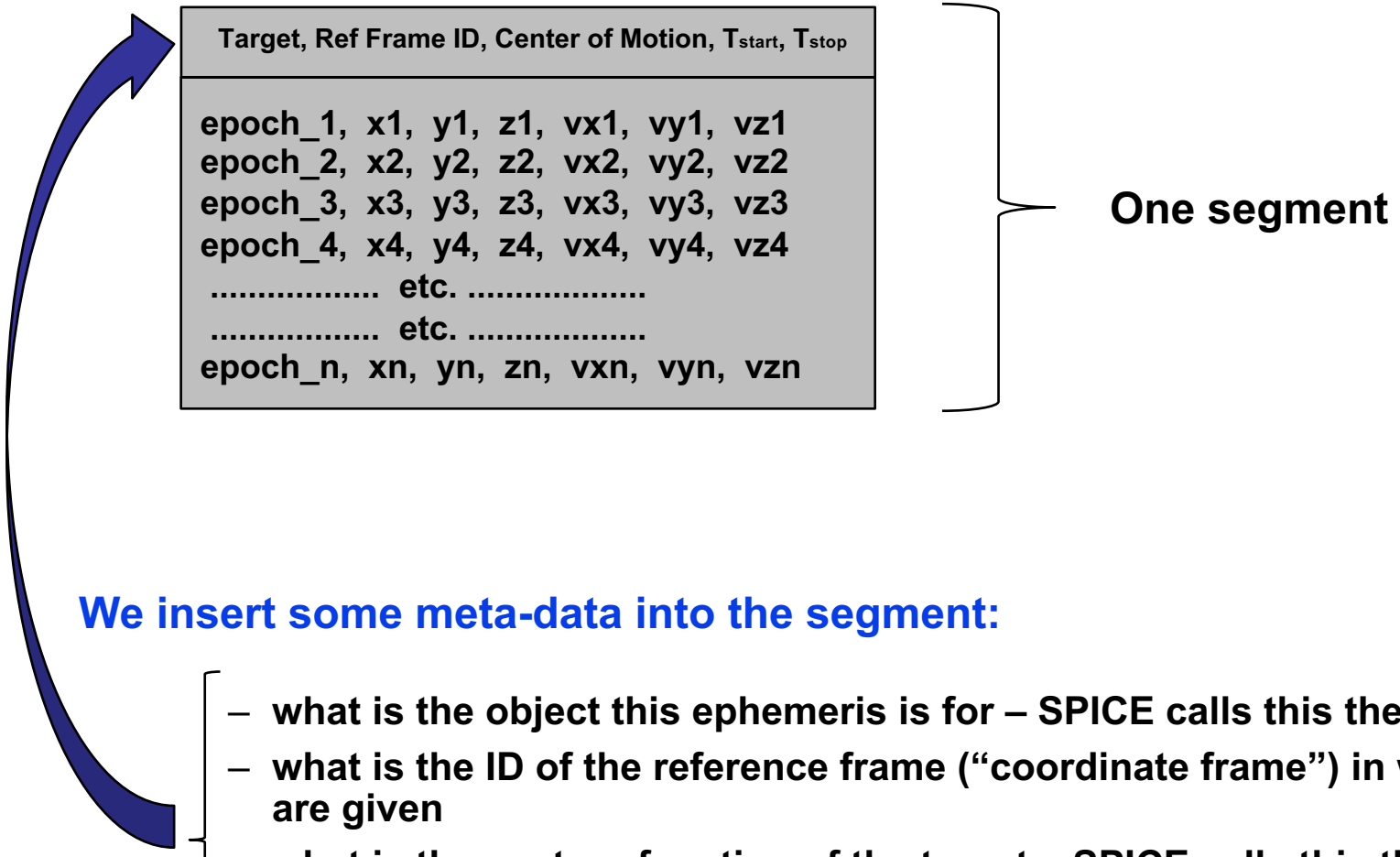
```
epoch_1,  x1,  y1,  z1,  vx1,  vy1,  vz1
epoch_2,  x2,  y2,  z2,  vx2,  vy2,  vz2
epoch_3,  x3,  y3,  z3,  vx3,  vy3,  vz3
epoch_4,  x4,  y4,  z4,  vx4,  vy4,  vz4
 ..................  etc. ..................
 ..................  etc. ..................
epoch_n,  xn,  yn,  zn,  vxn,  vyn,  vzn
```

We'll represent that simple ephemeris file as a "block" like this.

This becomes the basis of a "segment" in an SPK file.

