Chapter 6.
Model Topology

Topic: *Model Topology

Topology refers to the spatial relationships between the various entities in a model. Topology describes how geometric entities are connected (connectivity). On its own, topology defines a “rubber” model, whose position is not fixed in space. For example, a circular edge and an elliptical edge are topologically equivalent (but not geometrically). Likewise, a square face and a rhomboid face are topologically equivalent (but not geometrically). A topological entity’s position is fixed in space when it is associated with a geometric entity.

Topology can be bounded, unbounded, or semi-bounded, allowing for complete and incomplete bodies. A solid, for example, can have missing faces, and existing faces can have missing edges. Solids can have internal faces that divide the solid into cells. Bodies such as these are not physically realizable, but can be represented with ACIS.

ACIS *separately* represents the geometry (detailed shape) and the topology (connectivity) of objects. This concept is called *boundary representation*, or *B-rep*, modeling. This provides the ability to determine whether a position is inside, outside, or on the boundary of a volume (which distinguishes a solid modeler from surface or wireframe modelers). ACIS defines the boundary between solid material and empty space. This boundary is made from a closed set of surfaces.

Topology and Boundary Representation

Topic: *Model Topology

The ACIS boundary representation (B-rep) of a model is a hierarchical decomposition of the model’s topology:

Body . . . . The highest level of model object, and is composed of lumps.

Lump . . . . A 1D, 2D, or 3D set of points in space that is disjoint with all other lumps. It is bounded by shells.

Shell . . . . A set of connected faces and wires, and can bound the outside of a solid or an internal void (hollow). *Subshells* form a further decomposition of shells for internal efficiency purposes.

Face . . . . A connected portion of a surface bounded by a set of loops.

Loop . . . . A connected series of coedges. Generally, loops are closed, having no actual start or end point.

Wire . . . . A connected series of coedges that are not attached to a face.

Coedge . . . . Represents the use of an edge by a face. It may also represent the use of an edge by a wire.

Edge . . . . A curve bounded by *vertices*.

Figure 6-1 shows the relationship of the conceptual topology elements that define the boundary representation of an ACIS model.

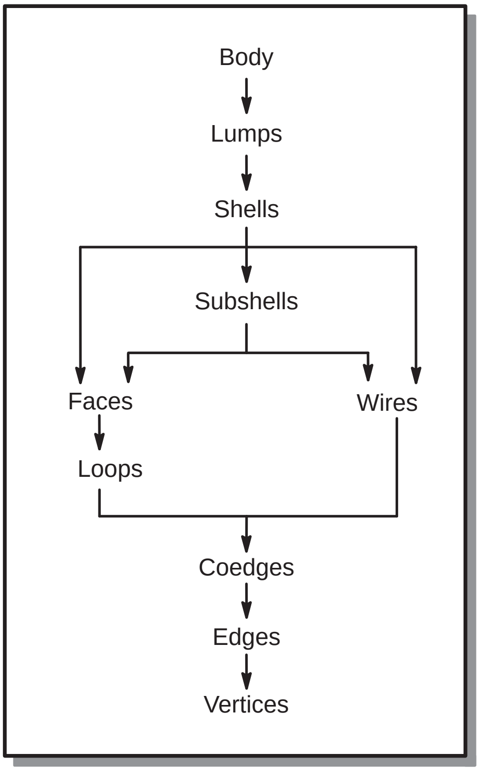

Figure 6-1. Topology

These elements are implemented in ACIS using the C++ classes BODY, LUMP, SHELL, SUBSHELL, FACE, LOOP, WIRE, COEDGE, EDGE, and VERTEX. Each of these classes is derived from the ENTITY class. The specific data and methods of each class is described in the class reference template in online help.

The B-rep topology specifies the *hierarchy* of elements involved. For example, the topology of a hollow cube would include one body, one lump, two shells (one outside and one forming the inner void), 12 loops, 24 edges, and 16 vertices. It also specifies the *relationships* of these elements. For example, the inner void could be centered within the cube, or could be a small rectangular space off to one side. Within these relationships, *transforms* mathematically specify relative location, rotation, scaling, reflection, and shear.

Bodies

Topic: *Model Topology

Bodies are the highest level entities in ACIS models. Typically, a body is a single solid or sheet component, such as a washer, a stripped-down engine block, a zero thickness plate, or a cross section. A body can also be several disjoint bodies treated as one. A transformation recorded with the body relates the local coordinate system of the wires and lumps to the global coordinate space of the body.

Bodies “own” any number of lumps: zero, one, or more. Figure 6-2 shows an example of a body with more than one lump. When a square block (solid lines) is cleared by a cylinder (dashed lines) whose diameter is slightly larger than the length of a side of the block, the block is separated into four lumps. Although the four lumps are not physically joined, they are still treated as a single body.

