Multimission Software Interface Specification (SIS)

SPICE
Events Kernel

EK

NAIF Document No. 371
Version 1.0

Prepared by: C. Acton

Navigation and Ancillary Information Facility (NAIF)
Jet Propulsion Laboratory
National Aeronautics and Space Administration

PURPOSE: This SIS describes the SPICE Events Kernel (EK) component files. The EK captures several kinds of "events" information useful to scientists and engineers in understanding what happened during the course of a mission.

CHANGE LOG

Version	Date	Page Nos.	Reason
1.0	11 SEP 2000	All	New multimission version.

List of Acronyms

ANSI	American National Standards Institute
ASCII	American Standard Code for Information Interchange
EK	Events Kernel
ENB	Events Kernel - Notebook Component
ESP	Events Kernel - Science Plan Component
ESQ	Events Kernel - Sequence Component
ET	Ephemeris Time
JPL	Caltech/Jet Propulsion Laboratory
FTP	File Transfer Protocol
NAIF	Navigation and Ancillary Information Facility
PDB	Project Data Base
PDS	Planetary Data System
SIS	Software Interface Specification
SPICE	S-, P-, I-, C- and E-kernels; the principal logical data components of a particular NASA ancillary information system
TDB	Barycentric Dynamical Time
TMOD	Telecommunications and Mission Operations Directorate
WWW	World Wide Web

SECTION 1
GENERAL DESCRIPTION

1.1 Purpose of Document

This Software Interface Specification (SIS) module describes the methods for producing and using Events kernel files, including use of related programs and ANSI FORTRAN 77 or C modules provided in the SPICE Toolkit. It outlines the kinds of information that is stored in each of these components, although specific details are mission-specific.

1.2 Scope

This is a multimission SIS, applicable to all projects and processes that use the SPICE ancillary information system.

Because the SPICE E-kernel system is highly tailorable, this SIS and its associated Applicable Documents cannot provide specific details pertinent to any particular mission; rather this SIS describes the EK functions and capabilities. Each flight project develops specific instances of the EK components to meet its needs, while preserving the fundamental EK design and allied Toolkit software.

1.3 Applicable Documents

No.	Name	NAIF Doc. No.
1	E-Kernel Required Reading	368
2	DAS Required Reading	367
3	PEF2EK User's Guide	280
4	INSPEKT User's Guide	384
5	EK - Introduction	SPICE Tutorial
6	EK - ESP	SPICE Tutorial
7	EK - ESQ	SPICE Tutorial
8	EK - ENB	SPICE Tutorial

Document Nos. 1-4 are contained in every distribution of the SPICE Toolkit. Document Nos. 5-8 are available from the NAIF server:

ftp://naif.jpl.nasa.gov/pub/naif/toolkit_docs/Tutorials/

or by contacting any NAIF staff member.

1.4 Functional Description

The SPICE E-kernel is a repository for instrument commands, spacecraft commands and a broad collection of additional “event”-type information that could be useful to scientists interpreting data returned from spacecraft instruments, and to engineers engaged in operation and analysis of the host spacecraft. The E-kernel can help instrument team members during the operations phase of a mission, and it will help scientists who will study flight project data after the mission has concluded. The E-kernel includes contributions provided by mission operations teams and scientists. Some E-kernel contributions are predictive in nature, while other contributions consist of after-the-fact information reflecting “what really happened.” An overview of the entire E-kernel system is provided in Applicable Document No 5.

The E-kernel has three distinct components.

SCIENCE PLAN (ESP) Contains statements of scientific objectives for observations scheduled over a specified period of time.

SEQUENCE (ESQ) Contains instrument commands, spacecraft commands and planned ground system events intended to carry out the observation objectives of the mission (as described in the SCIENCE PLAN collection).

NOTEBOOK (ENB) Contains after-the-fact notes, provided by scientists and mission operations teams, about execution of the planned observations. Should include notes describing unplanned or unexplained occurrences that could affect how, or how well, scientific data are interpreted.