An SPK "Segment"

Navigation and Ancillary Information Facility

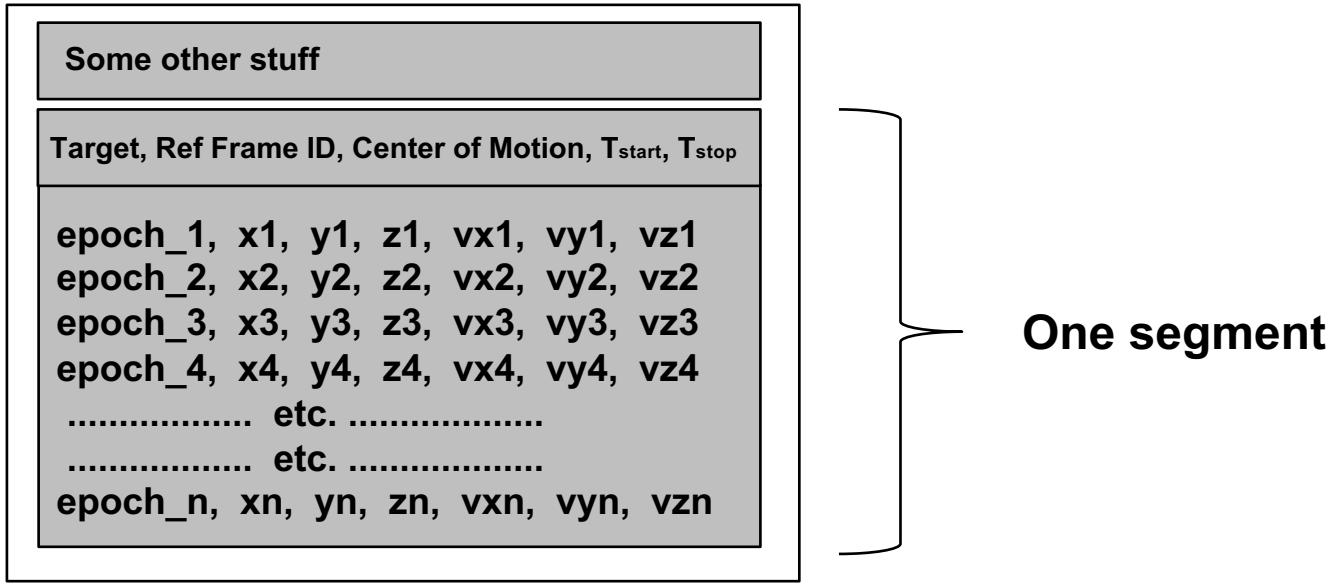


We insert some meta-data into the segment:

- what is the object this ephemeris is for – SPICE calls this the "target"
- what is the ID of the reference frame ("coordinate frame") in which the data are given
- what is the center of motion of the target – SPICE calls this the "observer"
- the start and stop times of the file, T_{start} and T_{stop}
  - » meaning, what are "epoch_1" and "epoch_n"

A Simple SPK File

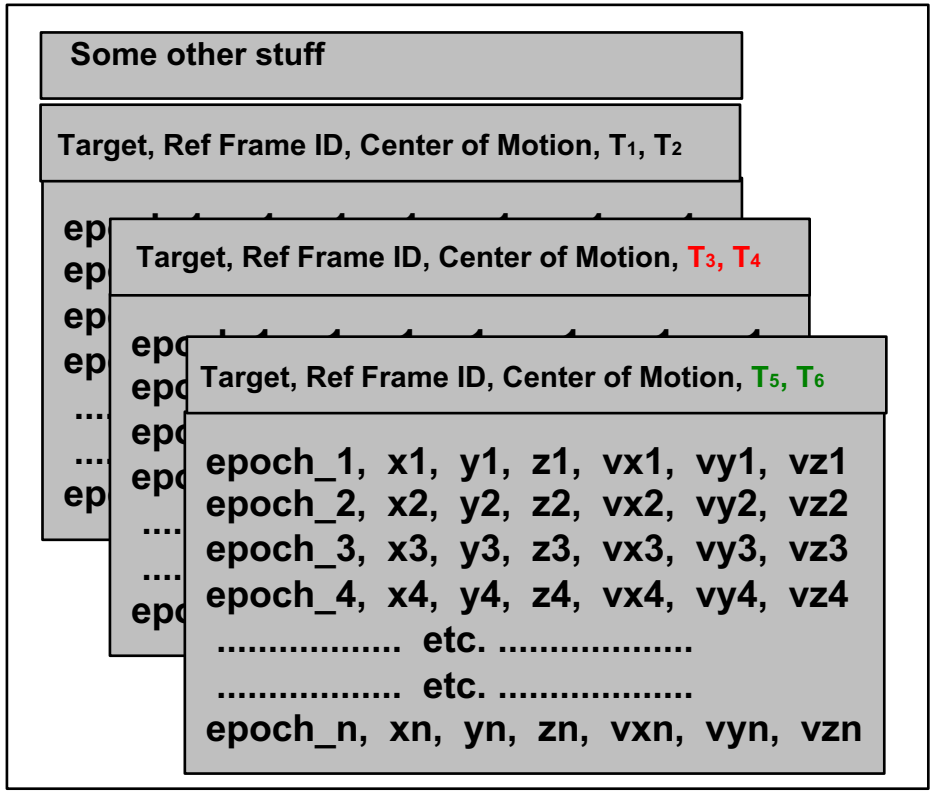
Navigation and Ancillary Information Facility