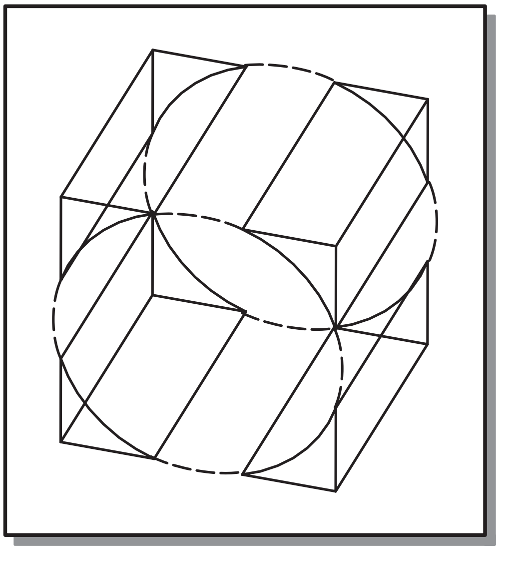

Figure 6-2. Body with Four Lumps

Lumps

Topic: *Model Topology

A lump represents a bounded, connected region in space. A lump is an entire connected set of points, whether the set is 3D, 2D, 1D, or a combination of dimensions. Thus, a solid block with a dangling outside face is one lump, as is a solid block with an internal cavity. Two disconnected sheets are represented as two lumps.

A body contains zero or more lumps, each of which represents a set of points that are disjoint from those represented by all other lumps in the body.

Figure 6-3 illustrates a body with two solid lumps. The large block represents one solid lump that completely encloses a void (dotted line). The second lump (small block with solid lines) is completely enclosed in the void.

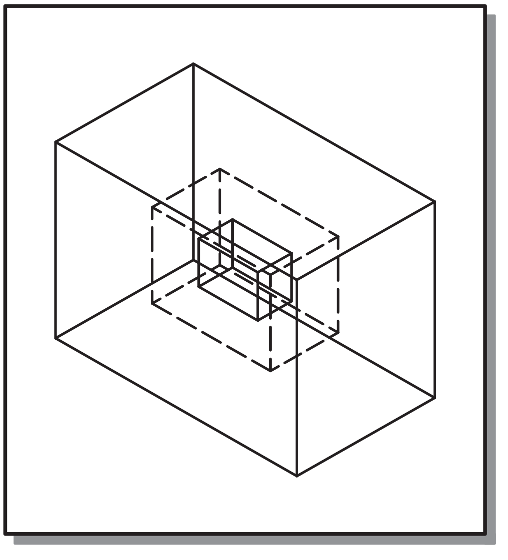

Figure 6-3. Body with Two Lumps

Figure 6-4 shows a body with a solid lump and a sheet lump. It is also possible to have solid and sheet regions within a single lump.

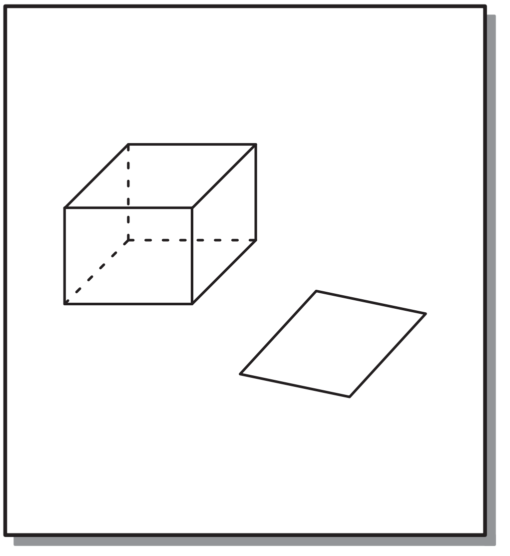

Figure 6-4. Body with Solid Lump and Sheet Lump

Shells

Topic: *Model Topology

A shell is an entire connected set of faces and/or wires, including connections through a nonmanifold vertex. Faces are connected together along common edges or at common vertices; wires may be connected to faces at end vertices.

A solid block with a dangling sheet is one shell, but a block with a cavity is two shells. A solid block with many embedded faces that are all connected through some path to the exterior faces is one shell, but a solid block with a disconnected “floating” embedded face is two shells (but one lump).

The most common type of shell is made up only of complete, finite single-sided faces (see section *Faces*), two of which meet at every edge, with compatible *insides* and *outsides*. Refer to Figure 6-5.

