The graPHIGS Programming Interface: ISO PHIGS Quick Reference

SC28-2705-02

The graPHIGS Programming Interface: ISO PHIGS Quick Reference

SC28-2705-02

Note
Before using this information and the product it supports, read the information in “Notices,” on page 49.

Third Edition (April 1994)

This edition applies to the AIXwindows Environment/6000 (1.2.5) AIXwindows/3D feature, Program Number 5601-257, and to all subsequent releases of this product until otherwise indicated in new editions.

Contents

About This Book . . . v
Who Should Use This Book . . . v
Highlighting . . . v
ISO 9000 . . . v
Related Publications . . . v

Chapter 1. Abbreviations Quick List . . . 1

Chapter 2. Listing by Function of Subroutines for C Binding . . . 3
Control Subroutines . . . 3
Output Primitives . . . 3
Attribute Specification . . . 4
Miscellaneous Structure Element Subroutines . . . 4
Structure Operation Subroutines . . . 4
Workstation Table Settings . . . 5
Structure Display Subroutines . . . 5
Structure Archiving Subroutines . . . 5
Transformation Subroutines . . . 6
Input Subroutines . . . 6
Utility Subroutines . . . 7
Error Control Subroutines . . . 8
Special Interface Subroutines . . . 8
Inquire Subroutines . . . 8

Chapter 3. Alphabetical Listing of Subroutines for C Binding . . . 13

Chapter 4. Listing by Function of Subroutines for FORTRAN Binding . . . 23
Control Subroutines . . . 23
Output Primitives . . . 23
Attribute Specification . . . 24
Miscellaneous Structure Element Subroutines . . . 25
Structure Operation Subroutines . . . 25
Workstation Table Settings . . . 25
Structure Display Subroutines . . . 26
Structure Archiving Subroutines . . . 26
Transformation Subroutines . . . 26
Input Subroutines . . . 26
Utility Subroutines . . . 27
Error Control Subroutines . . . 28
Special Interface Subroutines . . . 28
Inquire Subroutines . . . 28

Chapter 5. Alphabetical Listing of Subroutines for FORTRAN Binding . . . 33

Chapter 6. ISO PHIGS Enumerated Data Types . . . 41

Appendix. Notices . . . 49
Trademarks . . . 50

About This Book

This book provides a quick reference for the graPHIGS API. It is intended as a supplement to the *The graPHIGS Programming Interface: ISO PHIGS Subroutine Reference*, in which the subroutines are described in detail. To help you find information quickly in the reference book, each listed subroutine includes a page reference. This book contains both a reference-by-function listing as well as an alphabetical-by-subroutine listing for both the C and FORTRAN bindings.

Who Should Use This Book

This book is intended for application programmers who want a quick reference for the graPHIGS API.

Highlighting

The following highlighting conventions are used in this book:

Bold	Identifies commands, subroutines, keywords, files, structures, directories, and other items whose names are predefined by the system. Also identifies graphical objects such as buttons, labels, and icons that the user selects.
Italics	Identifies parameters whose actual names or values are to be supplied by the user.
`Monospace`	Identifies examples of specific data values, examples of text similar to what you might see displayed, examples of portions of program code similar to what you might write as a programmer, messages from the system, or information you should actually type.

ISO 9000

ISO 9000 registered quality systems were used in the development and manufacturing of this product.

Related Publications

The following books contain information on graPHIGS API products:

- *The graPHIGS Programming Interface: Customization and Problem Diagnosis*
- *The graPHIGS Programming Interface: ISO PHIGS Subroutine Reference*
- *The graPHIGS Programming Interface: Getting Started*
- *The graPHIGS Programming Interface: Messages and Codes*
- *The graPHIGS Programming Interface: Quick Reference*
- *The graPHIGS Programming Interface: Subroutine Reference*
- *The graPHIGS Programming Interface: Technical Reference*
- *The graPHIGS Programming Interface: Understanding Concepts*

Chapter 1. Abbreviations Quick List

The following abbreviations are used frequently throughout the graPHIGS API library and the ISO PHIGS Subroutine Reference.

ARCL	Archive Closed
AROP	Archive Open
ASAP	As Soon As Possible
ASF	Attribute Source Flag
ASTI	At Some Time
BNIG	Before Next Interaction Globally
BNIL	Before Next Interaction Locally
CIELUV	CIELUV color model system
CMY	Cyan-Magenta-Yellow color model
CSID	Character Set Identifier
CSS	Centralized Structure Store
EDF	External Defaults File
GDP	Generalized Drawing Primitive
GSE	Generalized Structure Element
HLHSR	Hidden Line Hidden Surface Removal
HSV	Hue-Saturation-Value color model
NROP	Non-Retained Structure Open
PDT	graPHIGS API Description Table
PET	Prompt and Echo Type
PHCL	graPHIGS Closed
PHOP	graPHIGS Open
PSL	graPHIGS API State List
RGB	Red-Green-Blue color model
STCL	Structure Closed
STOP	Structure Open
USL	Utility State List
WAIT	When Application Requests It
WDT	Workstation Description Table
WSCL	Workstation Closed
WSID	Workstation Identifier
WSL	Workstation State List
WSOP	Workstation Open
WSTYPE	Workstation Type

The following abbreviations for coordinate spaces are used:

MC	Modeling Coordinates
WC	World Coordinates

VC	Viewing Coordinates
NPC	Normalized Projection Coordinates
DC	Device Coordinates

Chapter 2. Listing by Function of Subroutines for C Binding

This section contains the following tables:

- Control Subroutines
- Output Primitives
- Attribute Specification
- Miscellaneous Structure Element Subroutines
- Structure Operation Subroutines
- Workstation Table Settings
- Structure Display Subroutines
- Structure Archiving Subroutines
- Transformation Subroutines
- Input Subroutines
- Utility Subroutines
- Error Control Subroutines
- Special Interface Subroutines
- Inquire Subroutines

Control Subroutines

Close PHIGS	**pclose_phigs ();**
Close Workstation	**pclose_ws (ws_id);**
Message	**pmessage (ws_id, message);**
Open PHIGS	**popen_phigs (err_file, mem_units);**
Open Workstation	**popen_ws (ws_id, conn_id, ws_type);**
Redraw All Structures	**predraw_all_structs (ws_id, ctrl_flag);**
Set Display Update State	**pset_disp_upd_st (ws_id, def_mode, mod_mode);**
Update Workstation	**pupd_ws (ws_id, regen_flag);**

Output Primitives

Annotation Text Relative	**panno_text_rel (ref_pt, offset, char_string);**
Annotation Text Relative 3	**panno_text_rel3 (ref_pt, offset, char_string);**
Cell Array	**pcell_array (rect, colr_array);**
Cell Array 3	**pcell_array3 (paral, colr_array);**
Fill Area	**pfill_area (point_list);**
Fill Area 3	**pfill_area3 (point_list);**
Fill Area Set	**pfill_area_set (point_list);**
Fill Area Set 3	**pfill_area_set3 (point_list_list);**
Generalized Drawing Primitive	**pgdp (point_list, gdp_id, gdp_data);**
Generalized Drawing Primitive 3	**pgdp3 (point_list, gdp3_id, gdp_data);**
Polyline	**ppolyline (point_list);**
Polyline 3	**ppolyline3 (point_list);**
Polymarker	**ppolymarker (point_list);**
Polymarker 3	**ppolymarker3 (point_list);**
Text	**ptext (text_pos, char_string);**
Text 3	**ptext3 (text_pos, text_dir, char_string);**

Attribute Specification

Add Names to Set	**padd_names_set (names);**
Remove Names from Set	**premove_names_set (names);**
Set Annotation Style	**pset_anno_style (anno_style);**
Set Annotation Text Alignment	**pset_anno_align (text_align);**
Set Annotation Text Character Height	**pset_anno_char_ht (char_ht);**
Set Annotation Text Character Up Vector	**pset_anno_char_up_vec (char_up_vec);**
Set Annotation Text Path	**pset_anno_path (text_path);**
Set Character Expansion Factor	**pset_char_expan (char_expan);**
Set Character Height	**pset_char_ht (char_ht);**
Set Character Spacing	**pset_char_space (char_space);**
Set Character Up Vector	**pset_char_up_vec (char_up_vec);**
Set Edge Color Index	**pset_edge_colr_ind (edge_colr_ind);**
Set Edge Flag	**pset_edge_flag (edge_flag);**
Set Edge Index	**pset_edge_ind (edge_ind);**
Set Edgetype	**pset_edgetype (edgetype);**
Set Edgewidth Scale Factor	**pset_edgewidth (edgewidth);**
Set HLHSR Identifier	**pset_hlhsr_id (hlhsr_id);**
Set Individual ASF	**pset_indiv_asf (asf_id, asf_source);**
Set Interior Color Index	**pset_int_colr_ind (int_colr_ind);**
Set Interior Index	**pset_int_ind (int_ind);**
Set Interior Style	**pset_int_style (int_style);**
Set Interior Style Index	**pset_int_style_ind (int_style_ind);**
Set Linetype	**pset_linetype (linetype);**
Set Linewidth Scale Factor	**pset_linewidth (linewidth);**
Set Marker Size Scale Factor	**pset_marker_size (marker_size);**
Set Marker Type	**pset_marker_type (marker_type);**
Set Pattern Reference Point	**pset_pat_ref_point (pat_ref_point);**
Set Pattern Reference Point and Vectors	**pset_pat_ref_point_vecs (pat_ref_point, pat_ref_vec);**
Set Pattern Size	**pset_pat_size (pat_size);**
Set Pick Identifier	**pset_pick_id (pick_id);**
Set Polyline Color Index	**pset_line_colr_ind (line_colr_ind);**
Set Polyline Index	**pset_line_ind (line_ind);**
Set Polymarker Color Index	**pset_marker_colr_ind (marker_colr_ind);**
Set Polymarker Index	**pset_marker_ind (marker_ind);**
Set Text Alignment	**pset_text_align (text_align);**
Set Text Color Index	**pset_text_colr_ind (text_colr_ind);**
Set Text Font	**pset_text_font (font);**
Set Text Index	**pset_text_ind (text_ind);**
Set Text Path	**pset_text_path (text_path);**
Set Text Precision	**pset_text_prec (prec);**
Set View Index	**pset_view_ind (view_ind);**

Miscellaneous Structure Element Subroutines

Application Data	**pappl_data (data);**
Execute Structure	**pexec_struct (struct_id);**
Generalized Structure Element	**pgse (id, gse_data);**

Structure Operation Subroutines

Change Structure Identifier	**pchange_struct_id (orig_struct_id, result_struct_id);**

Change Structure Identifier and References	**pchange_struct_id_refs (orig_struct_id, result_struct_id);**
Change Structure References	**pchange_struct_refs (orig_struct_id, result_struct_id);**
Close Structure	**pclose_struct ();**
Copy All Elements from Structure	**pcopy_all_elems_struct (struct_id);**
Delete All Structures	**pdel_all_structs ();**
Delete Element	**pdel_elem ();**
Delete Element Range	**pdel_elem_range (elem_ptr1_value, elem_ptr2_value);**
Delete Elements Between Labels	**pdel_elems_labels (label1_id, label2_id);**
Delete Structure	**pdel_struct (struct_id);**
Delete Structure Network	**pdel_struct_net (struct_id, ref_flag);**
Empty Structure	**pempty_struct (struct_id);**
Label	**plabel (label_id);**
Offset Element Pointer	**poffset_elem_ptr (elem_ptr_offset);**
Open Structure	**popen_struct (struct_id);**
Set Edit Mode	**pset_edit_mode (edit_mode);**
Set Element Pointer	**pset_elem_ptr (elem_ptr_value);**
Set Element Pointer at Label	**pset_elem_ptr_label (label_id);**

Workstation Table Settings

Set Color Model	**pset_colr_model (ws_id, colr_model);**
Set Color Representation	**pset_colr_rep (ws_id, colr_ind, colr_rep);**
Set Edge Representation	**pset_edge_rep (ws_id, edge_ind, edge_bundle);**
Set Highlighting Filter	**pset_highl_filter (ws_id, filter);**
Set HLHSR Mode	**pset_hlhsr_mode (ws_id, hlhsr_mode);**
Set Interior Representation	**pset_int_rep (ws_id, int_ind, int_bundle);**
Set Invisibility Filter	**pset_invis_filter (ws_id, filter);**
Set Pattern Representation	**pset_pat_rep (ws_id, pat_ind, pat_bundle);**
Set Polyline Representation	**pset_line_rep (ws_id, line_ind, line_bundle);**
Set Polymarker Representation	**pset_marker_rep (ws_id, marker_ind, marker_bundle);**
Set Text Representation	**pset_text_rep (ws_id, text_ind, text_bundle);**
Set View Representation	**pset_view_rep (ws_id, view_ind, view_rep);**
Set View Representation 3	**pset_view_rep3 (ws_id, view_ind, view_rep);**
Set View Transformation Input Priority	**pset_view_tran_in_pri (ws_id, view_ind, ref_view_ind, rel_pri);**

Structure Display Subroutines

Post Structure	**ppost_struct (ws_id, struct_id, pri);**
Unpost All Structures	**punpost_all_structs (ws_id);**
Unpost Structure	**punpost_struct (ws_id, struct_id);**

Structure Archiving Subroutines

Archive All Structures	**par_all_structs (archive_id);**
Archive Structure Networks	**par_struct_nets (archive_id, struct_ids);**
Archive Structures	**par_structs (archive_id, struct_ids);**
Close Archive File	**pclose_ar_file (archive_id);**
Delete All Structures from Archive	**pdel_all_structs_ar (archive_id);**
Delete Structure Networks from Archive	**pdel_struct_nets_ar (archive_id, struct_ids);**
Delete Structures from Archive	**pdel_structs_ar (archive_id, struct_ids);**
Open Archive File	**popen_ar_file (archive_id, archive_file);**
Retrieve All Structures	**pret_all_structs (archive_id);**

Retrieve Paths to Ancestors	**pret_paths_ances** (ar_id, struct_id, order, depth, store, paths);
Retrieve Paths to Descendants	**pret_paths_descs** (ar_id, struct_id, order, depth, store, paths);
Retrieve Structure Identifiers	**pret_struct_id** (archive_id, num_elems_appl_list, start_ind, ids, num_elems_impl_list);
Retrieve Structure Networks	**pret_struct_nets** (archive_id, struct_ids);
Retrieve Structures	**pret_structs** (archive_id, struct_ids);
Set Conflict Resolution	**pset_conf_res** (archive_res, retrieval_res);

Transformation Subroutines

Restore Modeling Clipping Volume	**prestore_model_clip_vol**();
Set Global Transformation	**pset_global_tran** (global_tran);
Set Global Transformation 3	**pset_global_tran3** (global_tran);
Set Local Transformation	**pset_local_tran** (local_tran, compose_type);
Set Local Transformation 3	**pset_local_tran3** (local_tran, compose_type);
Set Modeling Clipping Indicator	**pset_model_clip_ind** (clip_ind);
Set Modeling Clipping Volume	**pset_model_clip_vol** (op, half_spaces);
Set Modeling Clipping Volume 3	**pset_model_clip_vol3** (op, half_spaces);
Set Workstation Viewport	**pset_ws_vp** (ws_id, ws_vp_limits);
Set Workstation Viewport 3	**pset_ws_vp3** (ws_id, ws_vp_limits);
Set Workstation Window	**pset_ws_win** (ws_id, ws_win_limits);
Set Workstation Window 3	**pset_ws_win3** (ws_id, ws_win_limits);

Input Subroutines

Await Event	**pawait_event** (timeout, ws_id, dev_class, in_num);
Flush Device Events	**pflush_events** (ws_id, dev_class, in_num);
Get Choice	**pget_choice** (in_status, choice);
Get Locator	**pget_loc** (view_ind, loc_pos);
Get Locator 3	**pget_loc3** (view_ind, loc_pos);
Get Pick	**pget_pick** (depth, in_status, pick);
Get String	**pget_string** (string);
Get Stroke	**pget_stroke** (view_ind, stroke);
Get Stroke 3	**pget_stroke3** (view_ind, stroke);
Get Valuator	**pget_val** (value);
Initialize Choice	**pinit_choice** (ws_id, choice_num, init_status, init_choice, pet, echo_area, choice_data);
Initialize Choice 3	**pinit_choice3** (ws_id, choice_num, init_status, init_choice, pet, echo_vol, choice_data);
Initialize Locator	**pinit_loc** (ws_id, loc_num, init_view_ind, init_loc_pos, pet, echo_area, loc_data);
Initialize Locator 3	**pinit_loc3** (ws_id, loc_num, init_view_ind, init_loc_pos, pet, echo_vol, loc_data);
Initialize Pick	**pinit_pick** (ws_id, pick_num, init_status, init_pick, pet, echo_area, pick_data, order);
Initialize Pick 3	**pinit_pick3** (ws_id, pick_num, init_status, init_pick, pet, echo_vol, pick_data, order);
Initialize String	**pinit_string** (ws_id, string_num, init_string, pet, echo_area, string_data);
Initialize String 3	**pinit_string3** (ws_id, string_num, init_string, pet, echo_vol, string_data);

