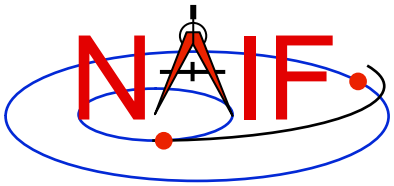
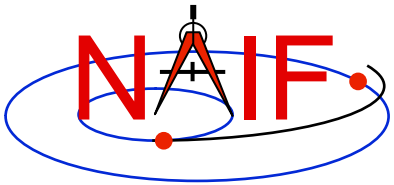


Navigation and Ancillary Information Facility

Shape Model Subsystem Preview

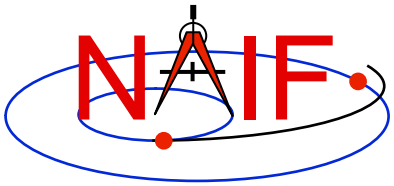
March 2010



SPICE DSK Topics

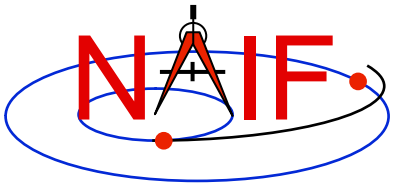
Navigation and Ancillary Information Facility

- Overview
- Requirements
- DSK Data Representations
- DSK System Components
- DSK Software Components
- DSK API Examples
- Using Shape with Orientation Data
- DSK Development Status



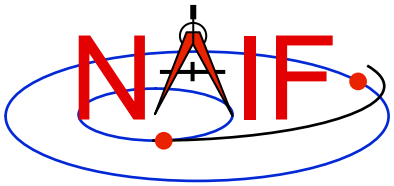
Overview

- NAIF is developing a new SPICE kernel type: DSK ("digital shape kernel")
- The SPICE DSK system deals with data sets describing topography of solar system objects, or more generally, shapes of 3-dimensional objects. Examples:
  - Digital elevation models (DEM) for the surfaces of Mars or the Moon
  - Tessellated plate model for the surface of a natural satellite, asteroid or comet nucleus
- The DSK system facilitates high-accuracy, SPICE-based geometric computations using "complex" shape data
  - Currently SPICE uses only triaxial ellipsoid shape models, which support low-accuracy computations



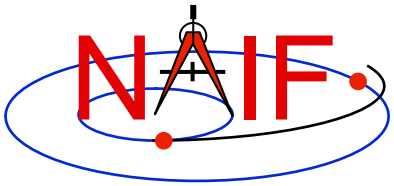
Requirements -1

- All “requirements” listed here are of an informal nature
  - Derived from customer interaction and NAIF team members’ experience using SPICE
- Overall requirement: facilitate high-accuracy geometry computations involving surfaces of extended bodies.
- Examples of computations that should be supported:
  - » Location of “sub-observer point” and height of observer above surface
  - » Ray-surface intercept point
  - » Occultation/transit state of a point target
  - » Limb and terminator location
  - » Illumination angles at a specified surface point
  - » Determine if a target is in an instrument’s field of view (FOV)



Requirements -2

- System should support efficient random access data search
  - For example: for a given (LONGITUDE, LATITUDE) coordinate pair, return radius (distance from body center) of the corresponding surface point
- System should support rapid, high volume data extraction ("bulk read")
  - Required for efficient use by graphics applications
- System should be able to use data sets spread across multiple files
  - Some current data sets exceed 2Gbytes in size
  - Larger data sets should be expected in the future
  - Impractical to store all needed data in one file
- System should be able to work with models for different bodies simultaneously.
  - For example: support simultaneous use of data sets for Mars and Phobos.

