Topic: Ignore

The *3D ACIS Glossary* contains definitions of some terms applicable to the ACIS components and environment, and to geometric modeling in general. Many terms have multiple meanings; in general, only meanings that are applicable to ACIS are included here. Some of the terms, or their definitions given in this glossary, are unique to ACIS.

ACIS

Glossary: ACIS Overview, Software Architecture

State-of-the-art 3D boundary representation modeler. See **ACIS 3D Geometric Modeler**.

ACIS 3D Geometric Modeler (ACIS)

Glossary: ACIS Overview, Software Architecture

ACIS 3D Geometric Modeler (ACIS) is an object-oriented geometric modeler, composed of libraries of C++ classes and functions, on top of which 3D modeling applications are built. ACIS consists of a variety of components that provide core modeling and other specialized support functionality.

acisinit.scm

Glossary: Scheme AIDE Application, Scheme Interface

This optional file is read by the Scheme ACIS Interface Driver Extension (Scheme AIDE) demonstration application at initialization time, and is a useful place to put Scheme defines, initializations and commonly used procedures.

active light list

Glossary: Lights and Shadows

The active light list is a list of known *active* lights maintained by the rendering library and applied during shaded modes of rendering. The active light list is empty by default. It is up to the application to set the members of the active light list by calling the api_rh_set_light_list function.

active primitive list

Glossary: Rendering Control

An internal list of geometric primitives maintained by the rendering library for rendering operations. The list must be explicitly specified by an application using the function LiSetPrimitiveList. Geometric primitives that have been created but are not part of the active primitive list will not be rendered or considered in any rendering operations, such as shadow map calculations.

affine transformation

Glossary: Mathematics

An affine transformation is any transformation that can be defined by a linear transformation plus translation. Affine transformations preserve collinearity and ratios; e.g., the midpoint of a line segment remains the midpoint. (Compare to a *perspective* transformation, which preserves collinearity but not ratios.) Affine transformations include translation, rotation, scaling, shear, and reflection. A bending transformation, however, is not affine.

AIDE, Scheme

Glossary: Scheme AIDE Application

Scheme ACIS Interface Driver Extension (Scheme AIDE) is a Scheme based ACIS demonstration application. See *Scheme AIDE*.

aliasing

Glossary: Image Output, Ray Tracing

Aliasing is a reduction in image quality caused by representing images with discrete pixels. More accurately, *aliasing* is a signal processing problem that results from the insufficient sampling of frequencies above the Nyquist limit. These high frequencies appear ‘under the alias of’ lower frequencies after sampling. In rendering, the term often refers to a type of aliasing known as *jaggies*. These occur when vectors or object edges are rendered with a jagged stair-case effect, to which near horizontal or near vertical lines are particularly prone. Such effects are called aliased artifacts. See also *anti-aliasing*.

anti–aliasing

Glossary: Image Output, Ray Tracing

Anti-aliasing techniques are applied in rendering to prevent or reduce aliased artifacts. For example, a curve displayed on a raster screen will appear jagged since the screen cannot physically display pixels in precisely the correct locations along that curve (aliasing). Anti-aliasing techniques alter the color and intensity of pixels immediately adjacent to the curve to create a visual blending effect which fools the viewer’s eye into seeing a cleaner curve.

Application Procedural Interface (API)

Glossary: C++ Interface

See *API*.

application supplied function

Glossary: Image Output, Viewing

Several external functions are called by ACIS renderers to produce image output, diagnostics, or handle interrupts. These functions are declared by the renderer, but the definitions must be supplied by the developer’s application.

API

Glossary: C++ Interface

The Application Procedural Interface is a set of functions that access ACIS functionality. An API function combines modeling functionality with application support features such as argument error checking and roll back. See also *DI* and *Scheme Interface*.

apropos

Glossary: Scheme AIDE Application, Scheme Interface

apropos is a Scheme procedure that can be used to print out a list of all Scheme commands that contain some character string in their name.

arc

Glossary: Model Geometry, Construction Geometry

An arc is either a circular edge or a circular curve, and can be either an entity data type or a curve data type.

aspect ratio

Glossary: Viewing

The aspect ratio is the ratio of the height of a rectangular region to its width. It is commonly used to refer to the aspect ratio of the screen.

attribute

Glossary: Attributes

Attributes are an inherent ACIS mechanism by which developers can define nongeometric data and attach that data to entities. ACIS implements name/value pairs using the ATTRIB class.

attribute state

Glossary: Attributes, Rendering Control

The information that determines how a geometric primitive will appear when rendered. In vector rendering modes, the attribute state defines the line style. In shaded rendering modes the appearance is derived from four shader functions that determine the color, transparency, illumination model and surface displacement of the primitive. The ratio of the height of a rectangular region to its width. Commonly used to refer to the aspect ratio of the screen.

autoload

Glossary: Scheme AIDE Application, Scheme Interface

autoload is a native Scheme command that will cause a file containing a procedure to be automatically loaded into Scheme when that procedure is first called. For example, help data files can remain unloaded until the user requests help, at which time the help procedure would autoload the data files.

B–spline

Glossary: Construction Geometry

Bézier spline. A concatenated sequence of Bézier curves.

back face culling

Glossary: Rendering Control, Faceting

The faces of any object represented by a solid polyhedron completely enclose its volume. For example, the faces have their surface normals pointing outwards and if none of the object's interior is exposed by the near clipping plane and the view point is outside the object, all faces whose surface normals point away from the view point will be completely hidden by closer faces. Back face culling eliminates all invisible back facing faces from view, and is particularly efficient in that no account need be taken of any other faces. The technique is often adopted during visibility determination.

background

Glossary: Backgrounds and Foregrounds

The background is comprised of all regions of an image not covered by any of the rendered geometry. The shading of the pixels is governed by the current background shader.

background shader

Glossary: Backgrounds and Foregrounds

A background shader calculates the color of pixels not covered by rendered geometry. This color is usually derived from the location of the pixel in image space. Background shaders include "none", "plain", "clouds", and "graduated".

Bezier patch primitive

Glossary: Viewing

A patch formed using Bezier splines. Bezier patches have the property that they pass through the corner points of the control mesh. The rendering library provides support for Bezier surfaces of any order.

bi–cubic patch primitive

Glossary: Viewing

A patch primitive defined by applying bi-cubic interpolation to a matrix of 16 control points. Whether the surface interpolates or approximates a particular control point depends on the basis matrix defined for each of the two parametric axes of the patch.

big–endian

Glossary: Modeler Control, SAT Save and Restore

See *endian*.

bi–linear patch primitive

Glossary: Viewing

The simplest of the patches supported by the rendering library. A bi-linear patch is defined by a control mesh of four points. The surface is defined by the bi-linear interpolation of the positions of its corners.

binding

Glossary: Scheme AIDE Application, Scheme Interface

Because Scheme is an interpretive language, it binds variables to values during execution. The actual point in time at which this binding occurs for a given variable varies with the Scheme *special form* (let, let*, begin, define, etc.) used. If the program tries to read a variable that has no value bound to it, the Scheme Interpreter will generate an error message "unbound variable".

blend

Glossary: Blending

A transition surface that provides a surface between regions of a model.

blending

Glossary: Blending

Replacing the sharp edges and vertices in models by faces in order to improve the appearance, reduce stress points, or assist manufacture.

body

Glossary: Model Topology

A body is a collection of lumps that have a common transform. It is the highest level entity in an ACIS model, and can own lumps or wires. See also *lump*, *edge*, *shell*, *face*, *wire*.

A *wire body* contains no faces, shells, or lumps. The minimal case of a wire body consists of a wire record, coedge, and edge with null geometry whose start and end are a single vertex with a point.

A *sheet body* is an infinitely thin body, with faces that never totally enclose a volume. This means that it is possible to "visit" both sides of all faces without passing through any face. Every edge of the model bounds at least one face. There are no free or dangling edges.

A *solid body* totally encloses a volume. It is not possible to "visit" both sides of any face without first passing through a face. Every boundary of every face is used twice, except in the cases of the analytic sphere and torus, which are closed and require no further topology to bound a volume. A solid body has no dangling faces or dangling edges. A single-sided face body is considered a solid body.

Boolean

Glossary: Booleans

TRUE or FALSE, #t or #f, on or off, 1 or 0. Boolean data has two possible states. Boolean functions have two possible outcomes. For example, a function that determines whether two lines intersect or not has a Boolean outcome: the lines do intersect, or the lines do no intersect. ACIS Boolean operations include subtract, unite, and intersect.

bound symbol

Glossary: Scheme AIDE Application, Scheme Interface

A bound symbol is a Scheme variable that has a value currently bound to (associated with) it. See also *binding*.

boundary representation modeler

Glossary: Model Topology

Defines the boundary, that is made from a closed set of surfaces, between solid material and empty space. Also called a *B-rep* modeler.

B–rep

Glossary: Model Topology

See *boundary representation modeler.*

branched history

Glossary: History and Roll

Branched history is a method of maintaining roll back data. It creates a new branch when one changes the model after roll back. Roll forward data is saved. In unbranched history, model changes after roll back result in deletion of roll forward data.

B–spline patch primitive

Glossary: Viewing

The most general form of polynomial patch supported by the rendering library. The B-spline patch uses B-spline curves in both parametric directions. A B-spline provides a smooth approximation to its control points. The most popular form of B-spline is NURBS (nonuniform rational B-spline), where the segment knots are spaced at non-equal intervals in parameter space.

bulletin

Glossary: History and Roll

ACIS has a subject for tracking all changes that occur in an ACIS session. This tracking happens at four levels of granularity: the bulletin, the bulletin board, the delta state, and the history stream. The finest level is the bulletin, which tracks the creation, deletion, or modification of a *single entity.* Individual bulletins are not used outside of the context of a bulletin board or delta state. See also *delta state* and *bulletin board*.

bulletin board

Glossary: History and Roll

A bulletin board is an intermediate level of model-change tracking. The bulletin board is used to track all bulletins (i.e., entity changes) that happen *within one API call*. (It can be used for other purposes as well.) Bulletin boards are composed of bulletins, and are commonly used in error recovery. See also *delta state* and *bulletin*.

bump mapping

Glossary: Displacement, Materials

See *displacement shader.*

C0 continuity

Glossary: Mathematics

Continuity of the zeroth derivative.

C1 continuity

Glossary: Mathematics

Continuity of the fist derivative; i.e., the tangent vector is continuous. The tangent of a curve at a parameter value is a vector. Two curves are *C1 continuous* at a point if both the magnitude and direction of the tangents are the same.

C2 continuity

Glossary: Mathematics

Continuity of the second derivative.

c–face

Glossary: Cellular Topology

In cellular topology, a c-face is a reference to one side of a face. It is implemented in the C++ class CFACE.

c–shell

Glossary: Cellular Topology

In cellular topology, a c-shell is is one portion of a cell's boundary. It is a connected set of c-faces that bound the 3D region of the cell. It is implemented in the C++ class CSHELL.

calibration curve

Glossary: Blending

In blending, a the calibration curve is used to calibrate the radius function of a variable–radius blend; i.e., to determine which radius value goes where. It is also used to parameterize the blend surface.

Because the calibration curve is sometimes used as a "defining curve," it is sometimes referred to by the term defining curve. However, in general terms, the calibration curve is not the same thing as a defining curve as defined in this glossary. See *defining curve*.

camera viewing

Glossary: Viewing

A method of view specification employed by rendering. Camera viewing is analogous to the way in which a photographer might position a camera in order to photograph some object, with the viewing position being situated at the camera lens. See also *viewing control*.

cap

Glossary: Blending

See *capping*.

capping

Glossary: Blending

The intersecting of the blend surface with existing body faces, or possibly the extension of existing body faces. See also *end cap* and *side cap*.

Catmull–Rom spline

Glossary: Mathematics

A form of polynomial spline that smoothly interpolates all but its first and last control points.

