SPICE: A REAL EXAMPLE OF DATA SYSTEM RE-USE TO REDUCE THE COSTS OF GROUND DATA SYSTEMS DEVELOPMENT AND MISSION OPERATIONS

Charles Acton, Nat Bachman, Lee Elson, Boris Semenov, Edward Wright

California Institute of Technology, Jet Propulsion Laboratory, 4800 Oak Grove Dr., Pasadena, CA 91109
Charles.H.Acton (at) jpl.nasa.gov

ABSTRACT

NASA's SPICE ancillary information system is used by engineers and scientists to capture and access a variety of space mission navigation and other ancillary information throughout the mission lifecycle. SPICE is used by mission architects in defining and analyzing a proposed mission. It is used by scientists to help evaluate a mission design, and to plan scientific observations. It is used by engineers to plan mission operations and to evaluate spacecraft operations. It is used by science instrument teams to help plan observations and produce data products. And it is used by the worldwide science community to help find archived science data of interest and to further analyze those data.

A primary requirement placed on the SPICE system is emphasis on multimission design, based on software and data structures that can be easily re-used for a wide set of functions. Similar strong emphasis is given to software portability, broad applicability, and quality control achieved through extensive testing. SPICE has been very successful in meeting these requirements, resulting in substantial community acceptance and very modest incremental cost for adaptation to a new mission.

1. INTRODUCTION

Scientists and engineers face ever-increasing pressure from NASA and other national space agencies to reduce the cost and to speed up the pace of flying space science missions. Infusion of new flight system technology to achieve more capability at less cost is an important component of this initiative. But re-use of standard, proven space system components–especially ground software system components–can also contribute to achieving cost containment while reducing risk. It behooves space agencies to find the right balance between using the new and the old in conducting affordable, successful missions.

An example of where re-use of "standard" software seems a good choice to contain costs–as well as to improve computational accuracy and customer productivity–is in the ancillary data sub-domain that is focused on computing observation geometry. Here, the basic equations for computing such geometry are pretty much fixed: there's not much new in the way one computes ranges, lighting angles or times of occultation. The re-use of proven software, with emphasis on making this use easier and more broadly applied, is the objective of the Navigation and Ancillary Information Facility (NAIF) team at the Jet Propulsion Laboratory.

2. ANCILLARY DATA SYSTEM STANDARDS

Why Establish Ancillary Data Standards?

Ancillary data standards help promote the exchange of good ideas for mission design and the validation of a selected design. They help distributed team members participate in constructing detailed observation sequence designs, or in simply understanding what the planned observations are. They help maximize the complete and precise interpretation of scientific data returned from those observations, including when cross correlation between datasets is attempted. They facilitate the access to such data by other scientists who might have new ideas or improved methods for analyzing old data. And they can reduce local data system development costs, particularly as re-use comes into play.

What Vehicles Should Be Supported?

The challenges of planning observations and providing ancillary data to help analyze the data returned from those observations are associated with every kind of robotic vehicle in use today and planned for the future exploration of the solar system–interplanetary spacecraft, orbiters, landers, rovers, balloons and airplanes. Vehicles examining the earth and the solar environment are equally good candidates for use of such standards.

Prime Components of an Ancillary Information System

Data are the fundamental component of an ancillary information system. Examples of ancillary data are vehicle position and velocity; target body size, shape and orientation; vehicle or instrument pointing; instrument aperture size, shape and orientation; and logs of observation plans, spacecraft and instrument commands, and notes detailing how things worked. Reference systems are another key component: coordinate systems and time systems are prime examples. Documents providing precise and complete definitions of the data and reference systems are a third component. Archives that provide easy and timely access to the data and other information system components are important and must exist for pre-flight, mission operations, and post-flight long-term data analysis phases.

Software that helps a scientist find, acquire and utilize ancillary data should also be considered a prime information system component. Adding software to this "mix" makes the job of building an ancillary information system much larger, but if done properly the payoff is well worth the extra investment. This software suite could include general application programs, utility programs and scripts, and subroutine libraries used by scientists in building their own applications.

Requirements on Ancillary Information System Components

A number of requirements are mandatory to ensure meeting expectations of the large and diverse customer community. Portable: data files and software must be useable on and easily moved between all popular computing platforms. Extensible: it must be easy to add or extend functionality. Correct: all components must be thoroughly tested and validated, including peer review where appropriate. Precise: generally all calculations must be done to meet the needs of the most demanding customer; the use of approximations must be carefully controlled. Documented: data and software must be clearly, fully documented. Convenient: all components must be freely available and easily obtained by all interested parties. Supported: professional help for customers must be available. It also seems that providing a fully open system–providing source code for software–invites extra confidence and participation of the user community.

