UF_GDT_add_datum_identifier (view source)

Defined in: uf_gdt.h

Overview

Add a datum identifier to an existent tolerance feature.

This function does not create a new tolerance feature. If an entirely
new tolerance feature is required use the UF_GDT_create_datum_identifier
routine. The tolerance_feature parameter should be a UF_tolerance_feature_type.

Note: More than one datum identifier could be added to a tolerance feature one
at a tome.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialize
UF_GDT_ERR_NULL_LABEL

Environment

Internal and External

See Also

UF_GDT_create_datum_identifier

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_add_datum_identifier
(
    tag_t tolerance_feature,
    UF_GDT_datum_identifier_t * datum_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to associate datum identifier
UF_GDT_datum_identifier_t *	**datum_data**	Input	Creation data for new datum identifier

UF_GDT_add_datum_to_feature (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to add a datum to tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NULL_LABEL

UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_add_datum_to_feature
(
    tag_t feature,
    tag_t datum
)
```

tag_t	**feature**	Input	Tolerance datum feature
tag_t	**datum**	Input	Datum to be added

UF_GDT_add_fcf (view source)

Defined in: uf_gdt.h

Overview

This function adds a geometric tolerance to an existing tolerance feature. This function does not create a new tolerance feature. If an entirely new tolerance feature is required use the UF_GDT_create_fcf routine.

Environment

Internal and External

See Also

UF_GDT_create_fcf

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_add_fcf
(
    tag_t tolerance_feature,
    UF_GDT_fcf_t * tolerance_data,
    tag_t * fcf
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to associate datum identifier
UF_GDT_fcf_t *	**tolerance_data**	Input	Creation data for new geometric tolerance

tag_t *	**fcf**	Output	Identifier of the geometric tolerances feature control frame

UF_GDT_add_leader (view source)

Defined in: uf_gdt.h

Overview

The following function will add a leader to a given instance based
on the given UF_GDT_leader_spec structure.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_INVALID_VIEW (View to attach leader is invalid)
UF_GDT_ERR_NO_ADD_PROFILE_LDR (Cannot add a profile leader)
UF_GDT_ERR_NO_ADD_TO_INSTANCE (Cannot add leaders to the following:
- instance with a tolerance of size
- instances with a profile leader
- datum targets
- datums without dot terminated leader
)
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_add_leader
(
    tag_t instance,
    UF_GDT_leader_spec_p_t spec
)
```

tag_t	**instance**	Input	Tag of instance to add leader to
UF_GDT_leader_spec_p_t	**spec**	Input	Spec structure of leader to create

UF_GDT_add_size_tolerance (view source)

Defined in: uf_gdt.h

Overview

This function adds a size tolerance to an existing tolerance feature.
This function does not create a new tolerance feature. If an entirely new
tolerance feature is required use the UF_GDT_create_size_tolerance routine.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_add_size_tolerance
(
    tag_t tolerance_feature,
    UF_GDT_size_tolerance_t * size_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to associate size tolerance
UF_GDT_size_tolerance_t *	**size_data**	Input	Creation data for new size tolerance

UF_GDT_ask_annotation_tags (view source)

Defined in: uf_gdt.h

Overview

The following function will query the given feature instance for:
dimension tags
feature control frame tags
appended text tags (of the feature control frames only)
number of non-dimensional leaders

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_FEATURE_INSTANCE (Object is not a feature instance)
UF_err_program_not_initialized

Environment

Internal and External

See Also

UF_GDT_free

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_annotation_tags
(
    tag_t instance,
    UF_GDT_annotation_tags_p_t tags
)
```

tag_t	**instance**	Input	Tag of the instance to query

UF_GDT_annotation_tags_p_t	**tags**	Output to UF_*free*	Data queried off of the instance. Use UF_GDT_free with UF_GDT_ANNOTATION_TAGS_TYPE to free this structure.

UF_GDT_ask_appended_text (view source)

Defined in: uf_gdt.h

Overview

Return the appended text at the specified location for the input feature control frame or dimension. This routines functionality has been extended to include tolerance feature tags. Passing a tolerance feature tag in as parameter 1 will query the size tolerance appended text data for that tolerance feature.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Modified in V16 to include tolerance feature tags

Required License(s)

gateway

```
int UF_GDT_ask_appended_text
(
    tag_t object,
    UF_GDT_text_location_t location,
    int * num_lines,
    char * * * appended_text
)
```

tag_t	**object**	Input	FCF, tolerance feature or dimension to query
UF_GDT_text_location_t	**location**	Input	Appended text location to query
int *	**num_lines**	Output	Number of appended text lines
char * * *	**appended_text**	Output to UF_*free*	Appended text at location requested. This array of strings must be freed by calling UF_free_string_array.

UF_GDT_ask_associated_datums (view source)

Defined in: uf_gdt.h

Overview

The following function will return the datum identifiers associated with the tolerance feature callout or dimension instance.

NOTE: The application is responsible for freeing the array of datums allocated by this function.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_associated_datums
(
    tag_t feature_instance,
    tag_t * * associated_datums,
    int * num_associated_datums
)
```

tag_t	**feature_instance**	Input	Tolerance feature display instance
tag_t * *	**associated_datums**	Output to UF_*free*	Associated datum features
int *	**num_associated_datums**	Output	Number of datum attached to the instance

UF_GDT_ask_association_type (view source)

Defined in: uf_gdt.h

Overview

Given a datum identifier, the following function will return the type of associative relationship for the datum. This is especially useful when the caller needs to know how datum feature symbols are attached to other display instances.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_association_type
(
    tag_t datum_instance,
    UF_GDT_datum_assoc_type_t * assoc_type
)
```

tag_t	**datum_instance**	Input	Datum identifier instance
UF_GDT_datum_assoc_type_t *	**assoc_type**	Input	Type of datum associativity

UF_GDT_ask_callout_strings (view source)

Defined in: uf_gdt.h

Overview

Return the callout strings corresponding to each of the tolerance feature's tolerance types.

If the input tolerance tag is a tolerance feature tag then all of the tolerance feature's callout strings are returned including:
its datum label (if any)
its datum target label (if any)
callout strings for each of its feature control frames (if any)
callout string for its size tolerance (if any)
callout string for its directed dimension (if any)

If the input tolerance tag is a single feature control frame then only the its callout string is returned.

If the input tolerance tag is a composite feature control frame then two callout strings are returned. The first representing the upper tolerance and the second representing the lower tolerance.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_SUBTOLERANCE_TYPE
UF_err_program_not_initialized

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_callout_strings
(
    tag_t tolerance,
    int * num_callout_strings,
    UF_GDT_callout_string_p_t * callout_strings
)
```

tag_t	**tolerance**	Input	Tolerance to query may be either a tolerance feature tag or a feature control frame tag

int *	**num_callout_strings**	Output	Number of callout strings
UF_GDT_callout_string_p_t *	**callout_strings**	Output to UF_*free*	List of callout strings. Use UF_GDT_free to free each element of this array and then UF_free to free the array.

UF_GDT_ask_characteristic (view source)

Defined in: uf_gdt.h

Overview

Given a feature control frame, this function returns the feature control frame's tolerance characteristic.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_characteristic
(
    tag_t fcf,
    UF_GDT_characteristic_t * characteristic
)
```

tag_t	**fcf**	Input	Feature control frame to inquire
UF_GDT_characteristic_t *	**characteristic**	Output	Result of query

UF_GDT_ask_complex_feature (view source)

Defined in: uf_gdt.h

Overview

Given a complex tolerance subfeature, return its parent complex feature.
A complex feature is a tolerance feature defined by a predefined set of subfeatures known as complex tolerance subfeatures.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_COMPLEX_SUB_FEAT

Environment

Internal and external

History

Originally released in V18.0

Required License(s)

gateway

```
int UF_GDT_ask_complex_feature
(
    tag_t complex_sub_feature,
    tag_p_t complex_feature
)
```

tag_t	**complex_sub_feature**	Input	Tag of the complex tolerance subfeature to query.
tag_p_t	**complex_feature**	Output	Complex feature tags containing the complex tolerance subfeature specified.

UF_GDT_ask_complex_sub_features (view source)

Defined in: uf_gdt.h

Overview

Given a complex tolerance feature, return an array of complex sub-features.
A complex sub-feature is a tolerance subfeature whose faces are a subset of the given complex tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_COMPLEX_TOL_FEAT

Environment

Internal and external

History

Originally released in V18.0

Required License(s)

gateway

```
int UF_GDT_ask_complex_sub_features
(
    tag_t complex_feature,
    int * num_complex_sub_features,
    tag_p_t * complex_sub_features
)
```

tag_t	**complex_feature**	Input	Tag of the complex tolerance feature to query.
int *	**num_complex_sub_features**	Output	Number of complex subfeatures returned.

tag_p_t *	**complex_sub_features**	Output to UF_*free*	Array of complex subfeature tags. This array must be freed by calling UF_free.

UF_GDT_ask_component_tolerance_index (view source)

Defined in: uf_gdt.h

Overview

Return the unique index number of a tolerance (fcf or size tolerance) in the component part file that it was pulled from. If the tolerance feature input into this routine has no tolerance or size data, or the tolerance is not a pulled tolerance, an error will be returned.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOLERANCE
UF_GDT_ERR_NO_SIZE_DATA_ON_FEATURE
UF_GDT_ERR_NOT_PULLED_SUBTOLERANCE
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_component_tolerance_index
(
    tag_t tolerance,
    int * index
)
```

tag_t	**tolerance**	Input	This can either be a tolerance feature or fcf tag that is a pulled tolerance.
int *	**index**	Output	Unique index for the tolerance that the given tag was pulled from.

UF_GDT_ask_composite_drf (view source)

Defined in: uf_gdt.h

Overview

This function asks the composite priority for a composite feature control frame.

Return Codes -

UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_composite_drf
(
    tag_t fcf,
    UF_GDT_precedence_t * priority
)
```

tag_t	**fcf**	Input	Feature control frame identifier
UF_GDT_precedence_t *	**priority**	Output	Precedence of lower datum frame

---

UF_GDT_ask_datum_by_label (view source)

Defined in: uf_gdt.h

Overview

This function returns the corresponding datum by a datum label.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_by_label
(
    const char * datum_label,
    tag_t * datum
)
```

const char *	**datum_label**	Input	Datum label string to inquire
tag_t *	**datum**	Output	Tag of the corresponding datum

UF_GDT_ask_datum_ident_parms (view source)

Defined in: uf_gdt.h

Overview

This function returns information on a specified datum identifier.