CCI

Glossary: Intersectors

Curve/curve intersection. CCI algorithms find points or intervals at which two curves meet or approach, to within a specified tolerance.

CCS

Glossary: Intersectors

Curve/curve intersection on a surface. The CCS algorithm improves the reliability and the performance of many curve/curve intersections when it is known that both of the curves lie on the same surface. The CCS algorithm uses the additional surface information to find more reliable results and to improve the efficiency of the intersection. See *CCI*.

cellular topology

Glossary: Cellular Topology

Organizes models with mixed dimensionality into the solid and sheet cells from which they are composed.

chamfer

Glossary: Blending

A beveled edge or corner, especially at a 45 degree angle.

class

Glossary: C++ Interface

A class is an object-oriented C++ construct defining data and functions that can be applied to that data. A *virtual* or *base* class is used by ACIS to specify the form of the class. A *first level derived* class is used by ACIS to specify ownership of the class. *Second level derived classes* can be instantiated.

class hierarchy

Glossary: Software Architecture

The class hierarchy shows the derivations of classes from their parent classes.

Class Interface

Glossary: C++ Interface

The Class Interface is a set of C++ classes that define ACIS models and their characteristics. These classes are used by both the API and DI interfaces and may be used directly by an application to interact with ACIS. See also *API*, *DI*, and *Scheme Interface*.

clipping

Glossary: Viewing

Clipping is the removal of those parts of objects that lie outside specified bounds from an image. See also *clipping plane*, *hither clipping plane*, and *yon clipping plane*.

clipping plane

Glossary: Viewing

A clipping plane is a world space plane lying perpendicular to the viewing direction and used to specify which objects are to be considered visible. Two clipping planes are used: the near plane and far plane also know as the hither clipping plane and yon clipping plane respectively. The planes are specified by a distance from the view point in the viewing direction. Under perspective projection, the near plane will always lie closest to the view point. Parts of objects that do not lie between the clipping planes are not rendered. See also *clipping*, *hither clipping plane*, and *yon clipping plane*.

coedge

Glossary: Model Topology

A coedge records the occurrence of an edge in a loop of a face. The introduction of a coedge permits edges to occur in one, two, or more faces, and so makes possible the modeling of sheets and solids, both manifold and nonmanifold. See also *edge*, *loop*, and *face*.

coherent noise function

Glossary: Rendering Control

A coherent noise function is a utility function to support shader functions, such as the "simple wood" color shader in the basic shader library. A coherent noise function will produce pseudo-random effects without discontinuities.

color model

Glossary: Color Patterns, Materials

A color model is some particular way of representing color values. Usually a color model defines a color as a position in a three dimensional coordinate system, such as the RGB (red, green, blue) model. The RGB model is common because it matches the way colors are produced on display monitors. However other color models, such as those based on hue, saturation and lightness (HSL) more closely match the manner in which people describe colors and are becoming popular in user interfaces.

color resolution

Glossary: Color Patterns, Materials

Color resolution is a measure of the range of discrete intensity values that a display can show or an image format can represent.

color source shader

Glossary: Color Patterns, Materials

Color source shaders are a class of shaders that define the color of a point lying on the surface of a primitive. A color source shader may be as simple as a plain color that is applied to all points on the surface of an object with uniform intensity or may define a complex pattern, such as wood or marble.

command

Glossary: Scheme AIDE Application

A command is an interpreted instruction, such as to an operating system or application. See also *compound*, *function*, *primitive (Scheme)*, and *procedure*.

command window

Glossary: Scheme AIDE Application

The Scheme ACIS Interface Driver Extension (Scheme AIDE) command window is a window into which the user can type Scheme commands (at the prompt), and in which Scheme Interpreter messages and errors are displayed. It is the “base” window of the Scheme AIDE demonstration application.

component (subgraph)

Glossary: Graph Theory

A maximum connected subgraph of G is called a component of G.

component (software)

Glossary: Software Architecture

A software component is a functionally specialized unit of software—a collection of software items (functions, classes, etc.) grouped together to serve some distinct purpose. It serves as a constituent part of a whole software system or product. An ACIS component maps to a top-level installed directory.

component shader

Glossary: Rendering Control

Component shader is a term for one of the four shaders that constitute an RH_MATERIAL. See also *color source shader*, *displacement shader*, *reflectance shader*, and *transparency shader*.

composite primitive

Glossary: Viewing

The name given to a primitive that comprises a collection of other primitives. The rendering library supports two types of composite primitives: the group primitive and instance primitive. See also *procedural primitive*.

compound

Glossary: Scheme AIDE Application, Scheme Interface

A compound Scheme procedure is a Scheme procedure defined at run time using the *lambda* primitive, such as (*define* my_proc (*lambda* (x) (display x))). See also *procedural primitive*.

conductor

Glossary: Transparency, Materials

A material that conducts electricity. Conductors include all metals. Their opposites, nonconductors, are referred to as dielectric materials, such as air or glass. Generally, conductors are opaque and dielectrics transparent. Some materials, such as certain types of plastics that could be considered as dielectrics, yet appear opaque, do so because structurally they are compound materials containing suspensions of conducting materials. This material classification is important when a shader uses a physically based illumination model.

connected

Glossary: Graph Theory

A graph G is called connected if given any two vertices a and b in G, there is a path in G from a to b.

constrained shading mode

Glossary: Rendering Control

See *shader mode*.

constructive solid geometry

Glossary: Model Geometry, Construction Geometry

Constructive solid geometry is a type of solid modeling in which complex forms are created by the composition and editing of simpler forms.

continuity

Glossary: Mathematics

In a general sense, continuity describes how two things come together. In ACIS, these items may be two curves that meet in some way, two portions of the same curve, etc. In ACIS, two types of continuity are generally discussed: C*n* and G*n*, where *n* refers to the *n*th derivative.

C*n* continuity refers to continuity of the *n*th derivatives of the equations underlying the entities. This means that the magnitude and direction of the *n*th derivative must be continuous. G*n* continuity refers to continuity of geometric, or parameterization-independent, properties, which means that only the direction of the *n*th derivative must be continuous.

See also *C0 continuity*, *C1 continuity*, *C2 continuity*, *G0 continuity*, *G1 continuity*, and *G2 continuity*.

covering

Glossary: Covering

Fits a surface over a closed loop of curves (wires); i.e., all the boundaries must be specified.

CSI

Glossary: Intersectors

Curve/surface intersection. CSI algorithms find the intersection between implicit curves and surfaces or the intersection between a parametric curve and a singular surface when the singularity is due to the degeneracy of the surface boundary to a single point.

curve

Glossary: Construction Geometry

A curve is a mathematical representation of a 3D curve, used in place of an edge when high computational efficiency is required. Curves hold a count of the number of edges that refer to the curve, and provide a route to the details of individual curve types (straight lines, ellipses, circles, and spline curves). Curves are implemented as both a C++ class and a Scheme data type.

curve parameter space

Glossary: Object and Parameter Spaces

Curve parameter space is the 1D real number space. A point in curve parameter space is expressed as a single coordinate, denoted by *t*. In ACIS, curve parameter space may also be referred to as *the real line*. See also *parameter space*.

cut edge

Glossary: Graph Theory

An edge e is called a cut edge of the graph G if removing the edge e from G has more components than G. When an edge is removed from a graph, its vertices are left in.

cut vertex

Glossary: Graph Theory

A vertex v is called a cut vertex of the graph G if removing the vertex v and the boundary edges from G has more components than G.

cycle

Glossary: Graph Theory

A cycle is a sequence of at vertices v_0, v_1, v_2,..., v_n such that $v_0 = v_n$ and v_0, v_1, v_2,..., v_{n-1} is a path. A graph is called a cycle if it is connected and non–empty and if every vertex is of degree two.

cycle vertex

Glossary: Graph Theory

A vertex v is called a cycle vertex of the graph G if v belongs to a cycle in the graph G.

DDE

Glossary: Ignore

Dynamic Data Exchange (DDE) is a means by which two PC applications can communicate. An application can send Scheme commands to the Scheme Interpreter and obtain results with DDE.

deep copy

Glossary: C++ Interface

A copy made by replicating the object plus any assets owned by the object, including objects pointed at by data members of the object being copied. The copied item does not share any data with the original. A deep copy allocates new storage for all member data and any pointers, so that all the information about that item is self–contained in its own memory block. See also *shallow copy.*

For a copy of an entity that does not share underlying information with the original, a deep copy should be used. One reason for using a deep copy of an entity is to be move the deep copy into a different history stream, breaking all ties with the previous history stream. A call to api_deep_copy_entity can be used to create the deep copy. There are some entities that can not be deep copied (by design). If such entities are present during a deep copy, a sys_error will be thrown. If *attributes* that can't be deep copied need to be skipped over when doing a deep copy, the logical flag dpcpy_skip should be passed in to the API as TRUE. Any non–attribute entities that cannot be deep copied will throw a sys_error regardless of this flag's setting.

define

Glossary: Scheme AIDE Application, Scheme Interface

The Scheme *special form* "define" associates a variable name with a value. The value can be any type of Scheme object, including numeric, alphanumeric, string, or even a Scheme procedure.

defining curve

Glossary: Blending

In blending, an internal curve called a defining curve is used for parameterizing a blend. When the blend is created, a defining curve is set up, and is subsequently used to parameterize the blend and to drive the evaluation algorithm. In some cases, the calibration curve is used as the defining curve. See *calibration curve*.

degree

Glossary: Graph Theory

The degree of a vertex is the number of adjacent edges to the vertex.

distance

Glossary: Graph Theory

The distance between two vertices in a graph is the length of the shortest path in the graph from one vertex to the other.

degree of continuity

Glossary: Mathematics

The derivative level, *n*, to which an object is continuous refers to its *degree of continuity.* If a given object is continuous at the *n*th derivative, it is said to have *n*th degree of continuity (or degree of continuity *n*). See also *continuity*.

delta state

Glossary: History and Roll

A delta state is a "checkpoint in time" of the geometric model. Each change to the model is recorded in a *bulletin.* The bulletins of each API call that results in changes to the model are collected in a *bulletin board*. These bulletin boards are collected together into a user-specified checkpoint called a *delta state*. Delta states are useful for model management tasks such as roll back and roll forward.

derivation

Glossary: C++ Interface

Derivation is the creation of a new C++ class from an existing one.

developer

Glossary: Ignore

A developer is a programmer/engineer who uses ACIS to create an application runnable by end users. See also *end user.*

DI

Glossary: C++ Interface

The Direct Interface (DI) is a set of functions that provide access to modeler functionality without the additional application support features of APIs. Not all functionality is accessible by DI functions. See also *API*, *Class Interface*, and *Scheme Interface*.

diagnostics

Glossary: Debugging

Information provided by the rendering library to the application when an error condition occurs. The information is passed by the rendering library by calling the application-supplied function LiDiagnostics.

dielectric

Glossary: Transparency, Materials

See *conductor*.

diffuse reflection

Glossary: Reflectance, Materials

Diffuse reflection is a mode of light behavior obeying Lambert's Law. This law states that light striking a surface is scattered with equal intensity in all directions, with an intensity that is proportional to the angle of the incident light. Generally this is not sufficient to model how most materials appear when illuminated by a particular light source. Rather a diffuse term is taken into account in more general illumination models. Surfaces where the diffuse term is dominant have a matte appearance.

dimensionality

Glossary: Dimensionality

Dimensionality refers to the number of dimensions occupied by entities within the model. One dimension (1D) refers to *wires* (such as a line), two dimensions (2D) refers to *sheets* (such as a plane), and three dimensions (3D) refers to *solids* (such as a block or sphere). Mixed dimensionality occurs when objects of different dimensionality are represented in the same model.