Initialize Stroke	**pinit_stroke** (ws_id, stroke_num, init_view_ind, init_stroke, pet, echo_area, stroke_data);
Initialize Stroke 3	**pinit_stroke3** (ws_id, stroke_num, init_view_ind, init_stroke, pet, echo_vol, stroke_data);
Initialize Valuator	**pinit_val** (ws_id, val_num, init_value, pet, echo_area, val_data);
Initialize Valuator 3	**pinit_val3** (ws_id, val_num, init_value, pet, echo_vol, val_data);
Request Choice	**preq_choice** (ws_id, choice_num, in_status, choice);
Request Locator	**preq_loc** (ws_id, loc_num, in_status, view_ind, loc_pos);
Request Locator 3	**preq_loc3** (ws_id, loc_num, in_status, view_ind, loc_pos);
Request Pick	**preq_pick** (ws_id, pick_num, depth, in_status, pick);
Request String	**preq_string** (ws_id, string_num, in_status, string);
Request Stroke	**preq_stroke** (ws_id, stroke_num, in_status, view_ind, stroke);
Request Stroke 3	**preq_stroke3** (ws_id, stroke_num, in_status, view_ind, stroke);
Request Valuator	**preq_val** (ws_id, val_num, in_status, value);
Sample Choice	**psample_choice** (ws_id, choice_num, choice_in_status, choice);
Sample Locator	**psample_loc** (ws_id, loc_num, view_ind, loc_pos);
Sample Locator 3	**psample_loc3** (ws_id, loc_num, view_ind, loc_pos);
Sample Pick	**psample_pick** (ws_id, pick_num, depth, pick_in_status, pick);
Sample String	**psample_string** (ws_id, string_num, string);
Sample Stroke	**psample_stroke** (ws_id, stroke_num, view_ind, stroke);
Sample Stroke 3	**psample_stroke3** (ws_id, stroke_num, view_ind, stroke);
Sample Valuator	**psample_val** (ws_id, val_num, value);
Set Choice Mode	**pset_choice_mode** (ws_id, choice_num, op_mode, echo_switch);
Set Locator Mode	**pset_loc_mode** (ws_id, loc_num, op_mode, echo_switch);
Set Pick Filter	**pset_pick_filter** (ws_id, pick_num, filter);
Set Pick Mode	**pset_pick_mode** (ws_id, pick_num, op_mode, echo_switch);
Set String Mode	**pset_string_mode** (ws_id, string_num, op_mode, echo_switch);
Set Stroke Mode	**pset_stroke_mode** (ws_id, stroke_num, op_mode, echo_switch);
Set Valuator Mode	**pset_valuator_mode** (ws_id, val_num, op_mode, echo_switch);

Utility Subroutines

Build Transformation Matrix	**pbuild_tran_matrix** (point, shift_vec, angle, scale_vec, err_ind, result_tran);
Build Transformation Matrix 3	**pbuild_tran_matrix3** (point, shift_vec, x_angle, y_angle, z_angle, scale_vec, err_ind, result_tran);
Compose Matrix	**pcompose_matrix** (tran_a, tran_b, err_ind, result_tran);
Compose Matrix 3	**pcompose_matrix3** (tran_a, tran_b, err_ind, result_tran);
Compose Transformation Matrix	**pcompose_tran_matrix** (tran, point, shift_vec, angle, scale_vec, err_ind, result_tran);
Compose Transformation Matrix 3	**pcompose_tran_matrix3** (tran, point, shift_vec, x_angle, y_angle, z_angle, scale_vec, err_ind, result_tran);
Create Store	**pcreate_store** (err_ind, store);
Delete Store	**pdel_store** (err_ind, store);
Evaluate View Mapping Matrix	**peval_view_map_matrix** (mapping, err_ind, result_tran);
Evaluate View Mapping Matrix 3	**peval_view_map_matrix3** (mapping, err_ind, result_tran);
Evaluate View Orientation Matrix	**peval_view_ori_matrix** (view_ref_point, view_up_vec, err_ind, result_tran);
Evaluate View Orientation Matrix 3	**peval_view_ori_matrix3** (view_ref_point, view_norm_vec, view_up_vec, err_ind, result_tran);
Rotate	**protate** (angle, err_ind, result_tran);
Rotate X	**protate_x** (angle, err_ind, result_tran);
Rotate Y	**protate_y** (angle, err_ind, result_tran);
Rotate Z	**protate_z** (angle, err_ind, result_tran);
Scale	**pscale** (scale_vec, err_ind, result_tran);

Scale 3	**pscale3** (scale_vec, err_ind, result_tran);
Transform Point	**ptran_point** (point, tran, err_ind, result);
Transform Point 3	**ptran_point3** (point, tran, err_ind, result);
Translate	**ptranslate** (trans_vec, err_ind, result_tran);
Translate 3	**ptranslate3** (trans_vec, err_ind, result_tran);

Error Control Subroutines

Emergency Close PHIGS	**pemergency_close_phigs** ();
Error Handling	**perr_hand** (error_num, func_num, error_file);
Error Logging	**perr_log** (error_num, func_num, error_file);
Set Error Handling	**pset_err_hand** (new_err_hand, old_err_hand);
Set Error Handling Mode	**pset_err_hand_mode** (error_mode);

Special Interface Subroutines

Escape	**pescape** (func_id, in_data, store, out_data);

Inquire Subroutines

Element Search	**pelem_search** (struct_id, struct_elem, dir, incl, excl, err_ind, status, found_elem_ptr)
Inquire All Conflicting Structures	**pinq_all_conf_structs** (ar_id, num_elems_appl_list, start_ind, err_ind, ids, num_elems_impl_list);
Inquire Annotation Facilities	**pinq_anno_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, styles, num_elems_impl_list, num_anno_char_hts, min_anno_char_ht, max_anno_char_ht);
Inquire Archive Files	**pinq_ar_files** (store, err_ind, ar_files);
Inquire Archive State Value	**pinq_ar_st** (ar_st);
Inquire Choice Device State	**pinq_choice_st** (ws_id, choice_num, store, err_ind, op_mode, echo_switch, init_status, init_choice, prompt_echo, echo_area, choice_data);
Inquire Choice Device State 3	**pinq_choice_st3** (ws_id, choice_num, store, err_ind, op_mode, echo_switch, init_status, init_choice, prompt_echo, echo_vol, choice_data);
Inquire Color Facilities	**pinq_colr_facs** (ws_type, err_ind, fac);
Inquire Color Model	**pinq_colr_model** (ws_id, err_ind, model);
Inquire Color Model Facilities	**pinq_colr_model_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, models, num_elems_impl_list, def);
Inquire Color Representation	**pinq_colr_rep** (ws_id, colr_ind, type, err_ind, colr_rep);
Inquire Conflict Resolution	**pinq_conf_res** (err_ind, archive_res, retrieve_res);
Inquire Conflicting Structures in Network	**pinq_conf_structs_net** (ar_id, struct_id, source, num_elems_appl_list, start_ind, err_ind, ids, num_elems_impl_list);
Inquire Current Element Content	**pinq_cur_elem_content** (store, err_ind, elem_data);
Inquire Current Element Type and Size	**pinq_cur_elem_type_size** (err_ind, elem_type, elem_size);
Inquire Default Choice Device Data	**pinq_def_choice_data** (ws_type, choice_num, store, err_ind, max_choices, pet_list, echo_area, choice_data);
Inquire Default Choice Device Data 3	**pinq_def_choice_data3** (ws_type, choice_num, store, err_ind, max_choices, pet_list, echo_vol, choice_data);
Inquire Default Display Update State	**pinq_def_disp_upd_st** (ws_type, err_ind, def_mode, mod_mode);
Inquire Default Locator Device Data	**pinq_def_loc_data** (ws_type, loc_num, store, err_ind, loc_pos, pet_list, echo_area, loc_data);

Inquire Default Locator Device Data 3	**pinq_def_loc_data3** (ws_type, loc_num, store, err_ind, loc_pos, pet_list, echo_vol, loc_data);
Inquire Default Pick Device Data	**pinq_def_pick_data** (ws_type, pick_num, store, err_ind, pet_list, echo_area, pick_data);
Inquire Default Pick Device Data 3	**pinq_def_pick_data3** (ws_type, pick_num, store, err_ind, pet_list, echo_vol, pick_data);
Inquire Default String Device Data	**pinq_def_string_data** (ws_type, string_num, store, err_ind, max_buf_size, pet_list, echo_area, string_data);
Inquire Default String Device Data 3	**pinq_def_string_data3** (ws_type, string_num, store, err_ind, max_buf_size, pet_list, echo_vol, string_data);
Inquire Default Stroke Device Data	**pinq_def_stroke_data** (ws_type, stroke_num, store, err_ind, max_buf_size, pet_list, echo_area, stroke_data);
Inquire Default Stroke Device Data 3	**pinq_def_stroke_data3** (ws_type, stroke_num, store, err_ind, max_buf_size, pet_list, echo_vol, stroke_data);
Inquire Default Valuator Device Data	**pinq_def_val_data** (ws_type, val_num, store, err_ind, def_value, pet_list, echo_area, val_data);
Inquire Default Valuator Device Data 3	**pinq_def_val_data3** (ws_type, val_num, store, err_ind, def_value, pet_list, echo_vol, val_data);
Inquire Display Space Size	**pinq_disp_space_size** (ws_type, err_ind, size);
Inquire Display Space Size 3	**pinq_disp_space_size3** (ws_type, err_ind, size);
Inquire Display Update State	**pinq_disp_upd_st** (ws_id, err_ind, def_mode, mod_mode, disp_surf_empty, vis_st);
Inquire Dynamics of Structures	**pinq_dyns_structs** (ws_type, err_ind, dyns);
Inquire Dynamics of Workstation Attributes	**pinq_dyns_ws_attrs** (ws_type, err_ind, attr);
Inquire Edge Facilities	**pinq_edge_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list);
Inquire Edge Representation	**pinq_edge_rep** (ws_id, index, type, err_ind, edge_rep);
Inquire Edit Mode	**pinq_edit_mode** (err_ind, edit_mode);
Inquire Element Content	**pinq_elem_content** (struct_id, elem_num, store, err_ind, elem_data);
Inquire Element Pointer	**pinq_elem_ptr** (err_ind, elem_ptr_value);
Inquire Element Type and Size	**pinq_elem_type_size** (struct_id, elem_num, err_ind, elem_type, elem_size);
Inquire Error Handling Mode	**pinq_err_hand_mode** (err_ind, err_mode);
Inquire Generalized Drawing Primitive	**pinq_gdp** (ws_type, gdp, err_ind, num_attr, attr);
Inquire Generalized Drawing Primitive 3	**pinq_gdp3** (ws_type, gdp, err_ind, num_attr, attr);
Inquire Generalized Structure Element Facilities	**pinq_gse_facs** (num_elems_appl_list, start_ind, err_ind, gse, num_elems_impl_list);
Inquire Highlighting Filter	**pinq_highl_filter** (ws_id, store, err_ind, highl_filter);
Inquire HLHSR Identifier Facilities	**pinq_hlhsr_id_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, hlhsr_ids, num_elems_impl_list);
Inquire HLHSR Mode	**pinq_hlhsr_mode** (ws_id, err_ind, upd_st, cur_mode, req_mode);
Inquire HLHSR Mode Facilities	**pinq_hlhsr_mode_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, hlhsr_modes, num_elems_impl_list);
Inquire Input Queue Overflow	**pinq_in_overf** (err_ind, ws_id, in_class, in_num);
Inquire Interior Facilities	**pinq_int_facs** (ws_type, hatch_num_elems_appl_list, hatch_start_ind, err_ind, int_facs, hatch_num_elems_impl_list);
Inquire Interior Representation	**pinq_int_rep** (ws_id, index, type, err_ind, int_rep);
Inquire Invisibility Filter	**pinq_invis_filter** (ws_id, store, err_ind, invis_filter);
Inquire List of Available Generalized Drawing Primitives	**pinq_list_avail_gdp** (ws_type, num_elems_appl_list, start_ind, err_ind, gdps, num_elems_impl_list);
Inquire List of Available Generalized Drawing Primitives 3	**pinq_list_avail_gdp3** (ws_type, num_elems_appl_list, start_ind, err_ind, gdps, num_elems_impl_list);
Inquire List of Available Generalized Structure Elements	**pinq_list_avail_gse** (ws_type, num_elems_appl_list, start_ind, err_ind, gses, num_elems_impl_list);
Inquire List of Available Workstation Types	**pinq_list_avail_ws_types** (num_elems_appl_list, start_ind, err_ind, types, num_elems_impl_list);

Inquire List of Color Indices	**pinq_list_colr_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, colr_ind, num_elems_impl_list);
Inquire List of Edge Indices	**pinq_list_edge_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_edge_ind, num_elems_impl_list);
Inquire List of Interior Indices	**pinq_list_int_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_int_ind, num_elems_impl_list);
Inquire List of Pattern Indices	**pinq_list_pat_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_pat_ind, num_elems_impl_list);
Inquire List of Polyline Indices	**pk.pinq_list_line_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_line_ind, num_elems_impl_list);
Inquire List of Polymarker Indicies	**pinq_list_marker_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_marker_ind, num_elems_impl_list);
Inquire List of Text Indices	**pinq_list_text_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, def_text_ind, num_elems_impl_list);
Inquire List of View Indices	**pinq_list_view_inds** (ws_id, num_elems_appl_list, start_ind, err_ind, view_inds, num_elems_impl_list);
Inquire Locator Device State	**pinq_loc_st** (ws_id, loc_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_loc_pos, prompt_echo, echo_area, loc_data);
Inquire Locator Device State 3	**pinq_loc_st3** (ws_id, loc_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_loc_pos, prompt_echo, echo_vol, loc_data);
Inquire Modeling Clipping Facilities	**pinq_model_clip_facs** (num_elems_appl_list, start_ind, err_ind, num_planes, ops, num_elems_impl_list);
Inquire More Simultaneous Events	**pinq_more_simult_events** (err_ind, simult_events);
Inquire Number of Available Logical Input Devices	**pinq_num_avail_in** (ws_type, err_ind, num_in);
Inquire Number of Display Priorities Supported	**pinq_num_disp_pris** (ws_type, err_ind, num_pri);
Inquire Open Structure	**pinq_open_struct** (err_ind, status, struct_id);
Inquire Paths to Ancestors	**pinq_paths_ances** (struct_id, order, depth, store, err_ind, paths);
Inquire Paths to Descendants	**pinq_paths_descs** (struct_id, order, depth, store, err_ind, paths);
Inquire Pattern Facilities	**pinq_pat_facs** (ws_type, err_ind, num_pred);
Inquire Pattern Representation	**pinq_pat_rep** (ws_id, index, type, store, err_ind, pat_rep);
Inquire PHIGS Facilities	**pinq_phigs_facs** (num_elems_appl_list, start_ind, err_ind, max_open_ws, max_open_ar, num_avail_names, char_sets, num_elems_impl_list, iss_norm_max, iss_inv_max);
Inquire Pick Device State	**pinq_pick_st** (ws_id, pick_num, type, store, err_ind, op_mode, echo_switch, pick_filter, init_status, init_pick, prompt_echo, echo_area, pick_data, order);
Inquire Pick Device State 3	**pinq_pick_st3** (ws_id, pick_num, type, store, err_ind, op_mode, echo_switch, pick_filter, init_status, init_pick, prompt_echo, echo_vol, pick_data, order);
Inquire Polyline Facilities	**pinq_line_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list);
Inquire Polyline Representation	**pinq_line_rep** (ws_id, index, type, errind, line_rep);
Inquire Polymarker Facilities	**pinq_marker_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list);
Inquire Polymarker Representation	**pinq_marker_rep** (ws_id, index, type, err_ind, marker_rep);
Inquire Posted Structures	**pinq_posted_structs** (ws_id, num_elems_appl_list, start_ind, err_ind, struct_ids, num_elems_impl_list);
Inquire Predefined Color Representation	**pinq_pred_colr_rep** (ws_type, colr_ind, err_ind, colr_rep);
Inquire Predefined Edge Representation	**pinq_pred_edge_rep** (ws_type, index, err_ind, bundle);
Inquire Predefined Interior Representation	**pinq_pred_int_rep** (ws_type, index, err_ind, bundle);
Inquire Predefined Pattern Representation	**pinq_pred_pat_rep** (ws_type, index, store, err_ind, pat_rep);
Inquire Predefined Polyline Representation	**pinq_pred_line_rep** (ws_type, index, err_ind, bundle);