The input tag object could be a tolerance feature or a datum object.
If it is a tolerance feature, it works the same way as before except that if the tolerance feature refers to more than one datum, an error called UF_GDT_ERR_FEAT_NOT_VALID will returned. If it is a datum object, it will return the information about this datum.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_BASE
UF_GDT_ERR_FEAT_NOT_VALID
UF_GDT_ERR_NOT_IDENT_DATUM

Environment

Internal and External

History

Original release was in 14.0, modified in release v18.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_ident_parms
(
    tag_t base,
    UF_GDT_datum_identifier_p_t * datum_data
)
```

tag_t	**base**	Input	Datum feature or datm object
UF_GDT_datum_identifier_p_t *	**datum_data**	Output to UF_*free*	Returned data concerning the datum identifier. Use UF_GDT_free to deallocate memory when done.

UF_GDT_ask_datum_keywords (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to query keywords on datum objects.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

UF_GDT_ERR_NOT_DATUM

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_keywords
(
    tag_t datum,
    int * num_keywords,
    UF_GDT_keyword_p_t * keywords
)
```

tag_t	**datum**	Input	The datum object
int *	**num_keywords**	Output	The number of keywords
UF_GDT_keyword_p_t *	**keywords**	Output to UF_*free*	The keywords list

---

UF_GDT_ask_datum_multiple_parms (view source)

Defined in: uf_gdt.h

Overview

This function returns information on a specified multiple datum.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_multiple_parms
(
    tag_t multiple_datum,
    UF_GDT_multiple_datum_p_t * datum_data
)
```

tag_t	**multiple_datum**	Input	Identifier of multiple datum queried
UF_GDT_multiple_datum_p_t *	**datum_data**	Output to UF_*free*	Returned data concerning the multiple datum. Use UF_GDT_free to deallocate memory when done.

UF_GDT_ask_datum_of_label (view source)

Defined in: uf_gdt.h

Overview

This function returns the identifier of the datum referred to by a datum label.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_of_label
(
    const char * datum_label,
    tag_t * datum_feature
)
```

const char *	**datum_label**	Input	Datum label string to inquire
tag_t *	**datum_feature**	Output	Identifier of datum referenced by label

UF_GDT_ask_datum_of_target (view source)

Defined in: uf_gdt.h

Overview

Find the tag of the datum parent of a given target.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_datum_of_target
(
    tag_t target,
    tag_t * datum
)
```

tag_t	**target**	Input	Datum target to query
tag_t *	**datum**	Output	Parent datum of the target

UF_GDT_ask_datum_referencers (view source)

Defined in: uf_gdt.h

Overview

Get the feature control frames that reference the datum.

Note: The input could be a datum feature or a datum object.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_GDT_ERR_FEATURE_NOT_DATUM
UF_err_program_not_initialized
UF_GDT_ERR_NOT_BASE

Environment

Internal and External

History

Original release was in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_datum_referencers
(
    tag_t base,
    int * num_fcfs,
    tag_p_t * fcfs
)
```

tag_t	**base**	Input	Datum feature or datum object to query
int *	**num_fcfs**	Output	Number of feature control frames referencing the datum
tag_p_t *	**fcfs**	Output to UF_*free*	List of feature control frames referencing the datum. This should be freed with UF_free.

UF_GDT_ask_datum_references (view source)

Defined in: uf_gdt.h

Overview

The following function returns the datum references of the given data frame in a given feature control frame. The number frames is only greater than one in the case of a composite tolerance. A non-composite tolerance composite frame has only one frame.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_datum_references
(
    tag_t fcf,
    int nth_frame,
    int * num_refs,
    UF_GDT_datum_reference_p_t * * datum_reference
)
```

tag_t	**fcf**	Input	Feature control frame to inquire
int	**nth_frame**	Input	The frame in the feature control frame to query
int *	**num_refs**	Output	Number of datum references returned
UF_GDT_datum_reference_p_t * *	**datum_reference**	Output to UF_*free*	Function_to_free = UF_GDT_free Datum references for the feature control frame. Use UF_GDT_free to deallocate memory when done.

UF_GDT_ask_datums_of_feature (view source)

Defined in: uf_gdt.h

Overview

This routine queries all the datums on the tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and External

History

Originally released in V18.0

Required License(s)

gateway

```
int UF_GDT_ask_datums_of_feature
(
    tag_t datum_feature,
    tag_p_t * datums,
    int * num_datums
)
```

tag_t	**datum_feature**	Input	Tolerance feature to inquire
tag_p_t *	**datums**	Output to UF_*free*	All the datums associated with the tolerance feature, Use UF_free to deallocate memory when done.
int *	**num_datums**	Output	Number of datums

UF_GDT_ask_depth_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to query the depth tolerance information (nominal value, +/- tolerance values, etc.) for a given tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized x

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_depth_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_depth_tolerance_p_t * depth_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_depth_tolerance_p_t *	**depth_data**	Output to UF_*free*	Information on the depth tolerance. The data allocated for this structure must

			be freed by calling UF_GDT_free.

UF_GDT_ask_description (view source)

Defined in: uf_gdt.h

Overview

This function returns the tolerance feature description and name information to the calling application.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_description
(
    tag_t tolerance_feature,
    UF_GDT_description_p_t * info
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
UF_GDT_description_p_t *	**info**	Output to UF_*free*	Description associated with tolerance feature or tolerance datum feature. This must be freed by calling UF_GDT_free.

UF_GDT_ask_directed_dimension (view source)

Defined in: uf_gdt.h

Overview

Finds information for a directed dimension tolerance on a tolerance feature.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

int UF_GDT_ask_directed_dimension
(
tag_t tolerance_feature,
UF_GDT_directed_dimension_p_t * data
)

tag_t	**tolerance_feature**	Input	Tolerance feature tag
UF_GDT_directed_dimension_p_t *	**data**	Output to UF_*free*	Data for the tolerance. This must be freed by calling UF_free.

UF_GDT_ask_directed_dimension1 (view source)

Defined in: uf_gdt.h

Overview

Required License(s)

gateway

int UF_GDT_ask_directed_dimension1
(
tag_t tolerance_feature,
UF_GDT_directed_dimension1_p_t * data
)

tag_t	**tolerance_feature**	Input	Tolerance feature tag
UF_GDT_directed_dimension1_p_t *	**data**	Output to UF_*free*	Data for the tolerance. This must be freed by calling UF_free.

UF_GDT_ask_drf (view source)

Defined in: uf_gdt.h

Overview

This function asks the data of the given datum reference frame.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_drf
(
    tag_t drf,
    UF_GDT_drf_data_p_t * drf_data
)
```

tag_t	**drf**	Input	Datum reference frame identifier
UF_GDT_drf_data_p_t *	**drf_data**	Output to UF_*free*	Data of a datum reference frame. This must be freed by calling UF_free

---

UF_GDT_ask_face_from_index (view source)

Defined in: uf_gdt.h

Overview

Return the geometry tag that corresponds to the geometry index specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

V18.0. This function was enhanced to support smart model attributes.

Required License(s)

gateway

```
int UF_GDT_ask_face_from_index
(
    int geometry_index,
    tag_t * geometry
)
```

int	**geometry_index**	Input	Index of geometry to lookup
tag_t *	**geometry**	Output	Geometry tag with the index specified

---

UF_GDT_ask_face_index (view source)

Defined in: uf_gdt.h

Overview

Find the unique geometry index of a piece of geometry referenced by a tolerance feature or smart model attribute in the part file.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

V18.0. This function was enhanced to support smart model attributes.

Required License(s)

gateway

```
int UF_GDT_ask_face_index
(
    tag_t geometry,
    int * geometry_index
)
```

tag_t	**geometry**	Input	Geometry tag to query index from
int *	**geometry_index**	Output	Geometry index

UF_GDT_ask_face_index_string (view source)

Defined in: uf_gdt.h

Overview

Returns the index of the given face as a string.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_face_index_string
(
    tag_t face_tag,
    char * * index_string
)
```

tag_t	**face_tag**	Input	Tag of the face to query.
char * *	**index_string**	Output to UF_*free*	Face index of the given tag. Use UF_free to deallocate the memory when done.

UF_GDT_ask_faces (view source)

Defined in: uf_gdt.h

Overview

Returns the set of faces associated with the tolerance feature or tolerance datum feature. If the feature is in an out of date or retained state, it is possible that no faces are returned to the caller.
Some valid datum features may not have any associated faces (for example, datum target points, line, areas, centerplanes, etc.).

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_faces
(
    tag_t tolerance_feature,
    int * num_faces,
    tag_p_t * faces
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
int *	**num_faces**	Output	Number of faces in face array
tag_p_t *	**faces**	Output to UF_*free*	Array of faces associated with tolerance feature. Use UF_free to deallocate memory when done.

UF_GDT_ask_fcf_drf (view source)

Defined in: uf_gdt.h

Overview

This function asks the datum reference frame of a feature control frame.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_fcf_drf
(
    tag_t fcf,
    tag_t * drf
)
```

tag_t	**fcf**	Input	Feature control frame identifier
tag_t *	**drf**	Output	Datum reference frame to use with FCF

UF_GDT_ask_fcf_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the defining data for a given single feature control frame. Characteristic, Tolerance Zone, and Datum Reference information is returned for each frame in the array.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_fcf_parms
(
    tag_t fcf,
    UF_GDT_fcf_p_t * fcf_data
)
```

tag_t	**fcf**	Input	Feature control frame to inquire
UF_GDT_fcf_p_t *	**fcf_data**	Output to UF_*free*	Feature control frame data. This should be freed by calling UF_GDT_free.

UF_GDT_ask_fcf_tags (view source)

Defined in: uf_gdt.h

Overview

This function returns the tags of the feature control frames that reference a given tolerance feature. The feature control frames are listed in the array in the order that they appear in their feature control frame.

In the case of multiple feature control frames the the first (topmost) frame is returned in the first element of the "fcfs" array. The second in the second element etc.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_fcf_tags
(
    tag_t tolerance_feature,
    int * num_fcfs,
    tag_p_t * fcfs
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
int *	**num_fcfs**	Output	Number of feature control frames in array
tag_p_t *	**fcfs**	Output to UF_*free*	Array of tags of the feature control frame identifiers associated with tolerance feature. Use UF_free to deallocate memory when done.

UF_GDT_ask_feature_of_datum (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to query datum feature associated with the given datum.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_DATUM
UF_GDT_ERR_DELETED_DATUM

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_feature_of_datum
(
    tag_t datum,
    tag_t * feature
)
```

tag_t	**datum**	Input	The datum object
tag_t *	**feature**	Output	The datum feature assoicated with the datum object

---

UF_GDT_ask_feature_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the tolerance feature creation parameters, given any tolerance feature.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_feature_parms
(
    tag_t tolerance_feature,
    UF_GDT_feature_parms_p_t * feat_parms
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
UF_GDT_feature_parms_p_t *	**feat_parms**	Output to UF_*free*	Tolerance feature creation parameters. UF_GDT_free must be used to free the space allocated for this argument.

UF_GDT_ask_feature_type (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function determines the exact type of feature and return it to the caller.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_feature_type
(
    tag_t tolerance_feature,
    UF_GDT_feature_type_t * feature
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
UF_GDT_feature_type_t *	**feature**	Output	Feature type of tolerance feature.

UF_GDT_ask_features_of_face (view source)

Defined in: uf_gdt.h

Overview

Returns all of the tolerance features associated with the face specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_features_of_face
(
    tag_t face,
    tag_t * * tolerance_features,
    int * num_tolerance_features
)
```

tag_t	**face**	Input	Face to query for associated tolerance features.
tag_t * *	**tolerance_features**	Output to UF_*free*	Array of tags of tolerance features associated with this face. This array must be freed by calling UF_free.
int *	**num_tolerance_features**	Output	Number of tolerance features in the array.

UF_GDT_ask_gdt_object_dfa_file (view source)

Defined in: uf_gdt.h

Overview

This routine queries the DFA file name on the given GDT object.

The GDT object could be one of the following:

Tolerance feature
Datum Reference Frame
Datum
Target
FCF
Tolerance

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_err_bad_parameter_number_1
UF_err_bad_parameter_number_2

Environment

Internal and External

History

Originally released in NX2.0

Required License(s)

gateway

```
int UF_GDT_ask_gdt_object_dfa_file
(
    tag_t gdt_object,
    char * * dfa_file
)
```

tag_t	**gdt_object**	Input	The GDT object to query
char * *	**dfa_file**	Output to UF_*free*	The DFA file name of the given object

UF_GDT_ask_gdt_view_matrix (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to query a view to determine the view matrix.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

NOTE: GDT views functionality is obsoleted in NX3. In NX3 and later, any view will have the same functionality as previous defined gdt view does, so this function will just return the given's view's orientation.

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_gdt_view_matrix
(
    tag_t view,
    double gdt_matrix [ 9 ]
)
```

tag_t	**view**	Input	The view tag to query
double	**gdt_matrix [9]**	Output	Matrix of the GD&T view

UF_GDT_ask_geometric_definition (view source)

Defined in: uf_gdt.h

Overview

Finds information about a geometric definition on a tolerance feature.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_geometric_definition
(
    tag_t tolerance_feature,
    UF_GDT_geometric_definition_p_t * data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
UF_GDT_geometric_definition_p_t *	**data**	Output to UF_*free*	Data for the definition. This must be freed by calling UF_free.

