Chapter 3.
Functions

Topic: Ignore

The function interface is a set of Application Procedural Interface (API) and Direct Interface (DI) functions that an application can invoke to interact with ACIS. API functions, which combine modeler functionality with application support features such as argument error checking and roll back, are the main interface between applications and ACIS. The DI functions provide access to modeler functionality, but do not provide the additional application support features, and, unlike APIs, are not guaranteed to remain consistent from release to release. Refer to the *3D ACIS Online Help User's Guide* for a description of the fields in the reference template.

api_initialize_precise_hidden_line

Function: Precise Hidden Line, Modeler Control

Action: Initializes the precise hidden line library.

Prototype:
```
outcome api_initialize_precise_hidden_line ();
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: System routine

api_phl_clean

Function: Precise Hidden Line

Action: Removes PHL attributes from a list of bodies.

Prototype:
```
outcome api_phl_clean (
    ENTITY_LIST const& bodies,  // body list
    int token                   // view token
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″kernel/kerndata/lists/lists.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
```

Description: This API removes and deletes any PHL attributes with a matching view token from the bodies list.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Changes model

api_phl_compute

Function: Precise Hidden Line

Action: Computes precise hidden line data for a given view and list of bodies, and optionally, stores the data on the model as attributes.

Prototype:
```
outcome api_phl_compute (
    ENTITY_LIST const& bodies,  // body list
    int token,                  // view token
    SPAposition const& eyepos,  // eye position
    SPAposition const& target,  // target position
    logical persp,              // perspective
                                // projection flag
    ENTITY_LIST& edges          // returns edge list
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″kernel/kerndata/lists/lists.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
#include ″baseutil/vector/position.hxx″
#include ″baseutil/logical.h″
```

Description: This API computes the precise hidden line data for the given viewing parameters and list of bodies.

If the view token is nonzero, the data is stored on the model as attributes and identified by token. Existing attributes identified by the same token are replaced. If the perspective projection flag persp is TRUE, data is calculated for display in a perspective projection; otherwise, calculations are done for a parallel projection.

All data returned is allocated with new. It is the user's responsibility to call lose for the returned camera and all entities in the list when they are no longer needed.

Errors: The distance between eyepos and target is less than SPAresabs.

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Changes model

api_phl_retrieve

Function: Precise Hidden Line

Action: Retrieves data stored in PHL attributes and returns it as an ENTITY_LIST of PHL_EDGEs.

Prototype:
```
outcome api_phl_retrieve (
    ENTITY_LIST const& bodies,  // body list
    int token,                  // view token
    ENTITY_LIST& edges,         // returns edge list
    PHL_CAMERA*& camera         // returns camera
                                // definition
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "kernel/kerndata/lists/lists.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
#include "phl_husk/phl_ent/phl_cam.hxx"
```

Description: This API retrieves the data stored in PHL attributes with a matching view token and returns it as an ENTITY_LIST of PHL_EDGEs. The attributes are placed on the body by api_phl_compute when the api_phl_compute's token is set to TRUE.

This API also returns the camera definition.

All data returned is a copy of the data stored in attributes on the body. It is the user's responsibility to call lose for the returned camera and all entities in the list, when they are no longer needed.

Errors: The bodies in the list did not have matching PHL_CAMERAs.

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Changes model

api_phl_set_cell_factor

Function: Precise Hidden Line

Action: Sets the cell factor value.

Prototype:
```
outcome api_phl_set_cell_factor (
    double cell_factor      // cell factor
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
```

Description: To accelerate the determination of the position of a point in relation to the faces or the mutual position of two sub–edges, the projection plane is divided into rectangular cells of the same width and height. The number of cells is determined by a *cell factor*. A larger cell factor value results in larger and fewer cells. This API sets the cell factor value.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_set_curve_resolution

Function: Precise Hidden Line

Action: Sets the curve resolution factor.

Prototype:
```
outcome api_phl_set_curve_resolution (
    double crv_res          // curve resolution
                            // factor
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
```

Description: The curve resolution factor (crv_res) determines in how many sub–edges a given curved edge is divided, depending on its curvature. If crv_res is set to *n*, a complete circle is divided into *n* sub–edges (the radius doesn't matter). The value of resolution is set to 24. Because the projection of a curved sub–edge is approximated by a circle for some operations (not for the intersections of the lines which are determined with the true curves), the quality of a drawing with bugs may be increased with a greater crv_res value; e.g., 36 or 48. The computing time to determine the visibility of the sub–edges increases roughly linearly with the value of crv_res. This API sets the curve resolution factor.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_set_occlusion

Function: Precise Hidden Line

Action: Sets the occlusion flag.

Prototype:
```
outcome api_phl_set_occlusion (
    logical occ             // occlusion flag
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
#include "baseutil/logical.h"
```

Description: This API sets a flag for the calculation of occlusions. If occ is TRUE, api_phl_compute finds edge segments that are obscured by other edge segments and classifies them as occluded (calculates obscured edge segments). If occ is FALSE, occlusion is not calculated and all invisible edge segments are classified as hidden.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_set_shooting

Function: Precise Hidden Line

Action: Sets the shooting precision factor.

Prototype:
```
outcome api_phl_set_shooting (
    double shooting         // shooting precision
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
```

Description: This API sets the precision of the shooting. The length of the difference between the true view vector and the approximated view vector must be smaller than sh. The approximated view vector is the unit vector in space joining the two points (one on each curve) whose projections are to coincide. The value of sh is set to 0.001. The quality of the shooting improves with a smaller value or diminishes with a greater value.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_set_smooth_angle

Function: Precise Hidden Line

Action: Sets the smooth angle.

Prototype:
```
outcome api_phl_set_smooth_angle (
    double angle            // smooth angle
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
```

Description: In order to reduce the computing time, it is possible to avoid determining the visibility of certain lines that are not to be drawn. These lines are classified as *smooth*.

A line is smooth if the following conditions are true:

- It belongs to two faces and only two (topological condition)
- The angle between the two correspondent surfaces is smaller than a given angle (geometrical condition)
- It is an inner line (both faces are projected on both sides of the line)

If the *smooth angle* is zero (default), no lines are smooth. If the smooth angle is 180, all the inner lines are smooth. The visibility of a smooth line is not determined. The correspondent segment is set as “undefined.”

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_show_cell_factor

Function: Precise Hidden Line

Action: Shows the cell factor value.

Prototype:
```
outcome api_phl_show_cell_factor (
    double& cell_factor     // cell factor
    );
```

Includes:
```
#include "kernel/acis.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "phl_husk/phl_api/phl_api.hxx"
```

Description: This API shows the value of the cell factor. For the meaning of cell factor, refer to api_phl_set_cell_factor.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_show_curve_resolution

Function: Precise Hidden Line

Action: Shows the curve resolution value.

Prototype:
```
outcome api_phl_show_curve_resolution (
    double& curve_resolution// curve resolution
    );
```

Includes:
```
#include ”kernel/acis.hxx”
#include ”kernel/kernapi/api/api.hxx”
#include ”phl_husk/phl_api/phl_api.hxx”
```

Description: This API shows the value of the curve resolution. For the meaning of curve resolution, refer to api_phl_set_curve_resolution.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_show_occlusion

Function: Precise Hidden Line

Action: Shows the occlusion flag.

Prototype:
```
outcome api_phl_show_occlusion (
    logical& occ            // occlusion flag
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
#include ″baseutil/logical.h″
```

Description: This API shows the flag for the calculation of occlusions (occ). For the meaning of the flag, refer to api_phl_set_occlusion.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_phl_show_shooting

Function: Precise Hidden Line

Action: Shows the shooting precision factor.

Prototype:
```
outcome api_phl_show_shooting (
    double& shooting        // shooting value
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
```

Description: This API shows the value of the shooting precision factor. For the meaning of shooting, refer to api_phl_set_shooting.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: Read–only

api_terminate_precise_hidden_line

Function: Precise Hidden Line, Modeler Control

Action: Terminates the precise hidden line library.

Prototype:
```
outcome api_terminate_precise_hidden_line ();
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″kernel/kernapi/api/api.hxx″
#include ″phl_husk/phl_api/phl_api.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_api/phl_api.hxx

