Appendix E. NAIF Toolkit Directory Structure

This appendix contains the software directory structure of the NAIF Toolkit for a SUN. It is an example of a platform-based model for a single platform. Note that the directory organization shown here does not strictly conform to the recommendations discussed in the *Volume Organization and Naming* chapter of this document.

**Chapter Contents**

Appendix E. NAIF Toolkit Directory Structure ................................................................. E-1  
E.1 NAIF Directory ......................................................................................................... E-2  
E.2 TOOLKIT Directory ................................................................................................ E-3  
E.3 Using the NAIF Toolkit ........................................................................................... E-12  
E.4 NAIF’s File Naming Conventions ............................................................................ E-13
E.1 NAIF Directory

The NAIF directory contains one subdirectory, TOOLKIT. The TOOLKIT tree contains all of the files that make up the NAIF Toolkit.

(directory under which you installed the NAIF Toolkit)

   naif
   
   toolkit
E.2 TOOLKIT Directory

The TOOLKIT directory contains the file `make_toolkit.csh`. This is a C shell script that builds all of the object libraries and executables in the TOOLKIT.

```
(directory under which you installed the NAIF Toolkit)
    |
    naif
    |
    toolkit
    |
    make_toolkit.csh
```

TOOLKIT also contains several subdirectories that will be described in more detail in the following sections.

```
(directory under which you installed the NAIF Toolkit)
    |
    naif
    |
    toolkit
    |
    src   lib   exe   doc   etc   example_data
```

1. **SRC**
The subdirectories of this directory contain all of the source code for the products in the TOOLKIT.

2. **LIB**
This directory contains all of the TOOLKIT object libraries.

3. **EXE**
This directory contains all of the TOOLKIT executables, and where applicable, scripts to run the executables.
4. DOC
This directory contains all of the TOOLKIT documentation. This includes user’s guides for the programs, “Required Reading” files for SPICELIB, documents describing the contents of SPICELIB such as the “Permuted Index and Module Summary,” and documents describing the contents and installation of the Toolkit.

5. ETC
The subdirectories of this directory contain product-specific files that are neither source, documentation, nor data. These include configuration files, set up files, and help files. The subdirectory build contains the C shell script that creates the toolkit object libraries and executables.

6. EXAMPLE_DATA
This directory contains example data for use with the cookbook and sptest programs. These files are to be used only with these programs.

E.2.1 SRC
The SRC directory contains one subdirectory for each product in the NAIF Toolkit. Each of these product directories contains the source code files and procedures to create the executable or object library.

(directory under which you installed the NAIF Toolkit)

```
    | naif
    | toolkit
    |   src
  spicelib  support  spacit  commnt  cookbook  sptest  inspekt
```
E.2.1.1 SPICELIB

SPICELIB is a Fortran source code library that contains approximately 650 functions, subroutines, and entry points.

This directory contains the SPICELIB source files.

(directory under which you installed the NAIF Toolkit)

    |
    |
    |
    naif

    |
    |
    |
    toolkit

    |
    |
    src

    |
    |
    spicelib

    *
    .f

E.2.1.2 SUPPORT

SUPPORT is a Fortran source code library that contains routines that support the Toolkit programs. These routines are not intended to be used by anyone except NAIF. These routines are not officially supported and may undergo radical changes such as calling sequence changes. They may even be deleted. Do not use them!

This directory contains the SUPPORT library source files.

(directory under which you installed the NAIF Toolkit)

    |
    |
    |
    naif

    |
    |
    toolkit

    |
    |
    src

    |
    |
    support

    *
    .f
E.2.1.3 SPACIT

Spacit is a utility program that performs three functions: it converts transfer format SPK, CK and EK files to binary format, it converts binary SPK, CK and EK files to transfer format, and it summarizes the contents of binary SPK, CK and EK files.

This directory contains the source code for the spacit main program and supporting routines.

(directory under which you installed the NAIF Toolkit)
   | naif
   | toolkit
   | src
   | spacit

spacit.main
   *.f

E.2.1.4 COMMNT

Commnt is a utility program that is used to add comments, extract comments, read comments, or delete comments in SPICE SPK, CK and EK files.