UF_GDT_ask_index_display (view source)

Defined in: uf_gdt.h

Overview

Returns the currently set index display method.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_index_display
(
    UF_GDT_index_display_type_t * index_display
)
```

UF_GDT_index_display_type_t *	**index_display**	Output	Current index display method.

UF_GDT_ask_instance_display_information (view source)

Defined in: uf_gdt.h

Overview

Queries the show feature info status for the work part

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_instance_display_information
(
    logical * preference
)
```

logical *	**preference**	Output	Current display preference setting

---

UF_GDT_ask_keyword_name (view source)

Defined in: uf_gdt.h

Overview

Return the name currently associated with the keyword specified in the input part. If the keyword does not exist in the part then an error will be returned.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_keyword_name
(
    tag_t part_tag,
    UF_GDT_keyword_id_t keyword_id,
    char * * keyword_name
)
```

tag_t	**part_tag**	Input	Part to query
UF_GDT_keyword_id_t	**keyword_id**	Input	Keyword whose name is requested
char * *	**keyword_name**	Output to UF_*free*	Name corresponding to keyword. This must be freed by calling UF_free.

UF_GDT_ask_keyword_text (view source)

Defined in: uf_gdt.h

Overview

Return the text currently associated with the keyword specified in the input part.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_keyword_text
(
    tag_t part_tag,
    const char * keyword_name,
    char * * keyword_text
)
```

tag_t	**part_tag**	Input	Part to query
const char *	**keyword_name**	Input	Keyword whose text is requested
char * *	**keyword_text**	Output to UF_*free*	Text corresponding to keyword. This must be freed by calling UF_free.

UF_GDT_ask_label_of_datum (view source)

Defined in: uf_gdt.h

Overview

Returns the datum label associated with a tolerance datum feature.

The input tag object could be a tolerance feature or a datum object.
If it is a tolerance feature, it works the same way as before except
that if the tolerance feature refers to more than one datum, an error
called UF_GDT_ERR_FEAT_NOT_VALID will returned. If it is a datum object,
it will return the information about this datum.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_BASE
UF_GDT_ERR_FEAT_NOT_VALID
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 14.0, modified in v18.0

Required License(s)

gateway

```
int UF_GDT_ask_label_of_datum
(
    tag_t base,
    char * * datum_label
)
```

tag_t	**base**	Input	Tolerance Datum Feature or datum object to inquire
char * *	**datum_label**	Output to UF_*free*	Label associated with tolerance datum/multiple datum feature. Use UF_free to deallocate memory when done.

UF_GDT_ask_leader (view source)

Defined in: uf_gdt.h

Overview

The following function will query a given instance for the nth
leader object and return information about that leader

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NO_INDEXED_LEADER (Instance's nth leader non-existent)
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_leader
(
    tag_t instance,
    int nth,
    UF_GDT_leader_spec_p_t spec
)
```

tag_t	**instance**	Input	Tag of instance to query
int	**nth**	Input	Index of the leader to query
UF_GDT_leader_spec_p_t	**spec**	Output	Spec structure of the nth leader

UF_GDT_ask_limits_and_fits_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

This routine allows the calling program to query the limits and fits
tolerance information for a given tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized
UF_GDT_ERR_NO_LIMITS_AND_FITS_TOLERANCE

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_ask_limits_and_fits_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_limits_and_fits_tolerance_p_t * data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_limits_and_fits_tolerance_p_t *	**data**	Output to UF_*free*	Information on the limits and fits tolerance. The data allocated for this structure must be freed by calling UF_GDT_free.

UF_GDT_ask_linked_features (view source)

Defined in: uf_gdt.h

Overview

Return an array of features linked to the given tolerance feature.

Environment

Internal and External

See Also

UF_GDT_is_linked_feature

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_linked_features
(
    tag_t tolerance_feature,
    int * num_linked,
    tag_p_t * linked_features
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
int *	**num_linked**	Output	Number of linked features
tag_p_t *	**linked_features**	Output to UF_*free*	Array of linked feature tags. This array must be freed by calling UF_free.

UF_GDT_ask_load_component_flag (view source)

Defined in: uf_gdt.h

Overview

Query the "auto load component" flag for the current
NX work part. This flag control whether component parts should be
automatically loaded when entering the tolerancing module.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_load_component_flag
(
    logical * flag
)
```

logical *	**flag**	Output	Current status of the load component flag. TRUE = Load component flag is set.

UF_GDT_ask_major_dia_feature_of_spline_gear (view source)

Defined in: uf_gdt.h

Overview

This routine returns the sub feature representing the major diameter of the
spline/gear tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in NX3.0

Required License(s)

gateway

```
int UF_GDT_ask_major_dia_feature_of_spline_gear
(
    tag_t tolerance_feature,
    tag_t * major_dia_feature
)
```

tag_t	**tolerance_feature**	Input	Spline/Gear tolerance feature to query
tag_t *	**major_dia_feature**	Output	Sub feature representing the major diameter of the spline/ gear tolerance feature

---

UF_GDT_ask_minor_dia_feature_of_spline_gear (view source)

Defined in: uf_gdt.h

Overview

This routine returns the sub feature representing the minor diameter of the spline/gear tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in NX3.0

Required License(s)

gateway

```
int UF_GDT_ask_minor_dia_feature_of_spline_gear
(
    tag_t tolerance_feature,
    tag_t * minor_dia_feature
)
```

tag_t	**tolerance_feature**	Input	Spline/Gear tolerance feature to query
tag_t *	**minor_dia_feature**	Output	Sub feature representing the minor diameter of the spline/ gear tolerance feature

---

UF_GDT_ask_modl_data (view source)

Defined in: uf_gdt.h

Overview

Ask the modeling feature data referenced by the tolerance feature. Modeling feature list will be NULL if the modeling features are not fully loaded.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release NX2.0.0

Required License(s)

gateway

```
int UF_GDT_ask_modl_data
(
    tag_t tolerance_feature,
    int * num_modl_sets,
    UF_GDT_modl_data_p_t * * modl_sets
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to set
int *	**num_modl_sets**	Output	Number of model sets
UF_GDT_modl_data_p_t * *	**modl_sets**	Output to UF_*free*	Function_to_free = UF_GDT_free Modeling feature data sets. Use UF_GDT_free to free the memory.

UF_GDT_ask_modl_features (view source)

Defined in: uf_gdt.h

Overview

Ask the modeling features referenced by the tolerance feature. Modeling feature list will be NULL if the modeling features are not fully loaded.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release NX2.0.0

Required License(s)

gateway

```
int UF_GDT_ask_modl_features
(
    tag_t tolerance_feature,
    int * num_modl_features,
    tag_t * * modl_features
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to set
int *	**num_modl_features**	Output	Number of modeling features
tag_t * *	**modl_features**	Output to UF_*free*	Modeling feature list. Use UF_free to the memory.

UF_GDT_ask_non_feature_edge_selection (view source)

Defined in: uf_gdt.h

Overview

Queries the feature edges selection mode for the GD&T in the session.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_non_feature_edge_selection
(
    UF_GDT_edge_select_type_t * allow_non_feature_edges
)
```

UF_GDT_edge_select_type_t *	**allow_non_feature_edges**	Output	Current selection preference

---

UF_GDT_ask_num_leaders (view source)

Defined in: uf_gdt.h

Overview

The following function will determine the number of leaders a given
instance has. This function will return a valid leader count for:
Feature instances without a dimension
Datum target instances
Datum instances on dot terminated leaders
For all other instances, a zero will return as the leader count.

Return Codes -
UF_GDT_NO_ERRORS

UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_num_leaders
(
    tag_t instance,
    int * num_leaders
)
```

tag_t	**instance**	Input	Tag of the instance to query
int *	**num_leaders**	Output	Number of leaders on the instance

---

UF_GDT_ask_pitch_dia_feature_of_spline_gear (view source)

Defined in: uf_gdt.h

Overview

This routine returns the sub feature representing the pitch diameter of the spline/gear tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in NX3.0

Required License(s)

gateway

```
int UF_GDT_ask_pitch_dia_feature_of_spline_gear
(
    tag_t tolerance_feature,
    tag_t * pitch_dia_feature
)
```

tag_t	**tolerance_feature**	Input	Spline/Gear tolerance feature to query

tag_t *	**pitch_dia_feature**	Output	Sub feature representing the pitch diameter of the spline/ gear tolerance feature

UF_GDT_ask_product_attributes (view source)

Defined in: uf_gdt.h

Overview

This routine queries the product attributes of the tolerance feature. Currently, only the Spline and Gear type tolerance features have product attributes.

Please refer UF_PD application for details about the product attributes.
Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in NX3.0

Required License(s)

gateway

```
int UF_GDT_ask_product_attributes
(
    tag_t tolerance_feature,
    int * num_product_atts,
    tag_t * * product_atts
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to query
int *	**num_product_atts**	Output	Number of product attributes
tag_t * *	**product_atts**	Output to UF_*free*	List of product attributes

UF_GDT_ask_profile_tol_data (view source)

Defined in: uf_gdt.h

Overview

Queries the profile type and outside zone for a profile of line or profile of surface tolerance. The outside zone is only meaningful if the profile type returned is UF_GDT_PROFILE_UNEQ_BILATERAL.

The outside value may indicate that the rule for the profile type UF_GDT_UNEQ_BILATERAL (0.0 < outside < total_tolerance) has been violated. In this case the tolerance feature of the fcf will be retained.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_INVALID_OBJECT (object is not an fcf)
UF_GDT_NOT_PROFILE_FCF (fcf does not contain a profile of
surface or profile of line
characteristic)
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_profile_tol_data
(
    tag_t fcf,
    UF_GDT_profile_type_p_t profile_type,
    UF_GDT_tolerance_value_p_t outside
)
```

tag_t	**fcf**	Input	profile feature control frame to modify
UF_GDT_profile_type_p_t	**profile_type**	Output	profile type
UF_GDT_tolerance_value_p_t	**outside**	Output	outside tolerance zone when type is UF_GDT_UNEQ_BILATERAL

UF_GDT_ask_pulled_tolerance_component (view source)

Defined in: uf_gdt.h

Overview

Return the component part from which a tolerance (fcf or size tolerance) was pulled. If the tolerance feature input into this routine has no tolerance or size data, or the tolerance is not a pulled tolerance, an error will be returned.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_SUBTOLERANCE
UF_GDT_ERR_NOT_PULLED_SUBTOLERANCE
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_pulled_tolerance_component
(
    tag_t tolerance,
    tag_t * component_part_occ
)
```

tag_t	**tolerance**	Input	This can either be a tolerance feature or fcf tag that is a pulled tolerance.
tag_t *	**component_part_occ**	Output	Part occurrence from which the given tolerance was pulled.

UF_GDT_ask_size_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to query the size tolerance information (nominal value, +/- tolerance values, etc.) for a given tolerance feature.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_size_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_size_tolerance_p_t * size_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_size_tolerance_p_t *	**size_data**	Output to UF_*free*	Information on the size tolerance. The data allocated for this structure must be freed by calling UF_GDT_free.

UF_GDT_ask_size_tolerance_tag (view source)

Defined in: uf_gdt.h

Overview

Returns the identifier of the dimension which defines a size tolerance for a given tolerance feature. Any calls to this routine will only return NULL_TAG. To query size data please use UF_GDT_ask_size_tolerance_parms. To query the appended text on a size tolerance please use UF_GDT_ask_appended_text with the tolerance feature tag as parameter 1.

Environment

Internal and External

History

Original release was in 14.0.
Modified in V16 to always return a NULL_TAG

Required License(s)

gateway

```
int UF_GDT_ask_size_tolerance_tag
(
    tag_t tolerance_feature,
    tag_t * size_tolerance
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
tag_t *	**size_tolerance**	Output	Identifier of size tolerance

UF_GDT_ask_sub_features (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature, return an array of sub-features. A sub-feature is a tolerance feature whose faces are a subset of the given tolerance feature.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_sub_features
(
    tag_t tolerance_feature,
    int * num_sub_features,
    tag_p_t * sub_features
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag

int *	**num_sub_features**	Output	Number of subfeatures returned
tag_p_t *	**sub_features**	Output to UF_*free*	Array of subfeature tags. This array must be freed by calling UF_free.

