Navigation and Ancillary Information Facility

Non-Toolkit Applications

April 2016

Summary

- **NAIF makes available a set of applications *not* included in the generic Toolkits. This set includes programs for:**

  - **making, modifying, validating, inspecting, and analyzing SPK files:**
    - » *pinpoint, dafcat, bspidmod, dafmod, spy*

  - **making and modifying CK files**
    - » *prediCkt, ckslicer, ckspanit, dafcat, cksmrg, dafmod*

  - **making SCLK files**
    - » *makclk*

  - **computing derived quantities**
    - » *orbnum, optics, spy*

  - **determining SPICE kernel type and binary format**
    - » *archtype, bff*

  - **converting between binary and text kernel formats**
    - » *bingo*

- **Executables and User's Guides for supported environments are available from the NAIF server at:**

  - **http://naif.jpl.nasa.gov/naif/utilities.html**
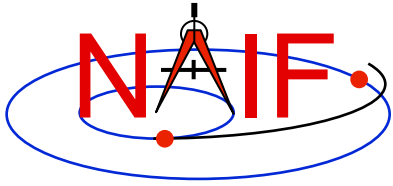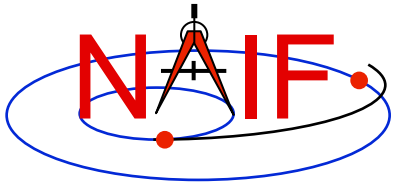
Using Non-Toolkit Apps

- **All of these apps are meant to be used as operating system shell executables**
   - **One generally cannot run these within IDL or MATLAB**
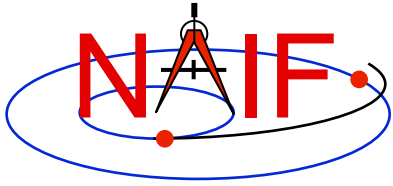
PINPOINT

- *pinpoint* creates SPK files and topocentric frames FK files for objects whose position is a constant offset with respect to another object

  – Ground stations

  – Landing sites, sites along a rover path

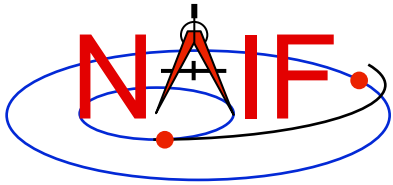 – Relative positions of manipulator joints, etc.

DAFCAT

- *dafcat* concatenates binary DAF files by copying all data segments from all input files, in the order they are provided, into the output file

  – Works on SPKs, CKs, and binary PCKs

  » Will not merge different types of kernels together, i.e. will not merge SPKs with CKs, CKs with PCKs, etc.

  » For merging SPKs, in most cases *spkmerge* should be used instead because it provides a much more powerful and sophisticated capability

- *dafcat* does not put any information into the comment area of the output file, leaving this responsibility to the user (use *commnt* to do so)
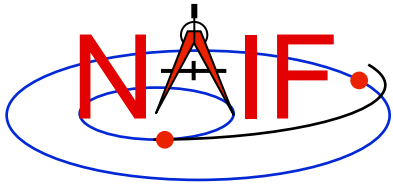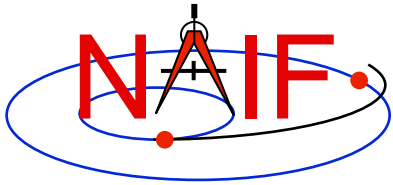
BSPIDMOD

- *bspidmod* alters the object IDs in a binary SPK file

    - Can be used to modify IDs in an SPK file(s) produced with a "bogus" spacecraft ID (or a simulation spacecraft ID)

    - Can be used to replace "good" IDs with "bogus" IDs if two different trajectories for the same object need to be used in the same program at the same time (for example for comparison)
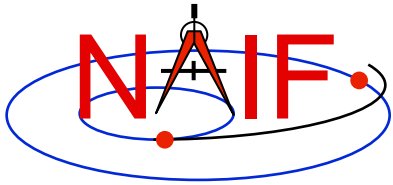
DAFMOD

