Chapter 2.
Finding Information

Topic: *Finding Information

The ACIS documentation consists of online help, which is in HTML format, and PDF files. All documentation is available in online help and may also be printed to your printer in "book chapter" form from PDF files provided on the 3D ACIS Online Help CD-ROM.

Spatial's documentation is divided into two kinds of information, each of which is presented in a format designed to communicate the information effectively:

Theory (Discussion) . . . . . Is presented in paragraph, or text, form. Theory includes general information about components, discussions of modeling concepts, tables, diagrams, examples, "how-to" information, etc. Theory is broken into chapters and sections (possibly with multiple levels) that each have a title heading. An online help page containing theory information is known as a "Discussion Topic."

Reference . . . . . Is presented in template form. Reference templates are primarily used for documentation of *code items*, such as C++ classes, functions, etc., but may also be used for such purposes as component descriptions. Each template has a title that is the name of the item documented, such as the class name, function name, or component name.

A reference template is a way of presenting material in a structured format with a standard set of information, using named data fields. For example, the template used to document C++ classes includes fields such as:

Purpose . . . . . Summarizes the intended purpose of the class.

Derivation . . . . Specifies the derivation of the class.

Filename . . . . . Specifies the name of the header file in which the class is declared.

Description . . . Describes the class, its use, and its structure.

There are two types of manuals:

General information manuals . . . . . Cover the entire product line; these manuals usually contain theory but may also contain some reference templates.

Component manuals . . . . . Contain all the documentation for a single software component, which may include theory and/or reference templates.

A manual may contain only theory or both theory and reference templates (refer to section *Manuals*). Within manuals, reference templates are organized by type, where *type* generally refers to a kind of code item, such as C++ class, function, Scheme extension, etc. All the templates of a given type are grouped together and ordered alphabetically.

In online help, information may be organized and displayed several different ways:

- Information can be displayed in the order it appears in a manual (use the Contents button in the navigation frame to display a list of all manuals).
- Information can be grouped by *subject* (use the Subjects button in the navigation frame to display a list of subjects—within subject groups).
- An alphabetical list of all of a specific type of reference item can be displayed (use the List of button in the navigation frame to display a list of types of reference items).
- A list of discussion or reference topics containing a specified keyword can be displayed (use the Index or Find button in the navigation frame to perform a search).

Information is displayed in online help as individual “chunks” called *topics*. This may be different from the way the information would appear in a printed manual. For example, the reference template for a function may span several pages in a printed manual, but it is one topic in online help; or, a single page in a printed manual may contain several paragraphs, each with its own heading, but these appear as separate discussion topics in online help. Each online help topic is stored in a separate HTML file.

ACIS Online Help CD–ROM

Topic: *Finding Information

The ACIS distribution set includes an 3D ACIS Online Help CD-ROM, which contains the files that form 3D ACIS Online Help (online help). This CD-ROM also contains PDF files for printing the documentation in “book form.”

Online help is a set of HTML files. You view the HTML files using an Internet Web browser—either Microsoft Internet Explorer, or Netscape Communicator. Refer to the *readme* file on the CD for information about specific browser versions supported. You may either install the files on your local system (or server) or access them directly from the CD.

Online help uses JavaScript to provide advanced functionality, such as expandable and collapsible icons, so JavaScript must be enabled. Online help also uses frames to display different types of information.

To find out more about using online help and setting up your browser, open the *3D ACIS Online Help User's Guide* by clicking the Using Online Help button on the home page.

Starting Online Help

Topic: *Finding Information

To start online help, you must load the HTML file ACIS.HTM into your browser. When you load this file, the "home" page is displayed. The online help screen contains three frames and a browser status bar:

Navigation frame . . . . . This frame at the top of the page allows you to choose the way you want to locate information in online help, to browse through topics, or to return to the home page.

Selection frame . . . . . Allows you to select a specific item (topic) to view.

Data frame . . . . . Displays the topic that you selected from the list in the selection frame.

Browser status bar . . . . . Displays messages, filenames, and brief link descriptions.

