

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_add_generic_named_attribute

Function: Attributes

Action: Creates a new generic named attribute owned by the given entity.

Prototype:

```
outcome api_add_generic_named_attribute (  
    ENTITY* ent,                // owning entity  
    const char* name,          // attribute name  
    ENTITY* value,             // attribute value  
    split_action split_opt     // SplitLose, SplitKeep,  
    = SplitKeep,              // SplitCopy, SplitCustom  
    merge_action merge_opt    // MergeLose,  
    = MergeKeepKept,         // MergeKeepKept,  
                                // MergeKeepLost,  
                                // MergeKeepOne,  
                                // MergeKeepAll,  
                                // MergeCustom  
    trans_action trans_opt    // TransLose,  
    = TransIgnore,           // TransIgnore,  
                                // TransApply,  
                                // TransCustom  
    copy_action copy_opt      // CopyLose, CopyKeep,  
    = CopyCopy,              // CopyCopy, CopyCustom  
    AcisOptions* ao = NULL    // acis options  
);
```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    const char* value,    // attribute value
    split_action split_opt // SplitLose, SplitKeep,
        = SplitKeep,     // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
        = MergeKeepKept, // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
        = TransIgnore,   // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt  // CopyLose, CopyKeep,
        = CopyCopy,     // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    double value,         // attribute value
    split_action split_opt // SplitLose, SplitKeep,
        = SplitKeep,     // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
        = MergeKeepKept, // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
        = TransIgnore,   // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt  // CopyLose, CopyKeep,
        = CopyCopy,     // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    int value,            // attribute value
    split_action split_opt // SplitLose, SplitKeep,
        = SplitKeep,     // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
        = MergeKeepKept, // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
        = TransIgnore,   // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt  // CopyLose, CopyKeep,
        = CopyCopy,     // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    SPAPosition& value,   // attribute value
    split_action split_opt // SplitLose, SplitKeep,
        = SplitKeep,     // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
        = MergeKeepKept, // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
        = TransIgnore,   // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt  // CopyLose, CopyKeep,
        = CopyCopy,     // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    split_action split_opt // SplitLose, SplitKeep,
        = SplitKeep,     // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
        = MergeKeepKept, // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
        = TransIgnore,   // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt   // CopyLose, CopyKeep,
        = CopyCopy,     // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

```

outcome api_add_generic_named_attribute (
    ENTITY* ent,           // owning entity
    const char* name,     // attribute name
    SPVector& value,      // attribute value
    split_action split_opt // SplitLose, SplitKeep,
                               // SplitCopy, SplitCustom
    merge_action merge_opt // MergeLose,
                               // MergeKeepKept,
                               // MergeKeepLost,
                               // MergeKeepOne,
                               // MergeKeepAll,
                               // MergeCustom
    trans_action trans_opt // TransLose,
                               // TransIgnore,
                               // TransApply,
                               // TransCustom
    copy_action copy_opt,  // CopyLose, CopyKeep,
                               // CopyCopy, CopyCustom
    AcisOptions* ao = NULL // acis options
);

```

Includes: `#include "kernel/acis.hxx"`
`#include "ga_husk/api/ga_api.hxx"`
`#include "ga_husk/attrib/at_name.hxx"`
`#include "kernel/kernapi/api/api.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`
`#include "baseutil/vector/position.hxx"`
`#include "baseutil/vector/vector.hxx"`
`#include "kernel/kernapi/api/acis_options.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/api/ga_api.hxx

Effect: Changes model

api_find_named_attribute

Function:

Attributes

Action: Finds the next named attribute with the same name.

Prototype:

```
outcome api_find_named_attribute (  
    ENTITY* ent,                // owning entity  
    const char* name,          // attribute name  
    ATTRIB_GEN_NAME*& ret_att, // returned attribute  
    AcisOptions* ao = NULL     // acis options  
);
```

Includes: `#include "kernel/acis.hxx"`
`#include "ga_husk/api/ga_api.hxx"`
`#include "ga_husk/attrib/at_name.hxx"`
`#include "kernel/kernapi/api/api.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`
`#include "kernel/kernapi/api/acis_options.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/api/ga_api.hxx

Effect: Read-only

api_initialize_generic_attributes

Function: Attributes, Modeler Control

Action: Initializes the generic attributes library.

Prototype: outcome api_initialize_generic_attributes ();

Includes: #include "kernel/acis.hxx"
#include "ga_husk/api/ga_api.hxx"
#include "kernel/kernapi/api/api.hxx"

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/api/ga_api.hxx

Effect: System routine

api_remove_generic_named_attribute

Function: Attributes

Action: Removes a named attribute attached to an entity.