- **This very simple SPK file is made up of a single segment containing ephemeris data:**
  - **for a single object (perhaps a spacecraft, an asteroid, or …whatever),**
  - **given in a single reference frame (“coordinate frame”),**
  - **having a single center of motion,**
  - **with data spanning from T_{start} to T_{stop}**

A More Substantial SPK File

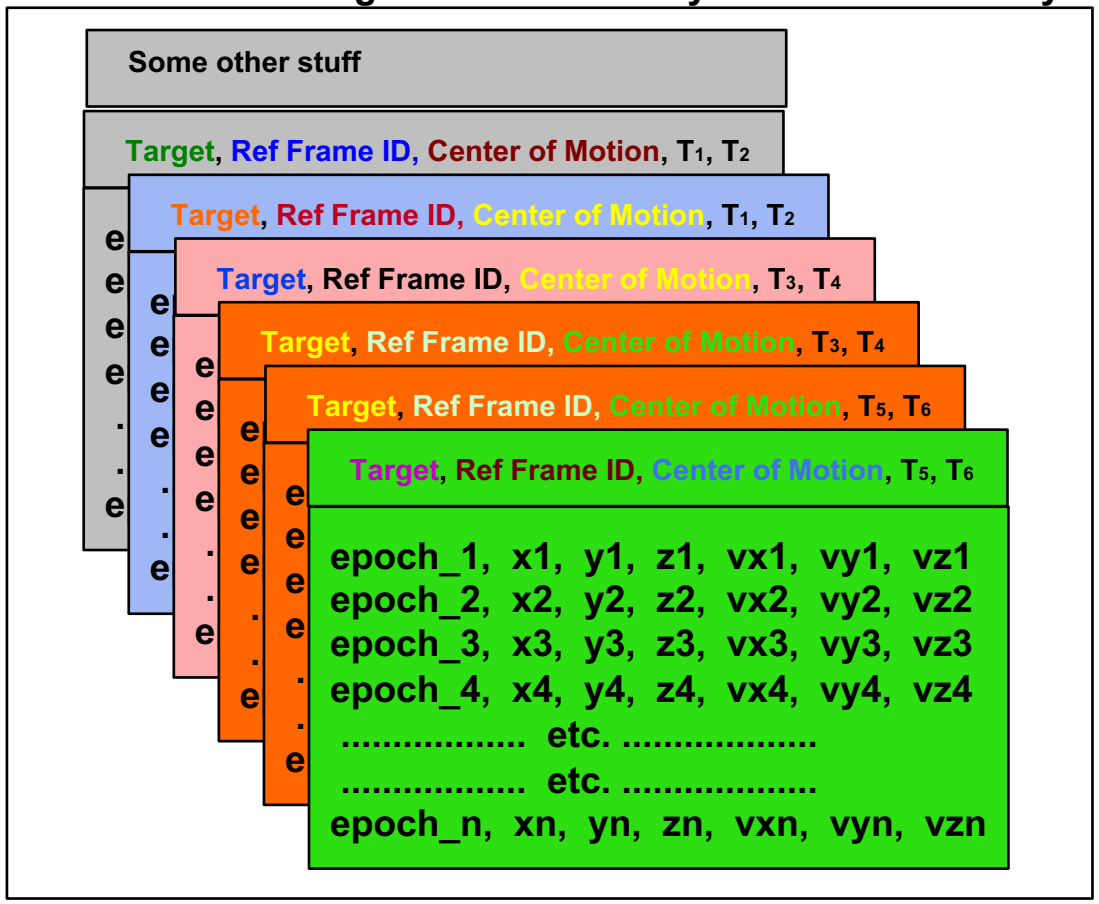
Navigation and Ancillary Information Facility



- **This more substantial SPK is made up of multiple segments containing ephemeris data:**
  - **for a single object (perhaps a spacecraft, an asteroid, or …???),**
  - **given in a single reference frame (“coordinate frame”),**
  - **having a single center of motion,**
  - **with data spanning from T_1 to T_6**