3. SPICE SYSTEM COMPONENTS

In 1982 the U.S. space science community made a strong recommendation [1] to NASA for the establishment of new information systems and their attendant standards to help make ready access to, and use of, archived scientific data become more of an expectation and less of a dream. While most of the discussion focused on science data obtained directly from instruments, included in this report was a recommendation for the establishment of standards and processes for obtaining and using the relevant ancillary/engineering data needed to help fully understand science instrument measurements, and to correlate results across instruments and missions. Out of this recommendation was born NASA's "SPICE" ancillary information system. The SPICE implementation strives to address the suggested requirements outlined earlier.

The primary SPICE data sets are often called "kernels." SPICE kernels comprise position, orientation and related ancillary information that has been organized and formatted for easy access and correct use by the space science and engineering communities. SPICE kernel contents are summarized in Table 1.

S	**S**pacecraft ephemeris, or more generally, state (position and velocity) of an observer, given as a function of time.
P	**P**lanet, satellite, comet, or asteroid ephemerides, or more generally, location of a target body, given as a function of time. The P kernel also logically includes certain physical, dynamical and cartographic constants for target bodies, such as size and shape specifications, and orientation of the spin axis and prime meridian.
I	**I**nstrument description kernel, containing descriptive and operational data peculiar to a particular scientific instrument, such as field-of-view model parameters and internal timing relative to the spacecraft clock.
C	Pointing kernel, containing a transformation traditionally called the **C**-matrix that provides time-tagged pointing (orientation) angles for a spacecraft structure upon which science instruments are mounted.
E	**E**vents kernel, summarizing mission activities–both planned and unanticipated. Three distinct classes of events data are defined: Science Plans, Sequences, and Experiment Notes.

Table 1 Primary SPICE Kernel Contents

Perhaps the "SPICE" acronym should have been "SPICES," with the final "S" standing for **S**oftware. The SPICE system includes the SPICE Toolkit, a large collection of observation geometry software available in both ANSI FORTRAN 77 and ANSI C. The principal component of this Toolkit is a library of portable routines used:

- to write kernels,

- to read kernels, and
- to calculate many commonly used observation geometry parameters derived from data provided in the kernels.

Customers integrate these SPICE Toolkit routines into their own application programs to compute observation geometry parameters and similar ancillary information.

Also parts of the SPICE Toolkit are the following components:

- Utility programs
  - for summarizing and managing SPICE data files
- Cookbook programs
  - provide basic examples of using SPICE
  - Toolkit software with SPICE data
- Documentation
  - extensive tutorials, user guides, reference documents and source code comments

4. USING SPICE

A SPICE customer integrates needed SPICE Toolkit routines into an application program. These routines include "readers" for needed SPICE files as well as routines that compute derived quantities. Combined with the user's own routines a complete, focused application is constructed, Figure 1. An update to a SPICE kernel could be one result.