Inquire Predefined Polymarker Representation	**pinq_pred_marker_rep** (ws_type, index, err_ind, bundle);
Inquire Predefined Text Representation	**pinq_pred_text_rep** (ws_type, index, err_ind, bundle);
Inquire Predefined View Representation	**pinq_pred_view_rep** (ws_type, index, err_ind, view);
Inquire Set of Open Workstations	**pinq_open_wss** (num_elems_appl_list, start_ind, err_ind, open_ws_ids, num_elems_impl_list);
Inquire Set of Workstations to Which Posted	**pinq_wss_posted** (struct_id, num_elems_appl_list, start_ind, err_ind, ws, num_elems_impl_list);
Inquire String Device State	**pinq_string_st** (ws_id, string_num, store, err_ind, op_mode, echo_switch, init_string, prompt_echo, echo_area, string_data);
Inquire String Device State 3	**pinq_string_st3** (ws_id, string_num, store, err_ind, op_mode, echo_switch, init_string, prompt_echo, echo_vol, string_data);
Inquire Stroke Device State	**pinq_stroke_st** (ws_id, stroke_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_stroke, prompt_echo, echo_area, stroke_data);
Inquire Stroke Device State 3	**pinq_stroke_st3** (ws_id, stroke_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_stroke, prompt_echo, echo_vol, stroke_data);
Inquire Structure Identifiers	**pinq_struct_ids** (num_elems_appl_list, start_ind, err_ind, struct_ids, num_elems_impl_list);
Inquire Structure State Value	**pinq_struct_st** (struct_st);
Inquire Structure Status	**pinq_struct_status** (struct_id, err_ind, status);
Inquire System State Value	**pinq_sys_st** (sys_st);
Inquire Text Extent	**pinq_text_extent** (ws_type, text_font, char_expan, char_space, char_ht, text_path, hor_text_align, vert_text_align, char_string, err_ind, rect, offset);
Inquire Text Facilities	**pinq_text_facs** (ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list);
Inquire Text Representation	**pinq_text_rep** (ws_id, index, type, err_ind, text_rep);
Inquire Valuator Device State	**pinq_val_st** (ws_id, val_num, store, err_ind, op_mode, echo_switch, init_value, prompt_echo, echo_area, val_data);
Inquire Valuator Device State 3	**pinq_val_st3** (ws_id, val_num, store, err_ind, op_mode, echo_switch, init_value, prompt_echo, echo_vol, val_data);
Inquire View Facilities	**pinq_view_facs** (ws_type, err_ind, num_view_ind);
Inquire View Representation	**pinq_view_rep** (ws_id, view_ind, err_ind, upd_st, cur_view, req_view);
Inquire Workstation Category	**pinq_ws_cat** (ws_type, err_ind, cat);
Inquire Workstation Classification	**pinq_ws_class** (ws_type, err_ind, ws_class);
Inquire Workstation Connection and Type	**pinq_ws_conn_type** (ws_id, store, err_ind, conn_id, ws_type);
Inquire Workstation State Table Lengths	**pinq_ws_st_table** (ws_type, err_ind, lengths);
Inquire Workstation State Value	**pinq_ws_st** (ws_st);
Inquire Workstation Transformation	**pinq_ws_tran** (ws_id, err_ind, upd_st, req_win_lim, cur_win_lim, req_vp_lim, cur_vp_lim);
Inquire Workstation Transformation 3	**pinq_ws_tran3** (ws_id, err_ind, upd_st, req_win_lim, cur_win_lim, req_vp_lim, cur_vp_lim);

Chapter 3. Alphabetical Listing of Subroutines for C Binding

Add Names to Set	**padd_names_set (**names**);**
Annotation Text Relative	**panno_text_rel (**ref_pt, offset, char_string**);**
Annotation Text Relative 3	**panno_text_rel3 (**ref_pt, offset, char_string**);**
Application Data	**pappl_data (**data**);**
Archive All Structures	**par_all_structs (**archive_id**);**
Archive Structure Networks	**par_struct_nets (**archive_id, struct_ids**);**
Archive Structures	**par_structs (**archive_id, struct_ids**);**
Await Event	**pawait_event (**timeout, ws_id, dev_class, in_num**);**
Build Transformation Matrix	**pbuild_tran_matrix (**point, shift_vec, angle, scale_vec, err_ind, result_tran**);**
Build Transformation Matrix 3	**pbuild_tran_matrix3 (**point, shift_vec, x_angle, y_angle, z_angle, scale_vec, err_ind, result_tran**);**
Cell Array	**pcell_array (**rect, colr_array**);**
Cell Array 3	**pcell_array3 (**paral, colr_array**);**
Change Structure Identifier	**pchange_struct_id (**orig_struct_id, result_struct_id**);**
Change Structure Identifier and References	**pchange_struct_id_refs (**orig_struct_id, result_struct_id**);**
Change Structure References	**pchange_struct_refs (**orig_struct_id, result_struct_id**);**
Close Archive File	**pclose_ar_file (**archive_id**);**
Close PHIGS	**pclose_phigs ();**
Close Structure	**pclose_struct ();**
Close Workstation	**pclose_ws (**ws_id**);**
Compose Matrix	**pcompose_matrix (**tran_a, tran_b, err_ind, result_tran**);**
Compose Matrix 3	**pcompose_matrix3 (**tran_a, tran_b, err_ind, result_tran**);**
Compose Transformation Matrix	**pcompose_tran_matrix (**tran, point, shift_vec, angle, scale_vec, err_ind, result_tran**);**
Compose Transformation Matrix 3	**pcompose_tran_matrix3 (**tran, point, shift_vec, x_angle, y_angle, z_angle, scale_vec, err_ind, result_tran**);**
Copy All Elements from Structure	**pcopy_all_elems_struct (**struct_id**);**
Create Store	**pcreate_store (**err_ind, store**);**
Delete All Structures	**pdel_all_structs ();**
Delete All Structures from Archive	**pdel_all_structs_ar (**archive_id>**);**
Delete Element	**pdel_elem ();**
Delete Element Range	**pdel_elem_range (**elem_ptr1_value, elem_ptr2_value**);**
Delete Elements Between Labels	**pdel_elems_labels (**label1_id, label2_id**);**
Delete Store	**pdel_store (**err_ind, store**);**
Delete Structure	**pdel_struct (**struct_id**);**
Delete Structure Network	**pdel_struct_net (**struct_id, ref_flag**);**
Delete Structure Networks from Archive	**pdel_struct_nets_ar (**archive_id, struct_ids**);**
Delete Structures from Archive	**pdel_structs_ar (**archive_id, struct_ids**);**
Element Search	**pelem_search (**struct_id, struct_elem, dir, incl, excl, err_ind, status, found_elem_ptr**)**
Emergency Close PHIGS	**pemergency_close_phigs ();**
Empty Structure	**pempty_struct (**struct_id**);**
Error Handling	**perr_hand (**error_num, func_num, error_file**);**
Error Logging	**perr_log (**error_num, func_num, error_file**);**
Escape	**pescape (**func_id, in_data, store, out_data**);**
Evaluate View Mapping Matrix	**peval_view_map_matrix (**mapping, err_ind, result_tran**);**

Evaluate View Mapping Matrix 3	**peval_view_map_matrix3 (**mapping, err_ind, result_tran**);**
Evaluate View Orientation Matrix	**peval_view_ori_matrix (**view_ref_point, view_up_vec, err_ind, result_tran**);**
Evaluate View Orientation Matrix 3	**peval_view_ori_matrix3 (**view_ref_point, view_norm_vec, view_up_vec, err_ind, result_tran**);**
Execute Structure	**pexec_struct (**struct_id**);**
Fill Area	**pfill_area (**point_list**);**
Fill Area 3	**pfill_area3 (**point_list**);**
Fill Area Set	**pfill_area_set (**point_list**);**
Fill Area Set 3	**pfill_area_set3 (**point_list_list**);**
Flush Device Events	**pflush_events (**ws_id, dev_class, in_num**);**
Generalized Drawing Primitive	**pgdp (**point_list, gdp_id, gdp_data**);**
Generalized Drawing Primitive 3	**pgdp3 (**point_list, gdp3_id, gdp_data**);**
Generalized Structure Element	**pgse (**id, gse_data**);**
Get Choice	**pget_choice (**in_status, choice**);**
Get Locator	**pget_loc (**view_ind, loc_pos**);**
Get Locator 3	**pget_loc3 (**view_ind, loc_pos**);**
Get Pick	**pget_pick (**depth, in_status, pick**);**
Get String	**pget_string (**string**);**
Get Stroke	**pget_stroke (**view_ind, stroke**);**
Get Stroke 3	**pget_stroke3 (**view_ind, stroke**);**
Get Valuator	**pget_val (**value**);**
Initialize Choice	**pinit_choice (**ws_id, choice_num, init_status, init_choice, pet, echo_area, choice_data**);**
Initialize Choice 3	**pinit_choice3 (**ws_id, choice_num, init_status, init_choice, pet, echo_vol, choice_data**);**
Initialize Locator	**pinit_loc (**ws_id, loc_num, init_view_ind, init_loc_pos, pet, echo_area, loc_data**);**
Initialize Locator 3	**pinit_loc3 (**ws_id, loc_num, init_view_ind, init_loc_pos, pet, echo_vol, loc_data**);**
Initialize Pick	**pinit_pick (**ws_id, pick_num, init_status, init_pick, pet, echo_area, pick_data, order**);**
Initialize Pick 3	**pinit_pick3 (**ws_id, pick_num, init_status, init_pick, pet, echo_vol, pick_data, order**);**
Initialize String	**pinit_string (**ws_id, string_num, init_string, pet, echo_area, string_data**);**
Initialize String 3	**pinit_string3 (**ws_id, string_num, init_string, pet, echo_vol, string_data**);**
Initialize Stroke	**pinit_stroke (**ws_id, stroke_num, init_view_ind, init_stroke, pet, echo_area, stroke_data**);**
Initialize Stroke 3	**pinit_stroke3 (**ws_id, stroke_num, init_view_ind, init_stroke, pet, echo_vol, stroke_data**);**
Initialize Valuator	**pinit_val (**ws_id, val_num, init_value, pet, echo_area, val_data**);**
Initialize Valuator 3	**pinit_val3 (**ws_id, val_num, init_value, pet, echo_vol, val_data**);**
Inquire All Conflicting Structures	**pinq_all_conf_structs (**ar_id, num_elems_appl_list, start_ind, err_ind, ids, num_elems_impl_list**);**
Inquire Annotation Facilities	**pinq_anno_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, styles, num_elems_impl_list, num_anno_char_hts, min_anno_char_ht, max_anno_char_ht**);**
Inquire Archive Files	**pinq_ar_files (**store, err_ind, ar_files**);**
Inquire Archive State Value	**pinq_ar_st (**ar_st**);**

Inquire Choice Device State	**pinq_choice_st (**ws_id, choice_num, store, err_ind, op_mode, echo_switch, init_status, init_choice, prompt_echo, echo_area, choice_data**);**
Inquire Choice Device State 3	**pinq_choice_st3 (**ws_id, choice_num, store, err_ind, op_mode, echo_switch, init_status, init_choice, prompt_echo, echo_vol, choice_data**);**
Inquire Color Facilities	**pinq_colr_facs (**ws_type, err_ind, fac**);**
Inquire Color Model	**pinq_colr_model (**ws_id, err_ind, model**);**
Inquire Color Model Facilities	**pinq_colr_model_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, models, num_elems_impl_list, def**);**
Inquire Color Representation	**pinq_colr_rep (**ws_id, colr_ind, type, err_ind, colr_rep**);**
Inquire Conflict Resolution	**pinq_conf_res (**err_ind, archive_res, retrieve_res**);**
Inquire Conflicting Structures in Network	**pinq_conf_structs_net (**ar_id, struct_id, source, num_elems_appl_list, start_ind, err_ind, ids, num_elems_impl_list**);**
Inquire Current Element Content	**pinq_cur_elem_content (**store, err_ind, elem_data**);**
Inquire Current Element Type and Size	**pinq_cur_elem_type_size (**err_ind, elem_type, elem_size**);**
Inquire Default Choice Device Data	**pinq_def_choice_data (**ws_type, choice_num, store, err_ind, max_choices, pet_list, echo_area, choice_data**);**
Inquire Default Choice Device Data 3	**pinq_def_choice_data3 (**ws_type, choice_num, store, err_ind, max_choices, pet_list, echo_vol, choice_data**);**
Inquire Default Display Update State	**pinq_def_disp_upd_st (**ws_type, err_ind, def_mode, mod_mode**);**
Inquire Default Locator Device Data	**pinq_def_loc_data (**ws_type, loc_num, store, err_ind, loc_pos, pet_list, echo_area, loc_data**);**
Inquire Default Locator Device Data 3	**pinq_def_loc_data3 (**ws_type, loc_num, store, err_ind, loc_pos, pet_list, echo_vol, loc_data**);**
Inquire Default Pick Device Data	**pinq_def_pick_data (**ws_type, pick_num, store, err_ind, pet_list, echo_area, pick_data**);**
Inquire Default Pick Device Data 3	**pinq_def_pick_data3 (**ws_type, pick_num, store, err_ind, pet_list, echo_vol, pick_data**);**
Inquire Default String Device Data	**pinq_def_string_data (**ws_type, string_num, store, err_ind, max_buf_size, pet_list, echo_area, string_data**);**
Inquire Default String Device Data 3	**pinq_def_string_data3 (**ws_type, string_num, store, err_ind, max_buf_size, pet_list, echo_vol, string_data**);**
Inquire Default Stroke Device Data	**pinq_def_stroke_data (**ws_type, stroke_num, store, err_ind, max_buf_size, pet_list, echo_area, stroke_data**);**
Inquire Default Stroke Device Data 3	**pinq_def_stroke_data3 (**ws_type, stroke_num, store, err_ind, max_buf_size, pet_list, echo_vol, stroke_data**);**
Inquire Default Valuator Device Data	**pinq_def_val_data (**ws_type, val_num, store, err_ind, def_value, pet_list, echo_area, val_data**);**
Inquire Default Valuator Device Data 3	**pinq_def_val_data3 (**ws_type, val_num, store, err_ind, def_value, pet_list, echo_vol, val_data**);**
Inquire Display Space Size	**pinq_disp_space_size (**ws_type, err_ind, size**);**
Inquire Display Space Size 3	**pinq_disp_space_size3 (**ws_type, err_ind, size**);**
Inquire Display Update State	**pinq_disp_upd_st (**ws_id, err_ind, def_mode, mod_mode, disp_surf_empty, vis_st**);**
Inquire Dynamics of Structures	**pinq_dyns_structs (**ws_type, err_ind, dyns**);**
Inquire Dynamics of Workstation Attributes	**pinq_dyns_ws_attrs (**ws_type, err_ind, attr**);**