An Even More Substantial SPK File

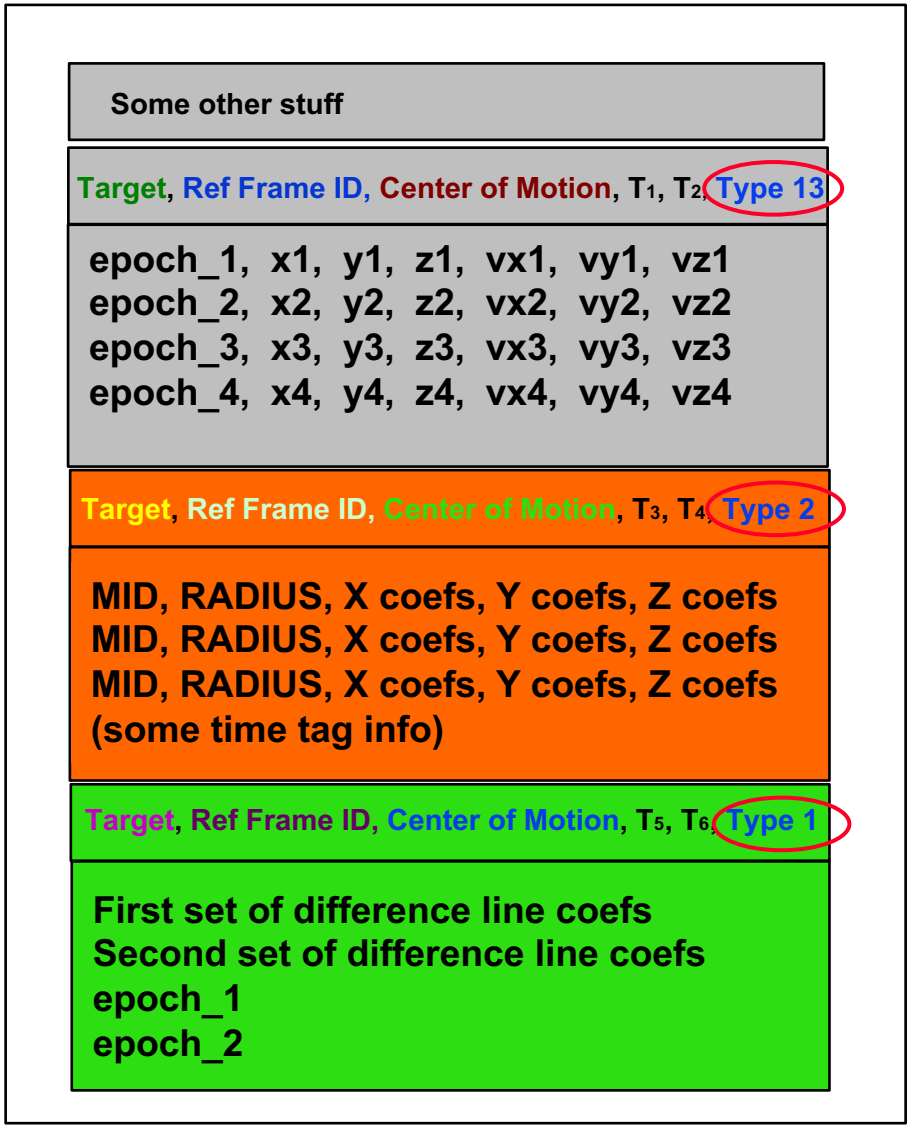
Navigation and Ancillary Information Facility



- **This even more substantial SPK contains multiple segments having:**
  - **several objects**
  - **several reference frames**
  - **several centers of motion**
  - **several pairs of start and stop times**

SPK "Type" Info in each Segment

Navigation and Ancillary Information Facility



- Each segment can contain a different type of ephemeris data (as long as it's been built into the SPK subsystem). Examples:
  - Discrete state vectors
  - Chebyshev polynomials
  - Difference lines (unique to JPL)
  - Etc., etc.
- Each segment has the SPK Type stored in its meta-data record.
- Toolkit software knows how to evaluate each Type – no worries for you!