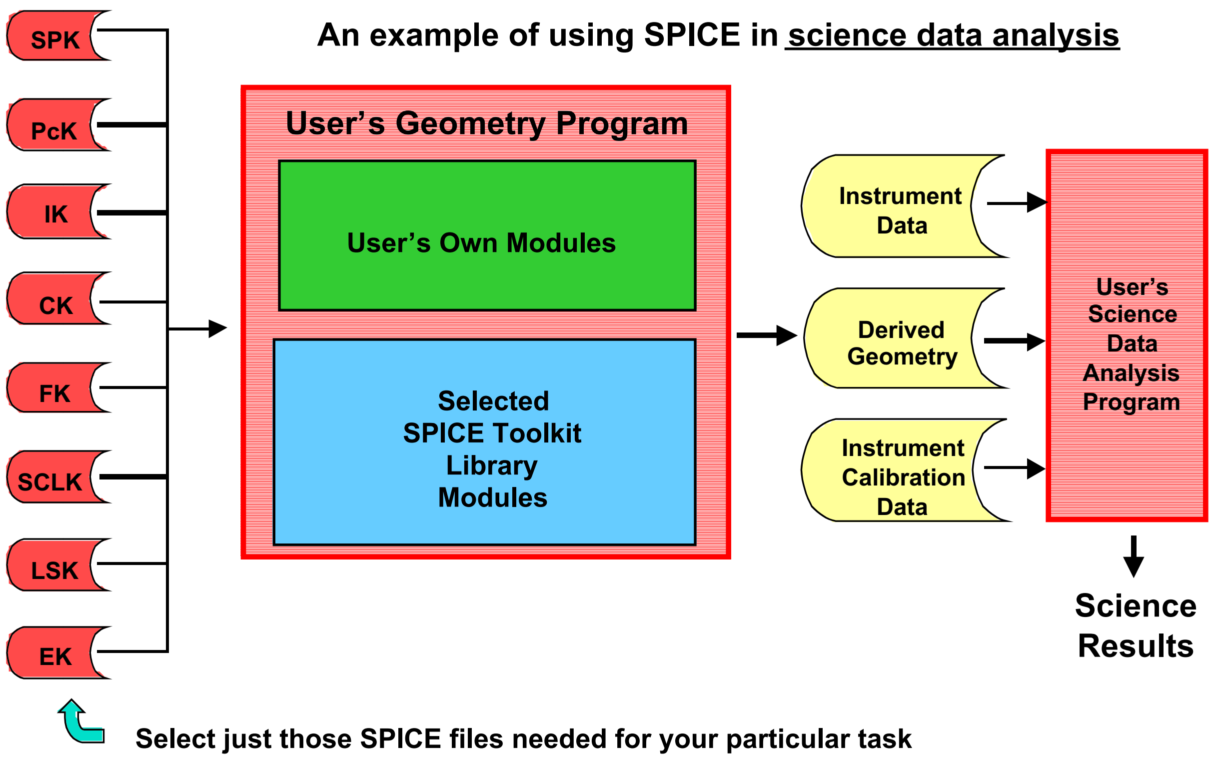

Figure 1 Integrating SPICE Software into an Application Program

The various elements of SPICE are used at JPL and numerous other institutions in mission design, observation planning, mission operations (including visualization), and science data analysis. The SPICE Toolkit is available in both ANSI FORTRAN 77 and ANSI C, and is supported on most popular platforms, including PC, Macintosh, Sun, SGI, HP, DEC Alpha and VAX.

The ephemeris component of SPICE is now a standard for customers of NASA's Deep Space Network, where it is used in long and short term scheduling of DSN antennas and in pointing these antennas and tuning transmitters and receivers during tracking passes.

The SPICE system has been evolving for over ten years, with new and improved capabilities being added while never eliminating or changing the operation of previously released functions. Recent new features include software to improve the portability of binary-format SPICE files and a high precision earth orientation capability. Currently under development are a generic sky catalog, tools for creating spacecraft orientation files (CK), a method for creating virtual ephemeris and orientation files

(SPK and CK) during the execution of an application program, the ability to construct dynamically defined reference frames, the inclusion of a suite of observation geometry event-finding modules (e.g. find all the epochs of periapsis passage, or find time intervals when the spacecraft altitude is within user-specified bounds), and provision of a convenient means for calling SPICE functions from the Interactive Data Language (IDL®) programming environment.

SPICE has many applications in space science. Depicted in Figure 2 is the use of SPICE to coordinate time, orientations and locations of a disparate suite of natural and man-made "objects," including the sun, earth and Mars, a Mars orbiting spacecraft, and a Mars lander and rover. Providing such positions, velocities and orientations, and the transformations between the reference systems in which the measurements are made, is routine work using SPICE.

5. Evolving SPICE to Meet Future Needs

The current version of SPICE is quite mature and has a great deal of capability, but more work remains to be done. Improvements can help to further reduce the cost of conducting scientifically challenging space missions while improving the science payoff from each mission. Examples of possible extensions to SPICE include:

- integration of digital shape models
- design and implementation of a tightly coupled surface features database
- design and implementation of a more robust instrument model
- design and implementation of routines to search for specified geometric conditions
- design and implementation of means to incorporate and properly utilize trajectory/orbit accuracy information

Examples of possible application programs to be added to the SPICE family include:

- an orbit characterization program
- tools to facilitate cooperative mission/observation planning in a distributed environment
- tools to facilitate planning for relay links
- visualization tools