This directory contains the commnt main program source file.

(directory under which you installed the NAIF Toolkit)
   | naif
   | toolkit
   | src
   | commnt

commnt.main
E.2.1.5 COOKBOOK

The COOKBOOK programs are sample programs that demonstrate how to use SPICELIB routines to obtain state vectors, convert between different time representations, manipulate the comments in binary SPK and CK files, and solve simple geometry problems.

This directory contains the COOKBOOK program source files.

(directory under which you installed the NAIF Toolkit)
    |
    naif
    |
    toolkit
    |
    src
    |
    cookbook
    |
    fstspk.main
    simple.main
    states.main
    subpt.main
    tictoc.main

E.2.1.6 INSPEKT

Inspekt is a program that allows you to examine the contents of an events component of an E-kernel.

This directory contains the source code for the inspekt main program and supporting routines.
E.2.1.7 SPTEST

*Sptest* is a utility program that tests the SPK file readers by comparing states read on the NAIF VAX with states read on the target machine.

This directory contains the *sptest* program source file.

(directory under which you installed the NAIF Toolkit)

```
  | naif
  | toolkit
  | src
  | inspekt

inspekt.main
*.*f
*.*inc
```

(directory under which you installed the NAIF Toolkit)

```
  | naif
  | toolkit
  | src
  | sptest

sptest.main
```
E.2.2 LIB
The LIB directory contains *spicelib.a*, the object library for SPICELIB. It also contains the object library *support.a*, but this library is for use by the Toolkit programs only. Do not link your applications with it!

(directory under which you installed the NAIF Toolkit)

```
naif
  toolkit
    lib
      spicelib.a
      support.a
```

E.2.3 EXE
The EXE directory contains the NAIF Toolkit executables and, where applicable, scripts to run executables.

(directory under which you installed the NAIF Toolkit)

```
naif
  toolkit
    exe
      commnt
      fstspk
      inspekt
      simple
      spacit
      sptest
      states
      subpt
      tictoc
```
E.2.4 DOC

The DOC directory contains all of the TOOLKIT documentation that is available on-line. This includes the user’s guides for the programs, all “Required Reading” files for SPICELIB, all documents describing the contents and porting of SPICELIB, and documents describing the installation and contents of the Toolkit. Please note that the INSPEKT user’s guide is not available on-line.

(directory under which you installed the NAIF Toolkit)

<table>
<thead>
<tr>
<th>naif</th>
</tr>
</thead>
<tbody>
<tr>
<td>toolkit</td>
</tr>
<tr>
<td>doc</td>
</tr>
</tbody>
</table>

- commnt.ug
- fstspk.ug
- simple.ug
- spacit.ug
- sptest.ug
- states.ug
- subpt.ug
- tictoc.ug
- *.req
- category.txt
- libsum.txt
- permuted_index.txt
- porting.txt
- toolkit_install.txt
- toolkit_description.txt
E.2.5 ETC
The ETC directory contains all files for the Toolkit products that are not source, documentation, or data such as set up files, configuration files or help files. It also contains the C shell script used to build the toolkit object libraries and executables.

(directory under which you installed the NAIF Toolkit)

    naif
    |
    toolkit
    |
    etc

spicelib  support  spacit  commnt  cookbook  sptest  build  build_it.csh

E.2.6 EXAMPLE_DATA
The EXAMPLE_DATA directory contains all of the NAIF Toolkit data. This data are intended only to be used with the TOOLKIT programs, and are included only to help you get started using the Toolkit.

(directory under which you installed the NAIF Toolkit)

    naif
    |
    example_data

    cook_01.tc
    cook_01.tls
    cook_01.tpc
    cook_01.tsc
    cook_01.tsp
    cook_02.tc
    cook_02.tsp
    sptest.gen
    sptest.rqs
    sptest.tsp
E.3 Using the NAIF Toolkit

After the installation has been completed successfully, there are a few things that you need to do to get started using SPICELIB. We recommend that you print out the source code for the cookbook programs (.naif/toolkit/src/cookbook/*.main) and examine it. Try running some of the cookbook programs yourself. The cookbook programs demonstrate how to use SPICELIB routines to obtain state vectors, convert between different time representations, manipulate the comments in binary SPK and CK files, and solve simple geometry problems.