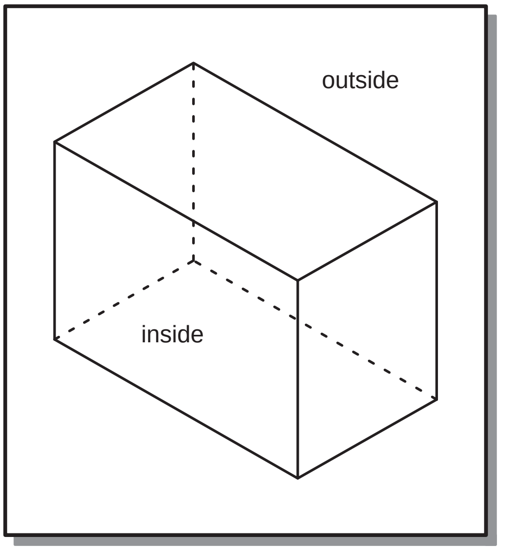

Figure 6-5. Complete Shell

Such a shell divides the whole of space into one finite region and one infinite one. If the finite region is inside the bounding faces, the shell is called *peripheral*, otherwise it represents a finite void in an object, and is called *void*. Refer to Figure 6-6.

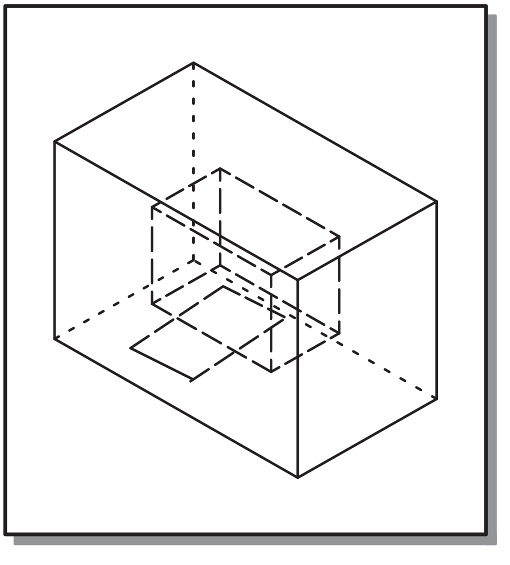

Figure 6-6. Solid with Void Bounded by One Shell

If one or more edges have only one face attached, they are described as *free* edges, and the shell is *open*. If all the faces are double-sided and all exterior or all interior, then the shell is a double-sided *sheet*. If any or all of the faces are single-sided, or there are double-sided faces of both sorts, points in space cannot be classified unambiguously as inside or outside the shell, and it is described as incompletely-bounded, or just *incomplete*. Refer to Figure 6-7.

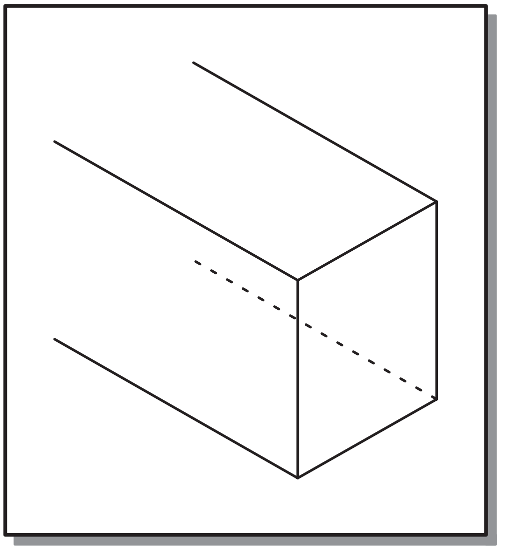

Figure 6-7. Incomplete Shell

More than two faces may meet along an edge, in which case the edge is said to be nonmanifold. Normally, if one collects together every face at a vertex that can be reached from a given face by crossing one or more edges starting or ending at the vertex, the collection contains all of the faces that meet at that vertex. If this is not the case, the vertex is said to be nonmanifold. One or more wires may be attached to a vertex that is already on the boundary of one or more faces. This again makes the vertex nonmanifold.

A shell that contains any nonmanifold edge or vertex is itself said to be nonmanifold. For these more complex shells, the test for whether they are complete or incomplete is correspondingly more complex. Points in space close to single-sided faces are inside or outside according to which side of the face they are on. Points close to double-sided faces or wire edges are inside or outside, according to whether the face or wire is *interior* or *exterior.* Any other point is classified the same as another already-classified point, if they can be joined by a continuous path which does not cross any face or wire. If every point in space can be classified unambiguously in this way, then the shell is complete; otherwise, it is incomplete. Any shell which contains a single-sided face with a free edge is automatically incomplete.

Bodies containing incomplete shells may only participate in modeling operations in regions where their shells are defined. Even then, some configurations give rise to ambiguous results, and so are disallowed.

Subshells

Topic: *Model Topology

ACIS groups shells into a hierarchy of subshells for internal purposes. By computing and testing boxes that enclose subshells, ACIS can avoid accessing details of subshells and their descendent subshells and faces, which improves performance. Subshells are not accessible via the API, and are not generally used directly by applications.

Faces

Topic: *Model Topology

A face is a portion of a single geometric surface in space—the two-dimensional analogue of the body.