SPK Data are Continuous Within a Segment

Navigation and Ancillary Information Facility

Cassini , Ref Frame ID, Saturn bc, T_1, T_2, Type 13
epoch_1, x1, y1, z1, vx1, vy1, vz1 epoch_2, x2, y2, z2, vx2, vy2, vz2 epoch_3, x3, y3, z3, vx3, vy3, vz3 epoch_4, x4, y4, z4, vx4, vy4, vz4

- Within the time bounds (T_1, T_2) of a segment, SPICE software will return a result–a state vector consisting of position and velocity–at any epoch… not just at the epochs of the ephemeris records (epoch_1, epoch_2, epoch_3, epoch_4)

- In the example above, SPICE will return the position and velocity–the state–of the Cassini spacecraft relative to the Saturn barycenter at any time *t* where: $T_1 \leq t \leq T_2$

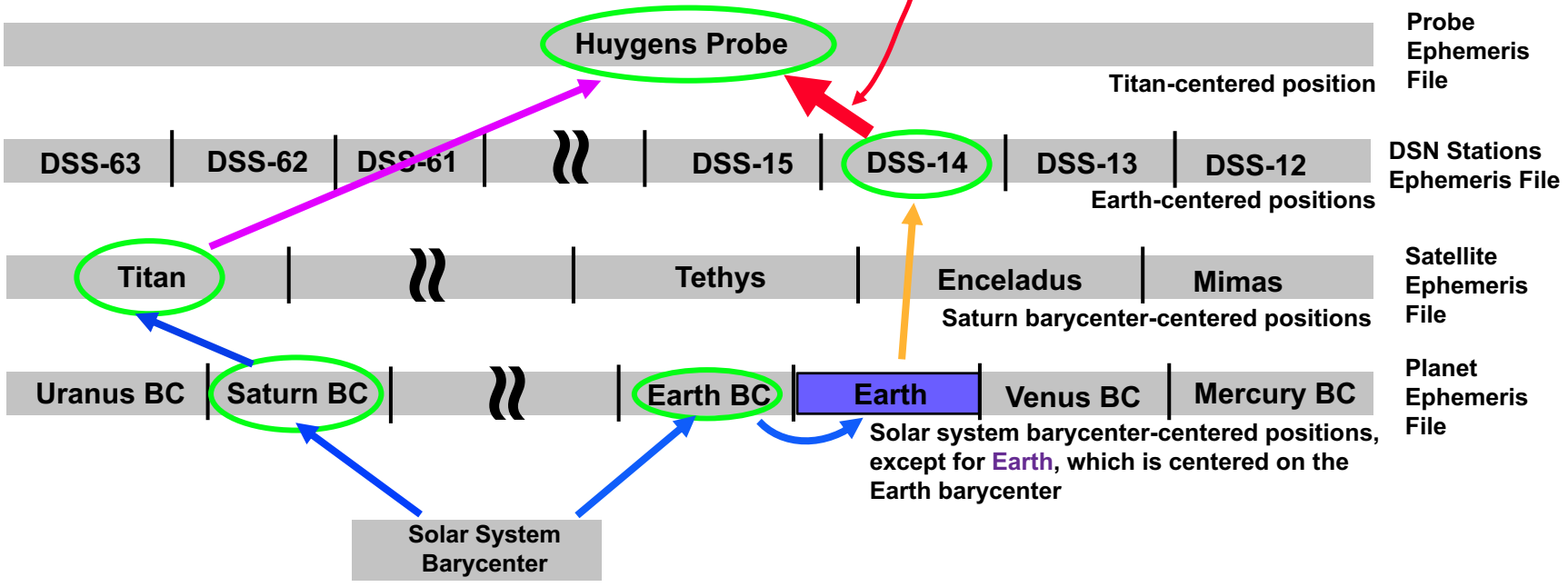
Chaining and Frame Transformation

- **Next we'll discuss "chaining" and "frame transformations"… features of the SPK subsystem that make it rather unique.**

SPICE Chains SPK Data

Navigation and Ancillary Information Facility

- **SPICE automatically searches across all loaded SPK files to find the segments needed to compute the vectors needed to obtain the result the customer has asked for. SPICE chains these together using vector addition and subtraction.**
  - **In this example the user wants the position of the Huygens probe sitting on the surface of Titan as seen from Deep Space Station 14.**
  - **SPICE computes this by chaining the gold, blue and violet chunks.**



SPICE Automates Frame Transformation

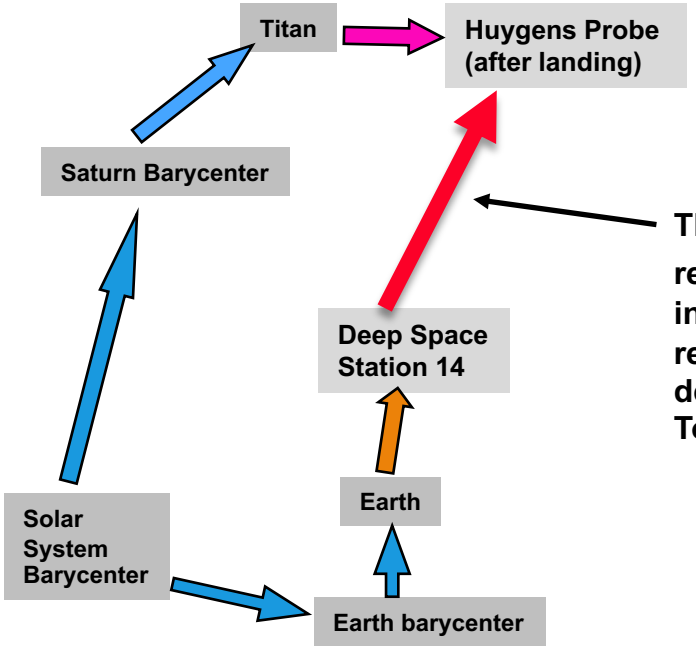
Navigation and Ancillary Information Facility