Inquire Edge Facilities	**pinq_edge_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list**);**
Inquire Edge Representation	**pinq_edge_rep (**ws_id, index, type, err_ind, edge_rep**);**
Inquire Edit Mode	**pinq_edit_mode (**err_ind, edit_mode**);**
Inquire Element Content	**pinq_elem_content (**struct_id, elem_num, store, err_ind, elem_data**);**
Inquire Element Pointer	**pinq_elem_ptr (**err_ind, elem_ptr_value**);**
Inquire Element Type and Size	**pinq_elem_type_size (**struct_id, elem_num, err_ind, elem_type, elem_size**);**
Inquire Error Handling Mode	**pinq_err_hand_mode (**err_ind, err_mode**);**
Inquire Generalized Drawing Primitive	**pinq_gdp (**ws_type, gdp, err_ind, num_attr, attr**);**
Inquire Generalized Drawing Primitive 3	**pinq_gdp3 (**ws_type, gdp, err_ind, num_attr, attr**);**
Inquire Generalized Structure Element Facilities	**pinq_gse_facs (**num_elems_appl_list, start_ind, err_ind, gse, num_elems_impl_list**);**
Inquire Highlighting Filter	**pinq_highl_filter (**ws_id, store, err_ind, highl_filter**);**
Inquire HLHSR Identifier Facilities	**pinq_hlhsr_id_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, hlhsr_ids, num_elems_impl_list**);**
Inquire HLHSR Mode	**pinq_hlhsr_mode (**ws_id, err_ind, upd_st, cur_mode, req_mode**);**
Inquire HLHSR Mode Facilities	**pinq_hlhsr_mode_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, hlhsr_modes, num_elems_impl_list**);**
Inquire Input Queue Overflow	**pinq_in_overf (**err_ind, ws_id, in_class, in_num**);**
Inquire Interior Facilities	**pinq_int_facs (**ws_type, hatch_num_elems_appl_list, hatch_start_ind, err_ind, int_facs, hatch_num_elems_impl_list**);**
Inquire Interior Representation	**pinq_int_rep (**ws_id, index, type, err_ind, int_rep**);**
Inquire Invisibility Filter	**pinq_invis_filter (**ws_id, store, err_ind, invis_filter**);**
Inquire List of Available Generalized Drawing Primitives	**pinq_list_avail_gdp (**ws_type, num_elems_appl_list, start_ind, err_ind, gdps, num_elems_impl_list**);**
Inquire List of Available Generalized Drawing Primitives 3	**pinq_list_avail_gdp3 (**ws_type, num_elems_appl_list, start_ind, err_ind, gdps, num_elems_impl_list**);**
Inquire List of Available Generalized Structure Elements	**pinq_list_avail_gse (**ws_type, num_elems_appl_list, start_ind, err_ind, gses, num_elems_impl_list**);**
Inquire List of Available Workstation Types	**pinq_list_avail_ws_types (**num_elems_appl_list, start_ind, err_ind, types, num_elems_impl_list**);**
Inquire List of Color Indices	**pinq_list_colr_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, colr_ind, num_elems_impl_list**);**
Inquire List of Edge Indices	**pinq_list_edge_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_edge_ind, num_elems_impl_list**);**
Inquire List of Interior Indices	**pinq_list_int_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_int_ind, num_elems_impl_list**);**
Inquire List of Pattern Indices	**pinq_list_pat_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_pat_ind, num_elems_impl_list**);**
Inquire List of Polyline Indices	**pk.pinq_list_line_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_line_ind, num_elems_impl_list**);**
Inquire List of Polymarker Indicies	**pinq_list_marker_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_marker_ind, num_elems_impl_list**);**

Inquire List of Text Indices	**pinq_list_text_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, def_text_ind, num_elems_impl_list**);**
Inquire List of View Indices	**pinq_list_view_inds (**ws_id, num_elems_appl_list, start_ind, err_ind, view_inds, num_elems_impl_list**);**
Inquire Locator Device State	**pinq_loc_st (**ws_id, loc_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_loc_pos, prompt_echo, echo_area, loc_data**);**
Inquire Locator Device State 3	**pinq_loc_st3 (**ws_id, loc_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_loc_pos, prompt_echo, echo_vol, loc_data**);**
Inquire Modeling Clipping Facilities	**pinq_model_clip_facs (**num_elems_appl_list, start_ind, err_ind, num_planes, ops, num_elems_impl_list**);**
Inquire More Simultaneous Events	**pinq_more_simult_events (**err_ind, simult_events**);**
Inquire Number of Available Logical Input Devices	**pinq_num_avail_in (**ws_type, err_ind, num_in**);**
Inquire Number of Display Priorities Supported	**pinq_num_disp_pris (**ws_type, err_ind, num_pri**);**
Inquire Open Structure	**pinq_open_struct (**err_ind, status, struct_id**);**
Inquire Paths to Ancestors	**pinq_paths_ances (**struct_id, order, depth, store, err_ind, paths**);**
Inquire Paths to Descendants	**pinq_paths_descs (**struct_id, order, depth, store, err_ind, paths**);**
Inquire Pattern Facilities	**pinq_pat_facs (**ws_type, err_ind, num_pred**);**
Inquire Pattern Representation	**pinq_pat_rep (**ws_id, index, type, store, err_ind, pat_rep**);**
Inquire PHIGS Facilities	**pinq_phigs_facs (**num_elems_appl_list, start_ind, err_ind, max_open_ws, max_open_ar, num_avail_names, char_sets, num_elems_impl_list, iss_norm_max, iss_inv_max**);**
Inquire Pick Device State	**pinq_pick_st (**ws_id, pick_num, type, store, err_ind, op_mode, echo_switch, pick_filter, init_status, init_pick, prompt_echo, echo_area, pick_data, order**);**
Inquire Pick Device State 3	**pinq_pick_st3 (**ws_id, pick_num, type, store, err_ind, op_mode, echo_switch, pick_filter, init_status, init_pick, prompt_echo, echo_vol, pick_data, order**);**
Inquire Polyline Facilities	**pinq_line_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list**);**
Inquire Polyline Representation	**pinq_line_rep (**ws_id, index, type, errind, line_rep**);**
Inquire Polymarker Facilities	**pinq_marker_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list**);**
Inquire Polymarker Representation	**pinq_marker_rep (**ws_id, index, type, err_ind, marker_rep**);**
Inquire Posted Structures	**pinq_posted_structs (**ws_id, num_elems_appl_list, start_ind, err_ind, struct_ids, num_elems_impl_list**);**
Inquire Predefined Color Representation	**pinq_pred_colr_rep (**ws_type, colr_ind, err_ind, colr_rep**);**
Inquire Predefined Edge Representation	**pinq_pred_edge_rep (**ws_type, index, err_ind, bundle**);**
Inquire Predefined Interior Representation	**pinq_pred_int_rep (**ws_type, index, err_ind, bundle**);**
Inquire Predefined Pattern Representation	**pinq_pred_pat_rep (**ws_type, index, store, err_ind, pat_rep**);**
Inquire Predefined Polyline Representation	**pinq_pred_line_rep (**ws_type, index, err_ind, bundle**);**
Inquire Predefined Polymarker Representation	**pinq_pred_marker_rep (**ws_type, index, err_ind, bundle**);**
Inquire Predefined Text Representation	**pinq_pred_text_rep (**ws_type, index, err_ind, bundle**);**
Inquire Predefined View Representation	**pinq_pred_view_rep (**ws_type, index, err_ind, view**);**

Inquire Set of Open Workstations	**pinq_open_wss (**num_elems_appl_list, start_ind, err_ind, open_ws_ids, num_elems_impl_list**);**
Inquire Set of Workstations to Which Posted	**pinq_wss_posted (**struct_id, num_elems_appl_list, start_ind, err_ind, ws, num_elems_impl_list**);**
Inquire String Device State	**pinq_string_st (**ws_id, string_num, store, err_ind, op_mode, echo_switch, init_string, prompt_echo, echo_area, string_data**);**
Inquire String Device State 3	**pinq_string_st3 (**ws_id, string_num, store, err_ind, op_mode, echo_switch, init_string, prompt_echo, echo_vol, string_data**);**
Inquire Stroke Device State	**pinq_stroke_st (**ws_id, stroke_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_stroke, prompt_echo, echo_area, stroke_data**);**
Inquire Stroke Device State 3	**pinq_stroke_st3 (**ws_id, stroke_num, type, store, err_ind, op_mode, echo_switch, init_view_ind, init_stroke, prompt_echo, echo_vol, stroke_data**);**
Inquire Structure Identifiers	**pinq_struct_ids (**num_elems_appl_list, start_ind, err_ind, struct_ids, num_elems_impl_list**);**
Inquire Structure State Value	**pinq_struct_st (**struct_st**);**
Inquire Structure Status	**pinq_struct_status (**struct_id, err_ind, status**);**
Inquire System State Value	**pinq_sys_st (**sys_st**);**
Inquire Text Extent	**pinq_text_extent (**ws_type, text_font, char_expan, char_space, char_ht, text_path, hor_text_align, vert_text_align, char_string, err_ind, rect, offset**);**
Inquire Text Facilities	**pinq_text_facs (**ws_type, num_elems_appl_list, start_ind, err_ind, fac, num_elems_impl_list**);**
Inquire Text Representation	**pinq_text_rep (**ws_id, index, type, err_ind, text_rep**);**
Inquire Valuator Device State	**pinq_val_st (**ws_id, val_num, store, err_ind, op_mode, echo_switch, init_value, prompt_echo, echo_area, val_data**);**
Inquire Valuator Device State 3	**pinq_val_st3 (**ws_id, val_num, store, err_ind, op_mode, echo_switch, init_value, prompt_echo, echo_vol, val_data**);**
Inquire View Facilities	**pinq_view_facs (**ws_type, err_ind, num_view_ind**);**
Inquire View Representation	**pinq_view_rep (**ws_id, view_ind, err_ind, upd_st, cur_view, req_view**);**
Inquire Workstation Category	**pinq_ws_cat (**ws_type, err_ind, cat**);**
Inquire Workstation Classification	**pinq_ws_class (**ws_type, err_ind, ws_class**);**
Inquire Workstation Connection and Type	**pinq_ws_conn_type (**ws_id, store, err_ind, conn_id, ws_type**);**
Inquire Workstation State Table Lengths	**pinq_ws_st_table (**ws_type, err_ind, lengths**);**
Inquire Workstation State Value	**pinq_ws_st (**ws_st**);**
Inquire Workstation Transformation	**pinq_ws_tran (**ws_id, err_ind, upd_st, req_win_lim, cur_win_lim, req_vp_lim, cur_vp_lim**);**
Inquire Workstation Transformation 3	**pinq_ws_tran3 (**ws_id, err_ind, upd_st, req_win_lim, cur_win_lim, req_vp_lim, cur_vp_lim**);**
Label	**plabel (**label_id**);**
Message	**pmessage (**ws_id, message**);**
Offset Element Pointer	**poffset_elem_ptr (**elem_ptr_offset**);**
Open Archive File	**popen_ar_file (**archive_id, archive_file**);**
Open PHIGS	**popen_phigs (**err_file, mem_units**);**
Open Structure	**popen_struct (**struct_id**);**
Open Workstation	**popen_ws (**ws_id, conn_id, ws_type**);**
Polyline	**ppolyline (**point_list**);**
Polyline 3	**ppolyline3 (**point_list**);**
Polymarker	**ppolymarker (**point_list**);**

Polymarker 3	**ppolymarker3 (**point_list**);**
Post Structure	**ppost_struct (**ws_id, struct_id, pri**);**
Redraw All Structures	**predraw_all_structs (**ws_id, ctrl_flag**);**
Remove Names from Set	**premove_names_set (**names**);**
Request Choice	**preq_choice (**ws_id, choice_num, in_status, choice**);**
Request Locator	**preq_loc (**ws_id, loc_num, in_status, view_ind, loc_pos**);**
Request Locator 3	**preq_loc3 (**ws_id, loc_num, in_status, view_ind, loc_pos**);**
Request Pick	**preq_pick (**ws_id, pick_num, depth, in_status, pick**);**
Request String	**preq_string (**ws_id, string_num, in_status, string**);**
Request Stroke	**preq_stroke (**ws_id, stroke_num, in_status, view_ind, stroke**);**
Request Stroke 3	**preq_stroke3 (**ws_id, stroke_num, in_status, view_ind, stroke**);**
Request Valuator	**preq_val (**ws_id, val_num, in_status, value**);**
Restore Modeling Clipping Volume	**prestore_model_clip_vol();**
Retrieve All Structures	**pret_all_structs (**archive_id**);**
Retrieve Paths to Ancestors	**pret_paths_ances (**ar_id, struct_id, order, depth, store, paths**);**
Retrieve Paths to Descendants	**pret_paths_descs (**ar_id, struct_id, order, depth, store, paths**);**
Retrieve Structure Identifiers	**pret_struct_id (**archive_id, num_elems_appl_list, start_ind, ids, num_elems_impl_list**);**
Retrieve Structure Networks	**pret_struct_nets (**archive_id, struct_ids**);**
Retrieve Structures	**pret_structs (**archive_id, struct_ids**);**
Rotate	**protate (**angle, err_ind, result_tran**);**
Rotate X	**protate_x (**angle, err_ind, result_tran**);**
Rotate Y	**protate_y (**angle, err_ind, result_tran**);**
Rotate Z	**protate_z (**angle, err_ind, result_tran**);**
Sample Choice	**psample_choice (**ws_id, choice_num, choice_in_status, choice**);**
Sample Locator	**psample_loc (**ws_id, loc_num, view_ind, loc_pos**);**
Sample Locator 3	**psample_loc3 (**ws_id, loc_num, view_ind, loc_pos**);**
Sample Pick	**psample_pick (**ws_id, pick_num, depth, pick_in_status, pick**);**
Sample String	**psample_string (**ws_id, string_num, string**);**
Sample Stroke	**psample_stroke (**ws_id, stroke_num, view_ind, stroke**);**
Sample Stroke 3	**psample_stroke3 (**ws_id, stroke_num, view_ind, stroke**);**
Sample Valuator	**psample_val (**ws_id, val_num, value**);**
Scale	**pscale (**scale_vec, err_ind, result_tran**);**
Scale 3	**pscale3 (**scale_vec, err_ind, result_tran**);**
Set Annotation Style	**pset_anno_style (**anno_style**);**
Set Annotation Text Alignment	**pset_anno_align (**text_align**);**
Set Annotation Text Character Height	**pset_anno_char_ht (**char_ht**);**
Set Annotation Text Character Up Vector	**pset_anno_char_up_vec (**char_up_vec**);**
Set Annotation Text Path	**pset_anno_path (**text_path**);**
Set Character Expansion Factor	**pset_char_expan (**char_expan**);**
Set Character Height	**pset_char_ht (**char_ht**);**
Set Character Spacing	**pset_char_space (**char_space**);**
Set Character Up Vector	**pset_char_up_vec (**char_up_vec**);**
Set Choice Mode	**pset_choice_mode (**ws_id, choice_num, op_mode, echo_switch**);**
Set Color Model	**pset_colr_model (**ws_id, colr_model**);**

Set Color Representation	**pset_colr_rep (**ws_id, colr_ind, colr_rep**);**
Set Conflict Resolution	**pset_conf_res (**archive_res, retrieval_res**);**
Set Display Update State	**pset_disp_upd_st (**ws_id, def_mode, mod_mode**);**
Set Edge Color Index	**pset_edge_colr_ind (**edge_colr_ind**);**
Set Edge Flag	**pset_edge_flag (**edge_flag**);**
Set Edge Index	**pset_edge_ind (**edge_ind**);**
Set Edge Representation	**pset_edge_rep (**ws_id, edge_ind, edge_bundle**);**
Set Edgetype	**pset_edgetype (**edgetype**);**
Set Edgewidth Scale Factor	**pset_edgewidth (**edgewidth**);**
Set Edit Mode	**pset_edit_mode (**edit_mode**);**
Set Element Pointer	**pset_elem_ptr (**elem_ptr_value**);**
Set Element Pointer at Label	**pset_elem_ptr_label (**label_id**);**
Set Error Handling	**pset_err_hand (**new_err_hand, old_err_hand**);**
Set Error Handling Mode	**pset_err_hand_mode (**error_mode**);**
Set Global Transformation	**pset_global_tran (**global_tran**);**
Set Global Transformation 3	**pset_global_tran3 (**global_tran**);**
Set Highlighting Filter	**pset_highl_filter (**ws_id, filter**);**
Set HLHSR Identifier	**pset_hlhsr_id (**hlhsr_id**);**
Set HLHSR Mode	**pset_hlhsr_mode (**ws_id, hlhsr_mode**);**
Set Individual ASF	**pset_indiv_asf (**asf_id, asf_source**);**
Set Interior Color Index	**pset_int_colr_ind (**int_colr_ind**);**
Set Interior Index	**pset_int_ind (**int_ind**);**
Set Interior Representation	**pset_int_rep (**ws_id, int_ind, int_bundle**);**
Set Interior Style	**pset_int_style (**int_style**);**
Set Interior Style Index	**pset_int_style_ind (**int_style_ind**);**
Set Invisibility Filter	**pset_invis_filter (**ws_id, filter**);**
Set Linetype	**pset_linetype (**linetype**);**
Set Linewidth Scale Factor	**pset_linewidth (**linewidth**);**
Set Local Transformation	**pset_local_tran (**local_tran, compose_type**);**
Set Local Transformation 3	**pset_local_tran3 (**local_tran, compose_type**);**
Set Locator Mode	**pset_loc_mode (**ws_id, loc_num, op_mode, echo_switch**);**
Set Marker Size Scale Factor	**pset_marker_size (**marker_size**);**
Set Marker Type	**pset_marker_type (**marker_type**);**
Set Modeling Clipping Indicator	**pset_model_clip_ind (**clip_ind**);**
Set Modeling Clipping Volume	**pset_model_clip_vol (**op, half_spaces**);**
Set Modeling Clipping Volume 3	**pset_model_clip_vol3 (**op, half_spaces**);**
Set Pattern Reference Point	**pset_pat_ref_point (**pat_ref_point**);**
Set Pattern Reference Point and Vectors	**pset_pat_ref_point_vecs (**pat_ref_point, pat_ref_vec**);**
Set Pattern Representation	**pset_pat_rep (**ws_id, pat_ind, pat_bundle**);**
Set Pattern Size	**pset_pat_size (**pat_size**);**
Set Pick Filter	**pset_pick_filter (**ws_id, pick_num, filter**);**
Set Pick Identifier	**pset_pick_id (**pick_id**);**
Set Pick Mode	**pset_pick_mode (**ws_id, pick_num, op_mode, echo_switch**);**
Set Polyline Color Index	**pset_line_colr_ind (**line_colr_ind**);**
Set Polyline Index	**pset_line_ind (**line_ind**);**
Set Polyline Representation	**pset_line_rep (**ws_id, line_ind, line_bundle**);**
Set Polymarker Color Index	**pset_marker_colr_ind (**marker_colr_ind**);**
Set Polymarker Index	**pset_marker_ind (**marker_ind**);**
Set Polymarker Representation	**pset_marker_rep (**ws_id, marker_ind, marker_bundle**);**
Set String Mode	**pset_string_mode (**ws_id, string_num, op_mode, echo_switch**);**
Set Stroke Mode	**pset_stroke_mode (**ws_id, stroke_num, op_mode, echo_switch**);**
Set Text Alignment	**pset_text_align (**text_align**);**