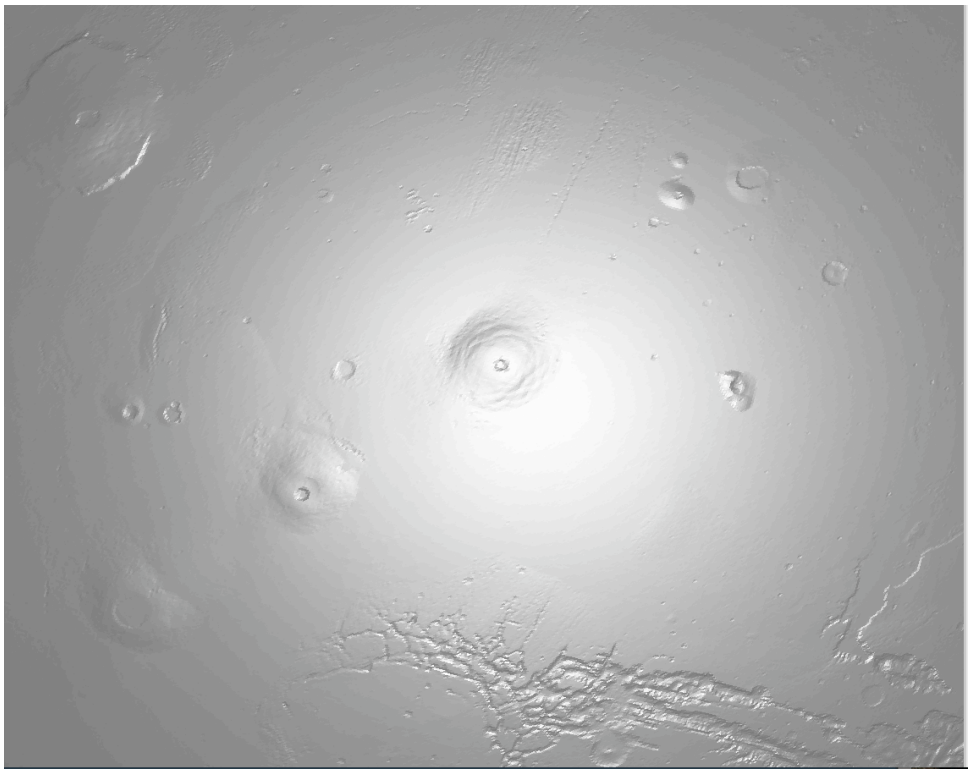


Requirements -3

- System should be able to work with multiple models for different parts of the surface of a specified body simultaneously.
  - Support simultaneous use of multiple data sets having different resolutions, or even different mathematical representations, for different regions of the surface.
- Data files should be portable
- Data files should support inclusion of metadata
- Tools should be provided for:
  - summarizing contents of data files
  - accessing metadata in data files
  - merging or subsetting data files
  - ingesting data from other types of files
    - » For example: Bob Gaskell's and Peter Thomas' shape models

DSK Data Representations -1

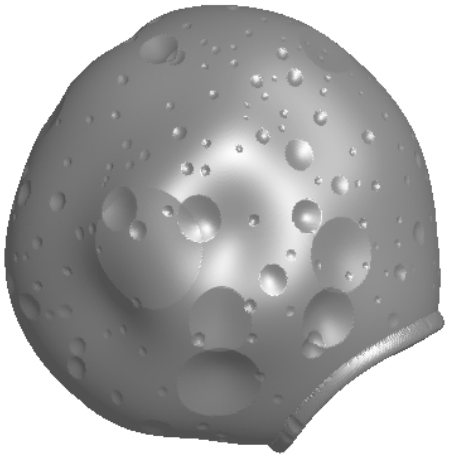
- **Digital elevation model (DEM)**
  - **Maps longitude/latitude to "elevation"**
    - » **Elevation of a surface point can be defined as distance from the origin of a body-fixed reference frame**
    - » **Elevation can be defined as height above a reference ellipsoid**
  - **Example: image created from MGS laser altimeter (MOLA) Mars DEM**