UF_GDT_ask_super_feature (view source)

Defined in: uf_gdt.h

Overview

For the given tolerance feature find the tolerance feature whose faces are a superset. If no such feature exists, then a NULL_TAG will be returned.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_super_feature
(
    tag_t tolerance_feature,
    tag_t * super_feature
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
tag_t *	**super_feature**	Output	Tag of the super feature

UF_GDT_ask_table_of_instance (view source)

Defined in: uf_gdt.h

Overview

This routine returns the table of the instance. The table is a type of tabular note.

Please refer UF_TABNOT application for a detailed description of tabular notes.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in NX3.0

Required License(s)

gateway

```
int UF_GDT_ask_table_of_instance
(
    tag_t feature_instance,
    tag_t * table
)
```

tag_t	**feature_instance**	Input	Instance of tolerance feature
tag_t *	**table**	Output	Identification tag of table referenced by the instance

UF_GDT_ask_tags (view source)

Defined in: uf_gdt.h

Overview

Return two arrays of tags which correspond to the "displayed elements" of a display instance. Note objects are created for feature control frames and feature control frame appended text. This function is most useful for applications performing translation of display instance information to another data format.

The input display instance could either be a tolerance feature instance or a product definition instance.

NOTE: The application may use UF_OBJ_delete_object to delete each of the tags created and returned by this routine.

NOTE ALSO: Users are responsible for deleting the output curves and annotations after calling this function and finish their operations.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_tags
(
    tag_t display_instance,
    tag_t * * curves,
    int * num_curves,
    tag_t * * annotations,
    int * num_annotations
)
```

tag_t	**display_instance**	Input	Display instance in which to generate curve or annotation tags.
tag_t * *	**curves**	Output to UF_*free*	Array of curves (either line or arc type curves) used to describe the annotation display. This must be freed by calling UF_free.
int *	**num_curves**	Output	Number of curves in array
tag_t * *	**annotations**	Output to UF_*free*	Array of annotations (either dimension, note, or crosshatching) used to describe the annotation display. This array must be freed by calling UF_free.
int *	**num_annotations**	Output	Number of annotations in array

UF_GDT_ask_target_area_xhatch (view source)

Defined in: uf_gdt.h

Overview

This function queries the crosshatching distance and angle parameters for the given datum target area instance.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_area_xhatch
(
    tag_t target_inst,
    double * distance,
    double * angle
)
```

tag_t	**target_inst**	Input	Tag of target instance
double *	**distance**	Output	Crosshatching distance
double *	**angle**	Output	Crosshatching angle

UF_GDT_ask_target_cyl_parms (view source)

Defined in: uf_gdt.h

Overview

Returns the data associated with the cylindrical datum target area.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_target_cyl_parms
(
    tag_t target,
    UF_GDT_target_cyl_area_p_t * data
)
```

tag_t	**target**	Input	Datum target tag
UF_GDT_target_cyl_area_p_t *	**data**	Output to UF_*free*	Cylindrical area data

UF_GDT_ask_target_dia_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the information which defines a circular datum target area.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_dia_parms
(
    tag_t target,
    UF_GDT_target_dia_area_p_t * data
)
```

tag_t	**target**	Input	Datum target tag

UF_GDT_target_dia_area_p_t *	**data**	Output to UF_*free*	Circular area data. Use UF_GDT_free to deallocate the memory when done.

UF_GDT_ask_target_line_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the information which defines a datum target line.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_line_parms
(
    tag_t target,
    UF_GDT_datum_target_line_p_t * data
)
```

tag_t	**target**	Input	Datum target tag
UF_GDT_datum_target_line_p_t *	**data**	Output to UF_*free*	Target line data. Use UF_GDT_free to deallocated the memory when done.

UF_GDT_ask_target_point_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the information which defines a datum target point.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_point_parms
(
    tag_t target,
    UF_GDT_datum_target_point_p_t * data
)
```

tag_t	**target**	Input	Datum target tag
UF_GDT_datum_target_point_p_t *	**data**	Output to UF_*free*	Target point data. Use UF_GDT_free to deallocate the memory allocated.

UF_GDT_ask_target_rect_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the information which defines a rectangular datum target area.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_rect_parms
(
    tag_t target,
    UF_GDT_target_rect_area_p_t * data
)
```

tag_t	**target**	Input	Datum target tag
UF_GDT_target_rect_area_p_t *	**data**	Output to UF_*free*	Rectangular area data. Use UF_GDT_free to free the memory when done.

UF_GDT_ask_target_udef_parms (view source)

Defined in: uf_gdt.h

Overview

This routine returns the information which defines a user defined datum target area.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_target_udef_parms
(
    tag_t target,
    UF_GDT_target_udef_area_p_t * data
)
```

tag_t	**target**	Input	Datum target tag
UF_GDT_target_udef_area_p_t *	**data**	Output to UF_*free*	User defined area data. Use UF_GDT_free to free the memory when done

UF_GDT_ask_targets_of_datum (view source)

Defined in: uf_gdt.h

Overview

Finds all targets for the given tolerance feature.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_targets_of_datum
(
    tag_t datum,
    int * num_targets,
    tag_p_t * targets
)
```

tag_t	**datum**	Input	Datum feature tag
int *	**num_targets**	Output	Number of targets returned

tag_p_t *	**targets**	Output to UF_*free*	Array of target tags. This array must be freed by calling UF_free.

UF_GDT_ask_thread_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to query the thread tolerance information for a given tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.5, NX1.0.1 and NX2.0.0

Required License(s)

gateway

```
int UF_GDT_ask_thread_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_thread_tolerance_p_t * data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_thread_tolerance_p_t *	**data**	Output to UF_*free*	Information on the thread tolerance parms. The data allocated for this structure must be freed by calling UF_GDT_free.

UF_GDT_ask_tol_feat_instance (view source)

Defined in: uf_gdt.h

Overview

This function returns all displayed instances created from the tolerance feature. 0..n instances may be returned.

Environment

Internal and External

History

Original release was in 14.0

Required License(s)

gateway

```
int UF_GDT_ask_tol_feat_instance
(
    tag_t tolerance_feature,
    int * num_instances,
    tag_p_t * feature_instances
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
int *	**num_instances**	Output	Number of instances returned in above array.
tag_p_t *	**feature_instances**	Output to UF_*free*	Array of tags of all instances created from the tolerance feature. Use UF_free to deallocate memory when done.

UF_GDT_ask_tol_feat_of_instance (view source)

Defined in: uf_gdt.h

Overview

Returns the tag of the tolerance feature associated with a given tolerance feature instance.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_tol_feat_of_instance
(
    tag_t feature_instance,
    tag_t * tolerance_feature
)
```

tag_t	**feature_instance**	Input	Tolerance Feature instance to inquire
tag_t *	**tolerance_feature**	Output	Identifier of the tolerance feature referenced by the instance

UF_GDT_ask_tol_feat_tag (view source)

Defined in: uf_gdt.h

Overview

Queries the unique tolerance index for the given feature control
frame or tolerance feature. A tolerance feature tag can only be passed into
this routine if the tolerance feature has a tolerance of size.

Environment

Internal and External

History

This function was originally released in V15.0.

Required License(s)

gateway

```
int UF_GDT_ask_tol_feat_tag
(
    tag_t tolerance,
    tag_t * tolerance_feature
)
```

tag_t	**tolerance**	Input	Tolerance type to inquire
tag_t *	**tolerance_feature**	Output	Tolerance feature identifier of the queried tolerance

UF_GDT_ask_tolerance (view source)

Defined in: uf_gdt.h

Overview

This routine queries the tolerance on the tolerance feature by given type.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT
UF_GDT_ERR_BAD_TOLERANCE_TYPE

Environment

Internal and External

History

Originally released in NX2.0

Required License(s)

gateway

```
int UF_GDT_ask_tolerance
(
    tag_t tol_feat,
    UF_GDT_tolerance_type_t type,
    tag_t * tolerance
)
```

tag_t	**tol_feat**	Input	The tolerance feature to inquire
UF_GDT_tolerance_type_t	**type**	Input	The tolerance type to inquire
tag_t *	**tolerance**	Output	The tolerance associated with the type

---

UF_GDT_ask_tolerance_from_index (view source)

Defined in: uf_gdt.h

Overview

Find the requirement tag for the index specified.

If the "fcf" parameter is a NULL_TAG then the index is returned is
either for size tolerance on a tolerance feature or a smart model
attribute. If an fcf tag is returned then the index is for the fcf
and the "requirement" is the parent feature of the fcf.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

V18.0. This function was enhanced to support smart model attributes.

Required License(s)

gateway

```
int UF_GDT_ask_tolerance_from_index
(
    int index,
    tag_t * requirement,
    tag_t * fcf
)
```

int	**index**	Input	Unique index number for the requirement
tag_t *	**requirement**	Output	Smart model attribute or tolerance feature for the index.
tag_t *	**fcf**	Output	Fcf for the index (if the requirement is a tolerance feature and the index is for a FCF.)

UF_GDT_ask_tolerance_index (view source)

Defined in: uf_gdt.h

Overview

Return the unique index, or requirement number of a tolerance (fcf or size tolerance) or smart model attribute in the part file.

If the tag input is a tolerance feature and this feature contains no tolerance or or size data, an error will be returned.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOLERANCE
UF_GDT_ERR_NO_SIZE_DATA_ON_FEATURE
UF_err_program_not_initialized

Environment

Internal and External

History

V18.0. This function was enhanced to support smart model attributes.

Required License(s)

gateway

```
int UF_GDT_ask_tolerance_index
(
    tag_t requirement,
    int * index
)
```

tag_t	**requirement**	Input	This can either be a tolerance feature, fcf tag or smart model attribute.
int *	**index**	Output	Unique index for the given tag

UF_GDT_ask_tolerance_index_string (view source)

Defined in: uf_gdt.h

Overview

Returns the index of the given tolerance as a string.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_ask_tolerance_index_string
(
    tag_t tolerance_tag,
    char * * index_string
)
```

tag_t	**tolerance_tag**	Input	Tag of the tolerance to query. This should be either the tag of an FCF object or a tolerance feature object.
char * *	**index_string**	Output to UF_*free*	Tolerance index of the given tag. Use UF_free to deallocate the memory when done.

UF_GDT_ask_tolerance_types (view source)

Defined in: uf_gdt.h

Overview

Returns an array of tolerance attributes which compose the given tolerance feature.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_tolerance_types
(
    tag_t tolerance_feature,
    int * num_types,
    UF_GDT_tolerance_type_p_t * types
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
int *	**num_types**	Output	Number of types in types array
UF_GDT_tolerance_type_p_t *	**types**	Output to UF_*free*	Pointer to an array of tolerance types. Use UF_free to deallocate memory when done.

UF_GDT_ask_tolerance_zone (view source)

Defined in: uf_gdt.h

Overview

Returns the information concerning the entire tolerance zone associated with a feature control frame.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_tolerance_zone
(
    tag_t fcf,
    int nth_zone,
    UF_GDT_tolerance_zone_p_t * zone,
    UF_GDT_modifier_data_p_t * frame_mod
)
```

tag_t	**fcf**	Input	Feature control frame to inquire
int	**nth_zone**	Output	Tolerance zone frame to query
UF_GDT_tolerance_zone_p_t *	**zone**	Output to UF_*free*	Pointer to tolerance zone data. Use UF_GDT_free to deallocate memory when done.
UF_GDT_modifier_data_p_t *	**frame_mod**	Output to UF_*free*	Modifier information for the tolerance zone. UF_GDT_free to deallocate memory when done.