Face Boundary Defined by Loops

Zero or more loops of edges constitute the boundary of a face. Figure 6-8 shows a loop of the top face (there are six faces) of a rectangular block.

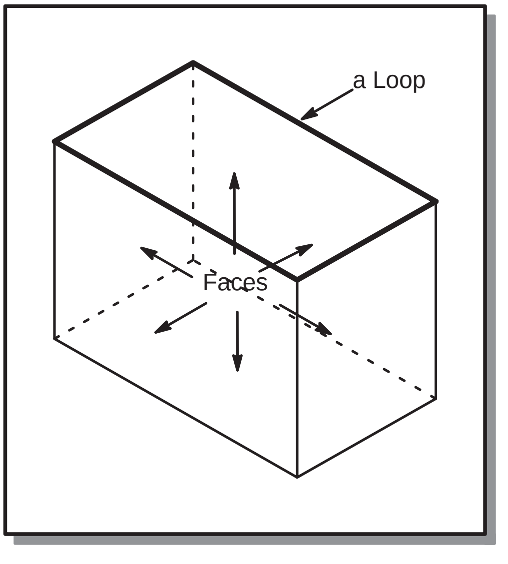

Figure 6-8. Faces

Each loop determines a portion of the surface that is *inside* the face and a portion that is *outside* the face. If the face has no loops, the whole of the surface on which it lies is considered to be inside the face.

If the inside and outside of all the loops are consistent (any line in the surface that joins a point inside to one outside must cross at least one loop) and the inside is connected, then the face is *incompletely defined*. A complete face whose inside is not bounded in space is said to be *infinite*. It is normally not meaningful to ask whether an incomplete face is finite or infinite. There are severe restrictions in ACIS on the operations that can be applied to infinite or incomplete faces.

Face Sidedness, Sense, and Containment

A face's *sidedness* indicates whether it is single-sided or double-sided. A *single-sided* face has material (a solid region) on one side (the "inside"), and is void on the other (the "outside"). A single-sided face either completely or partially bounds a solid region, and the face divides the inside from the outside. (A single-sided face is a solid body, not a sheet body.) 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 all inside, the face is an internal partition embedded in a solid.

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*. A single-sided face's normal always points *away* from the solid region (the material), regardless of the face's sense. Figure 6-9 shows a body with two single-sided faces bounding a solid region; the face normals point away from the material.

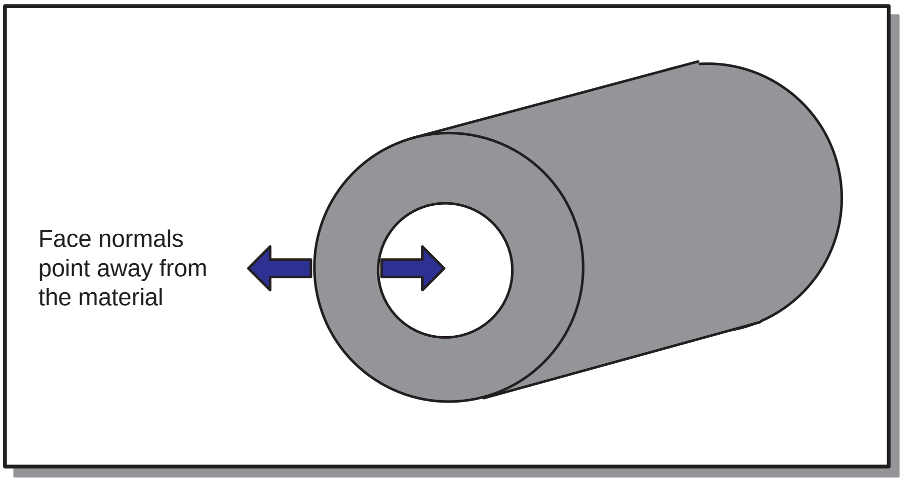

Figure 6-9. Single–Sided Face Normals

A double-sided face's *containment* indicates whether or not the double-sided face is embedded within another region (i.e., contained). It indicates where the solid region (material) is. If a double-sided face is embedded in a solid region, its containment is *both-inside*. If it is a 2D region, its containment is *both-outside*. Containment is not applicable to single-sided faces, because they bound solid regions.

Unlike representations that use a lamina (two back-to-back faces) to represent sheets, 2D regions and embedded faces are stored as a single face representation in ACIS. Faces that make up 2D regions and embedded surfaces are distinguished from faces that form the external boundary of a solid region by the face's sidedness, sense, and containment.

An ACIS face meets one of the following combinations of sidedness, sense, and containment:

Single-sided, either sense . . . . The face bounds a solid region (either completely or partially), so there is material on one side of the face and the other side is void. The "inside" of the face is solid and the "outside" of the face is void. The face normal points away from the material.