PDF Print Files

Topic: *Finding Information

Although all the ACIS manuals are available in online help, they may also be printed in book chapter form from files provided on the 3D ACIS Online Help CD-ROM. These are Adobe Portable Document Format (PDF) files that can be viewed and printed with the free Adobe Acrobat Reader. Refer to the *3D ACIS Online Help User's Guide* (and the *readme* file on the CD) for more information about getting Acrobat Reader and printing the PDF files.

Manuals

Topic: *Finding Information

The ACIS documentation set includes the following manuals:

- *3D ACIS Online Help User's Guide*
- *3D ACIS Getting Started Guide* (this manual)
- *3D ACIS Fundamental Concepts Guide*
- *3D ACIS Glossary*
- *3D ACIS Application Development Manual*
- *3D ACIS Release Notes*

- Component manuals

All manuals are available in online help and may also be printed from PDF files that are provided on the 3D ACIS Online Help CD-ROM. Individual manuals may be accessed in online help using the Contents button.

ACIS Online Help User's Guide

Topic: *Finding Information

The *3D ACIS Online Help User's Guide* describes how to use online help. It includes information about your system's requirements for using online help, tips on setting up your browser to work well with online help, what is available in online help, how to navigate through the documentation, and hints on how to find specific kinds of information. This manual also describes the subjects used to group information in online help and the templates used to document reference items.

ACIS Getting Started Guide

Topic: *Finding Information

The *3D ACIS Getting Started Guide* is *this* manual.

This manual introduces ACIS and the documentation set. It tells you how to start online help and find information, provides a general overview of ACIS, describes the ACIS architecture, provides information about the interfaces to ACIS, introduces the supplied demonstration applications and tells you how to use them, and discusses how to use global options to control ACIS operations. It also contains examples to help you get started creating models in Scheme.

ACIS Fundamental Concepts Guide

Topic: *Finding Information

This manual discusses many of the fundamental concepts of ACIS, including:

- basic concepts of boundary representation, dimensionality, continuity, etc.
- how ACIS uses C++
- geometry
- topology
- curves and surfaces
- model objects (including entities)
- tolerance and units
- model modification
- model analysis

ACIS Application Development Manual

Topic: *Finding Information

The *3D ACIS Application Development Manual* provides information to help you develop an application using ACIS. This manual contains information about such things as initializing and terminating the modeler, object libraries, error handling, debugging, extending ACIS, deriving classes, etc.

ACIS Glossary

Topic: *Finding Information

The glossary contains definitions of some terms applicable to the ACIS software and environment, and to geometric modeling in general.

ACIS Release Notes

Topic: *Finding Information

The *3D ACIS Release Notes* manual describes the changes to the **ACIS 3D Geometric Modeler** software and the documentation set. The *3D ACIS Release Notes* are also available on *Spatial*'s Web site for download by customers with valid customer IDs and passwords. If necessary, updates to the release notes may be published on the Web site as "information bulletins."

Note *The topics from the 3D ACIS Release Notes do not show up in the* Subjects *listings in online help.*

Component Manuals

Topic: *Finding Information

A *component manual* contains theory and reference templates for a specific software component (refer to Chapter 4, *Architecture*, for information about components).

ACIS 3D Geometric Modeler is comprised of many individual components, as described in Chapter 4, *Architecture*. The documentation set includes the following component manuals:

- *Precise Hidden Line Removal V5 Component Manual*
- *PowerFill Component Manual*
- *ACIS Deformable Modeling Component Manual*
- *ACIS Deformable Modeling Graphic Interaction Component Manual*
- *ACIS MFC Component Manual*
- *Advanced Blending Component Manual*
- *Advanced Surfacing Component Manual*