- **As part of the "chaining" process just mentioned…**
  - **position vectors are automatically rotated into a consistent reference frame to allow vector additions and subtractions**
  - **the final vector is rotated into the output reference frame requested by the user**

Reference Frames Used

- International Celestial Reference Frame (`J2000`)
- Titan body-fixed frame (`IAU_TITAN`)
- International Terrestrial Reference Frame (`ITRF93`)
- DSS-14 topocentric reference frame (`DSS-14_TOPO`)

Ephemeris Segments Used



The position of the Huygens Probe relative to DSS-14, at time *t*, given in the DSS-14 topocentric reference frame, can be determined using a SINGLE Toolkit subroutine call.

A single subroutine call does it all!

```
CALL SPKPOS ('HUYGENS_PROBE', t, 'DSS-14_TOPO', 'CN+S', 'DSS-14', POSITION, LT)
```

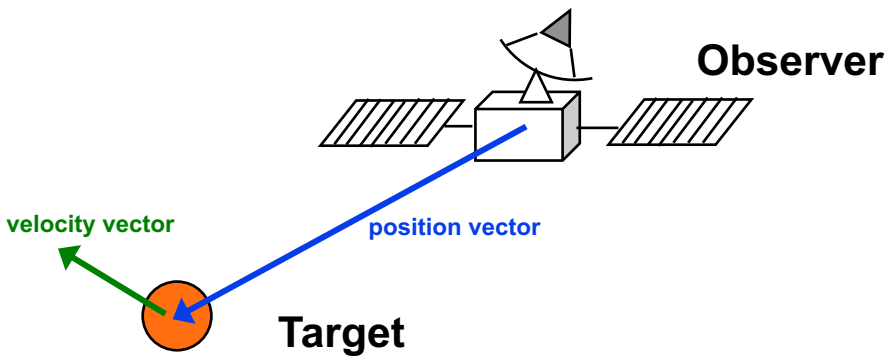
Details

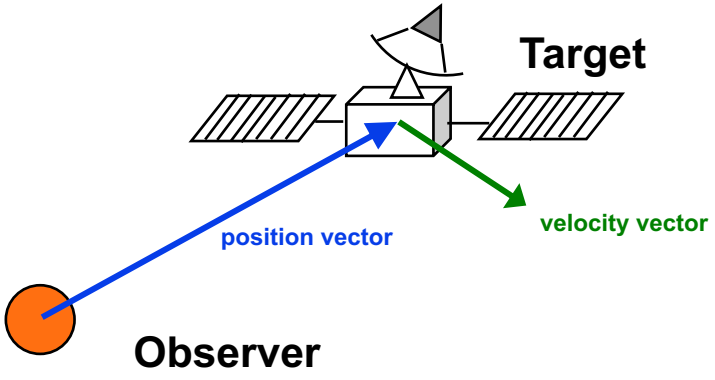
- **Now for some details.**
- **There's quite a lot… don't feel you need to grasp all of this immediately.**

Reading an SPK: Observers and Targets

Navigation and Ancillary Information Facility

- When you read an SPK file you specify which ephemeris object is to be the "target" and which is to be the "observer."
- The SPK system returns the state of the target relative to the observer.
  - The computed position data point from the "observer" to the "target."
  - The computed velocity is that of the "target" relative to the "observer."





- Any ephemeris object can be a target or an observer!