Direct Interface (DI)

Glossary: C++ Interface

See *DI*.

displacement shader

Glossary: Displacement, Materials

A displacement shader is one of four component shaders used to define an RH_MATERIAL. A displacement shader simulates the effects of small surface perturbations by modifying the surface normal vector that is applied to subsequent shading calculations. Also known as *bump mapping*, it provides visual effects such as roughness or undulations that are difficult or expensive to model discretely.

display list

Glossary: Viewing

A display list is a list of *figures* that are displayed on the screen. Figures contain the displayable graphics (silhouettes) of geometric entities and are used for quick display refresh.

dithering

Glossary: Image Output

Dithering is a class of techniques for representing images of one color-resolution at a lower resolution. Dithering techniques use groups of pixels to simulate higher color resolution, but at a cost of lower image resolution. An alternative approach to dithering is Floyd-Steinberg error diffusion.

DLL

Glossary: C++ Interface

Dynamically Linked Library (DLL) is a library that is linked with an executable at execution time rather than compile time. This permits many executables to load and link the same, single disk copy of a library, saving disk space at the cost of load-time overhead.

double–sided face

Glossary: Model Topology

A *double-sided* face means that the points on either side are either *all inside* or *all outside*. If they are all outside, the face is considered to represent an idealized, infinitely thin sheet (a 2D region). If they are inside, the face is an internal partition embedded in a solid.

driver

Glossary: Image Output

A driver is a low-level, hardware-, operating system-, or windowing system-dependent function or set of functions. For example, an application or toolkit might have a graphics driver to handle the output of data specifically to X Windows, and other drivers for Sun Windows, PC-DOS, PC Windows and PC Windows NT. By collecting such platform-specific code in drivers, the main body of functionality can be kept platform-independent and easily portable.

edge

Glossary: Model Topology

An ACIS edge consists of the boundary of a surface and the topology associated with a curve. An edge is bounded by one or more vertices. For example, by bounding an infinite plane with four orthogonal edges, you create a rectangular surface. Edges includes pointers to vertices and attributes, and support for display and roll back. Edges are used to represent the wires in a wireframe body and the boundaries of faces in a solid body. Edges are implemented in the class EDGE, which is derived from the class ENTITY.

effective view position

Glossary: Viewing

The view position used by a shader function. Normally this corresponds to the camera view position. However when ray tracing is invoked for secondary phenomena such as reflected views, the effective view position is the origin of the current ray.

Elk Scheme Interpreter

Glossary: Scheme AIDE Application, Scheme Interface

The Elk Scheme Interpreter is the software that implements the native Scheme language. Scheme has several versions available as shareware. The ACIS Scheme Interpreter uses and extends the Elk version.

encapsulation

Glossary: C++ Interface

Encapsulation places data and functions in a private area where they are not globally accessible. This can be achieved using by the private and protected keywords in a class declaration or by using static data and functions in source files.

end cap

Glossary: Blending

A cap that terminates a blend. See also *capping* and *side cap*.

endian

Glossary: Modeler Control, SAT Save and Restore

Binary save and restore data can be in either little-endian or big-endian byte format. In big-endian ordering, the most significant part of the word is first. In little-endian ordering, the least significant part of the word is first. Refer to the reference template for option binary_format for information on controlling byte swapping.

end user

Glossary: Ignore

An end user is a person who runs an application created with ACIS. See also *developer*.

entity

Glossary: Entity

An entity is the most basic class of ACIS object. All entities have a common set of functionality, such as the ability to save and restore themselves to/from a file, copy themselves, and debug themselves. Entities are implemented in the class ENTITY. All other geometric and higher level ACIS objects are derived from ENTITY.

entity filtering

Glossary: Filtering

Entity filtering is the process of selecting or rejecting entities from a list of entities based on criteria such as entity type or color.

entray

Glossary: Picking

An entray is a combination of an entity and a pick ray used for trimming or tangency operations.

environment

Glossary: Scheme AIDE Application, Scheme Interface

The Scheme environment is a top-level area that holds unbound variables and variables bound to values. A list of Scheme objects from the environment is available in oblist. Certain environment variables specific to the ACIS Scheme Interpreter, such as colors, text characteristics, views, and display characteristics can be queried and set using the (env:) group of Scheme procedures.

environment mapping

Glossary: Environment Maps

A rendering technique that creates an illusion of mirror-like reflections on the surface of objects. Environment mapping may simulate both the reflections of a surrounding environment lying at some remote distance from the objects being rendered and inter-object reflections. Environment mapping is also known as reflection mapping and is view-dependent.

event

Glossary: Debugging, Picking

Events are actions caused by stimuli external to the application, such as mouse button presses, releases, movements, and keyboard presses. The developer must supply a routine that handles asynchronous events that occur when ACIS is occupied. Events can be asked for using (read–event), and can be analyzed using the (event:) group of Scheme procedures.

exception

Glossary: Debugging

An exception is an error in processing. Applications typically use *exception handler functions* to trap and cope with such conditions.

exception handler

Glossary: Debugging

An exception handler is a routine that is dedicated to handling one or more errors that may occur during processing.

extensibility

Glossary: Software Architecture, Extending ACIS

Extensibility means that new classes and/or functionality can be added to the product in the future. For example, ACIS is extensible because users can derive new classes of data, define new methods for those classes, create their own functionality and attach it to ACIS.

extension (Scheme)

Glossary: Scheme AIDE Application, Scheme Interface

New functions, procedures or data types that extend the range and capability of the Scheme language. Extensions are tightly integrated so that a user need not know what is an extension and what is part of the native language.

eye position

Glossary: Viewing

A view of the geometric model is created by looking from the *eye position* in space, in a particular direction (usually at the model). A view can be "walked" around the model by changing the eye position and direction.

face

Glossary: Model Topology

A face is a surface bounded by edges. One or more loops of edges bound a face. A face can bound a solid region, be embedded in a solid region, or only represent a 2D region. It is implemented in the class FACE.

A face can be double-sided, in which case it is infinitely thin. It can also be single-sided, in which case the face normal points away from one side of the face, and solid material is present on the other side of the face. A single-sided face is a solid body, not a sheet body.

See also *double-sided face*, *edge*, *lump*, *wire*, *shell*, *single-sided face*.

facet

Glossary: Faceting

A facet, also known as *tessellation*, is a polygon. Facets are used to break down a surface into an approximate representation. See also *faceting*.

faceting

Glossary: Faceting

Faceting is an operation that generates approximate polygonal representations for the faces of entities. Edge consistency is maintained between adjacent faces. Face faceting is performed by subdividing the face in parameter space with a grid whose increments are determined through *refinements*. Refinements specify the accuracy of the faceted representation. Refinements also control how triangulation is performed and whether smoothing is used to improve the aspect ratios of triangles. The faceted representation of a face is also called a *mesh*. Faceted representations are used in rendering, in clearance analysis, and in operations where an approximation is acceptable in order to simplify calculations. See also *refinement*.

faceting artifact

Glossary: Faceting

The flat-faced nature of a faceted approximation to a high order surface sometimes remains evident in features of a shaded image. These features are called faceting artifacts and tend to be visually distracting. Techniques to avoid faceting artifacts include Gouraud shading and Phong shading.

fair surface or curve

Glossary: Faceting

A common term in CAGD literature used to describe surfaces and curves in deformable modeling. A fair surface or curve is a smoothly varying, doubly-curved surface or curve that tends to be aesthetically pleasing. A fair surface tends to have very few inflection points or lines. An open sail on a sailboat or soap bubbles stretched between wires are examples of fair surfaces.

far clipping plane

Glossary: Viewing

See *clipping* and *yon clipping plane*.

field of view

Glossary: Viewing

The field of view is a means of specifying which part of a scene is visible from the view point using the camera model of image control.

figure

Glossary: Viewing

A figure is a selected list of entities that are represented in a simple display list with just the polylines needed to display those entities. Figures are commonly used when performing rubberbanding.

fillet

Glossary: Blending

A fillet is a round blend on a concave edge that adds material to the body being blended.

filter

Glossary: Filtering, Rendering Control

An entity filter is a procedure that allows entities in a list to be selected or rejected based upon color, entity type, or potentially other criteria. Filters can be logically ORed, ANDed, and NOTed together. Also, part of a transparency shader that approximates clear or colored glass.

flat shaded rendering mode

Glossary: Rendering Control

A shaded rendering mode in which all the pixels of a polygon are shaded a constant color.

flat shading

Glossary: Rendering Control

Flat shading is a method of rendering the geometric model as a solid in which each face is assumed to be opaque with a single, uniform base color. Curved surfaces are approximated by a mesh of polygonal facets, and each facet is shaded uniformly with a single average color. Reflectance, displacement, transparency, texture spaces and shadows are not supported in this mode. See also *Gouraud shading*.

Floyd–Steinberg error diffusion

Glossary: Image Output

Floyd-Steinberg error diffusion is a method for displaying images of a given color resolution at a lower color resolution. The method distributes the error introduced during pixel quantization amongst neighboring pixels. Though slower than dithering, this approach can produce more acceptable results.

foreground shader

Glossary: Backgrounds and Foregrounds

An shader used to perform post-processing operations on an image during the shading pipeline. Foreground shaders are used to simulate atmospheric effects such as fog and depth cueing.

form

Glossary: Construction Geometry

The form of a curve/surface can be *open* with two distinct ends, *closed* where the start and end points/edges are the same, forming a loop curve such as a tear-drop, *periodic*, where it is smoothly closed with tangent continuity, or *unknown* when it has not yet been determined.

free form surface

Glossary: Construction Geometry

A free form surface cannot be defined by a simple formula. The human body, for example, is a free-form surface. Cylinders, cones, and cubes can be described by simple formulae and are not free-form surfaces.

Fresnel's Law

Glossary: Reflectance, Materials

Fresnel's Law defines the properties of light reflected from a perfectly smooth surface. It is used in physically-based illumination models to describe how light is reflected from micro-facets.

friend function

Glossary: C++ Interface

A friend function is a C++ function that is not a member (method) of a C++ class, but has direct access to the internal data of the class.

from point

Glossary: Viewing

A point in world space at which a virtual camera is positioned in the camera model of view control. It is equivalent to the *view point* or *eye point* in other literature. For an orthographic image projection, the from point is considered to be infinitely distant from the to point. For perspective projection, which shows a perceived *vanishing point* such as railroad tracks that seem to meet in the distance, the from point is not considered to be infinitely distant from the to point.

frustum

Glossary: Construction Geometry

A solid figure consisting of the bottom part of a cone or pyramid, the top of which has been cut off by a plane parallel to the base.

full ray traced rendering mode

Glossary: Rendering Control

Full ray traced rendering mode is a shaded rendering mode providing the same extensive range of photo-realistic rendering effects as the full render mode, but generally at a higher computational cost. The full ray tracing mode determines visibility *exclusively* by ray tracing to provide general shading of anti-aliased images. Anti-aliasing is performed based on shading properties of the scene being rendered. Full rendering mode only ray traces objects displaying reflection or refraction characteristics.

full rendering mode

Glossary: Rendering Control

A shaded rendering mode that supports general shading, with anti-aliasing and transparency. Anti-aliasing is performed based on geometric properties of the scene being rendered. The full mode is the fastest method for producing images of a photo-realistic quality. Like the preview rendering mode, the full mode makes selective use of ray tracing.

function

Glossary: C++ Interface

A C++ routine that optionally takes arguments and optionally returns a result, such as `int my_func (x, y) {. . .}`.

G0 continuity

Glossary: Mathematics

Geometric continuity of the zeroth derivative.

G1 continuity

Glossary: Mathematics

Geometric continuity of the first degree; i.e., the direction of the tangent vector is continuous. The tangent of a curve at a parameter value is a vector. Two curves are *G1 continuous* at that point if the directions of the vectors are the same, but the magnitude differ. G1 continuity is where tangent lines are smoothly connected.