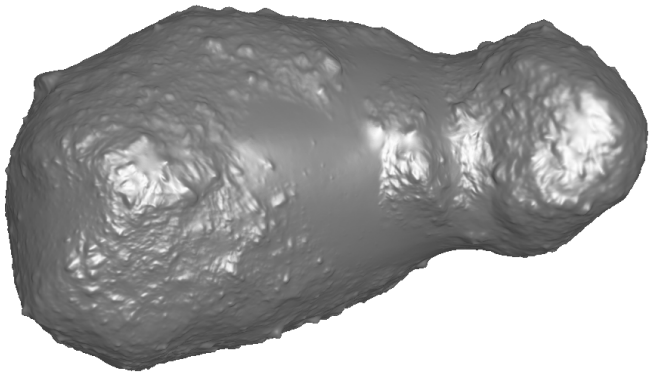
DSK Data Representations -2

Navigation and Ancillary Information Facility

- **Plate model**
  - **Surface of object is represented as a collection of triangular plates**
  - **More flexible than digital elevation model: arbitrary 3-D surface can be modeled**
    - » **Surface could be a complicated shape with multiple surface points having the same latitude and longitude**
      - **Examples: "dumbbell"-shaped asteroid, caves, arches**
  - **Less efficient than digital elevation model of similar resolution in terms of storage and computational speed**