Set Text Color Index	**pset_text_colr_ind (**text_colr_ind**);**
Set Text Font	**pset_text_font (**font**);**
Set Text Index	**pset_text_ind (**text_ind**);**
Set Text Path	**pset_text_path (**text_path**);**
Set Text Precision	**pset_text_prec (**prec**);**
Set Text Representation	**pset_text_rep (**ws_id, text_ind, text_bundle**);**
Set Valuator Mode	**pset_valuator_mode (**ws_id, val_num, op_mode, echo_switch**);**
Set View Index	**pset_view_ind (**view_ind**);**
Set View Representation	**pset_view_rep (**ws_id, view_ind, view_rep**);**
Set View Representation 3	**pset_view_rep3 (**ws_id, view_ind, view_rep**);**
Set View Transformation Input Priority	**pset_view_tran_in_pri (**ws_id, view_ind, ref_view_ind, rel_pri**);**
Set Workstation Viewport	**pset_ws_vp (**ws_id, ws_vp_limits**);**
Set Workstation Viewport 3	**pset_ws_vp3 (**ws_id, ws_vp_limits**);**
Set Workstation Window	**pset_ws_win (**ws_id, ws_win_limits**);**
Set Workstation Window 3	**pset_ws_win3 (**ws_id, ws_win_limits**);**
Text	**ptext (**text_pos, char_string**);**
Text 3	**ptext3 (**text_pos, text_dir, char_string**);**
Transform Point	**ptran_point (**point, tran, err_ind, result**);**
Transform Point 3	**ptran_point3 (**point, tran, err_ind, result**);**
Translate	**ptranslate (**trans_vec, err_ind, result_tran**);**
Translate 3	**ptranslate3 (**trans_vec, err_ind, result_tran**);**
Unpost All Structures	**punpost_all_structs (**ws_id**);**
Unpost Structure	**punpost_struct (**ws_id, struct_id**);**
Update Workstation	**pupd_ws (**ws_id, regen_flag**);**

Chapter 4. Listing by Function of Subroutines for FORTRAN Binding

This section contains the following tables:

- Control Subroutines
- Output Primitives
- Attribute Specification
- Miscellaneous Structure Element Subroutines
- Structure Operation Subroutines
- Workstation Table Settings
- Structure Display Subroutines
- Structure Archiving Subroutines
- Transformation Subroutines
- Input Subroutines
- Utility Subroutines
- Error Control Subroutines
- Special Interface Subroutines
- Inquire Subroutines

Note: When two mnemonics are listed after a given subroutine name, the first is FORTRAN and the one following is FORTRAN Subset.

Control Subroutines

Close PHIGS	**PCLPH**
Close Workstation	**PCLWK** (**wkid**)
Message	**PMSG** (**wkid**, **mess**)
	PMSGS (**wkid**, **lstr**, **mess**)
Open PHIGS	**POPPH** (**errfil**, **bufa**)
Open Workstation	**POPWK** (**wkid**, **conid**, **wtype**)
Redraw All Structures	**PRST** (**wkid**, **cofl**)
Set Display Update State	**PSDUS** (**wkid**, **defmod**, **modmod**)
Update Workstation	**PUWK** (**wkid**, **regfl**)

Output Primitives

Annotation Text Relative	**PATR** (**rpx**, **rpy**, **apx**, **apy**, **chars**)
	PATRS (**rpx**, **rpy**, **apx**, **apy**, **lstr**, **chars**)
Annotation Text Relative 3	**PATR3** (**rpx**, **rpy**, **rpz**, **apx**, **apy**, **apz**, **chars**)
	PATR3S (**rpx**, **rpy**, **rpz**, **apx**, **apy**, **apz**, **lstr**, **chars**)
Cell Array	**PCA** (**px**, **py**, **qx**, **qy**, **dimx**, **dimy**, **isc**, **isr**, **dx**, **dy**, **colia**)
Cell Array 3	**PCA3** (**cpxa**, **cpya**, **cpza**, **dimx**, **dimy**, **isc**, **isr**, **dx**, **dy**, **colia**)
Fill Area	**PFA** (**n**, **pxa**, **pya**)
Fill Area 3	**PFA3** (**n**, **pxa**, **pya**, **pza**)
Fill Area Set	**PFAS** (**npl**, **ixa**, **pxa**, **pya**,)
Fill Area Set 3	**PFAS3** (**npl**, **ixa**, **pxa**, **pya**, **pza**)
Generalized Drawing Primitive	**PGDP** (**n**, **pxa**, **pya**, **primid**, **ldr**, **datrec**)

Generalized Drawing Primitive 3	**PGDP3 (n, pxa, pya, pza, primid, ldr, datrec)**
Polyline	**PPL (n, pxa, pya)**
Polyline 3	**PPL3 (n, pxa, pya, pza)**
Polymarker	**PPM (n, pxa, pya)**
Polymarker 3	**PPM3 (n, pxa, pya, pza)**
Text	**PTX (px, py, chars)**
	PTXS (px, py, lstr, chars)
Text 3	**PTX3 (px, py, pz, tdx, tdy, tdz, chars)**
	PTX3S (px, py, pz tdx, tdy, tdz, lstr, chars)

Attribute Specification

Add Names to Set	**PADS (n, namset)**
Remove Names from Set	**PRES (n, namset)**
Set Annotation Style	**PSANS (astyle)**
Set Annotation Text Alignment	**PSATAL (atalh, atalv,)**
Set Annotation Text Character Height	**PSATCH (atchh)**
Set Annotation Text Character Up Vector	**PSATCU (atchux, atchuy)**
Set Annotation Text Path	**PSATP (atp)**
Set Character Expansion Factor	**PSCHXP (chxp)**
Set Character Height	**PSCHH (chh)**
Set Character Spacing	**PSCHSP (chsp)**
Set Character Up Vector	**PSCHUP (chux, chuy)**
Set Edge Color Index	**PSEDCI (coli)**
Set Edge Flag	**PSEDFG (edflag)**
Set Edge Index	**PSEDI (edi)**
Set Edgetype	**PSEDT (edtype)**
Set Edgewidth Scale Factor	**PSEWSC (ewidth)**
Set HLHSR Identifier	**PSHRID (hrid)**
Set Individual ASF	**PSIASF (aspcid, asfval)**
Set Interior Color Index	**PSICI (coli)**
Set Interior Index	**PSII (ii)**
Set Interior Style	**PSIS (ints)**
Set Interior Style Index	**PSISI (istyli)**
Set Linetype	**PSLN (ltype)**
Set Linewidth Scale Factor	**PSLWSC (lwidth)**
Set Marker Size Scale Factor	**PSMKSC (mszsf)**
Set Marker Type	**PSMK (mtype)**
Set Pattern Reference Point	**PSPARF (rfx, rfy)**
Set Pattern Reference Point and Vectors	**PSPRPV (rfx, rfy, rfz, (rfvy, rfvz)**
Set Pattern Size	**PSPA szx, szy)**
Set Pick Identifier	**PSPKID (pkid)**
Set Polyline Color Index	**PSPLCI (coli)**
Set Polyline Index	**PSPLI (pli)**
Set Polymarker Color Index	**PSPMCI (coli)**
Set Polymarker Index	**PSPMI (pmi)**
Set Text Alignment	**PSTXAL (txalh, txalv)**
Set Text Color Index	**PSTXCI (coli)**
Set Text Font	**PSTXFN (font)**
Set Text Index	**PSTXI (txi)**
Set Text Path	**PSTXP (txp)**

Set Text Precision	**PSTXPR (prec)**
Set View Index	**PSVWI (viewi)**

Miscellaneous Structure Element Subroutines

Application Data	**PAP (ldr, datrec)**
Execute Structure	**PEXST (strid)**
Generalized Structure Element	**PGSE (gseid, ldr, datrec)**

Structure Operation Subroutines

Change Structure Identifier	**PCSTID (oldsid, newsid)**
Change Structure Identifier and References	**PCSTIR (oldsid, newsid)**
Change Structure References	**PCSTRF (oldsid, newsid)**
Close Structure	**PCLST**
Copy All Elements from Structure	**PCELST (strid)**
Delete All Structures	**PDAS**
Delete Element	**PDEL**
Delete Element Range	**PDELRA (ep1, ep2)**
Delete Elements Between Labels	**PDELLB (label1, label2)**
Delete Structure	**PDST (strid)**
Delete Structure Network	**PDSN (strid, refhnf)**.
Empty Structure	**PEMST (strid)**
Label	**PLB (label)**
Offset Element Pointer	**POSEP (epo)**
Open Structure	**POPST (strid)**
Set Edit Mode	**PSEDM (editmo)**
Set Element Pointer	**PSEP (ep)**
Set Element Pointer at Label	**PSEPLB (ep)**

Workstation Table Settings

Set Color Model	**PSCMD (wkid, cmodel)**
Set Color Representation	**PSCR (wkid, ci, nccs, cspec)**
Set Edge Representation	**PSEDR (wkid, edi, edflag, edtype, ewidth, coli)**
Set Highlighting Filter	**PSHLFT (wkid, isn, is, esn, es)**
Set HLHSR Mode	**PSHRM (wkid, hrm)**
Set Interior Representation	**PSIR (wkid, ii, ints, styli, coli)**
Set Invisibility Filter	**PSIVFT (wkid, isn, is, esn, es)**
Set Pattern Representation	**PSPAR (wkid, pai, dimx, dimy, isc, isr, dx, dy, colia)**
Set Polyline Representation	**PSPLR (wkid, pli, ltype, lwidth, coli)**
Set Polymarker Representation	**PSPMR (wkid, pmi, mtype, mszsf, coli)**
Set Text Representation	**PSTXR (wkid, txi, font, prec, chxp, chsp, coli)**
Set View Representation	**PSVWR (wkid, viewi, vwormt, vwmpmt, vwcplm, xyclpi)**
Set View Representation 3	**PSVWR3 (wkid, viewi, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Set View Transformation Input Priority	**PSVTIP (wkid, viewi, rfvwix, relpri)**

Structure Display Subroutines

Post Structure	**PPOST (wkid, strid, priort)**
Unpost All Structures	**PUPAST (wkid)**
Unpost Structure	**PUPOST (wkid, strid)**

Structure Archiving Subroutines

Archive All Structures	**PARAST (afid)**
Archive Structure Networks	**PARSN (afid, n, lstrid)**
Archive Structures	**PARST (afid, n, lstrid)**
Close Archive File	**PCLARF (afid)**
Delete All Structures from Archive	**PDASAR (afid)**
Delete Structure Networks from Archive	**PDSNAR (afid, n, lstrid)**
Delete Structures from Archive	**PDSTAR (afid, n, lstrid)**
Open Archive File	**POPARF (afid, arcfil)**
Retrieve All Structures	**PRAST (afid)**
Retrieve Paths to Ancestors	**PREPAN (afid, strid, pthord, pthdep, ipthsz, n, ol, apthsz, paths)**
Retrieve Paths to Descendants	**PREPDE (afid, strid, pthord, pthdep, ipthsz, n, ol, apthsz, paths)**
Retrieve Structure Identifiers	**PRSID (afid, ilsize, n, lstrid)**
Retrieve Structure Networks	**PRESN (afid n, lstrid)**
Retrieve Structures	**PREST (afid, n, lstrid)**
Set Conflict Resolution	**PSCNRS (arccr, retcr)**

Transformation Subroutines

Restore Modeling Clipping Volume	**PRMCV**
Set Global Transformation	**PSGMT (xfrmt)**
Set Global Transformation 3	**PSGMT3 (xfrmt)**
Set Local Transformation	**PSLMT (xfrmt, ctype)**
Set Local Transformation 3	**PSLMT3 (xfrmt, ctype)**
Set Modeling Clipping Indicator	**PSMCLI (MCLIPI)**
Set Modeling Clipping Volume	**PSMCV (op,nhalfs,halfsp)**
Set Modeling Clipping Volume 3	**PSMCV3 (op,nhalfs,halfsp)**
Set Workstation Viewport	**PSWKV (wkid, xmin, xmax, ymin, ymax)**
Set Workstation Viewport 3	**PSWKV3 (wkid, wkvp)**
Set Workstation Window	**PSWKW (wkid, xmin, xmax, ymin, ymax)**
Set Workstation Window 3	**PSWKW3 (wkid, wkwn)**

Input Subroutines

Await Event	**PWAIT (tout, wkid, icl, idnr)**
Flush Device Events	**PFLUSH (wkid, icl, idnr)**
Get Choice	**PGTCH (stat, chnr)**
Get Locator	**PGTLC (viewi, lpx, lpy)**
Get Locator 3	**PGTLC3 (viewi, lpx, lpy, lpz)**
Get Pick	**PGTPK (ippd, stat, ppd, pp)**
Get String	**PGTST (lostr, str)**
	PGTST (lostr, str)
Get Stroke	**PGTSK (n, viewi, np, pxa, pya)**
Get Stroke 3	**PGTSK3 (n, viewi, np, pxa, pya, pza)**

Get Valuator	**PGTVL (val)**
Initialize Choice	**PINCH (wkid, chdnr, istat, ichnr, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Choice 3	**PINCH3 (wkid, chdnr, istat, ichnr, pet, evol, ldr, datrec)**
Initialize Locator	**PINLC (wkid, lcdnr, iviewi, ipx, ipy, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Locator 3	**PINLC3 (wkid, lcdnr, iviewi, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Initialize Pick	**PINPK (wkid, pkdnr, istat, ippd, pp, pet, xmin, xmax, ymin, ymax, ldr, datrec, ppordr)**
Initialize Pick 3	**PINPK3 (wkid, pkdnr, istat, ippd, pp, pet, evol, ldr, datrec, ppordr)**
Initialize String	**PINST (wkid, stdnr, lstr, istr, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
	PINST (wkid, stdnr, lstr, istr, pet, xmin, xmax, ymin, ymax, ldr, datrec)
Initialize String 3	**PINST3 (wkid, stdnr, lstr, istr, pet, evol, ldr, datrec)**
	PINST3 (wkid, stdnr, lstr, istr, pet, evol, ldr, datrec)
Initialize Stroke	**PINSK (wkid, skdnr, iviewi, n, ipx, ipy, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Stroke 3	**PINSK3 (wkid, skdnr, iviewi, n, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Initialize Valuator	**PINVL (wkid, vldnr, ival, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Valuator 3	**PINVL3 (wkid, vldnr, ival, pet, evol, ldr, datrec)**
Request Choice	**PRQCH (wkid, chdnr, stat, chnr)**
Request Locator	**PRQLC (wkid, lcdnr, stat, viewi, px, py)**
Request Locator 3	**PRQLC3 (wkid, lcdnr, stat, viewi, px, py, pz)**
Request Pick	**PRQPK (wkid, pkdnr, ippd, stat, ppd, pp)**
Request String	**PRQST (wkid, stdnr, stat, lostr, str)**
	PRQST (wkid, stdnr, stat, lostr, str)
Request Stroke	**PRQSK (wkid, skdnr, n, stat, viewi, np, pxa, pya)**
Request Stroke 3	**PRQSK3 (wkid, skdnr, n, stat, viewi, np, pxa, pya, pza)**
Request Valuator	**PRQVL (wkid, vldnr, stat, val)**
Sample Choice	**PSMCH (wkid, chdnr, stat, chnr)**
Sample Locator	**PSMLC (wkid, lcdnr, viewi, lpx, lpy)**
Sample Locator 3	**PSMLC3 (wkid, lcdnr, viewi, lpx, lpy, lpz)**
Sample Pick	**PSMPK (wkid, pkdnr, ippd, stat, ppd, pp)**
Sample String	**PSMST (wkid, stdnr, lostr, str)**
	PSMST (wkid, stdnr, lostr, str)
Sample Stroke	**PSMSK (wkid, skdnr, n, viewi, np, pxa, pya)**
Sample Stroke 3	**PSMSK3 (wkid, skdnr, n, viewi, np, pxa, pya, pza)**
Sample Valuator	**PSMVL (wkid, vldnr, val)**
Set Choice Mode	**PSCHM (wkid, chdnr, mode, esw)**
Set Locator Mode	**PSLCM (wkid, lcdnr, mode, esw)**
Set Pick Filter	**PSPKFT (wkid, pkdnr, isn, is, esn, es)**
Set Pick Mode	**PSPKM (wkid, pkdnr, mode, esw)**
Set String Mode	**PSSTM (wkid, stdnr, mode, esw)**
Set Stroke Mode	**PSSKM (wkid, skdnr, mode, esw)**
Set Valuator Mode	**PSVLM (wkid, vldnr, mode, esw)**