Double-sided, both-inside . . . . The face is embedded in a solid. Both sides of the face are solid (there is material on both sides). The face normals point to the inside.

Double-sided, both outside . . . . The face is a 2D region. Both sides of the face are void. Normals point outside.

The following methods (member functions) of the FACE class are used to determine a face's sidedness, sense, and containment:

FACE::sides . . . . Returns an indicator of the number of sides in the face, either SINGLE_SIDED or DOUBLE_SIDED.

FACE::sense . . . . Returns an indicator of the face's sense relative to the underlying surface normal, either FORWARD or REVERSED.

FACE::cont . . . . Returns an indicator of a double-sided face's containment, either BOTH_OUTSIDE or BOTH_INSIDE. This does not apply to single-sided faces.

Figure 6-10 shows an ACIS model that contains three types of faces as defined by their combinations of sidedness, sense, and containment:

- Face 1 is a spherical face that bounds a solid ball region; it is single-sided. The face normal points away from the sphere. The surface normal in this case also points away from the sphere, so the sense of the face is forward.

- Face 2 is embedded in a solid sphere; it is double-sided, both-inside.
- Face 3 is a 2D dangling flap; it is double-sided, both-outside.

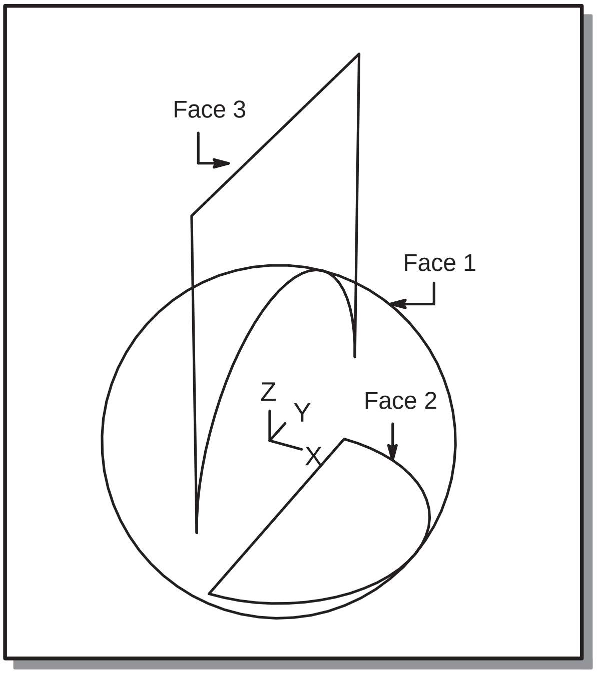

Figure 6-10. Three Face Types

Single–Sided Face Bodies versus Sheet Bodies

A *single-sided face body* is a solid body, not a sheet body. A *sheet body* is an infinitely thin body. ACIS considers a single-sided face body to be a solid that extends from the back side of the face out to infinity, with ill-defined boundaries extending where the edges of the original face extend backward. Because a single-sided face body is a solid, basic solid modeling concepts apply. Due to the ill-defined boundaries of a single-sided face body, subsequent solid modeling operations, such as Booleans, may not work, depending on how the single-sided face body is being used. When ACIS makes a body from a face, such as with API api_mk_by_faces, the body is a solid single-sided face, not a sheet body.

Loops

Topic: *Model Topology

A loop represents a connected portion of the boundary of a face. It consists of a set of coedges linked in a doubly-linked chain which may be *circular* or *open-ended*.

If either end of an open-ended loop is at a finite point, then the face containing the loop is necessarily *incomplete*. If either end is at infinity, then the face is *infinite*.

The illustration in Figure 6-11 contains three closed loops. Each loop is the boundary of a complete, finite face. In the actual physical structure, the adjacent parallel lines are coincident.

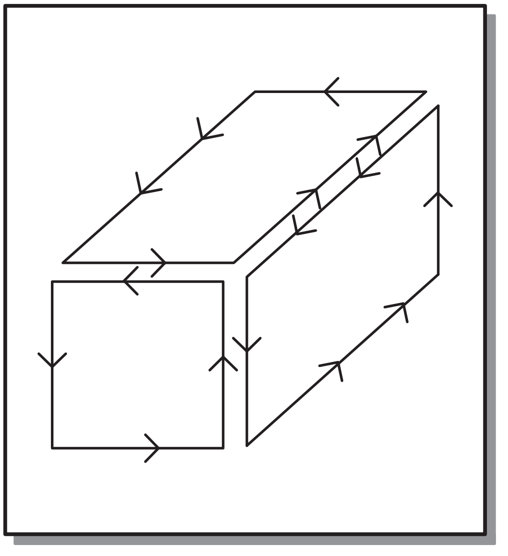

Figure 6-11. Loops of Edges

Wires

Topic: *Model Topology

A wire is a connected collection of edges that are not attached to faces and do not enclose any volume. Wires may represent abstract items like profiles, construction lines and center lines, or idealizations of rod or beam-like objects or internal passages. They are also commonly used to form wire frames to be surfaced to form solid-bounding shells.