Phobos

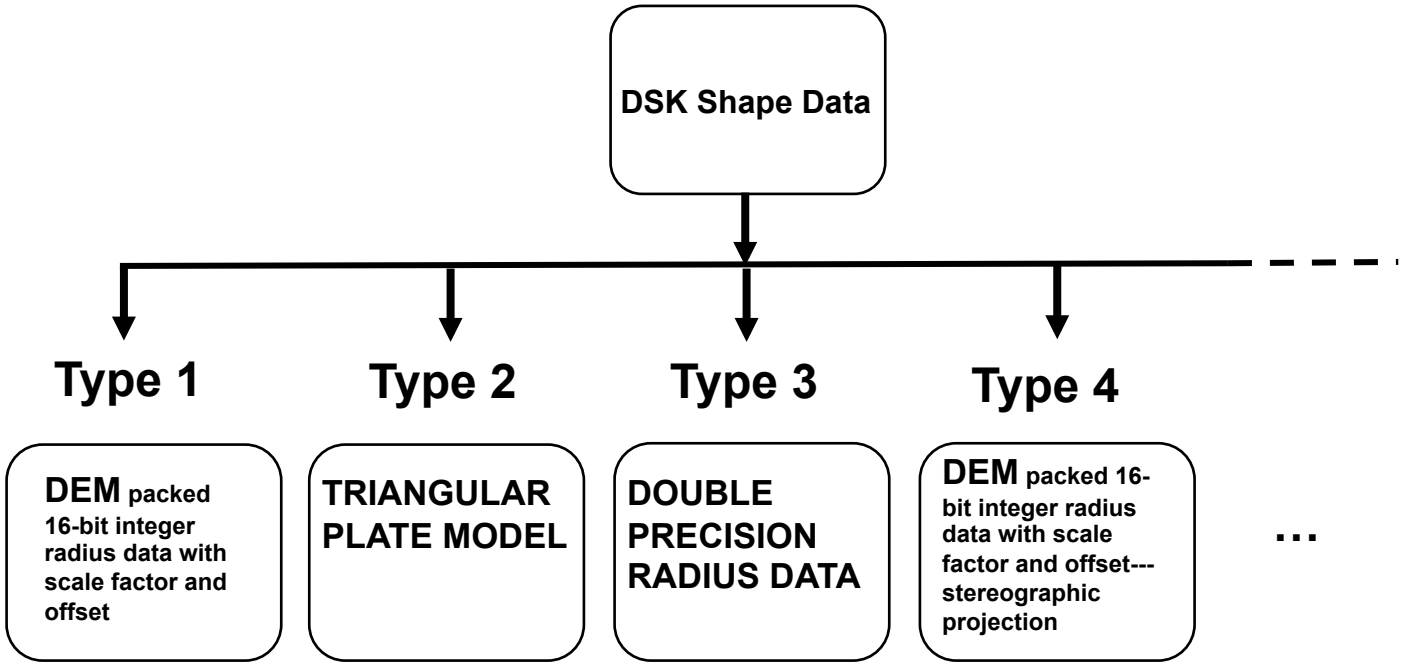


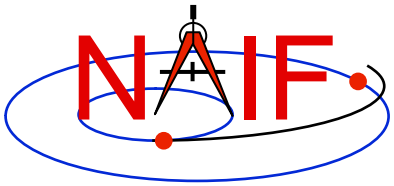
Itokowa

DSK Data Representations -3

Navigation and Ancillary Information Facility

- **DSK shape representations are polymorphic:**
  - DSK shape representations are called "Data Types."
  - Each data type has its own mathematical representation of a surface
  - Each data type has associated software that implements common functionality, such as the ability to return a radius (distance of surface point from body center) value for a specified latitude and longitude.
  - Each data type may have additional, unique functionality.
    - » For example, type 2 has accessor routines that return plate and vertex data. These functions are not applicable to other data types.





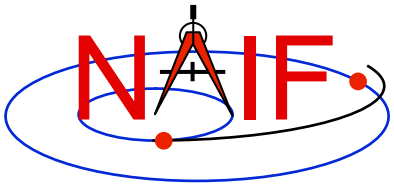
DSK System Components

- **DSK Files**
  - Use the SPICE DAS file architecture
    - » Binary, direct access
    - » System-independent buffering built in
    - » Comment area built in
- **DSK Software**
  - SPICE software which enables users to create and use DSK kernels
    - » Writer routines
    - » Reader routines
    - » High-level API routines
      - For example: routines dealing with observer-target geometry
    - » Supporting utility programs

DSK Software Components -1

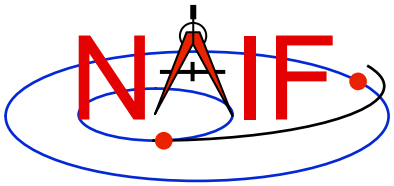
Navigation and Ancillary Information Facility

- **Writers**
  - Routines that enable a SPICE-based application to create a DSK kernel
    - » Open new DSK kernel for write access
    - » Open existing DSK kernel for write access
    - » Start new DSK segment ("segments" are partial DSK data sets containing data for a given region on a specified object)
    - » Add data to DSK segment
- **Readers**
  - Routines that extract data from a DSK file
    - » Return elevation of surface at given longitude/latitude
    - » Return specified attributes, for example the surface normal vector, for a specified longitude and latitude
    - » Rapidly obtain data for large portion of surface ("bulk read")
    - » Return DSK attributes such as number of plates, pixel size, min/max elevation, etc.
- **High-level functions (including, but not limited to, the following):**
  - Compute sub-observer point on surface and height of observer above surface
  - Compute intercept of ray with surface
  - Determine whether a portion of a target body's surface is within the FOV of specified instrument at specified time.
  - Determine occultation/transit state of a point target
  - Compute limb and terminator location
  - Compute Illumination angles at a specified surface point



DSK Software Components -2

- **Utility programs that**
  - **Create DSK files: import other surface shape data sets into SPICE DSK format**
  - **Port DSK files**
  - **Provide comment area access**
  - **Summarize DSK file contents**
  - **Subset or merge DSK files**
  - **Downsample DSK files**
  - **Convert one DSK data type to another**
    - » **Example: create type 2 DSK file from type 1**



DSK API Examples

Navigation and Ancillary Information Facility

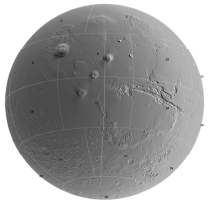
- **Get radius at surface point (inputs are in red, outputs in blue):**
  - **CALL DSKRAD (TARGET, LON, LAT, RADIUS)**
    - » **Inputs: target body name, longitude and latitude of point of interest**
    - » **Output: radius (distance from target center) at surface point**
- **Find sub-observer point on target:**
  - **CALL SUBPT (METHOD, TARGET, ET, ABCORR, OBSRVR, SPOINT, ALT)**
    - » **SUBPT is a generic, high-level API. SUBPT doesn't assume the surface is modeled by a DSK.**
    - » **Input "METHOD" indicates surface model and sub-point definition**
      - **For ellipsoids, METHOD may be set to 'near point' or 'intercept'**
      - **For DSKs, set METHOD to 'DSK intercept', indicating that the sub-point is defined as the closest intersection to the observer of the observer-target center ray with the surface, and DSK model is to be used.**
      - **Note that SPICE should not assume DSK is to be used just because a DSK for the target body is loaded; may be too inefficient for some applications. Caller must say which model is to be used.**
    - » **Other inputs: target body name, epoch, aberration correction, observer name.**
    - » **Outputs: sub-observer point in Cartesian coordinates, expressed in the body-fixed frame associated with the target, and altitude of the observer above the sub-point.**