G2 continuity

Glossary: Mathematics

Geometric continuity of the second derivative.

gamma correction

Glossary: Rendering Control

Gamma correction is a factor applied to pixel intensity values on a display in order to compensate for the nonlinear responses of monitors and of the human visual perception system.

garbage collection

Glossary: Scheme AIDE Application, Scheme Interface

Garbage collection is the process of recycling memory already used by the Scheme Interpreter. As the Scheme Interpreter runs, it creates objects and data structures, in available memory. No Scheme object is ever purged from memory, unless the interpreter can prove that it will never again be needed.

generator

Glossary: C++ Interface

A generator is a function or class that produces a new value each time it is called. Examples include random number generators and functions that read input data, producing datum for each call. See also *iterator*.

geometric continuity

Glossary: Mathematics

Geometric continuity refers to continuity of geometric, or parameterization-independent, properties, which means that only the direction of the *n*th derivative must be continuous. See also *continuity*, *C0 continuity*, *C1 continuity*, *C2 continuity*, *G0 continuity*, *G1 continuity*, and *G2 continuity.*

geometric normal vector

Glossary: Mathematics, Rendering Control

A geometric normal vector denotes the normal to the plane in which a polygon lies. It is also known as the surface normal vector. See also *shading normal vector.*

geometric picking

Glossary: Picking

The selection of geometric objects that appear beneath or near the cursor in the image. The operation is supported by the rendering library function LiPickPrimitives. This function selects all primitives from the active primitive list whose projections cover an application supplied image location. The selection operation is performed in the image coordinate system, defined by the current view specification.

geometric primitive

Glossary: Viewing

The lowest level of geometry recognized by the rendering library, for example, a sphere, polygon mesh, or bi-cubic patch. Geometric primitives or primitives are defined through the rendering library interface and stored internally in implicit form. The facets of primitives are generated internally by the rendering library when required. The rendering library provides a creation function for each primitive type, that returns a handle of type LiHandle. The application may refer to the primitive with this handle in subsequent operations.

geometric refinement

Glossary: Viewing

A geometric refinement or refinement is a data structure within the rendering library that defines how accurately nonpolygonal geometric primitives are represented when rendered. A refinement contains a number of criteria that govern the decomposition of the primitives into a faceted form and how many generators should be drawn in wire-frame mode. See also *refinement*.

geometry

Glossary: Construction Geometry, Model Geometry

Geometry refers to the physical *things* represented by the model, such as point, curve, and surface, and their derivations, straight, ellipse, interpolated curve, sphere, plane, cone, torus, and spline, independent of their topological (spatial) relationships.

GKS (Graphics Kernel System)

Glossary: Ignore, Software Architecture, Viewing

GKS is an ISO standard specification supporting 2D and through a later extension, 3D device independent graphics (GKS-3D).

global coordinate

Glossary: Mathematics, Viewing

See *world coordinate*.

global illumination model

Glossary: Lights and Shadows

In shading a surface, a global illumination model takes account of other scene objects, for example to render shadows, reflected views from mirroring surfaces such as metals, and refracted views through transmitting volumes such as glass. See also *illumination model*.

Gouraud rendering mode

Glossary: Viewing

A rendering mode that employs the Gouraud shading technique to shade polygons.

Gouraud shading

Glossary: Viewing

In the Gouraud shading technique, the color of pixels in the interior of a polygon are calculated by interpolating the values of the pixel intensities at the vertices. This produces an illusion of smoothness at relatively little computational cost. Also known as *intensity interpolated shading*. Proposed by Henri Gouraud.

graph

Glossary: Graph Theory

A graph is a symmetric relation on a set V representing the vertices of S. The ordered pairs of a graph are called edges of the graph. No distinction is made between the pair (a,b) and the pair (b,a). The vertices a and b are said to be adjacent to the edge (a,b). The edge (a,b) is said to be adjacent to the vertices a and b.

graphics driver

Glossary: Image Output

A graphics driver is a function tailored specifically to input and/or output graphics data in a device-specific format for devices such as an X Windows widget, a Windows metafile, an Interleaf file, a PostScript file, and a TARGA file.

graphics mode

Glossary: Image Output, Viewing

See *render mode*.

group primitive

Glossary: Model Geometry

A composite primitive that describes the geometry of an object in terms of one or more previously created primitives. A group primitive has no geometry of its own, serving only as a convenient means of referring to other primitives. Any attribute properties assigned to a group primitive will be propagated to all of its sub-primitives.

GUI

Glossary: Ignore

A Graphical User Interface (GUI) is the layer of an application that generates and controls windows, menus, buttons, sliders and other features with which the user directly interacts. Many GUIs and GUI-builder programs are available commercially.

gvector

Glossary: Scheme Interface

An ACIS-defined Scheme data type representing a (mathematical) vector that has both magnitude and direction. This is named gvector (because it's an ACIS "**g**eometric modeler vector") only to differentiate it from the inherent Scheme data type vector, which represents an array. See also *vector*.

handle

Glossary: Viewing

A reference to a data object. Although a handle may be a pointer to some data structure, the actual internal structure of the object is intended to be opaque to the application developer. Any property of the object should be set or retrieved through function calls. Handles are used extensively within the rendering library to refer to a variety of objects, such as geometric primitives or attribute states.

handler function

Glossary: Scheme AIDE Application, Scheme Interface

Each primitive Scheme command defined by the developer or by ACIS (a Scheme extension) has an associated handler function (in C++) that the Scheme Interpreter calls when that command is evaluated. This handler function implements that command, usually by translating Scheme data into C++, performing ACIS API calls, then translating the resulting data back into Scheme.

hashing

Glossary: Ignore

Hashing is a method for directly referencing records in a table by performing arithmetic transformations on keys into table addresses.

Hermite curve

Glossary: Mathematics

A polynomial curve defined by points and their tangent vectors.

hidden–line rendering mode

Glossary: Viewing

A vector rendering mode, in which those edges or parts of edges of facets that are obscured from view are not rendered.

history

Glossary: History and Roll

History permits rapid change between states of an ACIS model. See also *roll back and forward*.

hither clipping plane

Glossary: Viewing

The near clipping plane. Portions of objects between the viewer and the hither clipping plane are excluded from the view. See also *yon clipping plane* and *clipping*.

homogeneous coordinates

Glossary: Mathematics

Homogeneous coordinates is a form of coordinate representation that allows geometric transformations to be treated in a uniform way. The homogeneous representation of a point includes an extra coordinate in addition to its Cartesian coordinates. A two dimensional point is represented in homogeneous coordinates by three values, *x*, *y*, and *w*, and a three dimensional point by four.

hook

Glossary: Scheme AIDE Application, Scheme Interface, Picking, Debugging, Rubberbanding

A hook is a means to call a Scheme procedure when a specific event occurs. For example, mouse–down–hook is a procedure that gets called when any mouse button is pressed. Hooks by default are set to a null function and do nothing. A user or developer can set! a hook to some Scheme procedure. For example, a user can assign to mouse–down–hook a procedure that picks an entity, highlights it, and initiates rubber band drag mode. The user can also assign to mouse–up–hook a procedure that un-highlights the entity and terminates rubber band drag mode.

husk

Glossary: Software Architecture

The term “husk” has historical meaning in the ACIS product line, though it is being phased out beginning with Release 6.3. A husk is a package (of one or more software components) that performs some distinct set of specialized or advanced functionality. An “optional husk” was an add–on component purchased separately from the core **ACIS 3D Geometric Modeler** product.

illumination model

Glossary: Lights and Shadows

An illumination model is a particular equation modeling the interaction of light with the surface of some object, also known as a lighting model. See also *global illumination model* and *local illumination model*.

image coordinate system

Glossary: Modeler Control

A two-dimensional discrete coordinate system that corresponds to the pixels in an image. The origin is located in the top left corner, with x increasing to the right, and y increasing down the image.

image resolution

Glossary: Image Output

Image resolution is a measure of the granularity of an image defined in terms of the number of pixels per row and column.

image projection

Glossary: Viewing

Image projection projects a 3D image onto a 2D plane. A perspective projection imitates a pin hole camera, with the projection lines all meeting at the view point so that objects closest thereto appear larger. Orthographic projection is a parallel projection with all objects appearing the same size irrespective of distance from the view point.

implicit primitive

Glossary: Viewing

A collective term referring to geometric primitives whose surfaces can be described by an implicit function. These will often be algebraic surfaces, that is, surfaces that can be described with a polynomial equation in terms of their coordinates. The most common surfaces of this class are the quadric surfaces. Though surfaces may have a natural implicit form, in practice, it is often more convenient to adopt a parametric representation. In the rendering library, all quadric primitives and the torus primitive are implicit.

incident light

Glossary: Lights and Shadows

Light striking a surface.

information bulletin

Glossary: Finding Information

A Web–based publication of some specific information that may be helpful or critical to your use of *Spatial*'s products. 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

Information bulletins are posted on *Spatial*'s Web site.

inheritance

Glossary: C++ Interface

Inheritance is the process by which a derived, child C++ class can use the methods and data definitions of its parent class. In ACIS, for example, all child classes of ENTITY inherit methods from ENTITY, such save, restore, debug, and many other methods. See also *instance*.

instance

Glossary: C++ Interface

When a C++ class is defined or derived, that class is a template. Data is stored in an *instance* (object) of that class. The process of creating an instance is called *instantiation*. OOPS utilities allow similar operations to be performed in the Scheme language. See also *shader instance*.

instance primitive

Glossary: Model Geometry

A composite primitive that describes the geometry of an object in terms of one or more previously created primitives. An instance primitive has no geometry of its own, serving only as a reference to other primitives; however, it may contain a transformation that is applied to all of its sub-primitives.

instantiation

Glossary: C++ Interface

Instantiation is the process of creating an instance (that can hold data) from a C++ class (which is a template than cannot hold data). Instances are created using a *constructor* function and are destroyed using a *destructor* function. Destructor names begin with the ~ character.

intcurve

Glossary: Construction Geometry

Spatial's abbreviation for *interpolated curve*. An interpolated curve is represented in ACIS by the C++ classes intcurve and INTCURVE. See also *interpolated curve*.

interconnected data structure

Glossary: C++ Interface

An interconnected data structure consists of many elements that refer to each other using pointers. The user saves a pointer to some anchor point and finds other elements by traversal.

interpolated curve

Glossary: Construction Geometry

An interpolated curve is constructed by interpolating a smooth curve through a finite set of points. See also *intcurve*.

interpreter

Glossary: Scheme AIDE Application, Scheme Interface

See *Elk Scheme Interpreter*.

interrupt handler function

Glossary: Debugging

During rendering operations, the rendering library periodically calls a function named rh_check_interrupt that is be used by the application to inform the rendering library that an interrupt has occurred. This rendering library interrupt handler is distinct from an interrupt handling function that an application may use to handle interrupt signals that have been trapped by the operating system, for instance, using the UNIX system call *signal*.

intersection curve

Glossary: Intersectors

An intersection curve is generated by the intersection of two surfaces.

intersectors

Glossary: Intersectors

Algorithms for calculating the intersections between curves and surfaces are referred to as intersectors. An intersection may be a curve/curve intersection (CCI), curve/surface intersection (CSI), surface/surface intersection (SSI), or curve/curve intersection on a surface (CCS).

isometric

Glossary: Viewing

The isometric viewing mode displays the size of geometry independent of distance from the viewer. See also *perspective* and *orthographic projection*.

iterator

Glossary: C++ Interface

An iterator is a special kind of generator that traverses a data structure and returns a new data element each time called.

jaggies

Glossary: Image Output

See *aliasing*.

knot
Glossary: Spline Interface

A point in parameter space at the join between adjacent polynomial segments in a B-spline.

knot vector
Glossary: Spline Interface

A list of the parametric values corresponding to the knots of a B-spline, from which the continuity between adjacent segments can be derived.

lambda
Glossary: Scheme AIDE Application, Scheme Interface

Lambda is a Scheme *special form* that specifies a Scheme object that is a procedure.