- *Base Component Manual*
- *Basic Rendering Component Manual*
- *Blending Component Manual*
- *Boolean Component Manual*
- *CATIA Translator Component Manual*
- *Cellular Topology Component Manual*
- *Clearance Component Manual*
- *Constructors Component Manual*
- *Covering Component Manual*
- *Euler Operations Component Manual*
- *Examples Component Manual*
- *Faceter Component Manual*
- *Generic Attributes Component Manual*
- *Graphic Interaction Component Manual*
- *Healing Component Manual*
- *IGES Translator Component Manual*
- *Interactive Hidden Line Component Manual*
- *Interactive OpenGL Component Manual*
- *Intersectors Component Manual*
- *Kernel Component Manual*
- *Laws Component Manual*
- *Local Operations Component Manual*
- *Local Operation Tools Component Manual*
- *Offsetting Component Manual*
- *OpenGL Rendering Component Manual*
- *Operators Component Manual*
- *Part Management Component Manual*
- *Persistent ID Component Manual*
- *Precise Hidden Line Component Manual*
- *Pro/E Translator Component Manual*
- *Remove Faces Component Manual*
- *Rendering Base Component Manual*
- *Repair Body Intersections Component Manual*
- *Scheme AIDE Main Program Component Manual*
- *Scheme Support Component Manual*
- *Selective Booleans Component Manual*
- *Shelling Component Manual*

- *Space Warping Component Manual*
- *Standalone Deformable Modeling Component Manual*
- *STEP Translator Component Manual*
- *Stitch Component Manual*
- *Sweeping Component Manual*
- *Translation Geometry Component Manual*
- *Translator Utility Component Manual*
- *VDA–FS Translator Component Manual*
- *Visualization Manager Component Manual*

Examples in the Documentation

Topic: *Finding Information, *Examples

Examples appear throughout the documentation set. These may be C++ code examples, or Scheme examples. Some examples are only one or two lines, while others are quite large (possibly reproductions of entire source files, Scheme procedure files, etc.). Examples appear in a monospaced `Courier` font.

Each C++ code example is a *code fragment* that illustrates a particular concept or usage. **Examples are for illustration purposes and are not intended to be complete programs that can be compiled or run**. The #include statements in the code fragments may not contain the files or pathnames for the current release.

Sample input data files that are loaded by some Scheme examples are provided on the 3D ACIS Online Help CD-ROM, in directory DATA\SCHEME.

Additional examples are also available from *Spatial*'s Customer Support department for customers who have current maintenance agreements.

Examples Component

Topic: *Examples

The Examples Component provides Scheme extensions that can be used to illustrate how to use APIs. The C++ source code for these extensions is provided to all customers, so customers can look at the code to see how to properly call the APIs in a C++ application. Refer to the *Examples Component Manual* in online help for more information. The C++ source code is located in the examples directory.

Test Application Examples

Topic: *Examples

Scheme AIDE is the demonstration/test application that is provided with ACIS This application illustrates many ACIS modeling operations, and because the C++ source code for both is supplied to all customers, they can be used by developers as example ACIS applications for learning and prototyping. The source code for the application also illustrates such things as how to start and stop the modeler, initialize components, set options, call APIs, etc.

Scheme AIDE

Scheme AIDE (Scheme ACIS Interface Driver Extension) is a Scheme language based ACIS demonstration application. Scheme AIDE is a command–line program that accepts Scheme commands at a command prompt and displays the results in a separate view window. The commands may be either native Scheme commands or ACIS Scheme command extensions. The source code for the Scheme AIDE can be found in the following locations:

Component	Directory	Description
Scheme AIDE Main Program Component	tkmain	Contains the code for the Scheme AIDE main program
Scheme Support Component	scm	Contains the code for general Scheme support (contains several subdirectories)
Individual components	*_scm	Subdirectory in the component's main directory that contains the code for Scheme extensions specific to that component

Searchable Index

Topic: *Finding Information

The online documentation contains an index, which is used for the Index search in online help. The index contains names of code items (functions, classes, Scheme extensions, etc.), chapter and section headers, and subject information. This index allows a faster search of the documentation using a subset of the words in a "full text" search.

The index references information by:

Concept . . . . . Concepts are indexed where introduced and defined.