SPK File Coverage - 1

Navigation and Ancillary Information Facility

- **The time period over which an SPK file provides data for an ephemeris object is called the "coverage" or "time coverage" for that object.**
  - **An SPK file's coverage for an object consists of one or more time intervals.**
  - **Often the coverage for all objects in an SPK file is a single, common time interval.**

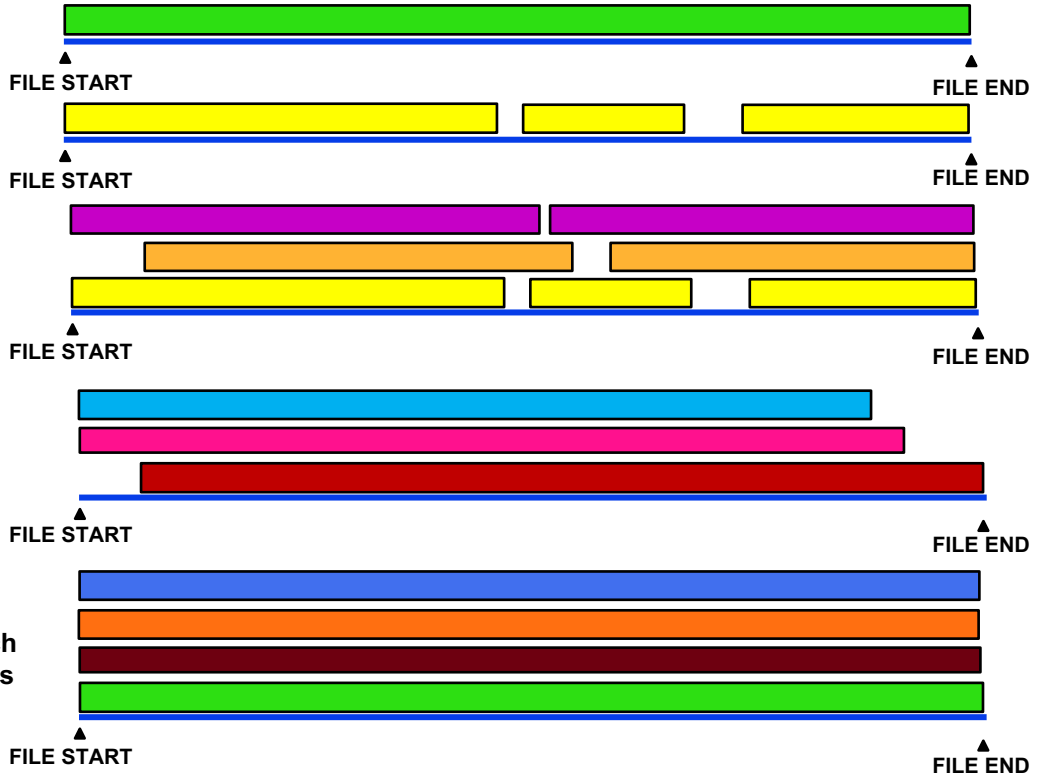
SPK file containing data for one object with no data gaps

SPK file containing data for one object, with two data gaps

SPK file containing data for three objects, each having different data gaps

SPK file containing data for three objects, each having different coverage but with no data gaps

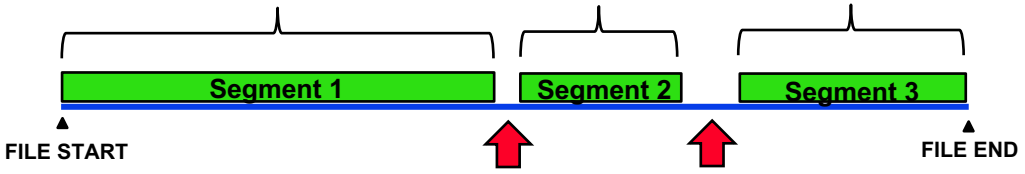
SPK file containing data for several objects, each having the same coverage and with no data gaps



SPK File Coverage - 2

- **For any request time within any time interval comprising the coverage for an object (i.e. the three green stripes shown below), the SPK subsystem can return a vector representing the state of that object relative to its center of motion.**
  - **The SPK system will automatically interpolate ephemeris data to produce a Cartesian state vector at the request time.**
  - **To a user's program, the ephemeris data appear to be continuous over each time interval, even if the data stored inside the SPK file are discrete.**
- **The SPK subsystem will *not* return a result for a request time falling within a data gap.**
  - **Data gaps can only occur between segments.**

"Results" will be returned by the SPK reader API for any request time falling within these three intervals.



Note: each of the green stripes above consists of one or more segments.