Lambert's Law
Glossary: Reflectance, Materials

See *diffuse light*.

light source
Glossary: Lights and Shadows

A light source is a model of some source of light radiation used by the rendering library. Light sources only approximate types of lights found in the real world. The effect of a light on a surface depends upon the reflectance model associated with the surface.

light source shader
Glossary: Lights and Shadows

Shader function that describes how much illumination from a light source falls on a surface. A light shader can take one of several light types such as point, distant or ambient. Light source shaders include "ambient," "distant, "eye," "point," and "spot."

LightWorks Image Format
Glossary: Viewing

A format used by LightWorks Design for representing images containing either pixel values or vectors. Pixel based images are run-length encoded. Files containing data in this format have a .lwi file name extension.

linear
Glossary: Graph Theory

A graph is called linear if it is a tree and if it does not contain a vertex of degree greater than two.

line style
Glossary: Image Output

An application supplied value that may be associated with an attribute state to be output with a line during vector rendering modes. It is up to the application to set and interpret that value. For example, the line style may be used as an index into a color lookup table or as an encoded dash pattern.

line type

Glossary: Image Output

A classification used in vector rendering modes to denote the categorization of a line. Line types, encoded using bit fields in the LiLineType data type, are provided so that edges that correspond to silhouettes, boundaries, interior, hidden, and visible edges may be distinguished.

little–endian

Glossary: Modeler Control, SAT Save and Restore

See *endian*.

loadpath

Glossary: Scheme AIDE Application, Scheme Interface, Scheme AIDE Application

When Scheme tries to load a file, it examines a variable called loadpath to determine which directories it should search to find that file.

load command

Glossary: Scheme AIDE Application, Scheme Interface

The load command is a Scheme primitive that reads a file from disk into the Scheme Interpreter and evaluates the contents of the file. *Spatial* identifies Scheme procedure files with a .scm file name extension.

load function

Glossary: Scheme AIDE Application, Scheme Interface

A load function is associated with a shader prototype and is called when an instance of the shader is created. The role of a load function is analogous to that of a constructor in object oriented languages, such as C++.

local coordinate system

Glossary: Viewing

A local coordinate system is defined with respect to some local frame of reference for a specific geometric primitive. All geometric primitives within the rendering library exist within their own local coordinate systems, but are located for rendering in the world coordinate system.

local illumination model

Glossary: Lights and Shadows

In shading a surface, a local illumination model takes account of the light sources in the active light list and that surface only. No other objects are taken into consideration. As such, global effects such as shadows are ignored. See also *illumination model*.

local operation

Glossary: Local Operations, Model Geometry

A local operation (“local op”) is a means of manipulating geometry in a prescribed way, without modifying topology significantly. The types of local operations include move, taper, tweak, and offset.

lofting

Glossary: Skinning and Lofting

Fits a surface through a series of curves (wires) and provides control over the tangents of the surface at the first and last curve. A lofted surface is a skinned surface with tangent constraints.

loop

Glossary: Model Topology

A loop represents a connected portion of the boundary of a *face*. Despite its name, a loop need not be closed. A loop can comprise a group of wire edges connected in a branched arrangement or in a simple open chain. See also *face*, *body*, *lump*, *shell*, *edge*, and *wire*.

lump

Glossary: Model Topology

A lump represents a connected 3D (solid) and/or 2D (sheet) region. A body can contain zero or more lumps. Each lump represents a disjoint set of points. One lump can be completely enclosed inside the void of another solid lump. Each lump must have at least one shell.

manifold

Glossary: Model Geometry, Model Topology

Manifold geometry is physically realizable. A cube, for example, is manifold, since you can physically create it. Two cubes sharing exactly one edge are nonmanifold, since they are not physically realizable.

material

Glossary: Materials

A material is a group of four shaders used for rendering a solid image of a geometric model. Color, displacement, reflectance and transparency shaders are assigned to a material, which can then be assigned to any number of entities.

member function

Glossary: C++ Interface

See *method*.

method

Glossary: C++ Interface

A method (also known as a member function) is a function associated with a C++ object, and that has direct access to that object's data. It is defined as part of a C++ class in the same way a data member is defined.

micro–facet

Glossary: Faceting, Lights and Shadows

A micro-facet is a microscopic planar polygon that models the surface structure of a material in a physically-based illumination model.

mitered blend

Glossary: Blending

A blend in which the edges do not meet tangentially, or there are nonsmooth nonblended edges at the vertex.

mixed dimensionality

Glossary: Dimensionality

Mixed dimensionality occurs when objects of different dimensionality are represented in the same model. See also *dimensionality*.

mixed dimensionality model

Glossary: Dimensionality

A geometric model that is composed of some combination of wire (1D), sheet (2D) and solid (3D) bodies. See also *body.*

model world

Glossary: Object and Parameter Spaces

The model world is a subset of object space in which ACIS objects are modeled. The origin, the point (*x*,*y*,*z*) = (0,0,0), is at the center of the model world. See also *object space*.

near clipping plane

Glossary: Viewing

See *clipping* and *hither clipping plane*.

nesting level

Glossary: Scheme AIDE Application, Scheme Interface

As a Scheme procedure is built, one or more Scheme commands can be nested using parenthesis. The command is evaluated only when the interpreter finds an equal number of open and close parenthesis. The Scheme ACIS Interface Driver Extension (Scheme AIDE) demonstration program's prompt shows the required number of close parenthesis (nesting level of Scheme commands) for commands that are not yet complete: **acis(3)>** means that three more close parenthesis are needed.

nonmanifold

Glossary: Model Geometry, Model Topology

Nonmanifold geometry is not physically realizable. For example, Klein bottles, two cubes that precisely share an edge, and a sphere whose surface precisely intersects the vertex of a cone are not physically realizable, and hence are nonmanifold.

normal

Glossary: Mathematics

A normal is a vector perpendicular to a surface. When dealing with a solid, the normal always points out of the material.

NTSC

Glossary: Viewing

National Television System Committee (NTSC) denotes an American television transmission standard also used for some video recorders. The NTSC luminance conversion criteria are used by the image utility library when converting pixel values from color to grey-scale.

NURBS

Glossary: Spline Interface

Non-Uniform Rational B-splines are a way of representing curves and surfaces.

Nyquist limit

Glossary: Viewing

The Nyquist limit is the highest frequency in a signal that may be reliably sampled at a given rate without aliasing. Signal processing theory shows this to be half the sampling rate.

object

Glossary: C++ Interface

VisMan introduces to ACIS the concept of coclasses and interfaces. This concept is similar to what is available in Java and Microsoft's Component Object Model (COM). Coclasses are C++ classes that are derived from interfaces. A coclass instance is considered an object.

The term "object" is used both by Scheme AIDE (see *Scheme object*) and by VisMan.

object space

Glossary: Object and Parameter Spaces

Object space is the 3D real number space. A point in object space is expressed as a coordinate triple, denoted by (*x*,*y*,*z*). In ACIS, object space may also be referred to as *3-space* or *xyz-space*. See also *parameter space*.

octtree

Glossary: Ray Tracing

An octtree is a three-dimensional data structure that partitions space into axis-aligned boxes called voxels. Octtrees are adopted in various divide and conquer rendering strategies. Starting with a root voxel enclosing the region to be partitioned, voxels are partitioned by bisection in each dimension, generating a tree with eight branches at each node. Solutions to various problems are generally simpler within each branch, since that branch addresses contains only a sub-region of the origin voxel. The partitioning continues until some termination criteria are met, typically when the solution to a problem becomes trivial.

OOPS

Glossary: Ignore, Scheme AIDE Application, Scheme Interface

Object-Oriented Programming System (OOPS) packages can be added to Scheme to implement a C++-like object-oriented structure of classes, instances and methods.

operator overloading

Glossary: C++ Interface

When one operator is assigned more than one function, it is overloaded. For example, the "+" operator can be overloaded to add integers and to add matrices.

orthographic projection

Glossary: Viewing

See *image projection*.

osculating torus

Glossary: Construction Geometry

An osculating torus is one in which the major and minor radii are the same. To construct one, think of drawing a circle, then drawing a line that touches the circle tangentially, and then rotating the circle all the way round this axis. It is referred to as osculating, or "touching closely," because the point of contact between the circle and line remains stationary as you rotate, and is the spot at the middle of the torus where the torus "touches itself in all directions." The osculating torus is the limiting case between the ordinary torus (donut shaped) where the major radius is bigger than the minor one, and the apple torus where the minor radius is a bit bigger than the major radius.

output mode

Glossary: Image Output, Rendering Control

The format of the scan-line data output by the function rh_image_scanline is dependent on the output mode. The output mode is set using function api_rh_set_output_mode. By default a scan-line will be output as an array of RGB triples for each pixel, with each color value being an unsigned 8-bit integer.

parameter position

Glossary: Mathematics, Object and Parameter Spaces

See *par-pos*.

parameter space

Glossary: Object and Parameter Spaces

Surface parameter space is the 2D real number space. A point in parameter space is expressed as a coordinate pair, denoted by (*u*,*v*). In ACIS, surface parameter space may also be referred to as *uv-parameter space*, *uv-space*, or simply as *parameter space*. A position in parameter space may be mapped to a position in object space. See also *curve parameter space* and *object space*.

parameter space curve

Glossary: Object and Parameter Spaces

A parameter space curve specifies the parameterization on a surface of a coedge or edge that represents a curve. This is also referred to as a curve in parameter space, or a surface parameter curve. See also *pcurve*.

SPApar_pos

Glossary: Mathematics, Object and Parameter Spaces

SPApar_pos is an abbreviation for *parameter position*, which is a position in the parameter space of a surface consisting of the two coordinates *u* and *v*. It represents a 2D parameter value that defines a (*u*, *v*) parameter space coordinate that, when evaluated on a surface, produces a 3D object space coordinate. A SPApar_pos is implemented in ACIS by the C++ class SPApar_pos.

part

Glossary: Part Management

A group of entities that can be manipulated as a single item.

part file

Glossary: Part Management

ACIS stores groups of entities as *parts* in part save files. These save files have a .sat or .sab file extension, depending on whether the data is stored as text or binary. See *SAT file*.

patch primitive

Glossary: Viewing

A group of geometric primitives based on polynomial patches. Patch primitives are the most versatile of the primitives supported by the rendering library and are able to encompass all common formats used in CAD/CAM systems, such as bi-linear, bi-cubic, Bezier, and B-spline patches. All patches may be created in rational or nonrational form. If a patch is rational, then control point coordinates must be provided in homogeneous form (*wx*, *wy*, *wz*, *w*) where *w* is the weight. The weight determines the relative influence of a control point on the spline.

path

Glossary: Graph Theory

A path is a distinct sequence of vertices v_0, v_1, v_2,..., v_n such that for all i<n, v_i is adjacent to v_i+1. The vertex v_0 is called the start of the path. The vertex v_n is called the end of the path. The integer n is called the length of the path. Sometimes the graph defined by the sequence of vertices along with the edges that connect them is called a path also.

patterns (pattern objects)

Glossary: Patterns

A pattern is a repetition of a feature or object arranged in a regular manner. Examples of patterns include the radial arrangement of holes in a shower head, the linear grating of ventilation holes on a computer monitor, or the treads on a tire. Implementing such features or objects can become unnecessarily burdensome, especially when the number of repetitive elements grows large. The ACIS patterns functionality is intended to reduce this burden by giving programmers tools that facilitate pattern creation and modification.

pcurve

Glossary: Object and Parameter Spaces

Spatial's abbreviation for *parameter space curve*. A parameter space curve is represented in ACIS by the C++ classes pcurve and PCURVE. See also *parameter space curve*.

perspective

Glossary: Viewing

Perspective is a viewing mode in which far objects are displayed smaller than near object. The size reduction is proportional to the distance from the viewer. See also *isometric*.

perspective projection

Glossary: Viewing

See *image projection*.

PHIGS

Glossary: Viewing

The Programmers Hierarchical Interactive Graphics System is an ANSI (American National Standards Institute) and ISO (International Standards Organization) standard for 3D graphics. PHIGS allows applications to create complex hierarchies of graphics data and manages the organization and display of that data in a central database. The PHIGS+ extension provides support for pseudo-realistic rendering.