Name . . . . . Code items (e.g., classes, methods, functions, commands, Scheme extensions, options, etc.) are indexed by their names. Code items such as these may also be indexed by individual substrings within their names. For example, the Scheme extension abl:fixed–width–rad is indexed by the substrings "abl," "fixed," "width," and "rad," in addition to its full name.

Filename . . . . . Where relevant, filenames are indexed.

Header . . . . . Chapter and section headers (titles) appear in the index (using both the full title and individual words in the title).

Information Bulletins

Topic: *Finding Information

Information bulletins are posted on *Spatial*'s Web site for customers who have valid customer keys and passwords. Examples of the kinds of information that may be published in bulletins include:

- release notes updates
- platform–specific problems
- specific changes to the product code that may affect you
- a response to a frequently–asked question
- corrections to errors in the documentation

Documentation Conventions

Topic: *Finding Information

Spatial Corp. (*Spatial*) has developed conventions for its technical manuals and online documentation. These conventions, which are used throughout all *Spatial* technical documentation, are used to ensure that information is presented consistently. The following sections define and illustrate these conventions.

Directory and Path Names

Topic: Finding Information

The documentation sometimes includes references to file pathnames. The ACIS pathnames begin with the ACIS root installation directory, which is denoted by the generic string *<install_dir>* in the documentation. In this root directory are the component directories, as well as directories for tools, executables, etc.

Some ACIS directories contain platform-specific subdirectories, which are denoted in the documentation by the generic string *<platform>*.

Pathnames are generally shown using the UNIX format unless the context is specific to another environment. The UNIX format includes a *forward* slash delimiter, /, after directory names (whereas PCs use a *backward* slash). The documentation might show a pathname as in this example:

```
<install_dir>/bool/boolean/kernutil/
```

However, on the PC, the path specification actually uses backward slashes:

```
<install_dir>\bool\boolean\kernutil\
```

Fonts and Typefaces

Topic: Finding Information

Spatial uses a variety of text fonts and typefaces in its documentation to highlight and differentiate information. These fonts are illustrated and explained below.

Serif	A proportional font is used in typical paragraphs and unspecified types of material. *Italics* text designates the first use of a term or phrase in a document. Where applicable, a definition follows. **Bold** and *italic* typefaces may also be used for emphasis.
. . . . .	A text string that has several possible values is indicated in italics within brackets, such as <*install_dir*>.
Sans Serif	A proportional font highlights filenames, functions, variables, macros, classes, keywords, and keyboard keys. It may also be used for menu names, data field names, buttons, etc., when describing operations within a specific application (such as Microsoft Developer Studio or Netscape).
Sans Serif Bold	A **boldface** proportional font indicates literals, such as commands, options, and keywords.
`Typewriter`	A monospaced font shows examples and code reproductions. Examples or reproductions are taken directly from a display and are presented as uppercase and lowercase characters that represent text exactly as seen on the display. Types include: C++ code fragments, Scheme examples or other commands. This font is also used in syntax definitions.
`Typewriter Bold`	A **`boldface`** monospaced font in a syntax definition shows the minimum portion of a literal, such as a command or keyword, that must be entered.

Class Names

Topic: *Finding Information

C++ class names are presented in a Sans Serif font. Examples include ENTITY, SPApar_pos, and SPAtransf. Although many class names correspond directly to a type of ACIS geometric modeling object, such as ENTITY, EDGE, and curve, the name is generally presented in a Sans Serif font only when referring specifically to the class or a specific instance of the class, rather than to the type of object in general. For example, “An entity is implemented in the class ENTITY, and a curve is implemented in the class curve.”

When the text refers to more than one object instantiated from a given class, the exact class name is presented in the Sans Serif font with an “s” (or “es”) appended to the end of the name in a Serif font, as in BODYs, VERTEXes, and SPAtransfs. For example, “The function takes two EDGEs as input arguments and returns two curves.”