UF_GDT_ask_tolerances (view source)

Defined in: uf_gdt.h

Overview

This routine queries all the size tolerances and associated tolerance types on the given tolerance feature.

NOTE: This function will not return any FCFs, to query FCFs, use function UF_GDT_ask_fcf_tags.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and External

History

Originally released in NX2.0

Required License(s)

gateway

```
int UF_GDT_ask_tolerances
(
    tag_t tol_feat,
    tag_p_t * tolerances,
    int * num_tolerances,
    UF_GDT_tolerance_type_p_t * types
)
```

tag_t	**tol_feat**	Input	The tolerance feature to inquire
tag_p_t *	**tolerances**	Output to UF_*free*	All the tolerances associated with the feature, Use UF_GDT_free to deallocate memory when done.
int *	**num_tolerances**	Output	Number of tolerances
UF_GDT_tolerance_type_p_t *	**types**	Output to UF_*free*	The corresponding tolerance types

UF_GDT_ask_tolerancing_standard (view source)

Defined in: uf_gdt.h

Overview

Returns the standard that is in effect in the current drawing.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_ask_tolerancing_standard
(
    UF_GDT_standard_t * standard
)
```

UF_GDT_standard_t *	**standard**	Output	Current engineering standard used for tolerancing

UF_GDT_ask_unit_basis (view source)

Defined in: uf_gdt.h

Overview

Interrogates the unit basis data off of a tolerance feature

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_FCF
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_ask_unit_basis
(
    tag_t fcf,
    UF_GDT_modifier_types_p_t modifier,
    UF_GDT_unit_basis_p_t data
)
```

tag_t	**fcf**	Input	Tag for FCF to interrogate
UF_GDT_modifier_types_p_t	**modifier**	Output	Characteristic type
UF_GDT_unit_basis_p_t	**data**	Output to UF_*free*	Unit basis data for fcf. This must be freed by calling UF_free

---

UF_GDT_ask_wall_thickness_parms (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to query the wall thickness information (nominal value, +/- tolerance values, etc.) for a given tolerance feature.

Environment

Internal and External

History

Original release was in 17.0.

Required License(s)

gateway

```
int UF_GDT_ask_wall_thickness_parms
(
    tag_t tolerance_feature,
    UF_GDT_size_tolerance_p_t * size_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire

UF_GDT_size_tolerance_p_t *	**size_data**	Output to UF_*free*	Information on the wall thickness. The data allocated for this structure must be freed by calling UF_GDT_free.

UF_GDT_break_relationship (view source)

Defined in: uf_gdt.h

Overview

Break a master model relationship between an assembly part and a component part. The following relation types are currently supported:

UF_GDT_PULL_RELATION_TYPE: If this type is specified, the tolerance feature at the assembly level must be supplied as input. After the operation, the assembly level feature will no longer be dependent on the component tolerance.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_INVALID_OBJECT (object is not a tolerance feature)
UF_GDT_FEATURE_NOT_PULLED (attempt to break pull relation on a non-pulled feature)
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_break_relationship
(
    tag_t tolerance_feature,
    UF_GDT_relation_type_t relationship_type
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to convert from an assembly tolerance to a component level tolerance.
UF_GDT_relation_type_t	**relationship_type**	Input	Type of relationship to break. Possible values are: UF_GDT_PULL_RELATION_TYPE

UF_GDT_create_datum_identifier (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to create a datum Identifier.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_create_datum_identifier
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_datum_identifier_p_t datum_data,
    tag_p_t datum_feature
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Number of identifiers in face list array
UF_GDT_datum_identifier_p_t	**datum_data**	Input	Creation information of tolerance datum feature
tag_p_t	**datum_feature**	Output	Identifier of created datum

UF_GDT_create_datum_multiple (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to create a multiple datum. The "num_faces" and "face_list" fields in the UF_GDT_feature_parms_s structure should be set to "0" and "NULL" respectively.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_create_datum_multiple
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_multiple_datum_p_t datum_data,
    tag_t * multiple_datum
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Creation information of tolerance datum feature

UF_GDT_multiple_datum_p_t	**datum_data**	Input	Datum defining data
tag_t *	**multiple_datum**	Output	Identifier of created datum

UF_GDT_create_drf (view source)

Defined in: uf_gdt.h

Overview

This routine creates a datum reference frame from the given data structure.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_create_drf
(
    UF_GDT_drf_data_p_t drf_data,
    tag_t * drf
)
```

UF_GDT_drf_data_p_t	**drf_data**	Input	Data of a datum reference frame
tag_t *	**drf**	Output	Identifier of created frame

UF_GDT_create_fcf (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to create a feature control frame. This does not create a displayed instance.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_create_fcf
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_fcf_t * tolerance_data,
    tag_t * tolerance_feature,
    tag_t * fcf
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Tolerance feature creation data
UF_GDT_fcf_t *	**tolerance_data**	Input	Creation information for the feature control frame
tag_t *	**tolerance_feature**	Output	Identifier of created tolerance feature
tag_t *	**fcf**	Output	Identifier of created FCF

UF_GDT_create_feature_with_product_att (view source)

Defined in: uf_gdt.h

Overview

This routine creates tolerance features with product attributes. Please
refer UF_PD application for details about product attributes.
Currently, the following tolerance feature types can be created with product
attributes -
UF_GDT_GEAR_FEATURE
UF_GDT_SPLINE_FEATURE

When UF_GDT_GEAR_FEATURE feature type is passed, this routine creates a gear
tolerance feature and its sub features.
The created tolerance features are returned in the following order -
Gear type tolerance feature
Pin/Hole type tolerance feature representing pitch diameter
Pin/Hole type tolerance feature representing major diameter
Pin/Hole type tolerance feature representing minor diameter
Circular Tooth Thickness type tolerance feature for external gear
Circular Space Width type tolerance feature for internal gear
Measurement type tolerance feature

Following product attribute type can be specified to create a gear type
tolerance feature -
UF_PD_SPUR_GEAR_TYPE
UF_PD_HELICAL_GEAR_TYPE

When UF_GDT_SPLINE_FEATURE feature type is passed, this routine creates a
spline tolerance feature and its sub features.
The created tolerance features are returned in the following order -
Spline type tolerance feature
Pin/Hole type tolerance feature representing pitch diameter
Pin/Hole type tolerance feature representing major diameter
Pin/Hole type tolerance feature representing minor diameter
Measurement type tolerance feature

Following product attribute type can be specified to create a gear type
tolerance feature -

UF_PD_STRAIGHT_SIDED_SPLINE_TYPE
UF_PD_INVOLUTE_SPLINE_TYPE

The function does not create a display instance.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_INVALID_PRODUCT_ATTRIBUTES
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in NX3.0

Required License(s)

geometric_tol

```
int UF_GDT_create_feature_with_product_att
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_product_attribute_p_t uf_atts,
    int * num_tol_feats,
    tag_t * * tol_feat_list
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Tolerance feature parms
UF_GDT_product_attribute_p_t	**uf_atts**	Input	Product attributes data
int *	**num_tol_feats**	Output	Number of tolerance features created
tag_t * *	**tol_feat_list**	Output to UF_*free*	List of tolerance features created. Use UF_free to free the memory.

UF_GDT_create_instance (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to create an instance of a tolerance feature. The tolerance feature must be of a displayable type.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_create_instance
(
    UF_GDT_tolerance_type_t type,
    tag_t tolerance_feature,
    tag_t view,
    tag_t edge,
    double origin [ 3 ] ,
    double attach_point [ 3 ] ,
    tag_t * feature_instance
)
```

UF_GDT_tolerance_type_t	**type**	Input	Type of tolerance feature to create
tag_t	**tolerance_feature**	Input	Referenced tolerance feature
tag_t	**view**	Input	Tag of the view where the instance is displayed
tag_t	**edge**	Input	Solid edge where the leader is located
double	**origin [3]**	Input	Location of instance origin
double	**attach_point [3]**	Input	Location of attachment point
tag_t *	**feature_instance**	Output	Identification tag of created feature instance

---

UF_GDT_create_modl_based_feature (view source)

Defined in: uf_gdt.h

Overview

Allows the calling program to create a model based tolerance feature. A model based tolerance feature references NX modeling features. This function does not create a display instance.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_INVALID_MODL_DATA
UF_GDT_ERR_MODL_FEATURES_NOT_FULLY_LOADED
UF_GDT_ERR_INVALID_FACE_MAP
UF_GDT_ERR_FACE_MAP_MISMATCH
UF_GDT_ERR_INVALID_MODL_FEATURE_TYPE
UF_GDT_ERR_INVALID_MODL_PARAM
UF_GDT_ERR_MODL_PARAM_MISMATCH

Environment

Internal and External

History

Original release NX2.0.0

Required License(s)

geometric_tol

```
int UF_GDT_create_modl_based_feature
(
    UF_GDT_feature_parms_p_t feat_parms,
    int * num_tol_feats,
    tag_t * * tol_feat_list
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Tolerance feature parms
int *	**num_tol_feats**	Output	Number of tolerance features created
tag_t * *	**tol_feat_list**	Output to UF_*free*	List of tolerance features created. Use UF_free to free the memory.

UF_GDT_create_size_tolerance (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to create a size tolerance. This does not create a displayed instance.

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_size_tolerance
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_size_tolerance_p_t size_data,
    tag_t * size_tolerance
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Tolerance feature creation data
UF_GDT_size_tolerance_p_t	**size_data**	Input	Creation information for the size tolerance
tag_t *	**size_tolerance**	Output	Identifier of created size tolerance

UF_GDT_create_table_instance (view source)

Defined in: uf_gdt.h

Overview

This routine creates a table instance of a tolerance feature. The table instance references a tabular note and displays the tolerance feature data in the table format.

Following tolerance feature types can have a table instance -
UF_GDT_SPLINE_FEATURE
UF_GDT_GEAR_FEATURE

Please refer UF_TABNOT application for a detailed description of tabular notes.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in NX3.0

Required License(s)

geometric_tol

```
int UF_GDT_create_table_instance
(
    tag_t tolerance_feature,
    tag_t view,
    double origin [ 3 ] ,
    tag_t * feature_instance
)
```

tag_t	**tolerance_feature**	Input	Referenced tolerance feature
tag_t	**view**	Input	Tag of the view where the instance is displayed
double	**origin [3]**	Input	Location of instance origin
tag_t *	**feature_instance**	Output	Identification tag of created feature instance

UF_GDT_create_target_cyl_area (view source)

Defined in: uf_gdt.h

Overview

Creates the a cylindrical datum target area on the part with the area data specified.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_create_target_cyl_area
(
    UF_GDT_target_cyl_area_p_t target_data,
    tag_t * target_area
)
```

UF_GDT_target_cyl_area_p_t	**target_data**	Input	Target area data
tag_t *	**target_area**	Output	Tag of target area

UF_GDT_create_target_dia_area (view source)