Phong rendering mode

Glossary: Viewing

A rendering mode that employs the Phong shading technique to shade polygons.

Phong shading

Glossary: Rendering Control

Phong shading is a method of rendering the geometric model as a solid in which each face is assumed to be opaque with a single, uniform base color. Curved surfaces are approximated by a mesh of polygonal facets, and facet colors are interpolated as the normals of a scanline across each facet (vector-normal interpolation).

physically–based illumination model

Glossary: Lights and Shadows

A physically-based illumination model is a model of light interaction with the surface of a material based on a physical model of the surface. Most basic lighting models, such as the Phong Model, are based on macroscopic properties of materials interacting with light and provide a limited range of ‘realistic’ rendering effects. Physically-based models adopt a statistical, microscopic model of the surface, that treats the surface as a collection of micro-facets. By considering the interaction of light with these micro-facets, it is possible to yield effects that more closely simulate the appearance of ‘real’ materials.

pick

Glossary: Picking

A graphical pick uses the mouse cursor and buttons to identify (pick) an entity that is displayed in a viewport, and return that entity’s id to the program for further processing. It is the primary way for a user to identify some displayed object of interest to the software.

picking

Glossary: Picking

See *geometric picking*.

pickray

Glossary: Picking

A pickray is a composite Scheme object consisting of a position and a gvector. In the process of picking an entity, a ray is fired from the specified position along the specified vector. When the ray intersects an entity, that entity is picked.

pipe surface

Glossary: Model Geometry

A surface that is the envelope of a fixed–radius circle centered on a point on a given curve; a surface like the outer surface of a pipe.

pixel

Glossary: Image Output

A pixel is a rectangular ‘picture element’. Images or displays are composed of arrays of pixels. Each pixel has a value that depends on the image format or display capability. Each pixel in a monochrome image has only an ‘on’ or ‘off’ state. Pixel values for color images are represented by RGB (red, green, blue) triples.

pixel aspect ratio

Glossary: Image Output

The pixel aspect ratio is the ratio of pixel height to width. The pixel aspect ratio can vary between displays. Images produced assuming square pixels may appear ‘squashed’ if rendered on displays whose pixel aspect ratio is not 1.

point

Glossary: Model Geometry

A point holds a count of the number of vertices that refer to the point, and records the coordinates of the point.

polygon mesh

Glossary: Viewing, Faceting

A polygon mesh is a means of describing a collection of polygons that exploits the connectivity between adjacent polygons in a more efficient representation than a list of disjoint polygons. More precisely, a polygon mesh can be defined as a collection of polygons connected such that an edge is shared by at most two polygons and that an edge joins two vertices. Each vertex must be shared by at least two edges and all polygons must consist of a closed loop of connected edges. In the rendering library, *n* meshes are described in terms of a collection of polygons whose edges are defined as an ordered vertex list.

polymorphism

Glossary: C++ Interface

Polymorphism is the ability to call a variety of functions using exactly the same interface, using, for example, virtual functions.

position

Glossary: Mathematics

A position is a set of (*x*,*y*,*z*) coordinates that identify a particular point in model space or in the Work Coordinate System. See also *work coordinate system*.

PostScript

Glossary: Image Output

PostScript is a proprietary device-independent page description language developed by Adobe Systems Incorporated. Postscript drivers are found on many raster devices, especially laser printers.

preview raytrace mode

Glossary: Ray Tracing, Rendering Control

Preview raytrace mode is a rendering mode based upon ray tracing that supports a general shading model and provides a similar range of rendering effects as the scan line preview rendering mode, such as mirroring and transmitting surfaces. The preview raytrace mode does not support anti-aliasing.

preview rendering mode

Glossary: Ray Tracing, Rendering Control

Preview rendering mode is a rendering mode that supports general shading without anti-aliasing or transparency shading effects. The preview mode primarily uses Phong shading with selective use of ray tracing.

primitive (Scheme)

Glossary: Scheme AIDE Application, Scheme Interface

A Scheme primitive is a Scheme procedure that is defined to the Scheme Interpreter at load time, either by being part of the native Scheme language, such as (*define* green 2), or by being a Scheme extension written in C++, such as (*solid:block* (position 0 0 0) (position 10 10 10)).

primitive (geometric)

Glossary: Viewing

See *geometric primitive*.

procedure

Glossary: Scheme AIDE Application, Scheme Interface

A Scheme procedure is a set of one or more Scheme primitives and/or compounds enclosed within parenthesis, to be evaluated by the Scheme Interpreter at run time, such as (map event:x (env:views)).

procedural primitive

Glossary: Viewing

A primitive whose geometry is described by some application supplied procedure. The procedure must be able to generate a description of an object's geometry in terms of a list of rendering library primitives, possibly including other procedural primitive. For instance, an application may define a procedural primitive that can generate a description of a nut or bolt.

projection

Glossary: Viewing

See *image projection*.

quadric primitive

Glossary: Mathematics, Construction Geometry

A class of geometric primitives described by polynomial surfaces of the second order, known as quadrics. Quadrics represent several common surface types, including spheres, cones, and cylinders.

quantization

Glossary: Image Output

Quantization is the process of mapping a continuous value onto one of a number of discrete values. For example, in image processing this might apply to the mapping of some continuous pixel intensity onto an integer in the range 0-255. The term can also apply to the display of a vector on a raster screen as a number of set pixels.

R4RS

Glossary: Scheme AIDE Application, Scheme Interface

R4RS is an abbreviation for Revised Report (version four) of the Algorithmic Language Scheme. This paper, edited by William Clinger and Jonathan Rees, is considered to be the base standard defining Scheme, and describes language constructs and concepts.

ray

Glossary: Picking, Ray Tracing

A ray is a combination of a position and a vector defining a line, commonly used for picking.

ray casting

Glossary: Ray Tracing

Ray casting is a single level of visibility determination from the view point by ray tracing.

ray traced shading

Glossary: Ray Tracing, Rendering Control

Ray tracing is a method of shading the geometric model as a solid in which a ray of light is followed from a visible object back to its source. Ray tracing supports displacement, reflectance, transparency, texture spaces and shadows, and generates a photo-realistic image.

ray tracing

Glossary: Ray Tracing, Rendering Control

A rendering technique in which the colors are computed by tracing imaginary rays of light through an environment. Ray tracing can be used to produce photo-realistic rendering effects, such as mirror reflections and refractions.

refinement

Glossary: Faceting

Refinements specify criteria that control the accuracy and types of polygons (facets) generated in faceting. Refinements also control how triangulation is performed and whether smoothing is used to improve the aspect ratios of triangles. See also *faceting.*

A refinement attached to a face applies only to that face. A refinement attached to a shell, a lump, or a body may apply to the contained faces. Based on the surface type of a face, the refinement is first searched from the face, then the shell, then the lump, then the body, then finally from the faceting defaults. Available refinements include:

- Surface tolerance

- Normal tolerance
- Grid aspect ratio
- Maximum edge length
- Maximum grid lines
- Grid mode
- Triangulation mode
- Adjust mode
- Surface type

After faceting has determined the applicable refinement for a face, shell, lump, or body during the search, the application is allowed to change the refinement for a face through a method (member function) of the mesh manager.

See also *faceting.*

reflectance model

Glossary: Reflectance, Materials

See *illumination model*.

reflectance shader

Glossary: Reflectance, Materials

A reflectance shader is one of the four component shaders that comprise an RH_MATERIAL. A reflectance shader models the interaction of light with a surface and in particular how much light is reflected towards the viewer. A reflectance shader can be thought of as defining the surface *finish*. Reflectance shaders can be used to model reflectance properties that simulate the effects of a surface being composed of specific types of material such as plastic or metal.

reflection, the law of

Glossary: Reflectance, Materials

The law of reflection is a physical law describing the direction of reflected light.

refraction

Glossary: Transparency, Materials

Refraction is the change in direction of light as it crosses the boundary between two transmitting media.

refraction, the law of

Glossary: Reflectance, Materials

The law of refraction is a physical law describing the direction of refracted light.

relation

Glossary: Graph Theory

A relation is a set of ordered pairs. If all of the elements of the ordered pairs come from a set S, then the relation may be said to be on the set S.

release notes

Glossary: Ignore

Release notes describe the current ACIS release, including limitations, bugs, and changes in the software or documentation. This may include changes or new documentation that was unavailable at the time the documentation set was prepared.

rendering

Glossary: Viewing

Rendering is the complete process of producing an image from a model. This comprises two major stages: visibility determination and shading.

repository

Glossary: Ignore

A repository is a computer facility that collects and disseminates software to users of Internet, usually using File Transfer Protocol (FTP). Numerous Scheme repositories around the world can be accessed using FTP, and are a valuable source of shareware for Scheme.

resabs

Glossary: Precision and Tolerance

Named for *resolution absolute*, it is the smallest representable quantity in ACIS. The default value is 10^{-6}. See also *resfit*, *resnor*, and *resmch*.

resfit

Glossary: Precision and Tolerance

Named for *resolution fit*, this is used as a guide to the fitting algorithms for the fit tolerance of an approximate curve or surface. The default value is 10^{-3}. See also *resabs*, *resnor*, and *resmch*.

resmch

Glossary: Precision and Tolerance

resmch is the largest number considered to be zero by the computer. The default value is 10^{-11}. This value can vary from platform to platform and should never be modified—it is for internal use only. See also *resabs*, *resfit*, and *resnor*.

resnor

Glossary: Precision and Tolerance

Named for *resolution normalized*, this is the ratio of the smallest meaningful quantity representable in ACIS (resabs) to the largest. The default value is 10^{-10}. See also *resabs*, *resfit*, and *resmch*.

right–hand rule

Glossary: Work Coordinate Systems

Point your right hand thumb up along the *z*-axis away form the WCS origin, and curl your fingers around the axis. Your fingers point in the direction of positive *x*-to-*y* rotation.

ring

Glossary: Model Topology

A ring is the collection of coedges that surround an edge. Rings allow the software to identify all the faces attached to an edge. This can be used in checking whether the edge is manifold (one or two faces) or nonmanifold (more than two faces). See also *edge*, *coedge*, and *loop*.

roll back and forward

Glossary: History and Roll

Roll is a model management subject that allows regression of the geometric and topological creation sequence. Roll back returns the geometric model to a previously defined state, and roll forward returns the model to a later state that has already been defined. Roll back is branchable. See also *delta state*.

round blend

Glossary: Blending

A round blend is a blend on a convex edge that removes material from the blank body.

rubberbanding

Glossary: Rubberbanding

Rubberbanding is a way to interactively see mouse movements on the display. On a high level, rubberbanding does not create or change anything. It merely displays graphically displays where the mouse is. However, rubberbanding can be used to control other operations that do create or change geometry, or view positions. Such rubberbanding tools that do perform such operations are called *rubberbanders*. Rubberbanders fall into two broad classes. View rubberbanders change the camera position or viewing parameters. Geometry rubberbanders display the geometry to be constructed.

run length encoding

Glossary: Viewing

Run length encoding is a method of encoding pixel based images in which spans of pixels with the same value (a run of pixels) are represented by a count and value pair, instead of repeated instances of the same value. Run length encoding is efficient for storing images that have a large degree of repetition. The LightWorks Image Format is run length encoded.

SAB

Glossary: SAT Save and Restore

Standard ACIS Binary save file (extension .sab). See *SAT file*.

SAB File

Glossary: SAT Save and Restore

See *SAT file*.

SAT

Glossary: SAT Save and Restore

Standard ACIS Text save file (extension .sat). See *SAT file*.