Once you’re ready to get your hands dirty, you should read the required reading files for SPICELIB. The required reading files are located in the directory .naif/toolkit/doc and have the extension “.req”. They are text files that describe families of subroutines and how they interact with the rest of SPICELIB.

The most important required reading files are: TIME, KERNEL, SPK, CK, SCLK, SPC, and NAIF_IDS. You should read at least these.

After you’ve done these things, you’re ready to start programming with SPICELIB!
### E.4 NAIF’s File Naming Conventions

NAIF follows a set of conventions for naming files based on the contents of the files. This allows you to find certain types of files in a directory tree quickly. The following table lists the current naming conventions.

<table>
<thead>
<tr>
<th>Extension</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*.for, *.f</td>
<td>Fortran-77 source code files</td>
</tr>
<tr>
<td>*.main</td>
<td>Source code files for program modules</td>
</tr>
<tr>
<td>*.inc</td>
<td>Fortran-77 include files</td>
</tr>
<tr>
<td>*.c</td>
<td>C source code files</td>
</tr>
<tr>
<td>*.o</td>
<td>Unix object files</td>
</tr>
<tr>
<td>*.obj</td>
<td>VAX/VMS object files</td>
</tr>
<tr>
<td>*.a</td>
<td>Unix object library files</td>
</tr>
<tr>
<td>*.olb</td>
<td>VAX/VMS object library files</td>
</tr>
<tr>
<td>*.tsp</td>
<td>Transfer format SPK (ephemeris) files</td>
</tr>
<tr>
<td>*.bsp</td>
<td>Binary format SPK (ephemeris) files</td>
</tr>
<tr>
<td>*.tc</td>
<td>Transfer format CK (pointing) files</td>
</tr>
<tr>
<td>*.bc</td>
<td>Binary format CK (pointing) files</td>
</tr>
<tr>
<td>*.ti</td>
<td>Text IK (instrument parameters) files</td>
</tr>
<tr>
<td>*.tls</td>
<td>Leapseconds kernel files</td>
</tr>
<tr>
<td>*.tpc</td>
<td>Physical and cartographic constants kernel files</td>
</tr>
<tr>
<td>*.tsc</td>
<td>Spacecraft clock coefficients kernel files</td>
</tr>
<tr>
<td>*.txt</td>
<td>Text format documentation files</td>
</tr>
<tr>
<td>*.ug</td>
<td>Text format User’s Guides</td>
</tr>
<tr>
<td>*.req</td>
<td>Text format SPICELIB Required Reading files</td>
</tr>
<tr>
<td>make_toolkit.csh, build_it.csh</td>
<td>Unix C shell script files for creating the toolkit object libraries and executables</td>
</tr>
<tr>
<td>make_toolkit.sh, build_it.sh (product name)</td>
<td>Unix Bourne shell script files for creating the toolkit object libraries and executables</td>
</tr>
<tr>
<td>make_(product name).com</td>
<td>VAX/VMS command procedures for creating products. For example, make_spicelib.com creates the object library spicelib.olb, while make_spacit.com creates the executable spatit.exe.</td>
</tr>
<tr>
<td>(product name).exe</td>
<td>VAX/VMS executable files. For example, spatit.exe is the executable file for the product spatit.</td>
</tr>
</tbody>
</table>

These conventions are preliminary. As coordination with AMMOS and the Planetary Data System (PDS) occurs, these conventions may be revised.
file names
  NAIF conventions............................................................................................................. E-13

NAIF
  file naming conventions .............................................................................................. E-13
  NAIF Toolkit
directory structure......................................................................................................... E-1
  using................................................................................................................................. E-12

software directory structure
  NAIF .................................................................................................................................. E-1
  SRC directory
    in NAIF distribution....................................................................................................... E-4