Defined in: uf_gdt.h

Overview

This function creates a circular (or annular) datum target area with the area data specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_target_dia_area
(
    UF_GDT_target_dia_area_p_t target_data,
    tag_t * target_area
)
```

UF_GDT_target_dia_area_p_t	**target_data**	Input	Target area data
tag_t *	**target_area**	Output	Tag of target area

UF_GDT_create_target_line (view source)

Defined in: uf_gdt.h

Overview

This function creates a datum target line with the line data specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1

UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_target_line
(
    UF_GDT_datum_target_line_p_t target_data,
    tag_t * target_line
)
```

UF_GDT_datum_target_line_p_t	**target_data**	Input	Target line data
tag_t *	**target_line**	Output	Tag of target line

UF_GDT_create_target_point (view source)

Defined in: uf_gdt.h

Overview

This function creates a datum target point with the data specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_target_point
(
    UF_GDT_datum_target_point_p_t target_data,
    tag_t * point
)
```

UF_GDT_datum_target_point_p_t	**target_data**	Input	Target point data
tag_t *	**point**	Output	Tag of target point

UF_GDT_create_target_rect_area (view source)

Defined in: uf_gdt.h

Overview

This function creates a rectangular datum target area with the area data specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_target_rect_area
(
    UF_GDT_target_rect_area_p_t target_data,
    tag_t * target_area
)
```

UF_GDT_target_rect_area_p_t	**target_data**	Input	Target area data
tag_t *	**target_area**	Output	Tag of target area

UF_GDT_create_target_udef_area (view source)

Defined in: uf_gdt.h

Overview

This function creates a user defined datum target area with the area data specified.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_target_udef_area
(
    UF_GDT_target_udef_area_p_t target_data,
    tag_t * target_area
)
```

UF_GDT_target_udef_area_p_t	**target_data**	Input	Target area data
tag_t *	**target_area**	Output	Tag of target area

UF_GDT_create_wall_thickness (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to create a wall thickness tolerance feature. This does not create a displayed instance.

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_create_wall_thickness
(
    UF_GDT_feature_parms_p_t feat_parms,
    UF_GDT_size_tolerance_p_t thickness_data,
    tag_t * thickness_feature
)
```

UF_GDT_feature_parms_p_t	**feat_parms**	Input	Tolerance feature creation data
UF_GDT_size_tolerance_p_t	**thickness_data**	Input	Creation information for the wall thickness tolerance
tag_t *	**thickness_feature**	Output	Identifier of created tolerance feature

UF_GDT_export_drawings (view source)

Defined in: uf_gdt.h

Overview

This routine exports the given set of drawings as well as their embedded GD&T to the given part.

PLEASE NOTE THE FOLLOWING WHEN USING THIS ROUTINE:

- The export will retain parametrics but will NOT export referenced objects just to accomplish this. This mimics the UF_PART_maintain_parms part export option.
- The export will deeply copy expressions. This mimics the UF_PART_copy_exp_deeply part export option.
- The export will load any parts needed to resolve references ON DEMAND.

Return Codes -

UG_GDT_NO_ERRORS
PART_err_failed_to_load_part

Environment

Internal and External

History

Created in v16.0.3

Required License(s)

geometric_tol

```
int UF_GDT_export_drawings
(
    int num_drawings,
    tag_p_t drawings,
    tag_t target_part
)
```

int	**num_drawings**	Input	Number of drawings to export
tag_p_t	**drawings**	Input	Array of drawing tags to export
tag_t	**target_part**	Input	Target part for the export operation

UF_GDT_free (view source)

Defined in: uf_gdt.h

Overview

Frees the memory allocated by the GDT module for its various data structures.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_free
(
    UF_GDT_data_type_t type,
    void * data
)
```

UF_GDT_data_type_t	**type**	Input	The type of data structure to free
void *	**data**	Input	Pointer to a data structure

UF_GDT_has_depth_tolerance (view source)

Defined in: uf_gdt.h

Overview

Queries if the given tolerance feature has a depth tolerance.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V18.0

Required License(s)

geometric_tol

```
int UF_GDT_has_depth_tolerance
(
    tag_t tolerance_feature,
    logical * has_depth_tolerance
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_depth_tolerance**	Output	TRUE if the tolerance feature has a depth tolerance.

UF_GDT_has_directed_dimension (view source)

Defined in: uf_gdt.h

Overview

Queries if the given tolerance feature has a directed dimension.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_has_directed_dimension
(
    tag_t tolerance_feature,
    logical * has_directed_dimension
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_directed_dimension**	Output	TRUE if the tolerance feature has a directed dimension.

UF_GDT_has_geometric_definition (view source)

Defined in: uf_gdt.h

Overview
Query if the given tolerance feature has a geometric definition.

Environment
Internal and External

History
Originally released in V16.0

Required License(s)
geometric_tol

```
int UF_GDT_has_geometric_definition
(
    tag_t tolerance_feature,
    logical * has_geometric_definition
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_geometric_definition**	Output	TRUE if the tolerance feature has a geometric definition.

UF_GDT_has_limits_and_fits_tolerance (view source)

Defined in: uf_gdt.h

Overview
Queries if the given tolerance feature has a limits and fits tolerance.

Environment
Internal and External

History
Originally released in V18.0

Required License(s)
geometric_tol

```
int UF_GDT_has_limits_and_fits_tolerance
(
    tag_t tolerance_feature,
    logical * has_limits_and_fits_tolerance
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_limits_and_fits_tolerance**	Output	TRUE if the tolerance feature has a limits and fits tolerance.

UF_GDT_has_size_tolerance (view source)

Defined in: uf_gdt.h

Overview

Queries if the given tolerance feature has a size tolerance.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V18.0

Required License(s)

geometric_tol

```
int UF_GDT_has_size_tolerance
(
    tag_t tolerance_feature,
    logical * has_size_tolerance
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_size_tolerance**	Output	TRUE if the tolerance feature has a size tolerance.

UF_GDT_has_wall_thickness (view source)

Defined in: uf_gdt.h

Overview

Queries if the given tolerance feature has a wall thickness tolerance.

Environment

Internal and External

History

Originally released in V17.0

Required License(s)

geometric_tol

```
int UF_GDT_has_wall_thickness
(
    tag_t tolerance_feature,
    logical * has_wall_thickness
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
logical *	**has_wall_thickness**	Output	TRUE if the tolerance feature has a directed dimension.

UF_GDT_inherit_model_gdt_to_drawing (view source)

Defined in: uf_gdt.h

Overview

Inherits the GD&T info from a model to drawing member view

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_inherit_model_gdt_to_drawing
(
    tag_t member_view
)
```

tag_t	**member_view**	Input	member view in which to inherit

UF_GDT_init (view source)

Defined in: uf_gdt.h

Overview

This function will initialize data structures in UF_GDT

Return Codes -

UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.4 and v19.0

Required License(s)

geometric_tol

```
int UF_GDT_init
(
    UF_GDT_data_type_t type,
    void * data
)
```

UF_GDT_data_type_t	**type**	Input	- Type of data structure to free
void *	**data**	Input / Output	data structure to be initialized to

---

UF_GDT_is_complex_feature (view source)

Defined in: uf_gdt.h

Overview

Returns TRUE if the tolerance feature is a Complex Tolerance Feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and external

History

Originally released in V18.0

Required License(s)

gateway

```
int UF_GDT_is_complex_feature
(
    tag_t tolerance_tag,
    logical * status
)
```

tag_t	**tolerance_tag**	Input	Tag of the tolerance feature to query
logical *	**status**	Output	Status of the query

UF_GDT_is_complex_sub_feature (view source)

Defined in: uf_gdt.h

Overview

Returns TRUE if the tolerance feature is a Complex Tolerance Subfeature.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and external

History

Originally released in V18.0

Required License(s)

gateway

```
int UF_GDT_is_complex_sub_feature
(
    tag_t tolerance_tag,
    logical * status
)
```

tag_t	**tolerance_tag**	Input	Tag of the tolerance feature to query
logical *	**status**	Output	Status of the query

UF_GDT_is_composite_tolerance (view source)

Defined in: uf_gdt.h

Overview

Given a feature control frame, this function determines if the feature control frame is a composite tolerance. This function does not return a true value for a non-composite tolerance composite feature control frame.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_is_composite_tolerance
(
    tag_t fcf,
    logical * is_composite_tol
)
```

tag_t	**fcf**	Input	Feature control frame to inquire
logical *	**is_composite_tol**	Output	Result of query

UF_GDT_is_datum (view source)

Defined in: uf_gdt.h

Overview

Determines if the feature is a datum feature or a tolerance feature. Datum features include: target points, target lines, target areas, centerplanes, and all other datum identifiers. Tolerance features include those tolerances with feature control frame callouts and datum features defined with a tolerance.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_is_datum
(
    tag_t tolerance_feature,
    logical * is_datum
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
logical *	**is_datum**	Output	Status of query

UF_GDT_is_datum_target (view source)

Defined in: uf_gdt.h

Overview

Determines if the feature is a datum target feature or not.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

UF_GDT_ERR_NOT_TOL_FEAT

Environment

Internal and External

History

Original release was in 19.0.

Required License(s)

gateway

```
int UF_GDT_is_datum_target
(
    tag_t tolerance_feature,
    logical * is_datum_target
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
logical *	**is_datum_target**	Output	Status of query

UF_GDT_is_feature_of_size (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function determines if the feature describes a feature of size.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

gateway

```
int UF_GDT_is_feature_of_size
(
    tag_t tolerance_feature,
    logical * feature_of_size
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
logical *	**feature_of_size**	Output	Flag indicating if the feature describes a feature of size.

UF_GDT_is_gdt_view (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to query a view to determine if it is a GD&T view.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

NOTE: Since GDT views functionality is obsoleted in NX3, this function will always return TRUE. Because in NX3 and later, any view will have the same functionality as previous defined gdt view does.

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_is_gdt_view
(
    tag_t view,
    logical * is_gdt_view
)
```

tag_t	**view**	Input	The view tag to query
logical *	**is_gdt_view**	Output	TRUE if the view is a GD&T view.

UF_GDT_is_linked_feature (view source)

Defined in: uf_gdt.h

Overview

Determine if the tolerance feature specified is a linked feature.

Environment

Internal and External

See Also

UF_GDT_ask_linked_features

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_is_linked_feature
(
    tag_t tolerance_feature,
    logical * is_linked
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to query
logical *	**is_linked**	Output	TRUE if feature is linked

UF_GDT_is_modl_based (view source)

Defined in: uf_gdt.h

Overview

Does this tolerance feature reference modeling features.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release NX2.0.0

Required License(s)

gateway

```
int UF_GDT_is_modl_based
(
    tag_t tolerance_feature,
    logical * is_modl_based
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
logical *	**is_modl_based**	Output	Status of query

UF_GDT_is_pulled_tolerance (view source)

Defined in: uf_gdt.h

Overview

Determines if the tolerance feature being queried has been pulled from a component part.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_INVALID_OBJECT (object is not a tolerance feature)
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_is_pulled_tolerance
(
    tag_t tolerance_feature,
    logical * status
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to query
logical *	**status**	Output	Status of query

UF_GDT_is_retained (view source)

Defined in: uf_gdt.h

Overview

Determines if the tolerance feature specified is currently retained.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_GDT_is_retained
(
    tag_t tolerance_feature,
    logical * is_retained
)
```

tag_t	**tolerance_feature**	Input	Tag of the tolerance feature to query.
logical *	**is_retained**	Output	Flag indicating whether the specified tolerance feature is currently retained.

UF_GDT_is_single_datum_reference_frame (view source)

Defined in: uf_gdt.h

Overview

Given a datum reference frame, this function determines if the frame is a single datum reference frame.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

gateway

```
int UF_GDT_is_single_datum_reference_frame
(
    tag_t frame,
    logical * is_single
)
```

tag_t	**frame**	Input	Tag of the datum reference frame to query
logical *	**is_single**	Output	True if frame is a single datum ref frame

---

UF_GDT_is_user_defined_keyword (view source)

Defined in: uf_gdt.h

Overview

Check whether or not the keyword specified in the input part is a standard keyword. If the keyword does not exist in the part then an error will be returned.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

gateway

```
int UF_GDT_is_user_defined_keyword
(
    tag_t part_tag,
    UF_GDT_keyword_id_t keyword_id,
    logical * status
)
```

tag_t	**part_tag**	Input	Part to query
UF_GDT_keyword_id_t	**keyword_id**	Input	Keyword to check if it is a user defined keyword
logical *	**status**	Output	TRUE if the keyword is user defined FALSE if the keyword is a standard keyword

---

UF_GDT_pull_tolerance (view source)

Defined in: uf_gdt.h

Overview

Pull a tolerance feature up the assembly. This function assumes the current work part is the part of the assembly and the tolerance feature specified resides in a component of that assembly. A tolerance feature can only be pulled from a component of the current assembly.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_DWG_DISPLAYED (can't create instances on drawings)
UF_GDT_INVALID_OBJECT (object is not a tolerance feature)
UF_GDT_ERR_INVALID_INSTANCE (invalid instance specified)
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_pull_tolerance
(
    tag_t tolerance_feature,
    tag_t instance,
    tag_t * new_feature_tag,
    logical create_instance
)
```

tag_t	**tolerance_feature**	Input	Tag of the tolerance feature in the component.
tag_t	**instance**	Input	Tag of the component where the tolerance feature resides.
tag_t *	**new_feature_tag**	Output	Tag of the pulled feature created at the assembly level.

logical	**create_instance**	Input	If TRUE, an instance of the tolerance feature will be created at the assembly level.