SAT File

Glossary: SAT Save and Restore

ACIS saves or stores, model information to ACIS save files (also known as *part save files* or *part files*). ACIS also restores model information from these files. These files have an open format so that applications not based on ACIS can have access to the ACIS model. There are two types of ACIS save files: text (file extension .sat, which stands for Standard ACIS Text) and binary (file extension .sab, which stands for Standard ACIS Binary). The only difference between these files is that the data is stored as ASCII text in a .sat file and in binary form in a .sab file. The organization of a .sat file and a ..sab file is identical. The term *SAT file* is generally used to refer to both.

scan–line

Glossary: Image Output

A horizontal row of pixels. The pixels of images are usually processed in scan-line order, from left to right and top to bottom.

scan–line rendering

Glossary: Image Output

Scan-line rendering employs a scan-line algorithm. Scan-line algorithms operate by scanning each row of pixels in sequence to determine the visible surface and pixel value. Scan-line rendering can produce images more efficiently than ray tracing methods.

Scheme

Glossary: Scheme AIDE Application, Scheme Interface

Scheme is an interpretive (as opposed to compilable) programming language based on LISP. The ACIS Scheme Interpreter uses the Elk Scheme language dialect. Scheme is particularly useful for rapid prototyping, algorithm development, and command and control.

Scheme AIDE

Glossary: Scheme AIDE Application

The Scheme ACIS Interface Driver Extension (Scheme AIDE) is a Scheme based ACIS demonstration application. This application was formerly known by the name TKMAIN.

Scheme Interface

Glossary: Scheme AIDE Application, Scheme Interface

The ACIS Scheme Interface allows a Scheme based application to use the API, DI, and Class interface and data of ACIS using Scheme commands and extensions.

Scheme Interpreter (SI)

Glossary: Scheme AIDE Application, Scheme Interface, Software Architecture

The ACIS Scheme Interpreter provides a Scheme language interpreter and Scheme extensions (commands) that support ACIS functionality.

Scheme object

Glossary: Scheme AIDE Application, Scheme Interface

A Scheme object can be any of the inherent Scheme data types, (number, string, character, list, pair, boolean, vector, procedure, symbol, continuation, environment, promise), or any of the ACIS Scheme data type extensions (background, entity, entity–with–ray, event, filter, direction, light, material, position, ray, texture–space, transform, view, etc.). Most Scheme procedures will take Scheme object(s) as parameters and return some type of Scheme object.

screen coordinate system

Glossary: Viewing

A homogeneous coordinate system into which objects are projected during rendering. Screen space is a left-handed coordinate system with its origin at the center of the image, *x* increasing to the right, *y* increasing up the image, and *z* increasing into the image.

SDK

Glossary: Software Architecture

Software Developer's Kit.

sense

Glossary: Model Geometry, Model Topology

Sense is a relative direction associated with some entity, with respect to some associated entity. If the directions are the same, the sense is said to be *forward*. If the directions are opposite, the sense is said to be *reversed*. Sense applies to face normals and their underlying surface normals, coedges and their underlying edges, edges and their underlying curves, etc.

For example, the normal to a face can be either the same direction as the normal of the underlying surface at any position, or it can be the reverse of the surface normal. If it is the same as the surface normal, the face's sense relative to the surface normal is forward; otherwise, its sense is reversed.

The sense of an entity is often specified by the ACIS typedef REVBIT (as FORWARD or REVERSED).

setback

Glossary: Blending

Setback is the vertex distance from which to stop blending edges. It defines a plane that is normal to the edge through a point on the edge and set back from the edge end by the given distance.

shader

Glossary: Rendering Control

A shader or shader function can loosely be defined as a generic C-code function that calculates some aspect of the rendering environment such as a pixel's background color or the effect of a surface's transparency on the color of a pixel. Each shader uses a named list of arguments. For example, the material reflectance shader named metal has three arguments, *ambient factor*, *specular factor,* and *roughness*. Shaders are categorized into several different shader classes, with each shader class being applied in a particular circumstance, such as the background shader class.

shader argument

Glossary: Rendering Control

One of the named arguments of a shader function.

shader creation function

Glossary: Rendering Control

A shader creation function is a function created by the shader compiler that must be called by the application in order to initialize a particular shader prototype and thus allow creation of any shader instances of that class.

shader global

Glossary: Rendering Control

S shader global is one of a group of global variables visible both to the rendering library and to shader functions. A shader function communicates its results to the rendering library and obtains information about the current rendering environment through such variables.

shader instance

Glossary: Rendering Control

See *shader prototype*.

shader mode

Glossary: Rendering Control

A shader mode governs the behavior of certain shader functions. By including a shader mode directive within the shader function definition, an applications developer can change the default behavior of a reflectance model shader that utilizes ray tracing. Two types of modal operation are supported; surface/solid mode and constrained/unconstrained mode. The choice of surface or solid mode affects how surface normal vectors are treated for double-sided surfaces. The constrained /unconstrained mode affects the treatment of normal vectors.

shader parameter

Glossary: Rendering Control

A shader parameter is one of the arguments used by an application to pass data to and from a shader function, such as the *color* of a light shader. A shader parameter is composed of a name, type, and value. The value is always passed as a Render_Arg.

shader prototype

Glossary: Rendering Control

A shader prototype or type can be considered as a template from which an instance of a shader function is created. Shader prototypes are analogous to *objects* or *classes* in an object oriented language such as C++. As with C++ classes, shader prototypes are not used directly themselves, but serve as a means of creating the instances that actually do the work.

shader uniforms

Glossary: Rendering Control

See *uniform variable*.

shading

Glossary: Rendering Control

Shading is the part of the rendering process that calculates the RGB intensity of a pixel.

shading language

Glossary: Rendering Control

The shading language is a C-like language supported by the shader compiler, in which an application developer can write a new shader function. The shading language includes a set of macros and utility functions to aid the application developer.

shading normal vector

Glossary: Rendering Control

A shading normal vector is a normal vector that an application may attach to the vertices of facets. This may be used in preference to the geometric normal vector in shading operations to avoid faceting artifacts.

shading pipeline

Glossary: Rendering Control

The shading pipeline is a sequence of distinct operations carried out during the shading part of the rendering process. Shader functions are applied in a fixed order, so that if applicable, a displacement shader will precede a transparency shader that in turn precedes a color source shader. The term pipeline is used because in general these operations cannot be carried out in parallel.

shadow

Glossary: Lights and Shadows

A shadow is that part of an object or region of space not directly illuminated by a particular light source. The flat and Gouraud shaded rendering modes do not support shadows.

shadow map

Glossary: Lights and Shadows

A depth-buffered image is computed from the view point of a light source to support the rendering of shadows on object surfaces. Shadow maps can be computed once by an application and reused for any set of viewing parameters until the position of a light or an object's geometry changes. As for other rendered images, shadow maps can be computed to an application specified image resolution.

shallow copy

Glossary: C++ Interface

A shallow copy copies an object, but instead of copying all the other objects it references (a *deep copy*) it references the same objects that the original uses. A shallow copy stores only the first instance of the item in memory, and increments a reference count for each copy.

sheet body

Glossary: Model Topology

A sheet body is an infinitely thin body that has at least one face, but does not enclose a solid area. Generally the faces are double sided, indicating that they are infinitely thin with no material on either side.

A special case is the *laminar sheet* in which two sheet bodies with single sided faces and opposing normals occupy the same space. Laminar sheets should be avoided. That fact that the faces are single sided tells the modeler to expect material on one side of each face. The fact that the sheets occupy the same space means that there can be no material between the sheets. These two conditions are contradictory and can confuse the modeler.

See also *body.*

shell

Glossary: Model Topology

A shell is a collection of connected faces that bound a volume or represent a sheet region. The faces can be connected along either edges or vertices. This is implemented in the class SHELL.

shelling

Glossary: Shelling

Creating a shelled (thin walled) solid by offsetting the faces of a sheet or solid body.

side cap

Glossary: Blending

A cap that does not cover the end of a blend face, but merely a side of it. See also *capping* and *end cap*.

sidedness, face

Glossary: Model Topology

A face's *sidedness* indicates whether it is single-sided or double-sided. A *single-sided* face means that points on one side of it are considered to be inside the shell, and points on the other side are considered to be on the outside. The selection of the inside and outside may be the same as, or converse to, that for the surface of which the face is part—depending on the sense of the face. A single-sided face either completely or partially bounds a solid region, with the face normal pointing away from the solid region. A *double-sided* face means that the points on either side are either *all inside* or *all outside*. If they are all outside, the face is considered to represent an idealized, infinitely thin sheet (a 2D region). If they are inside, the face is an internal partition embedded in a solid. (A single-sided face is a solid body, not a sheet body.)

sidedness, facet

Glossary: Model Topology

All facets have an orientation convention from which an *inside* and *outside* may be determined. By default, a facet is single-sided, meaning that only the outside surface will be visible when rendered. If a facet is double-sided both inside and outside surfaces will be rendered. Sidedness refers to whether a facet is double-sided, and is inherited from the ACIS entity to which the facets are attached; e.g., a body.

single–sided face

Glossary: Model Topology

A *single-sided* face means that points on one side of it are considered to be inside the shell, and points on the other side are considered to be on the outside. The selection of the inside and outside may be the same as, or converse to, that for the surface of which the face is part—depending on the sense of the face. A single-sided face either completely or partially bounds a solid region, with the face normal pointing away from the solid region. (A single-sided face is a solid body, not a sheet body.)

skinning

Glossary: Skinning and Lofting

Fits a surface through a series of curves (wires).

Snell's Law

Glossary: Transparency, Materials

See *refraction, the law of.*

software component

Glossary: Software Architecture

See *component (software).*

solid body

Glossary: Model Topology, Model Geometry

A *solid body* totally encloses a volume. It is not possible to *visit* both sides of any face without first passing through a face. Every boundary of every face is used twice, except in the cases of the analytic sphere and torus, which are closed and require no further topology to bound a volume. A solid body has no dangling faces or dangling edges. A single-sided face is a solid body, not a sheet body.

If a solid body, such as a cube, is missing a face, the body is said to be *unbounded* (infinitely long) in that direction.

See also *body,* and *sheet body.*

solid shading mode

Glossary: Rendering Control, Image Output

See *shader mode*.

solid texture

Glossary: Texture Spaces

A solid texture is a material shader that renders an object as if it were carved out of a single piece of some material. The *solid wood* and *blue marble* color shaders in the basic shader library are examples of solid textures. Solid textures calculate shading values for points on the surface of primitives as a function of their world coordinate. This is in contrast to the use of parametric coordinates by wrapped textures.

spline

Glossary: Construction Geometry

Spline refers to a curve or surface that cannot be directly represented in ACIS by one of the simple analytic surfaces (i.e., by a simple analytic formula), but that can be indirectly represented by an ordered list of analytic formulae, by segmenting the curve or surface with control points. ACIS implements some analytic surfaces, such as ellipsoids, as splines.

SSI

Glossary: Intersectors

Surface/surface intersection. SSI algorithms find the intersection between parametric surfaces, either of which can be a singular surface due to the degeneracy of the surface boundary to a single point.

state

Glossary: Attributes

See *attribute state*.

stencil

Glossary: Transparency, Materials

A stencil is the part of a transparency shader that is used to simulate the effect of holes being cut into a surface. A stencil may be considered to be a mask lying over a surface that describes the presence or absence of material.

subgraph

Glossary: Graph Theory

A graph S is called a subgraph of G if every vertex and edge in S is also in G.

sub–image

Glossary: Viewing

A sub-image is a rectangular sub-region of the current image defined by the function api_rh_set_subimage. A sub-image determines which part of the current image is rendered by subsequent rendering operations.

subshell

Glossary: Model Topology

ACIS can group shells into a hierarchy of shells, subshells, subshells of subshells, and/or faces. By computing and testing boxes that enclose subshells, ACIS can avoid accessing details of subshells and their descendent subshells and faces that are not needed.

super sampling

Glossary: Image Output

Super sampling is a form of anti-aliasing in which additional samples are taken to calculate a more accurate average. If the sampling rate is under the control of some quality estimation function, this is known as adaptive super sampling.

surface

Glossary: Model Geometry, Construction Geometry

Surfaces can be spheres, planes, elliptical cones (including circular cylinders and cones), tori, splines and offset surfaces. A surface record holds a count of the number of faces that refer to the surface, and provides a route to the details of individual surface types.

surface shading mode

Glossary: Rendering Control, Image Output

See *shader mode*.

symmetric

Glossary: Graph Theory

A relation R is called symmetric if for each ordered pair (a,b) in R, the ordered pair (b,a) is also in R.