Utility Subroutines

Build Transformation Matrix	**PBLTM (x0, y0, dx, dy, phi, fx, fy, errind, xfrmt)**
Build Transformation Matrix 3	**PBLTM3 (x0, y0, z0, dx, dy, dz, phix, phiy, phiz, fx, fy, fz, errind, xfrmt)**
Compose Matrix	**PCOM (xfrmta, xfrmtb, errind, xfrmto)**
Compose Matrix 3	**PCOM3 (xfrmta, xfrmtb, errind, xfrmto)**
Compose Transformation Matrix	**PCOTM (xfrmti, x0, y0, dx, dy, phi, fx, fy, errind, xfrmto)**
Compose Transformation Matrix 3	**PCOTM3 (xfrmti, x0, y0, z0, dx, dy, dz, phix, phiy, phiz, fx, fy, fz, errind, xfrmto)**

Evaluate View Mapping Matrix	**PEVMM (vwwnlm, pjvplm, errind, vwmpmt)**
Evaluate View Mapping Matrix 3	**PEVMM3 (vwwnlm, pjvplm, pjtype, pjrx, pjry, pjrz, vpld, bpld, fpld, errind, vwmpmt)**
Evaluate View Orientation Matrix	**PEVOM (vwrx, vwry, vupx, vupy, errind, vwormt)**
Evaluate View Orientation Matrix 3	**PEVOM3 (vwrx, vwry, vwrz, vpnx, vpny, vpnz, vupx, vupy, vupz, errind, vwormt)**
Pack Data Record	**PPREC (il, ia, rl, ra, sl, lstr, str, mldr, errind, ldr, datrec)**
	PPREC (il, ia, rl, ra, sl, lstr, str, mldr, errind, ldr, datrec)
Rotate	**PRO (rotang, errind, xfrmt)**
Rotate X	**PROX (rotang, errind, xfrmt)**
Rotate Y	**PROY (rotang, errind, xfrmt)**
Rotate Z	**PROZ (rotang, errind, xfrmt)**
Scale	**PSC (fx, fy, errind, xfrmt)**
Scale 3	**PSC3 (fx, fy, fz, errind, xfrmt)**
Transform Point	**PTP (xi, yi, xfrmt, errind, xo, yo)**
Transform Point 3	**PTP3 (xi, yi, zi, xfrmt, errind, xo, yo, zo)**
Translate	**PTR (dx, dy, errind, xfrmt)**
Translate 3	**PTR3 (dx, dy, dz, errind, xfrmt)**
Unpack Data Record	**PUREC (ldr, datrec, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PUREC (ldr, datrec, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)

Error Control Subroutines

Emergency Close PHIGS	**PECLPH**
Error Handling	**PERHND (errnr, fctid, errfil)**
Error Logging	**PERLOG (errnr, fctid, errfil)**
Set Error Handling Mode	**PSERHM (erhm)**

Special Interface Subroutines

Escape	**PESC (fctid, lidr, idr, mlodr, lodr, odr)**

Inquire Subroutines

Element Search	**PELS (strid, strtep, srcdir, eisn, eis, eesn, ees, errind, status, fndep)**
Inquire All Conflicting Structures	**PQCST NOWRAP(afid, n, errind, ol, ostrid)**
Inquire Annotation Facilities	**PQANF (wtype, n, errind, nas, as, nchh, minchh, maxchh)**
Inquire Archive Files	**PQARF (n, errind, number afid, arcfil)**
Inquire Archive State Value	**PQARS (arsta)**
Inquire Choice Device State	**PQCHS (wkid, chdnr, mldr, errind, mode, esw, istat, ichnr, pet, earea, ldr, datrec)**
Inquire Choice Device State 3	**PQCHS3 (wkid, chdnr, mldr, errind, mode, esw, istat, ichnr, pet, evol, ldr, datrec)**
Inquire Color Facilities	**PQCF (wtype, errind, ncoli, cola, npci, cc)**
Inquire Color Model	**PQCMD (wkid, errind, cmodel)**
Inquire Color Model Facilities	**PQCMDF (wtype, n, errind, ol, cmod, dfcmod)**
Inquire Color Representation	**PQCR (wkid, coli, ccsbsz, type, errind, ol, cspec)**
Inquire Conflicting Structures in Network	**PQCSTN (afid, strid, snsrc, n, errind, ol, ostrid)**
Inquire Conflict Resolution	**PQCNRS (errind, arccr, retcr)**

Inquire Current Element Content	**PQCECO (iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PQCECO (iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)
Inquire Current Element Type and Size	**PQCETS (erringd, eltype, il, rl, sl)**
Inquire Default Choice Device Data	**PQDCH (wtype, devno, n, mldr, errind, malt, ol, pet, earea, ldr, datrec)**
Inquire Default Choice Device Data 3	**PQDCH3 (wtype, devno, n, mldr, errind, malt, ol, pet, evol, ldr, datrec)**
Inquire Default Display Update State	**PQDDUS (wtype, errind, defmod, modmod)**
Inquire Default Locator Device Data	**PQDLC (wtype, devno, n, mldr, errind, dpx, dpy, ol, pet, earea, ldr, datrec)**
Inquire Default Locator Device Data 3	**PQDLC3 (wtype, devno, n, mldr, errind, dpx, dpy, dpz, ol, pet, evol, ldr, datrec)**
Inquire Default Pick Device Data	**PQDPK (wtype, devno, n, mldr, errind, ol, pet, earea, ldr, datrec)**
Inquire Default Pick Device Data 3	**PQDPK3 (wtype, devno, n, mldr, errind, ol, pet, evol, ldr, datrec)**
Inquire Default String Device Data	**PQDST (wtype, devno, n, mldr, errind, mbuff, ol, pet, earea, ldr, datrec)**
Inquire Default String Device Data 3	**PQDST3 (wtype, devno, n, mldr, errind, mbuff, ol, pet, evol, ldr, datrec)**
Inquire Default Stroke Device Data	**PQDSK (wtype, devno, n, mldr, errind, mbuff, ol, pet, earea, ldr, datrec)**
Inquire Default Stroke Device Data 3	**PQDSK3 (wtype, devno, n, mldr, errind, mbuff, ol, pet, evol, ldr, datrec)**
Inquire Default Valuator Device Data	**PQDVL (wtype, devno, n, mldr, errind, dval, ol, pet, earea, ldr, datrec)**
Inquire Default Valuator Device Data 3	**PQDVL3 (wtype, devno, n, mldr, errind, dval, ol, pet, evol, ldr, datrec)**
Inquire Display Space Size 3	**PQDSP3 (wtype, errind, dcunit, dx, dy, dz, rx, ry, rz)**
Inquire Display Update State	**PQDUS (wkid, errind, defmod, modmod, dempty, stofvr)**
Inquire Dynamics of Structures	**PQDSTR (wtype, errind, strcon, post, unpost, delete, refmod)**
Inquire Dynamics of Workstation Attributes	**PQDSWA (wtype, errind, plbun, pmbun, txbun, inbun, edbun, parep, colrep, vwrep, wktr, hlfltr, infltr, hlhsr)**
Inquire Edge Facilities	**PQEDF (wtype, n, errind, nedt, edt, nedw, nomedw, redwmn, npedi)**
Inquire Edge Representation	**PQEDR (wkid, edi, type, errind, edflag, edtype, ewidth, coli)**
Inquire Edit Mode	**PQEDM (errind, editmo)**
Inquire Element Content	**PQECO (strid, elenum, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PQECO (strid, elenum, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)
Inquire Element Pointer	**PQEP (errind, ep)**
Inquire Element Type and Size	**PQETS (strid, elenum, errind, eltype, il, rl, sl)**
Inquire Error Handling Mode	**PQERHM (errind, erhm)**
Inquire Generalized Drawing Primitive	**PQGDP (wtype, gdp, errind, nbnd, bndl)**
Inquire Generalized Drawing Primitive 3	**PQGDP3 (wtype, gdp, errind, nbnd, bndl)**
Inquire Generalized Structure Element Facilities	**PQGSEF (n, errind, ol, gseid, wsdind)**
Inquire Highlighting Filter	**PQHLFT (wkid, isbsz, esbsz, errind, isn, is, esn, es)**
Inquire HLHSR Identifier Facilities	**PQHRIF (wtype, ni, errind, nhrid, hrid)**
Inquire HLHSR Mode	**PQHRM (wkid, errind, hupd, chrm, rhrm)**
Inquire HLHSR Mode Facilities	**PQHRMF (wtype, nm, errind, nhrmd, hrmd)**
Inquire Display Space Size	**PQDSP (wtype, errind, dcunit, dx, dy, rx, ry)**
Inquire Input Queue Overflow	**PQIQOV (errind, wkid, icl, idn)**
Inquire Interior Facilities	**PQIF (wtype, ni, nh, errind, nis, is, nhs, hs, npfai)**
Inquire Interior Representation	**PQIR (wkid, ii, type, errind, ints, istyli, coli)**
Inquire Invisibility Filter	**PQIVFT (wkid, isbsz, esbsz, errind, isn, is, esn, es)**
Inquire List of Available Generalized Drawing Primitives	**PQEGDP (wtype, n, errind, ngdp, gdpl)**
Inquire List of Available Generalized Drawing Primitives 3	**PQEGD3 (wtype, n, errind, ngdp, gdpl)**
Inquire List of Available Generalized Structure Elements	**PQEGSE (wtype, n, errind, ngse, gsel)**
Inquire List of Available Workstation Types	**PQEWK (n, errind, number, wktyp)**
Inquire List of Color Indices	**PQECI (wkid, n, errind, ol, coli)**
Inquire List of Edge Indices	**PQEEDI (wkid, n, errind, ol, edi)**

Inquire List of Interior Indices	**PQEII (wkid, n, errind, ol, ii)**
Inquire List of Pattern Indices	**PQEPAI (wkid, n, errind, ol, pai)**
Inquire List of Polyline Indices	**PQEPLI (wkid, n, errind, ol, pli)**
Inquire List of Polymarker Indicies	**PQEPMI (wkid, n, errind, ol, pmi)**
Inquire List of Text Indices	**PQETXI (wkid, n, errind, ol, txi)**
Inquire List of View Indices	**PQEVWI (wkid, n, errind, nvwix, viewi)**
Inquire Locator Device State	**PQLCS (wkid, lcdnr, type, mldr, errind, mode, esw, iviewi, ipx, ipy, pet, earea, ldr, datrec)**
Inquire Locator Device State 3	**PQLCS3 (wkid, lcdnr, type, mldr, errind, mode, esw, iviewi, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Inquire Modeling Clipping Facilities	**PQMCLF (n,errind, ndpmcv,ol,mclpop)**
Inquire More Simultaneous Events	**PQSIM (errind, flag)**
Inquire Number of Available Logical Input Devices	**PQLI (wtype, errind, nlcd, nskd, nvld, nchd, npkd, nstd)**
Inquire Number of Display Priorities Supported	**PQDP (wtype, errind, nspsup)**
Inquire Open Structure	**PQOPST (errind, stype, strid)**
Inquire Paths to Ancestors	**PQPAN (strid, pthord, pthdep, ipthsz, n, errind, ol, apthsz, paths)**
Inquire Paths to Descendants	**PQPDE (strid, pthord, pthdep, ipthsz, n, errind, ol, apthsz, paths)**
Inquire Pattern Facilities	**PQPAF (wtype, errind, nppai)**
Inquire Pattern Representation	**PQPAR (wkid, pai, type, dimx, dimy, errind, dx, dy, colia)**
Inquire PHIGS Facilities	**PQPHF (ncs, errind, simopw, simopa, namesn olcs, cs, nfln, ifln)**
Inquire Pick Device State	**PQPKS (wkid, pkdnr, type, mldr, ipissz, ipessz, ippsz, errind, mode, esw, pissz, pins, pessz, pes, istat, ppd, pp, pet, earea, ldr, datrec, ppordr)**
Inquire Pick Device State 3	**PQPKS3 (wkid, pkdnr, type, mldr, ipissz, ipessz, ippsz, errind, mode, esw, pissz, pins, pessz, pes, istat, ppd, pp, pet, evol, ldr, datrec, ppordr)**
Inquire Polyline Facilities	**PQPLF (wtype, n, errind, nlt, lt, nlw, nomlw, rlwmin, rlwmax, nppli)**
Inquire Polyline Representation	**PQPLR (wkid, pli, type, errind, ltype, lwidth, coli)**
Inquire Polymarker Facilities	**PQPMF (wtype, n, errind, nmt, mt, nms, nomms, rmsmin, rmsmax, nppmi)**
Inquire Polymarker Representation	**PQPMR (wkid, pmi, type, errind, mtype, mszsf, coli)**
Inquire Posted Structures	**PQPOST (wkid, n, errind, number, strid, priort)**
Inquire Predefined Color Representation	**PQPCR (wtype, pci, ccsbsz, errind, ol, cspec)**
Inquire Predefined Edge Representation	**PQPEDR (wtype, pedi, errind, edflag, edtype, ewidth, coli)**
Inquire Predefined Interior Representation	**PQPIR (wtype, pii, errind, style, stylid, coli)**
Inquire Predefined Pattern Representation	**PQPPAR (wtype, ppai, dimx, dimy, errind, dx, dy, colia)**
Inquire Predefined Polyline Representation	**PQPPLR (wtype, pli, errind, ltype, lwidth, coli)**
Inquire Predefined Polymarker Representation	**PQPPMR (wtype, pmi, errind, mtype, mszsf, coli)**
Inquire Predefined Text Representation	**PQPTXR (wtype, ptxi, errind, font, prec, chxp, chsp, coli)**
Inquire Predefined View Representation	**PQPVWR (wtype, pvwi, errind, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Inquire Set of Open Workstations	**PQOPWK (n, errind, ol, wkid)**
Inquire Set of Workstations to Which Posted	**PQWKPO (strid, n, errind, ol, wkid)**
Inquire String Device State	**PQSTS (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, earea, ldr, datrec)**
	PQSTS (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, earea, ldr, datrec)

Inquire String Device State 3	**PQSTS3 (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, evol, ldr, datrec)**
	PQSTS3 (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, evol, ldr, datrec)
Inquire Stroke Device State	**PQSKS (wkid, skdnr, type, n, mldr, errind, mode, esw, iviewi, np, ipxa, ipya, pet, earea, ldr, datrec)**
Inquire Stroke Device State 3	**PQSKS (wkid, skdnr, type, n, mldr, errind, mode, esw, iviewi, np, ipxa, ipya, ipza, pet, evol, ldr, datrec)**
Inquire Structure Identifiers	**PQSID (n, errind, number, strid)**
Inquire Structure State Value	**PQSTRS (strsta)**
Inquire Structure Status	**PQSTST (strid, errind, strsti)**
Inquire System State Value	**PQSYS (syssta)**
Inquire Text Extent	**PQTXX (wktype, font, chxp, chsp, chh, txp, txalh, txalv, str, errind, txexrx, txexry, copx, copy)**
	PQTXXS (wktype, font, chxp, chsp, chh, txp, txalh, txalv, lstr, str, errind, txexrx, txexry, copx, copy)
Inquire Text Facilities	**PQTXF (wtype, n, errind, nfpp, font, prec, nchh, minchh, maxchh, nchx, minchx, maxchx, nptxi)**
Inquire Text Representation	**PQTXR (wkid, txi, type, errind, font, prec, chxp, chsp, coli)**
Inquire Valuator Device State	**PQVLS (wkid, vldnr, mldr, errind, mode, esw, ival, pet, earea, ldr, datrec)**
Inquire Valuator Device State 3	**PQVLS3 (wkid, vldnr, mldr, errind, mode, esw, ival, pet, evol, ldr, datrec)**
Inquire View Facilities	**PQVWF (wtype, errind, npvwi)**
Inquire View Representation	**PQVWR (wkid, viewi, curq, errind, vwupd, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Inquire Workstation Category	**PQWKCA (wtype, errind, wkcat)**
Inquire Workstation Classification	**PQWKCL (wtype, errind, vrtype)**
Inquire Workstation Connection and Type	**PQWKC (wkid, errind, conid, wtype)**
Inquire Workstation State Table Lengths	**PQWKSL (wtype, errind, mplbte, mpmbte, mtxbte, minbte, medbte, mpai, mcoli, vwtbi)**
Inquire Workstation State Value	**PQWKST (wksta)**
Inquire Workstation Transformation	**PQWKT (wkid, errind, tus, rwindo, cwindo, rviewp, cviewp)**
Inquire Workstation Transformation 3	**PQWKT3 (wkid, errind, tus, rwindo, cwindo, rviewp, cviewp)**

Chapter 5. Alphabetical Listing of Subroutines for FORTRAN Binding

Note: When two mnemonics are listed after a given subroutine name, the first is FORTRAN and the one following is FORTRAN Subset.