- *dafmod* alters selected segment attributes in an SPK, CK, or binary PCK
  - In an SPK file it can alter the target, center, or reference frame ID
  - In a CK or binary PCK file it can alter the object or reference frame ID
- *dafmod* puts into the comment area a warning note stating which items in which segments of the file were changed
- *dafmod* works only on files in native binary format
  - *bingo* may be used to convert a non-native binary kernel to native binary format

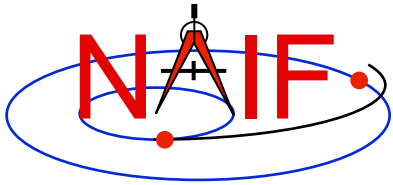- NAIF recommends you use *bspidmod* instead of *dafmod* whenever possible; it's safer to use

SPY

- *Spy* is a command-line utility for validating, inspecting, and analyzing SPK files
- Spy can:
  - Check SPK files
    - » Validate SPK structure
    - » Check sampled data for bounds violations
    - » Locate invalid double precision numbers
  - Sample data from a set of loaded kernels
    - » Sample position, distance, velocity, derived velocity, speed, acceleration, acceleration magnitude, osculating elements
  - Dump SPK file contents
    - » Data
    - » Summary information
   - » Comment area
    - » Bookkeeping information

- *prediCkt* makes CK files from a set of orientation specification rules and schedules defining when these rules are to be applied

- See the "Making a CK Tutorial" for more details

CKSLICER

- *ckslicer* subsets a CK file

- **It is useful in the situation when:**
  - only a portion of a CK covering a short interval of time is needed
    - » For example when the whole CK is not needed and it takes up a lot of space
  - you wish to cut parts from a few CKs with the intent to merge them together
    - » For example, if reconstructed CKs from different sources have too much overlap to simply "cat" them together

- **A note stating which subset was extracted is put into the comment area of the output CK file**
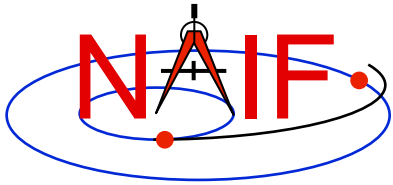
CKSPANIT

- *ckspanit* modifies interpolation interval information in Type 3 CK segments

  - It can also convert a Type 1 CK to a Type 2 or 3 CK

- *ckspanit* is useful when dealing with a Type 3 CK containing many small gaps within segments. It allows you to alter the CK in such a way that SPICE will interpolate over those gaps

CAUTION: before running *ckspanit*, make sure that interpolation over larger gaps is appropriate for the vehicle or structure you are dealing with. And don't forget to add appropriate comments to the newly created CK file.
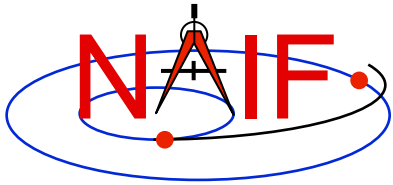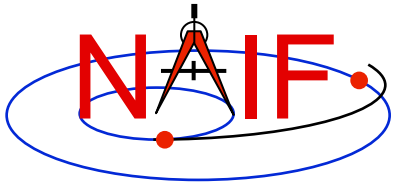
Navigation and Ancillary Information Facility

- *cksmrg* merges data from two or more uniform CK segments (same ID, base frame and type) provided in a single CK file

- It is used for:

  – eliminating gaps between segments(that cannot be removed by *ckspanit*

 – removing duplicate data points contained in different segments

CAUTION: cksmrg should not be used to merge CKs from different sources, nor should it be used to merge overlapping predict CKs
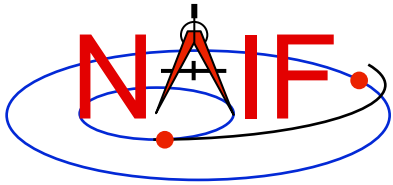
MAKCLK

- *makclk* converts a "SCLKSCET" file to a SPICE SCLK kernel

  – SCLKSCET is a time correlation file used by most JPL missions

    » It is an ASCII text file providing piece-wise linear clock correlation function as an array of triplets consisting of the reference on-board time, the reference UTC time and the clock rate