Writing Shape and Orientation Kernels

Navigation and Ancillary Information Facility

LAT/LON and height above ellipsoid or distance from center of frame

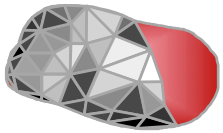
→ **MKDSK Program** (SPICE Toolkit) →



Digital Terrain Shape Model

Lists of plate model vertices and associated plates, and optionally, albedo data for each plate

→ **MKDSK Program** (SPICE Toolkit) →



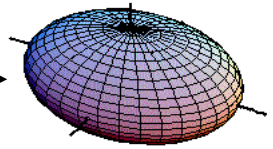
Tessellated Plates Shape Model

DSK

Digital shape kernel

Axes dimensions for tri-axial ellipsoid

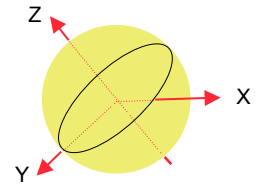
→ **Text editor** (Usually done by NAIF) →



Triaxial Ellipsoid Shape Model

Some source of rotation state information (pole RA/DEC and prime meridian location)

→ **Text editor** (Usually done by NAIF) →



Orientation

PCK

Planetary constants kernel containing rotation data for the body, and possibly tri-axial shape

Using Shape and Orientation Kernels

Navigation and Ancillary Information Facility

LAT/LON and height above ellipsoid or distance from center of frame

Program to make a DSK

(SPICE Toolkit)

Lists of plate model vertices and associated plates, and optionally, albedo data for each plate

MKPLAT Program

(SPICE Toolkit)

Axes dimensions for tri-axial ellipsoid

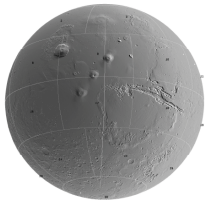
Text editor

(Usually done by NAIF)

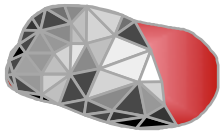
Some source of rotation state information (pole RA/DEC and prime meridian location)

Text editor

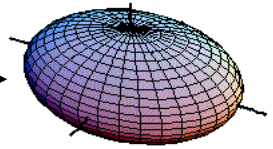
(Usually done by NAIF)



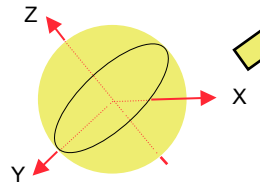
Digital Terrain Shape Model



Tessellated Plates Shape Model



Triaxial Ellipsoid Shape Model



Orientation

DSK

Digital shape kernel

Pick one* shape model

Shape

* Sometimes the triaxial model is needed in addition to one of the other two

Your Application Program

spicelib, cspice, Icy or Mice

Orientation

SPICE routines for obtaining rotation state and shape, and then computing derived quantities

PCK

Planetary constants kernel containing rotation data for the body, and possibly tri-axial shape

Other data as needed

DSK Development Status

Navigation and Ancillary Information Facility

- **History**
  - Precursor "Plate Model" system was delivered to NEAR and Hayabusa and used successfully on those missions
  - A prototype version of the DSK system was delivered to the DAWN project in November 2006. This software has been integrated into the SOA (Science Opportunity Analyzer) program.
    - » SOA uses DSK files to import shape model data for Vesta and Ceres.
    - » SOA also uses DSK software, along with custom, higher-level DSK-based software provided by NAIF, to perform geometric computations involving target body shape data.
  - This prototype has also been provided to a number of other interested groups.
- **Plans**
  - Development of the full DSK subsystem had been stalled, but has now started up again.
  - Release date of a full beta-test version of the DSK system is TBD.