No results will be returned by the SPK subsystem for any request time falling within these two data gaps

Reference Frames Used in Writing and Reading SPKs

Navigation and Ancillary Information Facility

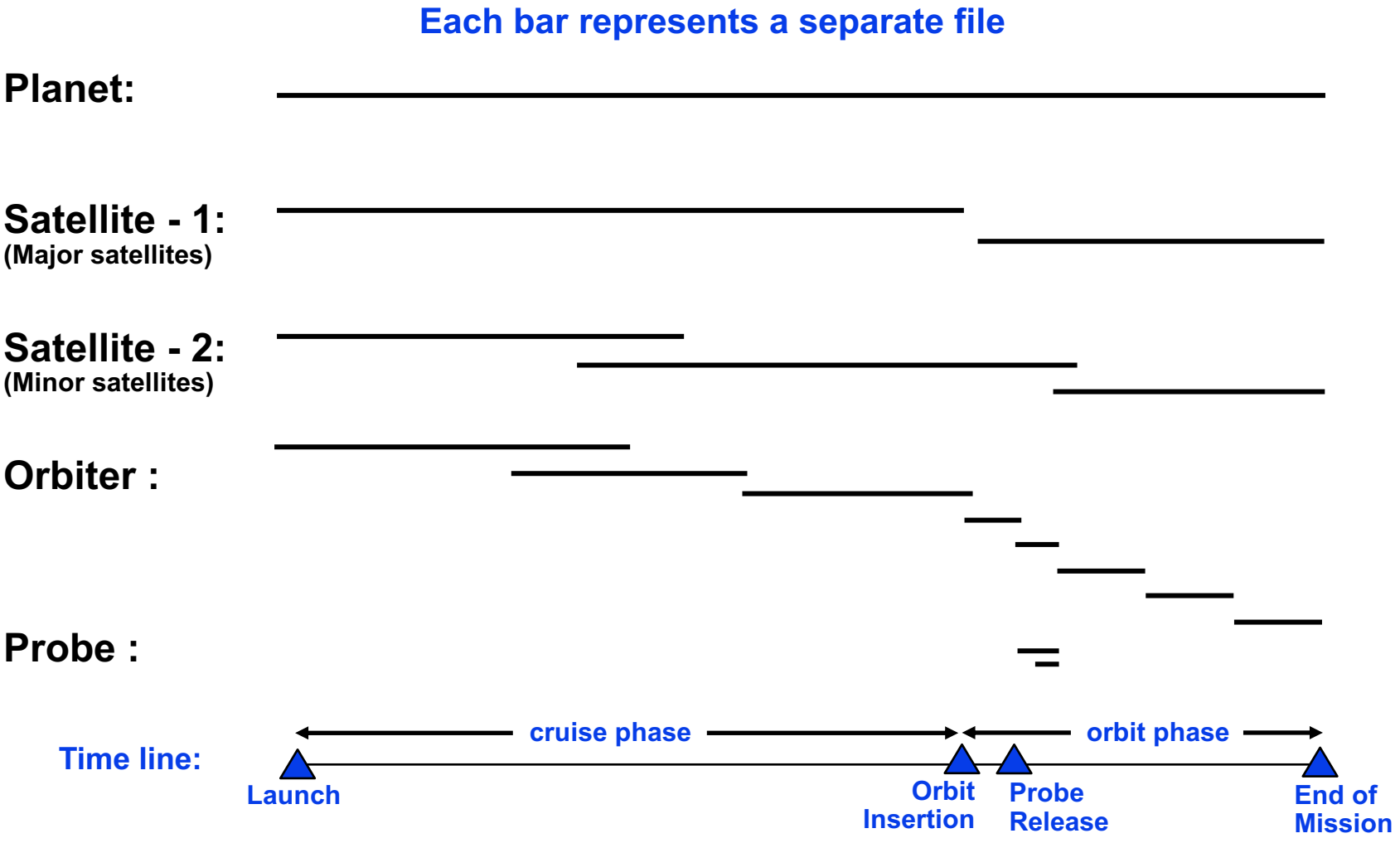
- **All ephemeris data have an associated reference frame***
  - **The frame specification is input by the SPK producer**
    - » **This input frame must be one known to the SPICE system**
  - **The frame can change from segment-to-segment**

- **A program reading an SPK file specifies relative to what reference frame the output state or position vectors are to be given; you're not stuck with using the frame the SPK producer used**
  - **This output frame you select must be known to your program**
    - » **"Known" means either a built-in frame (hard coded in SPICE) or one specified in a Frames Kernel**
    - » **The user's program may need to have access to additional SPICE data in order to construct the specified frame**

*** See the concepts tutorial for a discussion on reference frames**

Possible* SPK File Time Coverages for the Previous Example

Navigation and Ancillary Information Facility



* Note: This was not the real Cassini scenario–it is simply an illustration of some of the possibilities for ephemeris delivery on a planetary mission.

Understanding an SPK File

- The SPK producer should have provided descriptive meta-data inside an SPK file, in the “comment area”
  - The comments should say when, why, how and for what purpose the file was made
  - Additional useful information could also be provided by the producer
    - » Example: when and why any data gaps are present

- These comments may be extracted using an API (subroutine) or viewed using a SPICE utility program.
  - API: DAFEC
  - Utility program: commnt –r <spk_file_name>