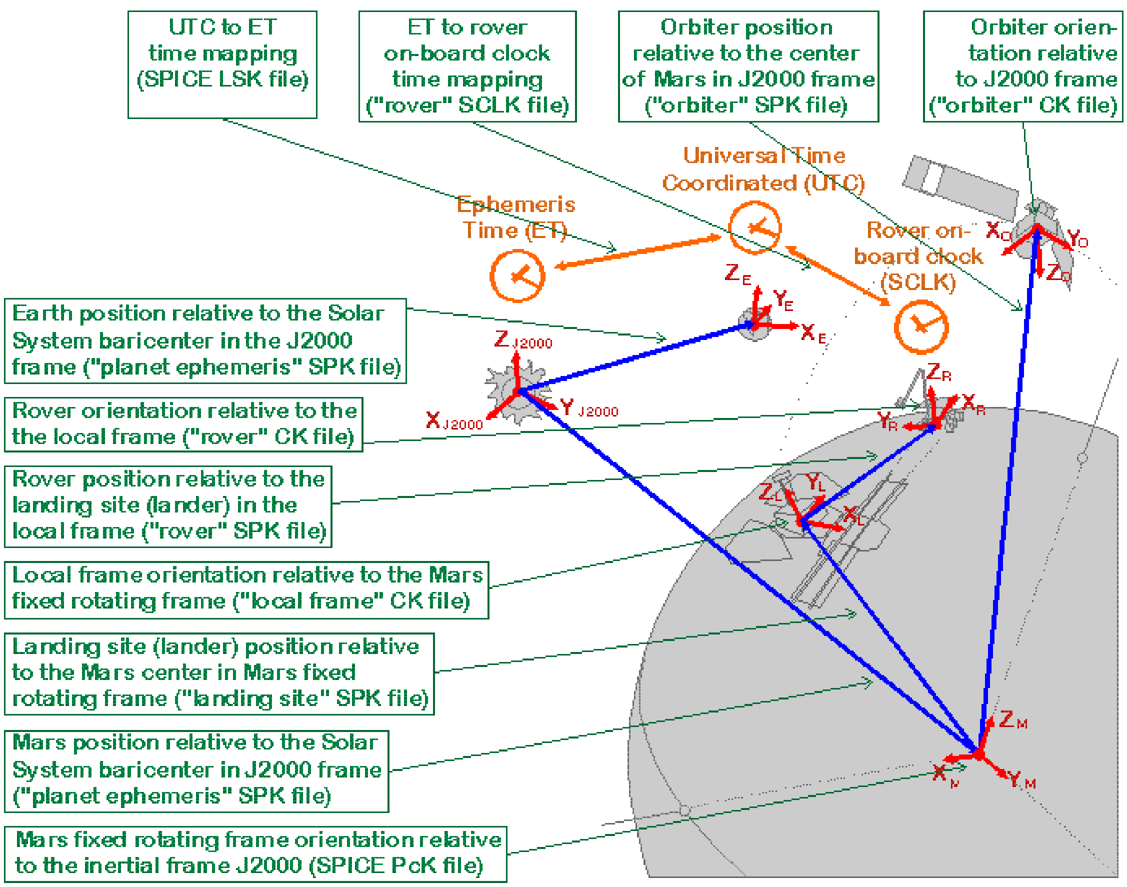

Figure 2 Connecting Geometry and Time Using SPICE

Examples of possible new data management capabilities include:

- implementation of a means for aggregating a collection of files into an information "library"
- implementation of a web-based SPICE products selection and distribution mechanism
- establishment and operation of mirror sites

Especially of interest to the user community is the provision of new methods of interfacing to SPICE. Users have requested:

- additional application program interface mechanisms, such as perl, Java and Matlab®;
- fully re-casting SPICE in an object-oriented language such as C++; and
- provision of a full-function SPICE "geometry engine" available in a client-server interface architecture.

The SPICE system is in use on a large assortment of space science missions as indicated in Table 2. While the principle use of SPICE has been in the planetary science discipline, astrophysics, space physics and Earth science projects are also using this technology.

Restorations	Past Customers	Current Customers	Pending
Apollo 15, 16 [P]	Magellan [P]	Galileo	Messenger
Mariner 9 [P]	Clementine (NRL)	Mars Global Surveyor	
Mariner 10 [P]	Mars Observer	Stardust	
Viking Orbiters [P]	Mars 96 (Russia)	Cassini/Huygens	**Future Possibilities**
Pioner 10/11 [P]	Hubble Telescope [S]	Mars Odyssey	NASA Mars Program
Haley armada [P]	ISO [S]	Mars Exploration Rover	Discovery Program
Phobos 2 [P] (Russia)	MSTI-3 (by ACT Corp.)	SIRTF [P]	New Horizons Prgm
Ulysses [P]	OTD (by MSFC)	Genesis	Explorers Program
Voyagers [P]	Mars Pathfinder	Mars Express (ESA)	Space Interferometry
	Mars Climate Orbiter	Deep Impact	
	Mars Polar Lander	Mars Recon. Orbiter	**???**
	NEAR		Muses-C (Japan)
	Deep Space 1	DSN Metric Predicts [S]	Rosetta (ESA)
	CONTOUR	Planetary Data System	
	Space VLBI [P]		

Table 2 Missions and Activities Using SPICE

[P] = partial use of SPICE [S] = special SPICE-based products

The NAIF Team believes there is room for new applications of these standards in support of space science endeavors. NAIF solicits suggestions from the community for such new applications as well as interest in teaming with NAIF in their implementations.

The authors wish to thank the project and program managers at JPL and NASA who have believed in this idea and provided the funding and opportunities for using SPICE. Thanks are also due to the hundreds of scientists and engineers around the world who have taken the chance on SPICE–investing precious time to learn how to use it and then supporting and guiding its further development.

6. REFERENCES

[1] Data Management and Computation, Volume 1: Issues and Recommendations, 1982, National Academy Press.

Development of the SPICE system was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.