Overviews of these three EK components are found in Applicable Document Nos. 6, 7 and 8. While all three components deal with "events" information, the content, format and access methods for each component could be unique. In that sense this SIS might have been divided into three distinct SISs, but a having single document was the chosen documentation approach.

1.4.1 Data Source, Destination and Transfer Method

Inputs for the ESQ may come from any source in which relevant event information are available–a Predicted Events File (PEF) is one such example. Such data are typically found in a project database, such as a File Interchange System (FIS) or Data Object Manager (DOM), and are transferred to an ESQ production workstation using ftp.

Inputs for the ENB component originate with any member of a flight team and are provided using either a WWW or an e-mail interface.

Inputs for the ESP may follow the path used for either the ENB or ESQ subsystems.

1.4.2 Labeling and Identification

ESQ files may contain comments in the "comment area" of the file. ENB entries are self-labeled.

SPICE makes no restrictions on file naming convention, but individual flight projects may impose some rules, and NASA's Planetary Data System places restrictions (currently "27.3" on names of files placed on archive volumes.

SECTION 2
ENVIRONMENT

2.1 Hardware Characteristics and Limitations

All three components may exist on any platform supported by the SPICE Toolkit, except that the ENB mechanism also requires a WWW server and CGI interface.

2.2 Interface Medium and Characteristics

ESQ data are accessed by copying them to the user's workstation and using SPICE Toolkit software built for accessing ESQ files. ENB data are accessed using a WWW browser.

2.3 Calling Sequence

Data in an ESQ component file are accessed using subroutines provided in the SPICE Toolkit, or by using the INSPEKT program, also provided in this Toolkit. Instructions for using SPICE subroutines to read ESQ files are hound in the headers for the EK family of subroutines, found in the EK source code modules provided in each copy of the SPICE Toolkit. Details for using the INSPEKT program are found in the User Guide, provided as part of each Toolkit delivery.

Data in an ENB are accessed using a WWW browser.

2.4 Failure Protection, Detection and Recovery Features

SPICE software provided by NAIF contains built-in error detection and reporting mechanisms that provide a measure of protection against incorrect use.

2.5 Utility Programs

The SPICE Toolkit utility programs toxfr and tobin are used to convert an ESQ file from binary to encoded text format, and back again, to allow EK file transfers between heterogeneous computers. SPACIT may also be used for this purpose, as well as to produce a summary of the contents or display any internal labels and comments that had been placed within an ESQ component file. (These utilities have no function with respect to the ENB mechanism.) An ESQ query/report generation tool named INSPEKT is described in Applicable Document No. 4.

SECTION 3
DATA FLOW CHARACTERISTICS

3.1 Operational Characteristics

3.1.1 Generation Method

ESQ kernels are made using custom software that includes SPICE Toolkit EK writing routines. ESQs can be made from nearly any kind of "data" that contains possibly useful events information. One typical source for making ESQ files is an "as-run" Predicted Events File (PEF). The use of this program is described in Applicable Document No. 3.

The ENB is built up from individual "notebook" entries supplied by scientists and mission operations team members, using either e-mail or a WWW interface. Typically NAIF provides the web server and underlying software needed to store and display (serve) ENB contents.

AS currently implemented, there is no unique ESP data structure or allied software; ESP inputs may be made using either the ESQ or the ENB mechanism described above.

3.1.2 Time Span of Product

The time span and frequency of generation for inputs to the three EK components is determined by mutual agreement of the producers and users. There is no inherent restriction on time span for these products.

3.2 Data Volume

The volume of EK data for a given period of time will depend somewhat on science and mission operations team contributions—factors that are unknown in advance. It is expected that the SCIENCE PLAN and NOTEBOOK components will be relatively small compared to other mission products, probably not exceeding 1 Mbyte per month. A very rough estimate of the volume of EVENTS component files is somewhere between once and twice the volume of the PEF file (or some equivalent) corresponding to the same period.