Some C++ classes have uppercase names and some have lowercase names. ACIS contains some pairs of related classes with similar names, differing only in their case. The uppercase class represents an object that is permanent in the model, while the lowercase class represents an object that is temporary in nature and may be used for a variety of purposes, including constructing a corresponding permanent object. For example, a SPHERE is a permanent object in the model that is constructed using a sphere.

Reference Templates

Topic: Finding Information

Spatial presents reference material, which includes functions, classes, Scheme extensions, etc., in standard formats called *reference templates*. Each type of reference material is presented in a template that contains a set of named information fields applicable to the type of material. For example, all functions are documented in a template containing a *Prototype* field that lists the function’s prototype(s), and a *Filename* field that lists the source file in which the function is declared.

The template name, or heading, is generally the name of the reference item, such as a function name or class name. *Explanatory templates* that define the contents of each field, including the template name, are included in the *3D ACIS Online Help User’s Guide*. These explanatory templates are named Function Template, Class Template, etc.

Keyboard Keys, Mouse Buttons, and Menu Names

Topic: Finding Information

In some parts of the documentation, special instructions may be given that include keyboard keys to press, mouse buttons to click, menu items to select, data fields to fill in, etc. Examples include installation instructions, instructions on starting online help, or a description of a specific development environment. *Spatial* uses the following conventions in these cases:

<Shift> . . . . . Keyboard keys to press are in a Sans Serif font, enclosed within angle brackets. Key names are those names or abbreviations found on standard English computer keyboards and are in mixed case.

<Left Mouse> . . . . . Mouse buttons to click are in a Sans Serif font, enclosed within angle brackets. The mouse buttons are: <Left Mouse> <Middle Mouse>, and <Right Mouse>, as found on a right-handed 3-button mouse.

<Ctrl+Y> . . . . . Key *combinations* (or key and mouse click combinations) are joined with a plus sign. This indicates the keys should be pressed simultaneously. The entire key combination is enclosed within angle brackets. The individual key names are not enclosed within angle brackets, unless necessary in order to distinguish them.

<Ctrl+X> <E> . . . . . Key *sequences* are joined with a space separator. This indicates the individual keys (or key combinations) should be pressed in sequence. Individual key names (or key combinations) are enclosed within angle brackets. For example, <Ctrl+X> <E> means to press the key combination <Ctrl+X> followed by the single key <E>.

Preprocessor . . . . . Names of application menus, “tabs,” data fields, radio buttons, etc. are in a Sans Serif font.

Build | Settings . . . . . Multi-level menu selections are shown in a Sans Serif font, with the levels separated by a vertical bar. For example, the menu description Build | Settings means to select the Build menu, then select the Settings submenu.

Filenames

Topic: *Finding Information

Files in ACIS are named using conventions, many of which are standard in the software industry. The following list shows the filename (or extension) convention for some of the files commonly used in ACIS:

*.a . . . . . UNIX static object library.

*.c, *.cxx . . . . . C/C++ source code file.

*.dbg . . . . . Contains debug information.

<*dir_name*>.msg . . . . . ACIS error message file, where <*dir_name*> is usually the name of the directory in which the file is located. This file is edited by developers to add new error messages.

<*dir_name*>.err . . . . . ACIS error code definition file, where <*dir_name*> is usually the name of the directory in which the file is located. This file is generated automatically by the build tool from the <*dir_name*>.msg file.

*.dll . . . . . Windows DLL object library.

e<*dir_name*>.cxx . . . . . ACIS error message source file, where <*dir_name*> is usually the name of the directory in which the file is located. This file is generated automatically by the build tools from the <*dir_name*>.msg file.

*.h, *.hxx . . . . . C/C++ header file. Each header filename *must* be unique.

*.jrl . . . . . Scheme journal file.

*.lib . . . . . Windows object library.

*.o . . . . . UNIX object code file.

*.obc . . . . . Windows object code file.

*.sab . . . . . ACIS binary part save file.

*.sat . . . . . ACIS text part save file.

*.scm . . . . . ACIS Scheme procedure file.

*.sl . . . . . UNIX shared object library on HP700 systems.

*.so . . . . . UNIX shared object library on most systems.