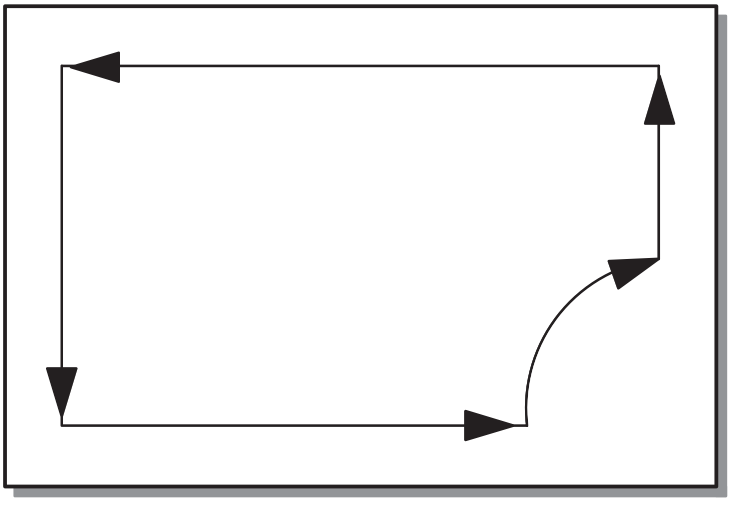

Figure 6-12. Wire Body

A shell may contain a single wire or faces with multiple wires attached to them at vertices. A shell with just a wire is called a *wire shell*, a lump with only a wire shell is a *wire lump*, and a body with only wire lumps is a *wire body.*

Each wire is classified as being *exterior* representing an infinitesimally thin piece of material, or *interior* representing an infinitesimally thin passageway within bulk material.

Coedges

Topic: *Model Topology

A coedge records the occurrence of an edge in a loop of a face. The introduction of coedges permits edges to occur in one, two or more faces, and so makes possible the modeling of sheets and solids (manifold or not). A loop refers to one coedge in the loop, from which pointers lead to the other coedges of the loop. Coedges in a loop are ordered in a continuous path around the loop and are doubly-linked. If a loop is not a circular list, the loop points to the first coedge.

In a manifold solid body shell, each edge is adjacent to exactly two faces; therefore, the edge has two coedges, each associated with a loop in one of the faces (the two faces can be the same, and even the loops can be the same). In this case, the two coedges always go in opposite directions along the edge. In a nonmanifold body shell, there may be more than two coedges associated with an edge. In this case, they are ordered counterclockwise about the edge. Refer to section *Edges* in this chapter for more information.

In Figure 6-13, an isometric view of a solid shows three faces. Each face is bounded by a loop of coedges. Each edge (corner of the block) has two coedges, one for each face that is adjacent to the edge. Each coedge is coincidental with the edge adjacent and parallel to it.

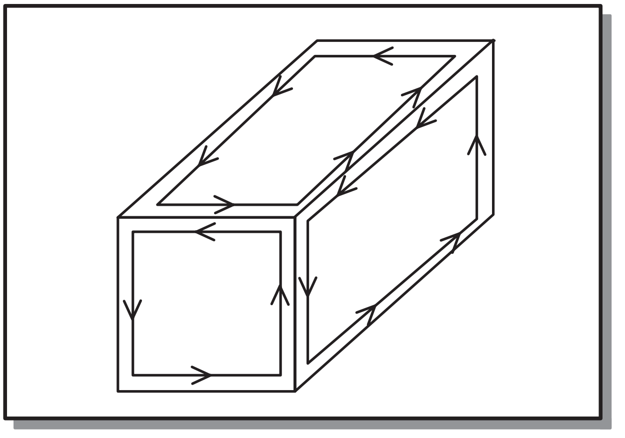

Figure 6-13. Loops of Coedges on Faces

Coedges in a loop are oriented so that looking along the coedge with the outward pointing face normal upwards, the face is on the left. In Figure 6-14, the directions of the face normals (blue arrows) and coedges (red arrows) for several faces of a body are shown to illustrate this orientation.