Effect: System routine

is_ATTRIB_PHL

Function: Precise Hidden Line

Action: Determines if the given entity is an ATTRIB_PHL.

Prototype:
```
logical is_ATTRIB_PHL (
    const ENTITY* e          // entity to test
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_att/att_phl.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_att/att_phl.hxx

Effect: Read–only

is_ATTRIB_PHL_VW

Function: Precise Hidden Line

Action: Determines if the given entity is an ATTRIB_PHL_VW..

Prototype:
```
logical is_ATTRIB_PHL_VW (
    const ENTITY* e          // entity to test
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_att/att_vw.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_att/att_vw.hxx

Effect: Read–only

is_ENTITY_PHL

Function: Precise Hidden Line

Action: Determines if the given entity is an ENTITY_PHL.

Prototype:
```
logical is_ENTITY_PHL (
    const ENTITY* e          // entity to test
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_ent/ent_phl.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_ent/ent_phl.hxx

Effect: Read–only

is_PHL_CAMERA

Function: Precise Hidden Line

Action: Determines if the given entity is a PHL_CAMERA.

Prototype:
```
logical is_PHL_CAMERA (
    const ENTITY* e          // entity to test
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_ent/phl_cam.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_ent/phl_cam.hxx

Effect: Read–only

is_PHL_EDGE

Function: Precise Hidden Line

Action: Determines if the given entity is a PHL_EDGE.

Prototype:
```
logical is_PHL_EDGE (
    const ENTITY* e          // entity to test
    );
```

Includes:
```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_ent/phl_edge.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_ent/phl_edge.hxx

Effect: Read–only

is_PHL_SEGMENT

Function: Precise Hidden Line

Action: Determines if the given entity is a PHL_SEGMENT.

Prototype:
```
logical is_PHL_SEGMENT (
    const ENTITY* e          //
    );
```

Includes:

```
#include ″kernel/acis.hxx″
#include ″baseutil/logical.h″
#include ″kernel/kerndata/data/entity.hxx″
#include ″phl_husk/phl_ent/phl_seg.hxx″
```

Description: Refer to Action.

Errors: None

Limitations: None

Library: phl_husk

Filename: phl/phl_husk/phl_ent/phl_seg.hxx

Effect: Read–only