Prototype: outcome api_remove_generic_named_attribute (
ENTITY* ent, // owning entity
const char* name, // attribute name
AcisOptions* ao = NULL // acis options
);

Includes: #include "kernel/acis.hxx"
#include "ga_husk/api/ga_api.hxx"
#include "kernel/kernapi/api/api.hxx"
#include "kernel/kerndata/data/entity.hxx"
#include "kernel/kernapi/api/acis_options.hxx"

Description: Refer to Action.

Errors: None
Limitations: None
Library: ga_husk
Filename: ga/ga_husk/api/ga_api.hxx
Effect: System routine

api_terminate_generic_attributes

Function: Attributes, Modeler Control
Action: Terminates the generic attributes library.
Prototype: `outcome api_terminate_generic_attributes ();`
Includes: `#include "kernel/acis.hxx"`
`#include "ga_husk/api/ga_api.hxx"`
`#include "kernel/kernapi/api/api.hxx"`
Description: Refer to Action.
Errors: None
Limitations: None
Library: ga_husk
Filename: ga/ga_husk/api/ga_api.hxx
Effect: System routine

is_ATTRIB_GENERIC

Function: Attributes
Action: Determines if an ENTITY is an ATTRIB_GENERIC.
Prototype: `logical is_ATTRIB_GENERIC (
 const ENTITY* e // entity to test
);`
Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_gen.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.
Errors: None
Limitations: None
Library: ga_husk
Filename: ga/ga_husk/attrib/at_gen.hxx
Effect: Read-only

is_ATTRIB_GEN_ENTITY

Function: Attributes
Action: Determines if an ENTITY is an ATTRIB_GEN_ENTITY .
Prototype:

```
logical is_ATTRIB_GEN_ENTITY (  
    const ENTITY* e          // entity to test  
);
```


Includes:

```
#include "kernel/acis.hxx"  
#include "baseutil/logical.h"  
#include "ga_husk/attrib/at_ent.hxx"  
#include "kernel/kerndata/data/entity.hxx"
```


Description: Refer to Action.
Errors: None
Limitations: None
Library: ga_husk
Filename: ga/ga_husk/attrib/at_ent.hxx
Effect: Read-only

is_ATTRIB_GEN_INTEGER

Function: Attributes
Action: Determines if an ENTITY is an ATTRIB_GEN_INTEGER .
Prototype:

```
logical is_ATTRIB_GEN_INTEGER (  
    const ENTITY* e          // entity to test  
);
```

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_int.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_int.hxx

Effect: Read-only

is_ATTRIB_GEN_NAME

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_NAME .

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_name.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_name.hxx

Effect: Read-only

is_ATTRIB_GEN_POINTER

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_POINTER .

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_ptr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_ptr.hxx

Effect: Read-only

is_ATTRIB_GEN_POSITION

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_POSITION .

Prototype:

```
logical is_ATTRIB_GEN_POSITION (  
    const ENTITY* e          // entity to test  
);
```

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_pos.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_pos.hxx

Effect: Read-only

is_ATTRIB_GEN_REAL

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_REAL .

Prototype:

```
logical is_ATTRIB_GEN_REAL (  
    const ENTITY* e          // entity to test  
);
```

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_real.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_real.hxx

Effect: Read-only

is_ATTRIB_GEN_STRING

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_STRING .

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_str.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_str.hxx

Effect: Read-only

is_ATTRIB_GEN_VECTOR

Function: Attributes

Action: Determines if an ENTITY is an ATTRIB_GEN_VECTOR .

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/attrib/at_vec.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/attrib/at_vec.hxx

Effect: Read-only

is_NAMED_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nm_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nm_attr.hxx

Effect: Read-only

is_NAMED_INT_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_INT_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nmi_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nmi_attr.hxx

Effect: Read-only

is_NAMED_LOGICAL_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_LOGICAL_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nml_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nml_attr.hxx

Effect: Read-only

is_NAMED_POS_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_POS_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nmp_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nmp_attr.hxx

Effect: Read-only

is_NAMED_REAL_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_REAL_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nmr_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nmr_attr.hxx

Effect: Read-only

is_NAMED_STRING_ATTRIB

Function: Attributes

Action: Determines if an ENTITY is a NAMED_STRING_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nms_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: Refer to Action.

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nms_attr.hxx

Effect: Read-only

is_NAMED_VEC_ATTRIB

Function: *Attributes*

Action: Determines if an ENTITY is a NAMED_VEC_ATTRIB.

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

Includes: `#include "kernel/acis.hxx"`
`#include "baseutil/logical.h"`
`#include "ga_husk/pmhusk/nmv_attr.hxx"`
`#include "kernel/kerndata/data/entity.hxx"`

Description: None

Errors: None

Limitations: None

Library: ga_husk

Filename: ga/ga_husk/pmhusk/nmv_attr.hxx

Effect: Read-only