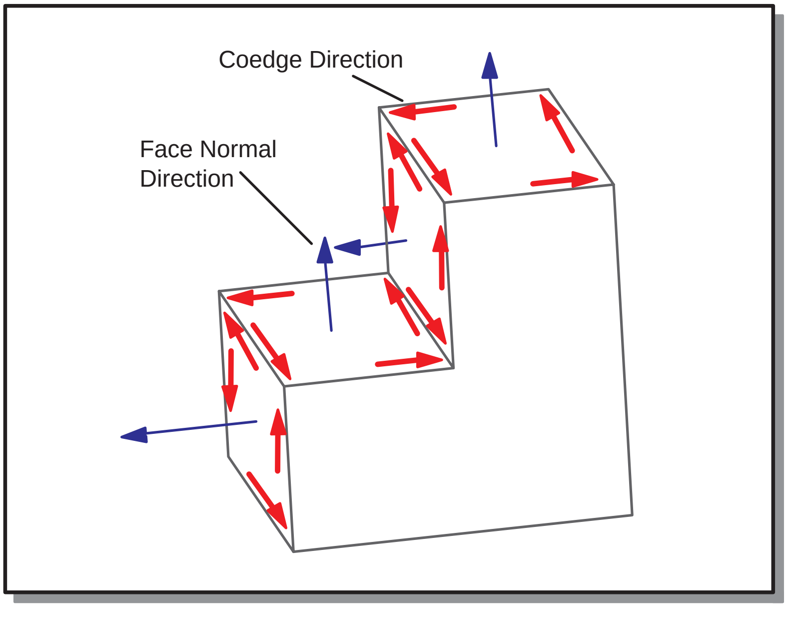

Figure 6-14. Coedge Orientation with Face Normals

In a sheet body, there may be edges that have only one coedge. These are knows as *free edges*, and they mark the boundary of a sheet. If the face attached to the coedge is single-sided, the inside and outside of the associated shell are not well-defined near the edge, and so the shell is necessarily *incomplete*. (Refer to section *Shells* in this chapter for more information.)

Even though the edges of a wire meet no faces, each is given an associated coedge whose pointers link the edge to its neighbors.

The direction of a coedge can be either the same direction as its edge, or it can be the reverse of the edge direction. If it is the same as the edge direction, the coedge's *sense* relative to the edge direction is *forward*; otherwise, its sense is *reversed*. Each coedge contains a record of its sense relative to its edge.

Edges

Topic: *Model Topology

An edge is the topology associated with a curve. The direction of an edge can be either the same direction as its underlying curve, or it can be the opposite direction. If it is the same as the curve direction, the edge's *sense* relative to the underlying curve is *forward*; otherwise, its sense is *reversed*. Each edge contains a record of its sense relative to its underlying curve.

An edge is bounded by one or more vertices, referring to one vertex at each end. If the reference at either or both ends is NULL, the edge is *unbounded* in that direction. If the underlying curve is *infinite*, so is the unbounded edge. When the curve is closed, the vertex references must both be the same or both NULL. If both vertex references are NULL, the edge is the whole curve; unless the curve is open and bounded, in which case the vertices must coincide with the curve ends.

A special case occurs when the geometry pointer is NULL and both vertex pointers point to the same vertex; for example, at the apex of a cone. Few ACIS operations work correctly with edges that have one or both end vertex pointers NULL.

An important feature of ACIS edge representation is the arrangement of the coedges around an edge. (Refer to section *Coedges* in this chapter for more information.) If only two faces meet at an edge, the two coedges from those faces point to each other through the coedge partner pointers. (If there is only one coedge, its partner pointer is NULL.) If more than two faces meet at an edge, the coedges are in a circular linked list. The order of the list is important, because it represents the *radial ordering* of the faces about the edge in a counterclockwise direction. Figure 6-15 shows a sheet body that contains a nonmanifold (more than two faces incident) edge. The edge is directed such that it is coming toward the reader, and the coedges are ordered counterclockwise about it in the partner list as indicated by the arrow.