UF_GDT_remove_leader (view source)

Defined in: uf_gdt.h

Overview

The following function will remove an instance's nth leader

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NO_INDEXED_LEADER (Instance's nth leader non-existent)
UF_GDT_ERR_NO_REMOVE_ONLY_LDR (Can't remove instance's only leader)
UF_err_program_not_initialized

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_remove_leader
(
    tag_t instance,
    int nth
)
```

tag_t	**instance**	Input	Tag of instance to remove a leader from
int	**nth**	Input	Index of the leader to be removed

UF_GDT_reset_callout_rules (view source)

Defined in: uf_gdt.h

Overview

Reset the specified part's Geometric Tolerancing Callout Rules Table
using the appropriate callout rules file.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_reset_callout_rules
(
    tag_t part
)
```

tag_t	**part**	Input	Part to update rules of

UF_GDT_set_appended_text (view source)

Defined in: uf_gdt.h

Overview

Modify the appended text at the specified location for the input feature control frame or dimension.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_appended_text
(
    tag_t fcf_or_dim,
    UF_GDT_text_location_t location,
    int num_lines,
    char * * appended_text
)
```

tag_t	**fcf_or_dim**	Input	FCF of dimension to query
UF_GDT_text_location_t	**location**	Input	Appended text location to modify
int	**num_lines**	Input	Number of appended text lines
char * *	**appended_text**	Input	Appended text at location requested

UF_GDT_set_characteristic (view source)

Defined in: uf_gdt.h

Overview

Given a feature control frame, this function changes the tolerance type to the specified characteristic.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_characteristic
(
    tag_t fcf,
    UF_GDT_characteristic_t * characteristic
)
```

tag_t	**fcf**	Input	Feature control frame to set
UF_GDT_characteristic_t *	**characteristic**	Input	Feature control frame type

UF_GDT_set_composite_drf (view source)

Defined in: uf_gdt.h

Overview

This function sets the composite priority for a composite feature control frame.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_composite_drf
(
    tag_t fcf,
    UF_GDT_precedence_t priority
)
```

tag_t	**fcf**	Input	Feature control frame identifier
UF_GDT_precedence_t	**priority**	Input	Precedence of lower datum frame

UF_GDT_set_datum_identifier (view source)

Defined in: uf_gdt.h

Overview

Edits the parameters of a datum identifier. If the user specifies a "NULL"
for the datum data, the current datum information will be removed from the
tolerance feature. In this case a size tolerance or a feature control frame
must still be part of the tolerance feature.

The input tag object could be a tolerance feature or a datum object.
If it is a tolerance feature, it works the same way as before except that if the
tolerance feature refers to more than one datum, an error called
UF_GDT_ERR_FEAT_NOT_VALID
will returned. If it is a datum object, it will set the information about this datum.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_FEAT_NOT_VALID
UF_GDT_ERR_NULL_LABEL
UF_GDT_ERR_NOT_BASE

Environment

Internal and External

History

Original release was in 14.0. Modified in v18.0

Required License(s)

geometric_tol

```
int UF_GDT_set_datum_identifier
(
    tag_t base,
    UF_GDT_datum_identifier_t * datum_data
)
```

tag_t	**base**	Input	Either a tolerance feature or a datum object, if it is a tolerance feature and the tolerance feature refers to more than one datums, an error will returned.
UF_GDT_datum_identifier_t *	**datum_data**	Input	Information to be edited

UF_GDT_set_datum_keywords (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to set keywords on datum objects.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized
UF_GDT_ERR_INVALID_KEYWORD
UF_GDT_ERR_NOT_DATUM

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_datum_keywords
(
    tag_t datum,
    int num_keywords,
    UF_GDT_keyword_p_t keywords
)
```

tag_t	**datum**	Input	The datum object
int	**num_keywords**	Input	The number of keywords
UF_GDT_keyword_p_t	**keywords**	Input	The keywords list

UF_GDT_set_datum_label (view source)

Defined in: uf_gdt.h

Overview

This function changes the label for a datum feature. The label must be unique in the context of the part file. Duplicate labels are not permitted.
The characters defining the label must be in the set of valid characters as defined in the customer defaults file.

The input tag object could be a tolerance feature or a datum object.
If it is a tolerance feature, it works the same way as before except that if the tolerance feature refers to more than one datum, an error called UF_GDT_ERR_FEAT_NOT_VALID will returned. If it is a datum object, it will set the information about this datum.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_BASE
UF_GDT_ERR_FEAT_NOT_VALID
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 14.0, modified in release v18.0

Required License(s)

geometric_tol

```
int UF_GDT_set_datum_label
(
    tag_t base,
    const char * datum_label
)
```

tag_t	**base**	Input	Tolerance datum feature or datum object to set
const char *	**datum_label**	Input	New label for datum feature

UF_GDT_set_datum_multiple (view source)

Defined in: uf_gdt.h

Overview

This function edits the parameters of a multiple datum.

Note: The input tag could be a multiple datum feature or a multiple datum object.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_BASE
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 14.0, modified in release v18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_datum_multiple
(
    tag_t multiple_datum,
    UF_GDT_multiple_datum_t * datum_data
)
```

tag_t	**multiple_datum**	Input	Multiple datum to set
UF_GDT_multiple_datum_t *	**datum_data**	Input	Information to be edited

UF_GDT_set_datum_references (view source)

Defined in: uf_gdt.h

Overview

Sets the datum references of a given frame for the specified feature control frame.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_datum_references
(
    tag_t fcf,
    int nth_frame,
    int num_frames,
    UF_GDT_datum_reference_t * datum_reference
)
```

tag_t	**fcf**	Input	Feature control frame to edit
int	**nth_frame**	Input	FCF data frame to set
int	**num_frames**	Input	Number of datum references in array
UF_GDT_datum_reference_t *	**datum_reference**	Input	Primary datum reference of feature control frame

UF_GDT_set_depth_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function allows for the resetting of the depth tolerance parameters.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_GDT_ERR_TOL_VALIDATION_FAILED
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_depth_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_depth_tolerance_t * depth_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to set
UF_GDT_depth_tolerance_t *	**depth_data**	Input	Data to be applied to tolerance feature.

UF_GDT_set_description (view source)

Defined in: uf_gdt.h

Overview

Allows the user to change the name of a datum feature or tolerance feature and assign a new description. If a field of the UF_GDT_description_t structure is null that field is not updated.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_description
(
    tag_t tolerance_feature,
    UF_GDT_description_t * info
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to inquire
UF_GDT_description_t *	**info**	Input	Design information to associated with the feature

---

UF_GDT_set_directed_dimension (view source)

Defined in: uf_gdt.h

Overview

Put the information for a directed dimension tolerance on a tolerance feature.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_directed_dimension
(
    tag_t tolerance_feature,
    UF_GDT_directed_dimension_p_t data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
UF_GDT_directed_dimension_p_t	**data**	Input	Data for the tolerance

UF_GDT_set_drf (view source)

Defined in: uf_gdt.h

Overview

This routine sets the data of a datum reference frame from the given data structure.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_drf
(
    tag_t drf,
    UF_GDT_drf_data_p_t drf_data
)
```

tag_t	**drf**	Input	Datum reference frame identifier
UF_GDT_drf_data_p_t	**drf_data**	Input	Data of a datum reference frame

UF_GDT_set_faces (view source)

Defined in: uf_gdt.h

Overview

Allows the calling application to reassociate the set of faces associated with the tolerance feature or datum feature. Some tolerance features return an error if the set of faces is modified (target points, target lines, centerplanes, etc.)

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_faces
(
    tag_t tolerance_feature,
    int num_faces,
    tag_t * faces
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to set
int	**num_faces**	Input	Number of faces in face array
tag_t *	**faces**	Input	Array of faces to associate with the tolerance feature

UF_GDT_set_fcf (view source)

Defined in: uf_gdt.h

Overview

Allows the calling application to set the the type of profile tolerance and the dimensions of the outer and inner portions of the tolerance zone for unilateral and unequally disposed bilateral tolerances.
If the profile type is not UF_GDT_PROFILE_UNEQ_BILATERAL, the outside parameter is ignored.

An error of UF_GDT_ERR_NOT_PROFILE_FCF is returned if the input feature control frame is not a profile of line or profile of surface callout.

An error of UF_GDT_ERR_INVALID_PROFILE_TOL is returned if the input profile type is UF_GDT_PROFILE_UNEQ_BILATERAL and outside is not in the range (0, total_tolerance).

Environment

Internal and External

History

This function was originally released in V15.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_fcf
(
    tag_t fcf,
    UF_GDT_fcf_t * fcf_data
)
```

tag_t	**fcf**	Input	Feature control frame to set
UF_GDT_fcf_t *	**fcf_data**	Input	Feature control frame data to set the tolerance feature with.

UF_GDT_set_fcf_drf (view source)

Defined in: uf_gdt.h

Overview

This function sets the datum reference frame for a feature control frame.

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_fcf_drf
(
    tag_t fcf,
    tag_t drf
)
```

tag_t	**fcf**	Input	Feature control frame identifier
tag_t	**drf**	Input	Datum reference frame to use with FCF

UF_GDT_set_geometric_definition (view source)

Defined in: uf_gdt.h

Overview

Create the information for a geometric definition on a tolerance feature.
If the geometric definition is that of a region the input structure,
data, contains a trimmed sheet and a thickened sheet. The thickened

sheet must be a thickening of the input trimmed sheet. Creating associative trimmed sheets and thickened sheets for regions can be done by using the following routines: UF_WAVE_create_linked_face, UF_MODL_create_sew, UF_MODL_create_trimmed_sheet, UF_MODL_ask_feat_body, and UF_MODL_create_thickened_sheet.

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_geometric_definition
(
    tag_t tolerance_feature,
    UF_GDT_geometric_definition_p_t data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature tag
UF_GDT_geometric_definition_p_t	**data**	Input	Data for the definition

UF_GDT_set_index_display (view source)

Defined in: uf_gdt.h

Overview

Sets the index display method.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_index_display
(
    UF_GDT_index_display_type_t index_display
)
```

UF_GDT_index_display_type_t	**index_display**	Input	Desired index display method.