SECTION 4
DETAILED DATA OBJECT DEFINITIONS

4.1 SEQUENCE Component (ESQ)

ESQ files are a specific instance of the SPICE Database Kernel (DBK). As this name suggests, ESQ files have a relational database organization. Like any SPICE DBK, ESQ files are built using an underlying custom data structure named Direct Access Segregated (DAS). Applicable Document No. 2 (DAS Required Reading) provides a complete description of the format of a DAS file. Applicable Document No. 1 (EK Required Reading) provides a detailed description of how the DAS structure is utilized in making an ESQ file. More importantly, this document also provides a detailed description of the SPICE Toolkit software provided for writing and reading ESQ files. To minimize documentation maintenance, none of that information is repeated here.

An ESQ query and report generation utility named INSPEKT is available in each SPICE Toolkit distribution. The operation of this software is described in Applicable Document No. 4.

The subroutines used by INSPEKT to query an EK file are also available to the user in SPICELIB or CSPICE; these may be integrated into any application program where event information is needed.

Both INSPEKT and the EK query subroutines use an SQL-like query language. An example of an EK query is shown below. The items shown in lower case are only examples, to be replaced with items ("column names") contained within a real ESQ file. Also, the SELECT and FROM clauses are the only ones absolutely required: all other directives, such as "WHERE" and "ORDER BY" are optional.

```
SELECT event_time, command_name, parameter_name, parameter_value
FROM sequence
WHERE command_name like “36*” and event_time between
“1985 Jan 1” and “January 1, 2000”
ORDER BY event_time DESCENDING, command_name
```

4.2 NOTEBOOK Component (ENB)

Inputs to the ENB mechanism may be provided using two methods: e-mail or World Wide Web (WWW).

4.2.1 NOTEBOOK Entries Using E-mail

The format used for e-mail-inputs to the E-kernel NOTEBOOK component (ENB) are shown below. It consists of a small amount of metadata provided using a KEYWORD = VALUE format, followed by free form ASCII text. MIME attachments may be included with an e-mail ENB entry.

```
\header
SYSTEM_NAME        = system ID
*PROTECTION        = PRIVATE (default is PUBLIC)
AUTHOR             = somebody (best given as last name, first
name)
INSTITUTION        = institute
START_TIME         = YYYY Mon dd hh:mm:ss
*STOP_TIME         = YYYY Mon dd hh:mm:ss
*TARGET            = body name
*SEQUENCE          = sequence ID
*ACKNOWLEDGMENT    = YES (default is NO)
*EMAIL_ADDRESS     = somebody@computer.institution.country

\text

  << write (or cut-and-paste) your notes here >>
```

The header items marked with an asterisk are optional; the others are required, including the "\header" and "\text" tokens. Syntax rules for constructing ENB e-mail inputs are provided in Appendix A, and are also found in the on-line HELP associated with each instance of an ENB subsystem, accessible using a WWW browser. (Hint: the on-line HELP takes precedence over Appendix A.)

One or more MIME documents may be attached to an ENB e-mail entry.

4.2.2 NOTEBOOK Entries Using WWW Browser

ENB inputs made using a WWW browser use a fill-in form for the metadata, followed by free-form text for the note itself. The text may be typed or cut-and-pasted from some other source. No formatting, including TAB characters, will be retained–only ASCII printing characters (ASCII 32 through 126) and blanks are retained.

4.2.3 NOTEBOOK Structure

ENB data are held in two or three structures, briefly noted below. Since access to these data are provided using a WWW browser, knowledge of the formats and organization of the underlying data buckets is not needed by ENB users (data providers or data consumers).

The metadata that accompany each ENB entry are stored in an instance of the SPICE DBK, so to provide query capability.

The ASCII text comprising each "note" is stored in a text file.

If an optional MIME attachment accompanies an ENB entry, that is stored as a separate, native-format file.

The DBK holding the ENB metadata also contains pointers to the "note" text file and any MIME attachments.

All of the above are integrated under the ENB subsystem software, providing users easy WWW access to ENB data without need for programming.