    » NAIF found that in many cases it is much easier to write an application to first make a SCLKSCET file and then convert it to an SCLK kernel using *makclk* than to write an application to make an SCLK kernel from "scratch"

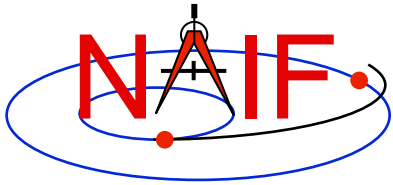- Very few people will need to use this program
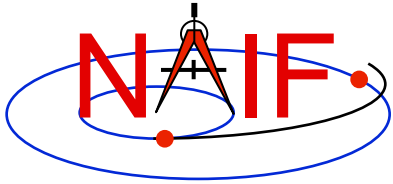
ORBNUM

- *orbnum* generates a SPICE orbit number file

    - Contains orbit start/stop times and orbit numbers

    - Also contains some derived quantities (orbital elements and coordinates of sub-spacecraft and sub-solar points)

    - The orbit number increment can be specified as occurring at one of these events: periapsis or apoapsis, ascending or descending equatorial node crossing, min or max value for the s/c position's Z-coordinate, or min or max value of the s/c latitude

- An orbnum file in not considered a SPICE kernel

    - It's just a convenient, derived product that NAIF offers to make for orbital missions that wish to have it

OPTIKS

- *optiks* generates information about instrument fields of view (FOV) from parameters present in IK and FK files
  - The FOVs must be defined using the keywords required by the GETFOV routine

- The output data are organized in three tables
  - The first table lists the angular extents (size) of circular, elliptical, and rectangular FOVs. Using command line options "-units" and "-half" the user can select the unit of measure for the angular measurements, and whether half or full FOV angular extents are listed
  - The second table contains FOV boresights in a user specified frame at a particular epoch, specified using the "-epoch" option
  - The third shows FOV boundary vectors and boresights as returned from the GETFOV API, or unitized and rotated into a user-specified frame at a particular epoch
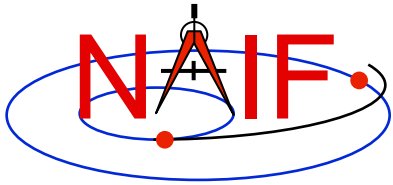
```
Terminal Window                                                    ☒

$ optiks -frame CASSINI_SC_COORD cas_iss_v09.ti cas_v37.tf naif0007.tls
cas00084.tsc

. . .

Kernels Loaded:

. . .

FOV full-angular extents computed in RADIANS


Field of View          Shape              Length           Width
-------------          -----              ------           -----

CASSINI_ISS_NAC        RECTANGULAR        +0.006108652382  +0.006108652382
CASSINI_ISS_NAC_RAD    CIRCULAR           +3.141592653590  +3.141592653590
CASSINI_ISS_WAC        RECTANGULAR        +0.060737457969  +0.060737457969
CASSINI_ISS_WAC_RAD    CIRCULAR           +3.141592653590  +3.141592653590


FOV boresights computed at epoch 2001-JAN-01 12:00
FOV boresights computed in frame CASSINI_SC_COORD
Field of View          Boresight Vector
-------------          ----------------
CASSINI_ISS_NAC        ( +0.000575958621, -0.999999819520, -0.000170972424 )
CASSINI_ISS_NAC_RAD    ( +1.000000000000, -0.000000000000, +0.000000000000 )
CASSINI_ISS_WAC        ( +0.001218344236, -0.999999225446, +0.000254451360 )
CASSINI_ISS_WAC_RAD    ( +1.000000000000, -0.000000000000, +0.000000000000 )
```

Three more utilities

- **archtype**
  - Displays the file architecture and type of a SPICE kernel; it is useful for scripting applications

- **bff**
  - Displays the binary file format of one or a few SPICE kernels

- **bingo**
  - Converts binary SPICE kernels between IEEE big endian and little endian formats
  - Converts text format SPICE kernels between DOS and UNIX text formats