UF_GDT_set_instance_display_information (view source)

Defined in: uf_gdt.h

Overview

Sets the show feature info status for the work part to be true or false.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_instance_display_information
(
    logical preference
)
```

logical	**preference**	Input	Display preference setting

UF_GDT_set_keyword_text (view source)

Defined in: uf_gdt.h

Overview

Modify the text associated with the keyword specified in the input part.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_keyword_text
(
    tag_t part_tag,
    const char * keyword_name,
    const char * keyword_text
)
```

tag_t	**part_tag**	Input	Part to query
const char *	**keyword_name**	Input	Keyword whose text to modify
const char *	**keyword_text**	Input	New text for keyword

UF_GDT_set_limits_and_fits_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function allows for
the resetting of the limits and fits tolerance parameters.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized
UF_GDT_ERR_NOT_VALID_FEAT_TYPE_FOR_LIM_FITS
UF_GDT_ERR_NOT_VALID_FEAT_TYPE_FOR_LIM_FITS_ENGLISH
UF_GDT_ERR_NOT_VALID_DATA_FOR_PART_UNIT
UF_GDT_ERR_INVALID_LIM_FITS_DATA
UF_GDT_ERR_INCH_PART_WITH_ISO_STANDARD
UF_GDT_ERR_UNVAILABLE_DATA_FOR_CURRENT_SETTING

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_limits_and_fits_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_limits_and_fits_data_p_t data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to set
UF_GDT_limits_and_fits_data_p_t	**data**	Input	Data to be applied to tolerance feature.

UF_GDT_set_load_component_flag (view source)

Defined in: uf_gdt.h

Overview

Set the "auto load component" flag for the current NX work part. This flag
controls whether component parts should be automatically loaded when entering
the tolerancing module.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_load_component_flag
(
    logical flag
)
```

logical	**flag**	Input	If set to TRUE then component parts will be automatically loaded when entering the tolerancing module.

UF_GDT_set_modl_data (view source)

Defined in: uf_gdt.h

Overview

Modify the reference to the modeling feature data.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized
UF_GDT_ERR_MODL_FEATURES_NOT_FULLY_LOADED
UF_GDT_ERR_INVALID_MODL_FEATURE_TYPE
UF_GDT_ERR_MODL_TOL_FEAT_FEATURE_TYPE_MISMATCH

Environment

Internal and External

History

Original release NX2.0.0

Required License(s)

geometric_tol

```
int UF_GDT_set_modl_data
(
    tag_t tolerance_feature,
    int num_modl_sets,
    UF_GDT_modl_data_p_t * modl_sets
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to set
int	**num_modl_sets**	Input	Number of model sets
UF_GDT_modl_data_p_t *	**modl_sets**	Input	Modeling feature data sets for tolerance creation

---

UF_GDT_set_non_feature_edge_selection (view source)

Defined in: uf_gdt.h

Overview

Sets the selection of non-feature edges for the GD&T session to be true or false.

Environment

Internal and external

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_non_feature_edge_selection
(
    UF_GDT_edge_select_type_t allow_non_feature_edges
)
```

UF_GDT_edge_select_type_t	**allow_non_feature_edges**	Input	Should selection of non-feature edges be allowed when creating display instances?

UF_GDT_set_profile_tol_data (view source)

Defined in: uf_gdt.h

Overview

Sets the profile type and outside zone for a profile of line or profile of surface tolerance. The outside zone is only used when the profile type is set to UF_GDT_PROFILE_UNEQ_BILATERAL.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_INVALID_OBJECT (object is not an fcf)
UF_GDT_NOT_PROFILE_FCF (fcf does not contain a profile of
surface or profile of line
characteristic)
UF_GDT_INVALID_PROFILE_ZON (if profile type = UF_GDT_UNEQ_BILATERAL
and outside does not follow the rule
0.0 < outside < total_tolerance
- ASME Y14.5M 1994)
UF_GDT_INVALID_PROFILE_TOL (if profile type is not supported for
current standard or feature)
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_profile_tol_data
(
    tag_t fcf,
    UF_GDT_profile_type_t profile_type,
    UF_GDT_tolerance_value_p_t outside
)
```

tag_t	**fcf**	Input	profile feature control frame to modify
UF_GDT_profile_type_t	**profile_type**	Input	profile type
UF_GDT_tolerance_value_p_t	**outside**	Input	outside tolerance zone when type is UF_GDT_UNEQ_BILATERAL

UF_GDT_set_region_parameters (view source)

Defined in: uf_gdt.h

Overview

This function will set the following parameters for regions of tolerance features: region thickness, whether or not the region is to be crosshatched, the crosshatch angle, and the crosshatch distance

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NON_POSITIVE_PARAMETERS

Environment

Internal and External

History

Originally released in V16.0

Required License(s)

geometric_tol

```
int UF_GDT_set_region_parameters
(
    double region_thickness,
    logical crosshatch_on,
    double crosshatch_angle,
    double crosshatch_distance
)
```

double	**region_thickness**	Input	The region thickness which is to be used when regions for GDTs are created
logical	**crosshatch_on**	Input	TRUE if the region of a GDT is to be crosshatched.

double	**crosshatch_angle**	Input	The angle to be used when crosshatching a region
double	**crosshatch_distance**	Input	The distance between hatch lines to be used when crosshatching a region

UF_GDT_set_size_data (view source)

Defined in: uf_gdt.h

Overview

This routine sets the size data based on the limits and fits information.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized

Environment

Internal and External

History

Original release was in 18.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_size_data
(
    tag_t tolerance_feature,
    UF_GDT_limits_and_fits_tolerance_p_t lim_fits
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_limits_and_fits_tolerance_p_t	**lim_fits**	Input / Output to UF_*free*	Information on the limits and fits tolerance. The data allocated for this structure must be freed by calling UF_GDT_free.

UF_GDT_set_size_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function allows for the resetting of the size tolerance parameters.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_size_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_size_tolerance_t * size_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to set
UF_GDT_size_tolerance_t *	**size_data**	Input	Data to be applied to tolerance feature.

UF_GDT_set_target_area_xhatch (view source)

Defined in: uf_gdt.h

Overview

This function sets the crosshatching distance and angle parameters for the given datum target area instance.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_err_bad_parameter_number_1
UF_err_bad_parameter_number_2
UF_err_bad_parameter_number_3
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_target_area_xhatch
(
    tag_t target_inst,
    double distance,
    double angle
)
```

tag_t	**target_inst**	Input	Tag of target instance
double	**distance**	Input	Crosshatching distance
double	**angle**	Input	Crosshatching angle

UF_GDT_set_thread_tolerance_parms (view source)

Defined in: uf_gdt.h

Overview

This function allows the calling program to set the thread tolerance information for a given tolerance feature.

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_TOL_FEAT
UF_err_program_not_initialized
UF_GDT_ERR_INVALID_THREAD_TOL
UF_GDT_ERR_TOL_VALIDATION_FAILED
UF_GDT_ERR_INVALID_FEAT_TYPE_FOR_THREAD_TOL

Environment

Internal and External

History

Original release was in 18.0.5, NX1.0.1 and NX2.0.0

Required License(s)

geometric_tol

```
int UF_GDT_set_thread_tolerance_parms
(
    tag_t tolerance_feature,
    UF_GDT_thread_tolerance_p_t data
)
```

tag_t	**tolerance_feature**	Input	Tolerance feature to inquire
UF_GDT_thread_tolerance_p_t	**data**	Input	Thread tolerance parms to apply to the tolerance feature.

UF_GDT_set_tolerance_zones (view source)

Defined in: uf_gdt.h

Overview

This function edits the tolerance zone of a given feature control frame.

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_ERR_NOT_FCF
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_tolerance_zones
(
    tag_t fcf,
    int num_zones,
    UF_GDT_tolerance_zone_p_t zones,
    UF_GDT_modifier_data_p_t mod_data
)
```

tag_t	**fcf**	Input	FCF to set
int	**num_zones**	Input	Number of zones to set
UF_GDT_tolerance_zone_p_t	**zones**	Input	Tolerance zone data to set
UF_GDT_modifier_data_p_t	**mod_data**	Input	Modifier data to set

UF_GDT_set_tolerancing_standard (view source)

Defined in: uf_gdt.h

Overview

This function allows the user to change the value of the tolerancing standard used by the GDT Module. The standard can only be set if there are no tolerance features created on the part.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_tolerancing_standard
(
    UF_GDT_standard_t standard
)
```

UF_GDT_standard_t	**standard**	Input	Desired engineering standard used for tolerancing

UF_GDT_set_unit_basis (view source)

Defined in: uf_gdt.h

Overview

Sets unit basis data for the given fcf

Return

Return Codes -
UF_GDT_NO_ERRORS
UF_GDT_null_object
UF_GDT_invalid_object
UF_err_program_not_initialized

Environment

Internal and External

Required License(s)

geometric_tol

```
int UF_GDT_set_unit_basis
(
    tag_t fcf,
    UF_GDT_modifier_types_t modifier,
    UF_GDT_unit_basis_p_t data
)
```

tag_t	**fcf**	Input	Tag for FCF to set
UF_GDT_modifier_types_t	**modifier**	Input	Characteristic type
UF_GDT_unit_basis_p_t	**data**	Input	Unit basis data to set

UF_GDT_set_wall_thickness_parms (view source)

Defined in: uf_gdt.h

Overview

Given a tolerance feature or datum feature, this function allows for the resetting of the wall thickness tolerance parameters.

Environment

Internal and External

History

Original release was in 17.0.

Required License(s)

geometric_tol

```
int UF_GDT_set_wall_thickness_parms
(
    tag_t tolerance_feature,
    UF_GDT_size_tolerance_t * thickness_data
)
```

tag_t	**tolerance_feature**	Input	Tolerance Feature to set
UF_GDT_size_tolerance_t *	**thickness_data**	Input	New wall thickness data.

UF_GDT_update_features (view source)

Defined in: uf_gdt.h

Overview

This routine updates the given list of tolerance features. If update is currently in progress, or any of the entites in the list are not tolerance features, no update will be done.

Return

0 = Success
error code otherwise

Environment

Internal and External

History

Created in v16.0.4.2

Required License(s)

geometric_tol

```
int UF_GDT_update_features
(
    int num_features,
    tag_p_t features
)
```

int	**num_features**	Input	Number of tolerance features to update
tag_p_t	**features**	Input	List of tolerance feature tags to update

---

UF_GDT_upgrade_legacy_feature (view source)

Defined in: uf_gdt.h

Overview

Allows an "old" style Geometric Tolerancing symbol or feature to be upgraded to be included in the new tolerancing module. If the "old" symbol or feature is inconsistent with the current model, the feature may be created in a retained state. Retained annotations cannot be upgraded.

Environment

Internal and External

History

Original release was in 14.0.

Required License(s)

geometric_tol

```
int UF_GDT_upgrade_legacy_feature
(
    int num_annot,
    tag_t * annotation_features,
    int num_faces,
    tag_t * face_list,
    UF_GDT_feature_type_t feature,
    double origin [ 3 ] ,
    UF_GDT_description_t * descript,
    tag_t * tolerance_feature
)
```

int	**num_annot**	Input	Number of annotations to be upgraded
tag_t *	**annotation_features**	Input	Identifier of the set of old dimensions and symbols that get converted
int	**num_faces**	Input	Number of identifiers in face list array
tag_t *	**face_list**	Input	Array of identifiers that define the face_list of the tolerance feature
UF_GDT_feature_type_t	**feature**	Input	Feature type of the tolerance feature
double	**origin [3]**	Input	Origin of tolerance feature instance
UF_GDT_description_t *	**descript**	Input	Description of the noew tolerance feature
tag_t *	**tolerance_feature**	Output	Identifier of upgraded feature