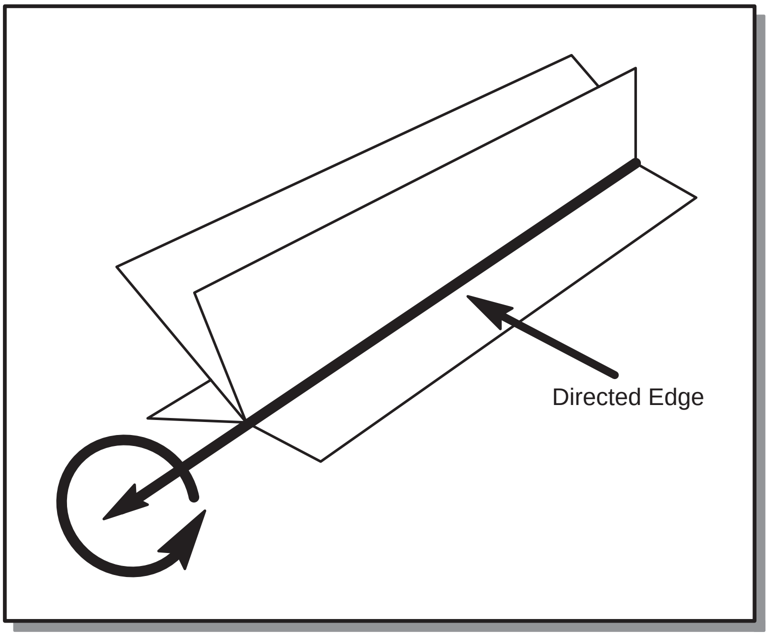

Figure 6-15. Radial Coedge Ordering

The purpose of radial coedge ordering becomes clear during traversal of an ACIS model that contains faces embedded with a solid region, such as in a 3D finite element model. Moving from a face across an edge to the radially closest adjacent face is a simple list operation, rather than a geometric operation as when coedges are not ordered. For example, moving to the face that is closest to the front (outward normal) side of a face is done by moving to the next partner pointer, if the original face coedge is FORWARD. If the original face coedge is REVERSED, moving to the closet face is done by moving to the previous partner pointer (obtained by walking all the way around the list).

In a nonmanifold body shell, the edge may be adjacent to more than two faces. A nonmanifold body is shown in Figure 6-16. An edge common to four faces is at each point where the cylinder is tangent to the outside edge of the block. In the example, edge *ab* is common to two faces on the inside wall of the cylinder (*abcd* and *abfe*) and two faces on the outside of the block (*abgh* and *abjk*). Edge *ab* is common to all four faces. When there are more than two coedges associated with an edge, they are ordered counterclockwise about the edge.

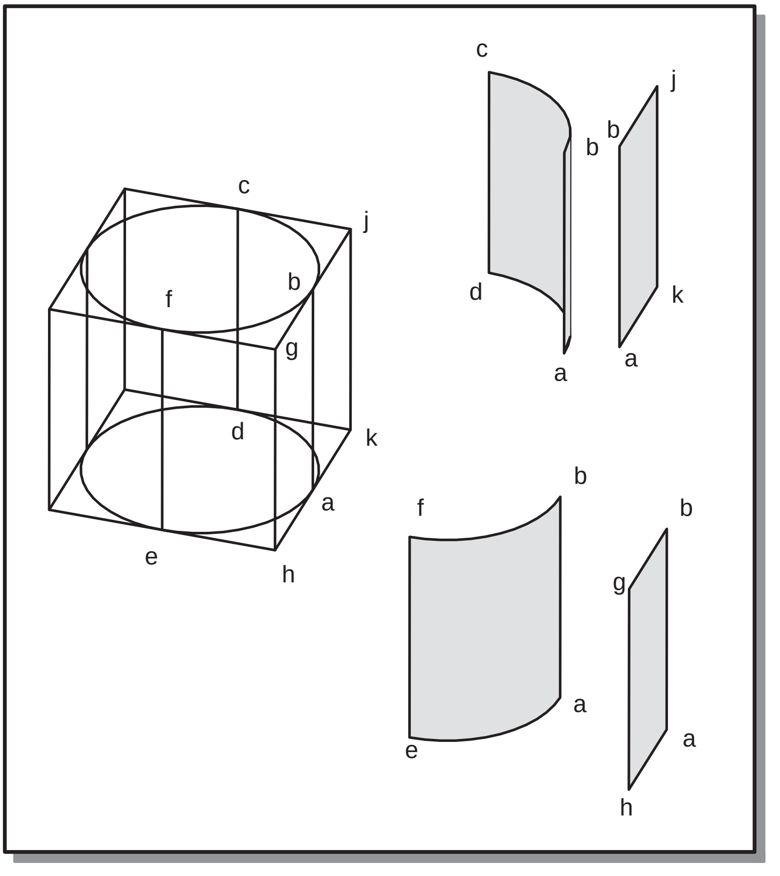

Figure 6-16. Nonmanifold Body With Common Edge

Vertices

Topic: *Model Topology

A vertex is the corner of either a face (Figure 6-17) or a wire. Vertex refers to a point in object space and to the edges that it bounds. The other edges are found by following pointers through the coedges.

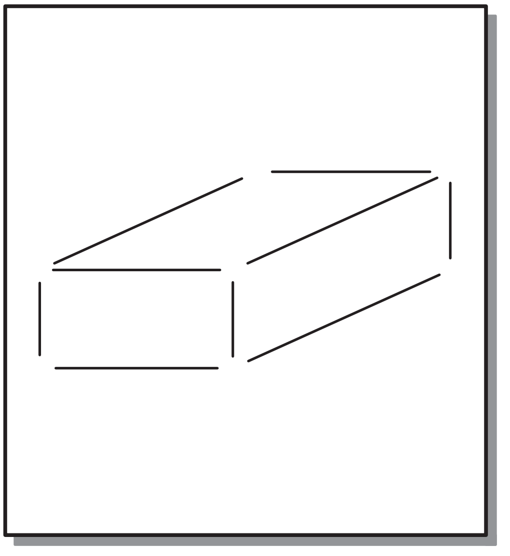

Figure 6-17. Vertices