Add Names to Set	**PADS (n, namset)**
Annotation Text Relative	**PATR (rpx, rpy, apx, apy, chars)**
	PATRS (rpx, rpy, apx, apy, lstr, chars)
Annotation Text Relative 3	**PATR3 (rpx, rpy, rpz, apx, apy, apz, chars)**
	PATR3S (rpx, rpy, rpz, apx, apy, apz, lstr, chars)
Application Data	**PAP (ldr, datrec)**
Archive All Structures	**PARAST (afid)**
Archive Structure Networks	**PARSN (afid, n, lstrid)**
Archive Structures	**PARST (afid, n, lstrid)**
Await Event	**PWAIT (tout, wkid, icl, idnr)**
Build Transformation Matrix	**PBLTM (x0, y0, dx, dy, phi, fx, fy, errind, xfrmt)**
Build Transformation Matrix 3	**PBLTM3 (x0, y0, z0, dx, dy, dz, phix, phiy, phiz, fx, fy, fz, errind, xfrmt)**
Cell Array	**PCA (px, py, qx, qy, dimx, dimy, isc, isr, dx, dy, colia)**
Cell Array 3	**PCA3 (cpxa, cpya, cpza, dimx, dimy, isc, isr, dx, dy, colia)**
Change Structure Identifier	**PCSTID (oldsid, newsid)**
Change Structure Identifier and References	**PCSTIR (oldsid, newsid)**
Change Structure References	**PCSTRF (oldsid, newsid)**
Close Archive File	**PCLARF (afid)**
Close PHIGS	**PCLPH**
Close Structure	**PCLST**
Close Workstation	**PCLWK (wkid)**
Compose Matrix	**PCOM (xfrmta, xfrmtb, errind, xfrmto)**
Compose Matrix 3	**PCOM3 (xfrmta, xfrmtb, errind, xfrmto)**
Compose Transformation Matrix	**PCOTM (xfrmti, x0, y0, dx, dy, phi, fx, fy, errind, xfrmto)**
Compose Transformation Matrix 3	**PCOTM3 (xfrmti, x0, y0, z0, dx, dy, dz, phix, phiy, phiz, fx, fy, fz, errind, xfrmto)**
Copy All Elements from Structure	**PCELST (strid)**
Delete All Structures	**PDAS**
Delete All Structures from Archive	**PDASAR (afid)**
Delete Element	**PDEL**
Delete Element Range	**PDELRA (ep1, ep2)**
Delete Elements Between Labels	**PDELLB (label1, label2)**
Delete Structure	**PDST (strid)**
Delete Structure Network	**PDSN (strid, refhnf)**.
Delete Structure Networks from Archive	**PDSNAR (afid, n, lstrid)**
Delete Structures from Archive	**PDSTAR (afid, n, lstrid)**
Element Search	**PELS (strid, strtep, srcdir, eisn, eis, eesn, ees, errind, status, fndep)** ELESEA.
Emergency Close PHIGS	**PECLPH**
Empty Structure	**PEMST (strid)**
Error Handling	**PERHND (errnr, fctid, errfil)**
Error Logging	**PERLOG (errnr, fctid, errfil)**
Escape	**PESC (fctid, lidr, idr, mlodr, lodr, odr)**
Evaluate View Mapping Matrix	**PEVMM (vwwnlm, pjvplm, errind, vwmpmt)**

Evaluate View Mapping Matrix 3	**PEVMM3 (vwwnlm, pjvplm, pjtype, pjrx, pjry, pjrz, vpld, bpld, fpld, errind, vwmpmt)**
Evaluate View Orientation Matrix	**PEVOM (vwrx, vwry, vupx, vupy, errind, vwormt)**
Evaluate View Orientation Matrix 3	**PEVOM3 (vwrx, vwry, vwrz, vpnx, vpny, vpnz, vupx, vupy, vupz, errind, vwormt)**
Execute Structure	**PEXST (strid)**
Fill Area	**PFA (n, pxa, pya)**
Fill Area 3	**PFA3 (n, pxa, pya, pza)**
Fill Area Set	**PFAS (npl, ixa, pxa, pya,)**
Fill Area Set 3	**PFAS3 (npl, ixa, pxa, pya, pza)**
Flush Device Events	**PFLUSH (wkid, icl, idnr)**
Generalized Drawing Primitive	**PGDP (n, pxa, pya, primid, ldr, datrec)**
Generalized Drawing Primitive 3	**PGDP3 (n, pxa, pya, pza, primid, ldr, datrec)**
Generalized Structure Element	**PGSE (gseid, ldr, datrec)**
Get Choice	**PGTCH (stat, chnr)**
Get Locator	**PGTLC (viewi, lpx, lpy)**
Get Locator 3	**PGTLC3 (viewi, lpx, lpy, lpz)**
Get Pick	**PGTPK (ippd, stat, ppd, pp)**
Get String	**PGTST (lostr, str)**
	PGTST (lostr, str)
Get Stroke	**PGTSK (n, viewi, np, pxa, pya)**
Get Stroke 3	**PGTSK3 (n, viewi, np, pxa, pya, pza)**
Get Valuator	**PGTVL (val)**
Initialize Choice	**PINCH (wkid, chdnr, istat, ichnr, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Choice 3	**PINCH3 (wkid, chdnr, istat, ichnr, pet, evol, ldr, datrec)**
Initialize Locator	**PINLC (wkid, lcdnr, iviewi, ipx, ipy, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Locator 3	**PINLC3 (wkid, lcdnr, iviewi, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Initialize Pick	**PINPK (wkid, pkdnr, istat, ippd, pp, pet, xmin, xmax, ymin, ymax, ldr, datrec, ppordr)**
Initialize Pick 3	**PINPK3 (wkid, pkdnr, istat, ippd, pp, pet, evol, ldr, datrec, ppordr)**
Initialize String	**PINST (wkid, stdnr, lstr, istr, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
	PINST (wkid, stdnr, lstr, istr, pet, xmin, xmax, ymin, ymax, ldr, datrec)
Initialize String 3	**PINST3 (wkid, stdnr, lstr, istr, pet, evol, ldr, datrec)**
	PINST3 (wkid, stdnr, lstr, istr, pet, evol, ldr, datrec)
Initialize Stroke	**PINSK (wkid, skdnr, iviewi, n, ipx, ipy, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Stroke 3	**PINSK3 (wkid, skdnr, iviewi, n, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Initialize Valuator	**PINVL (wkid, vldnr, ival, pet, xmin, xmax, ymin, ymax, ldr, datrec)**
Initialize Valuator 3	**PINVL3 (wkid, vldnr, ival, pet, evol, ldr, datrec)**
Inquire All Conflicting Structures	**PQCST (afid, n, errind, ol, ostrid)**
Inquire Annotation Facilities	**PQANF (wtype, n, errind, nas, as, nchh, minchh, maxchh)**
Inquire Archive Files	**PQARF (n, errind, number afid, arcfil)**
Inquire Archive State Value	**PQARS (arsta)**
Inquire Choice Device State	**PQCHS (wkid, chdnr, mldr, errind, mode, esw, istat, ichnr, pet, earea, ldr, datrec)**

Inquire Choice Device State 3	**PQCHS3 (wkid, chdnr, mldr, errind, mode, esw, istat, ichnr, pet, evol, ldr, datrec)**
Inquire Color Facilities	**PQCF (wtype, errind, ncoli, cola, npci, cc)**
Inquire Color Model	**PQCMD (wkid, errind, cmodel)**
Inquire Color Model Facilities	**PQCMDF (wtype, n, errind, ol, cmod, dfcmod)**
Inquire Color Representation	**PQCR (wkid, coli, ccsbsz, type, errind, ol, cspec)**
Inquire Conflict Resolution	**PQCNRS (errind, arccr, retcr)**
Inquire Conflicting Structures in Network	**PQCSTN (afid, strid, snsrc, n, errind, ol, ostrid)**
Inquire Current Element Content	**PQCECO (iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PQCECO (iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)
Inquire Current Element Type and Size	**PQCETS (erringd, eltype, il, rl, sl)**
Inquire Default Choice Device Data	**PQDCH (wtype, devno, n, mldr, errind, malt, ol, pet, earea, ldr, datrec)**
Inquire Default Choice Device Data 3	**PQDCH3 (wtype, devno, n, mldr, errind, malt, ol, pet, evol, ldr, datrec)**
Inquire Default Display Update State	**PQDDUS (wtype, errind, defmod, modmod)**
Inquire Default Locator Device Data	**PQDLC (wtype, devno, n, mldr, errind, dpx, dpy, ol, pet, earea, ldr, datrec)**
Inquire Default Locator Device Data 3	**PQDLC3 (wtype, devno, n, mldr, errind, dpx, dpy, dpz, ol, pet, evol, ldr, datrec)**
Inquire Default Pick Device Data	**PQDPK (wtype, devno, n, mldr, errind, ol, pet, earea, ldr, datrec)**
Inquire Default Pick Device Data 3	**PQDPK3 (wtype, devno, n, mldr, errind, ol, pet, evol, ldr, datrec)**
Inquire Default String Device Data	**PQDST (wtype, devno, n, mldr, errind, mbuff, ol, pet, earea, ldr, datrec)**
Inquire Default String Device Data 3	**PQDST3 (wtype, devno, n, mldr, errind, mbuff, ol, pet, evol, ldr, datrec)**
Inquire Default Stroke Device Data	**PQDSK (wtype, devno, n, mldr, errind, mbuff, ol, pet, earea, ldr, datrec)**
Inquire Default Stroke Device Data 3	**PQDSK3 (wtype, devno, n, mldr, errind, mbuff, ol, pet, evol, ldr, datrec)**
Inquire Default Valuator Device Data	**PQDVL (wtype, devno, n, mldr, errind, dval, ol, pet, earea, ldr, datrec)**
Inquire Default Valuator Device Data 3	**PQDVL3 (wtype, devno, n, mldr, errind, dval, ol, pet, evol, ldr, datrec)**
Inquire Display Space Size	**PQDSP (wtype, errind, dcunit, dx, dy, rx, ry)**
Inquire Display Space Size 3	**PQDSP3 (wtype, errind, dcunit, dx, dy, dz, rx, ry, rz)**
Inquire Display Update State	**PQDUS (wkid, errind, defmod, modmod, dempty, stofvr)**
Inquire Dynamics of Structures	**PQDSTR (wtype, errind, strcon, post, unpost, delete, refmod)**
Inquire Dynamics of Workstation Attributes	**PQDSWA (wtype, errind, plbun, pmbun, txbun, inbun, edbun, parep, colrep, vwrep, wktr, hlfltr, infltr, hlhsr)**
Inquire Edge Facilities	**PQEDF (wtype, n, errind, nedt, edt, nedw, nomedw, redwmn, npedi)**
Inquire Edge Representation	**PQEDR (wkid, edi, type, errind, edflag, edtype, ewidth, coli)**
Inquire Edit Mode	**PQEDM (errind, editmo)**
Inquire Element Content	**PQECO (strid, elenum, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PQECO (strid, elenum, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)
Inquire Element Pointer	**PQEP (errind, ep)**
Inquire Element Type and Size	**PQETS (strid, elenum, errind, eltype, il, rl, sl)**

Inquire Error Handling Mode	**PQERHM (errind, erhm)**
Inquire Generalized Drawing Primitive	**PQGDP (wtype, gdp, errind, nbnd, bndl)**
Inquire Generalized Drawing Primitive 3	**PQGDP3 (wtype, gdp, errind, nbnd, bndl)**
Inquire Generalized Structure Element Facilities	**PQGSEF (n, errind, ol, gseid, wsdind)**
Inquire Highlighting Filter	**PQHLFT (wkid, isbsz, esbsz, errind, isn, is, esn, es)**
Inquire HLHSR Identifier Facilities	**PQHRIF (wtype, ni, errind, nhrid, hrid)**
Inquire HLHSR Mode	**PQHRM (wkid, errind, hupd, chrm, rhrm)**
Inquire HLHSR Mode Facilities	**PQHRMF (wtype, nm, errind, nhrmd, hrmd)**
Inquire Input Queue Overflow	**PQIQOV (errind, wkid, icl, idn)**
Inquire Interior Facilities	**PQIF (wtype, ni, nh, errind, nis, is, nhs, hs, npfai)**
Inquire Interior Representation	**PQIR (wkid, ii, type, errind, ints, istyli, coli)**
Inquire Invisibility Filter	**PQIVFT (wkid, isbsz, esbsz, errind, isn, is, esn, es)**
Inquire List of Available Generalized Drawing Primitives	**PQEGDP (wtype, n, errind, ngdp, gdpl)**
Inquire List of Available Generalized Drawing Primitives 3	**PQEGD3 (wtype, n, errind, ngdp, gdpl)**
Inquire List of Available Generalized Structure Elements	**PQEGSE (wtype, n, errind, ngse, gsel)**
Inquire List of Available Workstation Types	**PQEWK (n, errind, number, wktyp)**
Inquire List of Color Indices	**PQECI (wkid, n, errind, ol, coli)**
Inquire List of Edge Indices	**PQEEDI (wkid, n, errind, ol, edi)**
Inquire List of Interior Indices	**PQEII (wkid, n, errind, ol, ii)**
Inquire List of Pattern Indices	**PQEPAI (wkid, n, errind, ol, pai)**
Inquire List of Polyline Indices	**PQEPLI (wkid, n, errind, ol, pli)**
Inquire List of Polymarker Indicies	**PQEPMI (wkid, n, errind, ol, pmi)**
Inquire List of Text Indices	**PQETXI (wkid, n, errind, ol, txi)**
Inquire List of View Indices	**PQEVWI (wkid, n, errind, nvwix, viewi)**
Inquire Locator Device State	**PQLCS (wkid, lcdnr, type, mldr, errind, mode, esw, iviewi, ipx, ipy, pet, earea, ldr, datrec)**
Inquire Locator Device State 3	**PQLCS3 (wkid, lcdnr, type, mldr, errind, mode, esw, iviewi, ipx, ipy, ipz, pet, evol, ldr, datrec)**
Inquire Modeling Clipping Facilities	**PQMCLF (n,errind, ndpmcv,ol,mclpop)**
Inquire More Simultaneous Events	**PQSIM (errind, flag)**
Inquire Number of Available Logical Input Devices	**PQLI (wtype, errind, nlcd, nskd, nvld, nchd, npkd, nstd)**
Inquire Number of Display Priorities Supported	**PQDP (wtype, errind, nspsup)**
Inquire Open Structure	**PQOPST (errind, stype, strid)**
Inquire Paths to Ancestors	**PQPAN (strid, pthord, pthdep, ipthsz, n, errind, ol, apthsz, paths)**
Inquire Paths to Descendants	**PQPDE (strid, pthord, pthdep, ipthsz, n, errind, ol, apthsz, paths)**
Inquire Pattern Facilities	**PQPAF (wtype, errind, nppai)**
Inquire Pattern Representation	**PQPAR (wkid, pai, type, dimx, dimy, errind, dx, dy, colia)**
Inquire PHIGS Facilities	**PQPHF (ncs, errind, simopw, simopa, namesn olcs, cs, nfln, ifln)**
Inquire Pick Device State	**PQPKS (wkid, pkdnr, type, mldr, ipissz, ipessz, ippsz, errind, mode, esw, pissz, pins, pessz, pes, istat, ppd, pp, pet, earea, ldr, datrec, ppordr)**
Inquire Pick Device State 3	**PQPKS3 (wkid, pkdnr, type, mldr, ipissz, ipessz, ippsz, errind, mode, esw, pissz, pins, pessz, pes, istat, ppd, pp, pet, evol, ldr, datrec, ppordr)**
Inquire Polyline Facilities	**PQPLF (wtype, n, errind, nlt, lt, nlw, nomlw, rlwmin, rlwmax, nppli)**
Inquire Polyline Representation	**PQPLR (wkid, pli, type, errind, ltype, lwidth, coli)**
Inquire Polymarker Facilities	**PQPMF (wtype, n, errind, nmt, mt, nms, nomms, rmsmin, rmsmax, nppmi)**
Inquire Polymarker Representation	**PQPMR (wkid, pmi, type, errind, mtype, mszsf, coli)**
Inquire Posted Structures	**PQPOST (wkid, n, errind, number, strid, priort)**
Inquire Predefined Color Representation	**PQPCR (wtype, pci, ccsbsz, errind, ol, cspec)**