TARGA

Glossary: Image Output

TARGA is an image format developed by Truevision Inc.

target

Glossary: Viewing

To define a view the user specifies a point in space that represents the eye position, and a direction towards some target position at which the eye is looking.

tessellation

Glossary: Faceting

See *faceting*.

testing

Glossary: Debugging, Precision and Tolerance

Functionality testing insures that all required functionality is implemented and that it works. *Limits testing* insures that parameters have documented limits or ranges or acceptable values, values within valid ranges are properly accepted, values outside of valid ranges are rejected gracefully, and hard-coded limits test values are documented. *Precision and tolerance testing* insures that precision and tolerance settings (resabs, resnor, resfit, and resmch) are known and work properly within the program. *Error generation testing* insures that errors are properly detected and messages are properly output.

texture mapping

Glossary: Texture Spaces, Textures

Texture mapping is a technique for rendering fine surface details on an object without explicitly modeling the geometry. In texture mapping, a two-dimensional image, known as a texture map is projected onto the surface of an object. A texture maps may be either a procedurally defined function or an image, typically scanned or pre-computed. The rendering library provides both texture space shaders and color shaders to support texture mapping.

texture space

Glossary: Texture Spaces

A texture space is a two dimensional coordinate system used to map a wrapped texture onto the surface of a primitive.

texture space shader

Glossary: Texture Spaces

Texture space shaders differ from other classes of shaders in that they do not affect the final intensity of a pixel directly. Rather, they have an indirect effect by defining the texture space that may be used by other shaders.

thin void

Glossary: Model Topology, Model Geometry

Thin voids or *slits* are embedded faces in a solid region; i.e., 2D regions that are *not* in the point set of the model.

3–space

Glossary: Object and Parameter Spaces

See *object space*.

TIFF

Glossary: Image Output

(Tag Image File Format) A general purpose data format for interchange of digital image data. This nonproprietary industry standard is widely used in Macintosh in PC products. TIFF was designed by Aldus Corporation, in cooperation with Microsoft, Hewlett-Packard, Microtek, Datacopy, DEST, and New Image Technology.

TIGA

Glossary: Image Output

Texas Instruments Graphics Architecture (TIGA) is a software interface for communication between applications and graphics hardware based upon the Texas Instruments TMS340 family of hardware. TMS340-based boards provide high resolution graphics on IBM-compatible personal computers.

token

Glossary: Viewing

An enumerated type or integer denoting the type of data elements being passed to a function in the rendering library. Token arrays are employed to enable an application to pass a variable number of parameters to a function through a constant number of arguments in much the same manner as the variable argument array construct in C (argv[ ]). The rendering library function LiCreatePolygon is an example of such a function.

to point

Glossary: Viewing

The “to” point is a point in world space at which the camera is aimed in the camera model of view control.

topology

Glossary: Model Topology

Topology is the set of spatial *relationships* between the various geometric entities, recorded in the classes body, wire, lump, shell, subshell, face, loop, coedge, edge, and vertex.

torus, osculating

Glossary: Construction Geometry

See *osculating torus*.

total internal reflection

Glossary: Reflectance, Materials

When transmitted from a dense medium to a sparse medium, the direction of light is refracted away from the surface normal in accordance with Snell’s Law. As the angle of incidence increases, the angle of refraction increases at a faster rate. Eventually, the refracted direction lies in the tangent plane of the surface. The incident angle at this point is called the critical angle, at which the refracted direction has reached a limit, unable to bend further from the surface normal. If the angle of incidence is increased beyond the critical angle, the incident light is reflected off the interface of the media in accordance with Newton’s Law. This effect is called total internal reflection.

transform

Glossary: Transforms, Model Geometry

A transform is a matrix used to modify the size, orientation, and/or position of a geometric item (gvector, position, entity) relative to the world coordinate system or to another geometric item.

transformation

Glossary: Mathematics, Transforms

A transformation, also known as transform, provides a means of defining the position and orientation of primitives in the rendering library. Transforms may also scale or shear the geometry of an object. Functions are available to create transforms specially for translation, scaling, and rotation. In addition, a skew transform provides a means of applying a shear to an object. Transforms may be concatenated into a single transform.

transparency shader

Glossary: Transparency, Materials

A transparency shader defines the transparency of surface and thus how much light can pass through it. A transparency shader deals with two separate phenomena: filters and stencils. Respectively, these define the color and presence of light passing through the surface.

tree

Glossary: Graph Theory

A graph is called a tree if it is connected and does not contain a cycle.

traversal

Glossary: C++ Interface

Traversal is the visitation of all elements in an interconnected data structure.

unbound symbol

Glossary: Scheme AIDE Application, Scheme Interface

An unbound symbol is a Scheme variable that does not have a value currently bound to (associated with) it. See also *binding*.

unconstrained shading mode

Glossary: Rendering Control

See *shader mode*.

uniform variables

Glossary: Rendering Control

A uniform variable, or uniform, is a shader variable whose value is invariant for a given set of parameters. In particular they are variables whose values are independent of shader global variables and thus need only be calculated once per image.

uniforms function

Glossary: Rendering Control

A uniforms function is associated with a shader and is called to perform shader initialization, including calculation of uniform variables. In general it will be called only when a shader is initialized or the parameters of a shader have been modified.

unit vector

Glossary: Mathematics

A *unit vector* is a vector that always has a magnitude of one. See also *vector*.

unload function

Glossary: Rendering Control

An unload function is associated with a shader prototype and is called when a shader instance is deleted. The role of an unload function is similar to that of a destructor in object oriented languages, such as C++.

up–vector

Glossary: Viewing

An up-vector describes the orientation of the camera in the camera model of view control. The up-vector is passed as a parameter to function api_rh_set_view.

uv–space

Glossary: Object and Parameter Spaces

See *parameter space*.

vector

Glossary: Mathematics, Scheme Interface, Rendering Control

In mathematics, a vector is an expression that has both magnitude and direction (also known as a directed quantity). A vector may be used to represent quantities such as velocity or force. A *unit vector* always has a magnitude of one. Vectors can be represented geometrically as directed line segments.

In Scheme (and some other programming languages), a vector is a fixed-length indexed array (list) with random access to all elements. In Scheme, vector is an inherent data type. In ACIS, a Scheme data type named gvector has been defined to represent an entity that has both magnitude and direction; it is named gvector to differentiate it from the inherent Scheme data type vector. See also *gvector*.

In vector graphics (including rendering), digital images are created through a series of mathematical statements that place lines and shapes in a given two-dimensional or three-dimensional space. A vector graphic file describes a series of points to be connected.

vertex

Glossary: Model Topology

A vertex is the corner of either a face or a wire. A vertex refers to a point in object space and to the edges that it bounds. Other edges can be found by following pointers through the coedges.

vertex normal vector

Glossary: Rendering Control

A normal vector is associated with the vertex of a polygon or polygonal mesh. It may take the value of the geometric normal of the plane in which the polygon lies, or the shading normal vector if this is supplied by the application.

view

Glossary: Viewing

A view consists of a particular position in space relative to the model and a direction in which the viewer is looking at (or away from) the model. View eye position, center, distance, direction, draw mode, eye position, color, font characteristics, and other criteria can all be specified independently using view Scheme calls.

view plane

Glossary: Viewing

The view plane is the virtual plane of projection in the computer graphics analogy of a pinhole camera.

view screen

Glossary: Viewing

A view screen is the projection of the field of view onto the view plane.

view volume

Glossary: Viewing

The view volume is the region within the clipping planes.

viewing control

Glossary: Viewing

Viewing control refers to the specification of those values that affect the projection of three-dimensional objects onto a two-dimensional image. A complete view specification includes the location of the camera, its orientation and field of view of its lens.

viewport

Glossary: Viewing

The view space of the model is mapped to a *viewport* rectangle on the operating system window (display surface). Viewports can be sized, moved, refreshed, etc.

visibility determination

Glossary: Faceting, Viewing, Rendering Control

Visibility determination is the part of the rendering process that calculates which surfaces or parts of surfaces are visible at a pixel. Also known as visible surface calculation.

visit

Glossary: C++ Interface

When traversing an interconnected data structure, each element is visited in order to find other elements.

void

Glossary: Model Geometry

A void (in addition to being a C++ type) refers to an enclosed hollow region within a solid. For example, if a small sphere is subtracted from the center of a larger sphere, the resulting hollow region is called a void.

voxel

Glossary: Ray Tracing

A voxel is an axis-aligned box representing an element of space. The 3D analogue of a 2D pixel.

widget

Glossary: Extending ACIS

A widget is a user interface component in the X Windows Intrinsics. It consists of a window and methods that act on the window. Examples include dialog windows, buttons, and text fields.

window

Glossary: Viewing

An operating system window (the display surface) is referred to as a *display window* or *base window*. Viewports are then created within this window to display specific views of the model.

Windows (Microsoft)

Glossary: Ignore

Windows is a proprietary suite of systems software for IBM-compatible computers, developed by Microsoft Corporation, that provides a multi-window, device-independent graphical interface for application developers.

Windows DIB

Glossary: Image Output

DIB (Device Independent Bitmap) is a format used for recording pixels in Microsoft Windows in a device-independent form.

Windows NT

Glossary: Ignore

An operation system developed by Microsoft Corporation.

Win32

Glossary: Ignore

A 32-bit API for programming Windows applications.

wire

Glossary: Model Topology

A wire is a collection of edges and vertices that typically represent profiles, construction lines, and center lines of swept shapes. Wires can also represent wireframes that, when surfaced, form shells. This is implemented in the class WIRE.

wire body

Glossary: Model Topology

A wire body contains no faces, shells, or lumps. The minimal case of a wire body consists of a wire record, coedge, and edge with null geometry whose start and end are a single vertex with a point. See also *body.*

wireframe body

Glossary: Model Topology

A wireframe body is a representation of the geometric model as a collection of edges (wires) and vertices (but no faces), as opposed to a representation of the model as a solid (with faces). See also *body.*

wireframe rendering mode

Glossary: Rendering Control

A vector rendering mode in which the edges of primitives are output as a series of line segments. In addition, generators may be drawn on the faces of nonpolygonal primitives. The frequency of the generators depends upon the values in the geometric refinement attached to that primitive.

work coordinate system

Glossary: Model Geometry, Work Coordinate Systems

A Work Coordinate System (WCS) is a plane specified in relation to model space in which all entity creation and modification operations take place.

world coordinate system

Glossary: Mathematics, Model Geometry

An application defined Cartesian coordinate system in which camera and objects to be rendered are located. Also known as the *global coordinate system*. The rendering library adopts a right-handed coordinate system where the *x*, *y*, and *z*-axes are in the same relative orientation as the thumb, forefinger, and second finger respectively of a pointing right hand.

working plane

Glossary: Model Geometry, Work Coordinate Systems

The working plane is the *xy*-plane of the active working coordinate system.

wrapped texture

Glossary: Texture Spaces

A wrapped texture is an RH_MATERIAL that supports texture mapping, through the use of an appropriate shader. Wrapped texture states define the color of a point on the surface of a primitive in terms of its texture coordinate as opposed to the world space coordinates employed in solid texture states.

X Windows

Glossary: Ignore

X Windows is an industry standard windowing system developed at MIT, and is commonly found on UNIX work stations.

xyz–space

Glossary: Object and Parameter Spaces

See *object space*.

yon clipping plane

Glossary: Viewing

The yon clipping plane is the far clipping plane. Any portions of objects on the far side of the yon clipping plane from the viewer are excluded (clipped) from the view. See also *hither clipping plane* and *clipping*.