4.3 Science Plan Component (ESP)

While logically occurring first in the EK family, the ESP is described last in this SIS because there is not a separate structure/format defined for this component. Rather, each flight project will pick either the ESQ or ENB mechanism described above. Specific implementation details are left up to each project.

APPENDIX A

ENB E-mail Entry Construction Rules

The principal formatting rules are listed below. See the ENB on-line HELP for further details. This is available with any project instance of an ENB subsystem, using a WWW browser.

- The "\header" and "\text" tokens must occur in pairs, with "\header" appearing first. Mixed case may be used for these tokens.
- Use only printing characters (ASCII 32 through 126) and blanks. Note that the TAB character (ASCII character 9) is not included in this range because TABS do not have a unique interpretation.
- Do not exceed 80 characters on any line in the "\header" or "\text" sections.
- Each keyword=value pair and the "\header" and "\text" tokens must appear on separate lines. They may begin in any column as long as they are preceded only by blanks.
- The "\" character may not be the first non-blank character of any line except for occurrences of the \header and text tokens.
- White space (blanks) occurring between a keyword and the "=" (assignment operator), and between the "=" and a value field are allowed and are not significant.
- Those keywords identified as required must have non-blank, bona fide value fields. N/A and UNK are not valid values.
- For those keywords identified as optional the value field may be left blank, or you may enter either N/A or UNK.
- Only those keywords included in this original template will be validated; any added keywords will be simply passed on as is.
- The keyword=value assignments may appear in any order.
- Value fields and text may be entered in mixed case. Value fields will be converted to upper case when the SCIENCE PLAN or NOTEBOOK E-kernel component is made. Value fields with embedded white space or non-alphanumeric characters do not require quotes. Quotation marks (single or double) are never needed

APPENDIX B

Examples of NOTEBOOK Entries

B.1 From a Mission Operations Team

```
\header

   SYSTEM_NAME      = DSN
   AUTHOR           = Flinder, Evan
   INSTITUTION      = DSS-43 Canberra
   START_TIME       = 1998-07-23T08:23:02
   STOP_TIME        = 1998-07-23T08:37:00
   ACKNOWLEDGEMENT = YES
   EMAIL_ADDRESS    = eflinder@dsn-dispatch.jpl.nasa.gov


\text

Failure on X-band Receiver-A caused loss of data. Switched in backup unit from B
string and locked up again. Loss of Galileo playback data appears to be about 14
minutes long.
```

B.2 From a Science Team, used as a Note Mechanism

```
\header
SYSTEM_NAME        = MOC
AUTHOR             = SMITH, JOE
INSTITUTION        = MSSS
START_TIME         = 1999 MAR 6 11:23
STOP_TIME          = 1999 MAR 6 11:24
TARGET             = MARS

\text
Attempted to image the Mars Pathfinder (MPF) landing site. Preliminary analysis
of image data does not reveal anything obvious. Further analysis should be done.
```

B.3 From a Science Team, used as a Science Plan Mechanism

```
\header
SYSTEM_NAME        = TES
AUTHOR             = JOHNSON, MARTHA
INSTITUTION        = ASU
```

```
START_TIME        = 1999 SEP 26 15:52:24
STOP_TIME         = 1999 SEP 27 15:24:10
TARGET            = MARS

\text
4103    1) optical switch 00
        2) surface = -80 to +40 degrees lat., full spectral, full spatial(day)
           night = +60 to -60 degrees lat., spectral mask, full spatial
           poles = rest of lats., spectral mask, spatial mask (all detectors
                averaged together)
                - short packets except for space/reference sets
        3) for realtime orbits: full spatial, full spectral for entire orbit
                - long packets
                - maximum 4 interferograms near subsolar point, detector 2,
                        one minute between interferograms [for mini_tes]
        4) limb: all forward looking, space, +75 degrees, limb set (100 km
                height, 1/2 degree steps, padded by one look at surface angle)
                - all short packets
        5) limbs every 10 degrees, alternate orbits (a and b)
        6) continuous limb look - with 3 nadir looks between set for
                once per week repeat. cont_limb 90S - 30N
```