Inquire Predefined Edge Representation	**PQPEDR (wtype, pedi, errind, edflag, edtype, ewidth, coli)**
Inquire Predefined Interior Representation	**PQPIR (wtype, pii, errind, style, stylid, coli)**
Inquire Predefined Pattern Representation	**PQPPAR (wtype, ppai, dimx, dimy, errind, dx, dy, colia)**
Inquire Predefined Polyline Representation	**PQPPLR (wtype, pli, errind, ltype, lwidth, coli)**
Inquire Predefined Polymarker Representation	**PQPPMR (wtype, pmi, errind, mtype, mszsf, coli)**
Inquire Predefined Text Representation	**PQPTXR (wtype, ptxi, errind, font, prec, chxp, chsp, coli)**
Inquire Predefined View Representation	**PQPVWR (wtype, pvwi, errind, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Inquire Set of Open Workstations	**PQOPWK (n, errind, ol, wkid)**
Inquire Set of Workstations to Which Posted	**PQWKPO (strid, n, errind, ol, wkid)**
Inquire String Device State	**PQSTS (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, earea, ldr, datrec)**
	PQSTS (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, earea, ldr, datrec)
Inquire String Device State 3	**PQSTS3 (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, evol, ldr, datrec)**
	PQSTS3 (wkid, stdnr, mldr, errind, mode, esw, lostr, istr, pet, evol, ldr, datrec)
Inquire Stroke Device State	**PQSKS (wkid, skdnr, type, n, mldr, errind, mode, esw, iviewi, np, ipxa, ipya, pet, earea, ldr, datrec)**
Inquire Stroke Device State 3	**PQSKS (wkid, skdnr, type, n, mldr, errind, mode, esw, iviewi, np, ipxa, ipya, ipza, pet, evol, ldr, datrec)**
Inquire Structure Identifiers	**PQSID (n, errind, number, strid)**
Inquire Structure State Value	**PQSTRS (strsta)**
Inquire Structure Status	**PQSTST (strid, errind, strsti)**
Inquire System State Value	**PQSYS (syssta)**
Inquire Text Extent	**PQTXX (wktype, font, chxp, chsp, chh, txp, txalh, txalv, str, errind, txexrx, txexry, copx, copy)**
	PQTXXS (wktype, font, chxp, chsp, chh, txp, txalh, txalv, lstr, str, errind, txexrx, txexry, copx, copy)
Inquire Text Facilities	**PQTXF (wtype, n, errind, nfpp, font, prec, nchh, minchh, maxchh, nchx, minchx, maxchx, nptxi)**
Inquire Text Representation	**PQTXR (wkid, txi, type, errind, font, prec, chxp, chsp, coli)**
Inquire Valuator Device State	**PQVLS (wkid, vldnr, mldr, errind, mode, esw, ival, pet, earea, ldr, datrec)**
Inquire Valuator Device State 3	**PQVLS3 (wkid, vldnr, mldr, errind, mode, esw, ival, pet, evol, ldr, datrec)**
Inquire View Facilities	**PQVWF (wtype, errind, npvwi)**
Inquire View Representation	**PQVWR (wkid, viewi, curq, errind, vwupd, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Inquire Workstation Category	**PQWKCA (wtype, errind, wkcat)**
Inquire Workstation Classification	**PQWKCL (wtype, errind, vrtype)**
Inquire Workstation Connection and Type	**PQWKC (wkid, errind, conid, wtype)**
Inquire Workstation State Table Lengths	**PQWKSL (wtype, errind, mplbte, mpmbte, mtxbte, minbte, medbte, mpai, mcoli, vwtbi)**
Inquire Workstation State Value	**PQWKST (wksta)**
Inquire Workstation Transformation	**PQWKT (wkid, errind, tus, rwindo, cwindo, rviewp, cviewp)**
Inquire Workstation Transformation 3	**PQWKT3 (wkid, errind, tus, rwindo, cwindo, rviewp, cviewp)**
Label	**PLB (label)**

Message	**PMSG (wkid, mess)**
	PMSGS (wkid, lstr, mess)
Offset Element Pointer	**POSEP (epo)**
Open Archive File	**POPARF (afid, arcfil)**
Open PHIGS	**POPPH (errfil, bufa)**
Open Structure	**POPST (strid)**
Open Workstation	**POPWK (wkid, conid, wtype)**
Pack Data Record	**PPREC (il, ia, rl, ra, sl, lstr, str, mldr, errind, ldr, datrec)**
	PPREC (il, ia, rl, ra, sl, lstr, str, mldr, errind, ldr, datrec)
Polyline	**PPL (n, pxa, pya)**
Polyline 3	**PPL3 (n, pxa, pya, pza)**
Polymarker	**PPM (n, pxa, pya)**
Polymarker 3	**PPM3 (n, pxa, pya, pza)**
Post Structure	**PPOST (wkid, strid, priort)**
Redraw All Structures	**PRST (wkid, cofl)**
Remove Names from Set	**PRES (n, namset)**
Request Choice	**PRQCH (wkid, chdnr, stat, chnr)**
Request Locator	**PRQLC (wkid, lcdnr, stat, viewi, px, py)**
Request Locator 3	**PRQLC3 (wkid, lcdnr, stat, viewi, px, py, pz)**
Request Pick	**PRQPK (wkid, pkdnr, ippd, stat, ppd, pp)**
Request String	**PRQST (wkid, stdnr, stat, lostr, str)**
	PRQST (wkid, stdnr, stat, lostr, str)
Request Stroke	**PRQSK (wkid, skdnr, n, stat, viewi, np, pxa, pya)**
Request Stroke 3	**PRQSK3 (wkid, skdnr, n, stat, viewi, np, pxa, pya, pza)**
Request Valuator	**PRQVL (wkid, vldnr, stat, val)**
Restore Modeling Clipping Volume	**PRMCV**
Retrieve All Structures	**PRAST (afid)**
Retrieve Paths to Ancestors	**PREPAN (afid, strid, pthord, pthdep, ipthsz, n, ol, apthsz, paths)**
Retrieve Paths to Descendants	**PREPDE (afid, strid, pthord, pthdep, ipthsz, n, ol, apthsz, paths)**
Retrieve Structure Identifiers	**PRSID (afid, ilsize, n, lstrid)**
Retrieve Structure Networks	**PRESN (afid, n, lstrid)**
Retrieve Structures	**PREST (afid, n, lstrid)**
Rotate	**PRO (rotang, errind, xfrmt)**
Rotate X	**PROX (rotang, errind, xfrmt)**
Rotate Y	**PROY (rotang, errind, xfrmt)**
Rotate Z	**PROZ (rotang, errind, xfrmt)**
Sample Choice	**PSMCH (wkid, chdnr, stat, chnr)**
Sample Locator	**PSMLC (wkid, lcdnr, viewi, lpx, lpy)**
Sample Locator 3	**PSMLC3 (wkid, lcdnr, viewi, lpx, lpy, lpz)**
Sample Pick	**PSMPK (wkid, pkdnr, ippd, stat, ppd, pp)**
Sample String	**PSMST (wkid, stdnr, lostr, str)**
	PSMST (wkid, stdnr, lostr, str)
Sample Stroke	**PSMSK (wkid, skdnr, n, viewi, np, pxa, pya)**
Sample Stroke 3	**PSMSK3 (wkid, skdnr, n, viewi, np, pxa, pya, pza)**
Sample Valuator	**PSMVL (wkid, vldnr, val)**
Scale	**PSC (fx, fy, errind, xfrmt)**
Scale 3	**PSC3 (fx, fy, fz, errind, xfrmt)**
Set Annotation Style	**PSANS (astyle)**
Set Annotation Text Alignment	**PSATAL (atalh, atalv,)**

Set Annotation Text Character Height	**PSATCH (atchh)**
Set Annotation Text Character Up Vector	**PSATCU (atchux, atchuy)**
Set Annotation Text Path	**PSATP (atp)**
Set Character Expansion Factor	**PSCHXP (chxp)**
Set Character Height	**PSCHH (chh)**
Set Character Spacing	**PSCHSP (chsp)**
Set Character Up Vector	**PSCHUP (chux, chuy)**
Set Choice Mode	**PSCHM (wkid, chdnr, mode, esw)**
Set Color Model	**PSCMD (wkid, cmodel)**
Set Color Representation	**PSCR (wkid, ci, nccs, cspec)**
Set Conflict Resolution	**PSCNRS (arccr, retcr)**
Set Display Update State	**PSDUS (wkid, defmod, modmod)**
Set Edge Color Index	**PSEDCI (coli)**
Set Edge Flag	**PSEDFG (edflag)**
Set Edge Index	**PSEDI (edi)**
Set Edge Representation	**PSEDR (wkid, edi, edflag, edtype, ewidth, coli)**
Set Edgetype	**PSEDT (edtype)**
Set Edgewidth Scale Factor	**PSEWSC (ewidth)**
Set Edit Mode	**PSEDM (editmo)**
Set Element Pointer	**PSEP (ep)**
Set Element Pointer at Label	**PSEPLB (ep)**
Set Error Handling Mode	**PSERHM (erhm)**
Set Global Transformation	**PSGMT (xfrmt)**
Set Global Transformation 3	**PSGMT3 (xfrmt)**
Set Highlighting Filter	**PSHLFT (wkid, isn, is, esn, es)**
Set HLHSR Identifier	**PSHRID (hrid)**
Set HLHSR Mode	**PSHRM (wkid, hrm)**
Set Individual ASF	**PSIASF (aspcid, asfval)**
Set Interior Color Index	**PSICI (coli)**
Set Interior Index	**PSII (ii)**
Set Interior Representation	**PSIR (wkid, ii, ints, styli, coli)**
Set Interior Style	**PSIS (ints)**
Set Interior Style Index	**PSISI (istyli)**
Set Invisibility Filter	**PSIVFT (wkid, isn, is, esn, es)**
Set Linetype	**PSLN (ltype)**
Set Linewidth Scale Factor	**PSLWSC (lwidth)**
Set Local Transformation	**PSLMT (xfrmt, ctype)**
Set Local Transformation 3	**PSLMT3 (xfrmt, ctype)**
Set Locator Mode	**PSLCM (wkid, lcdnr, mode, esw)**
Set Marker Size Scale Factor	**PSMKSC (mszsf)**
Set Marker Type	**PSMK (mtype)**
Set Modeling Clipping Indicator	**PSMCLI (MCLIPI)**
Set Modeling Clipping Volume	**PSMCV (op,nhalfs,halfsp)**
Set Modeling Clipping Volume 3	**PSMCV3 (op,nhalfs,halfsp)**
Set Pattern Reference Point	**PSPARF (rfx, rfy)**
Set Pattern Reference Point and Vectors	**PSPRPV (rfx, rfy, rfz, rfvx, rfvy, rfvz)**
Set Pattern Representation	**PSPAR (wkid, pai, dimx, dimy, isc, isr, dx, dy, colia)**
Set Pattern Size	**PSPA (szx, szy)**
Set Pick Filter	**PSPKFT (wkid, pkdnr, isn, is, esn, es)**
Set Pick Identifier	**PSPKID (pkid)**
Set Pick Mode	**PSPKM (wkid, pkdnr, mode, esw)**
Set Polyline Color Index	**PSPLCI (coli)**
Set Polyline Index	**PSPLI (pli)**
Set Polyline Representation	**PSPLR (wkid, pli, ltype, lwidth, coli)**
Set Polymarker Color Index	**PSPMCI (coli)**

Set Polymarker Index	**PSPMI (pmi)**
Set Polymarker Representation	**PSPMR (wkid, pmi, mtype, mszsf, coli)**
Set String Mode	**PSSTM (wkid, stdnr, mode, esw)**
Set Stroke Mode	**PSSKM (wkid, skdnr, mode, esw)**
Set Text Alignment	**PSTXAL (txalh, txalv)**
Set Text Color Index	**PSTXCI (coli)**
Set Text Font	**PSTXFN (font)**
Set Text Index	**PSTXI (txi)**
Set Text Path	**PSTXP (txp)**
Set Text Precision	**PSTXPR (prec)**
Set Text Representation	**PSTXR (wkid, txi, font, prec, chxp, chsp, coli)**
Set Valuator Mode	**PSVLM (wkid, vldnr, mode, esw)**
Set View Index	**PSVWI (viewi)**
Set View Representation	**PSVWR (wkid, viewi, vwormt, vwmpmt, vwcplm, xyclpi)**
Set View Representation 3	**PSVWR3 (wkid, viewi, vwormt, vwmpmt, vwcplm, xyclpi, bclipi, fclipi)**
Set View Transformation Input Priority	**PSVTIP (wkid, viewi, rfvwix, relpri)**
Set Workstation Viewport	**PSWKV (wkid, xmin, xmax, ymin, ymax)**
Set Workstation Viewport 3	**PSWKV3 (wkid, wkvp)**
Set Workstation Window	**PSWKW (wkid, xmin, xmax, ymin, ymax)**
Set Workstation Window 3	**PSWKW3 (wkid, wkwn)**
Text	**PTX (px, py, chars)**
	PTXS (px, py, lstr, chars)
Text 3	**PTX3 (px, py, pz, tdx, tdy, tdz, chars)**
	PTX3S (px, py, pz tdx, tdy, tdz, lstr, chars)
Transform Point	**PTP (xi, yi, xfrmt, errind, xo, yo)**
Transform Point 3	**PTP3 (xi, yi, zi, xfrmt, errind, xo, yo, zo)**
Translate	**PTR (dx, dy, errind, xfrmt)**
Translate 3	**PTR3 (dx, dy, dz, errind, xfrmt)**
Unpack Data Record	**PUREC (ldr, datrec, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)**
	PUREC (ldr, datrec, iil, irl, isl, errind, il, ia, rl, ra, sl, lstr, str)
Unpost All Structures	**PUPAST (wkid)**
Unpost Structure	**PUPOST (wkid, strid)**
Update Workstation	**PUWK (wkid, regfl)**

Chapter 6. ISO PHIGS Enumerated Data Types

Data Type	C Binding	FORTRAN Binding
Annotation Style	1 = PANNO_STYLE_UNCONNECTED	1 = PUNCON
	2 = PANNO_STYLE_LEAD_LINE	2 = PLDLN
Archive State	0 = PST_ARCL	0 = PARCL
	1 = PST_AROP	1 = PAROP
Aspect Identifier	0 = PASPECT_LINETYPE	0 = PLN
	1 = PASPECT_LINEWIDTH	1 = PLWSC
	2 = PASPECT_LINE_COLR_IND	2 = PPLCI
	3 = PASPECT_MARKER_TYPE	3 = PMK
	4 = PASPECT_MARKER_SIZE	4 = PMKSC
	5 = PASPECT_MARKER_COLR_IND	5 = PPMCI
	6 = PASPECT_TEXT_FONT	6 = PTXFN
	7 = PASPECT_TEXT_PREC	7 = PTXPR
	8 = PASPECT_CHAR_EXPAN	8 = PCHXP
	9 = PASPECT_CHAR_SPACE	9 = PCHSP
	10 = PASPECT_TEXT_COLR_IND	10 = PTXCI
	11 = PASPECT_IND_STYLE	11 = PIS
	12 = PASPECT_INT_STYLE_IND	12 = PISI
	13 = PASPECT_INT_COLR_IND	13 = PICI
	14 = PASPECT_EDGE_FLAG	14 = PEDFG
	15 = PASPECT_EDGETYPE	15 = PEDT
	16 = PASPECT_EDGEWIDTH	16 = PEWSC
	17 = PASPECT_EDGE_COLR_IND	17 = PEDCI
Aspect Source	0 = PASF_BUNDLED	0 = PBUNDL
	1 = PASF_INDIV	1 = PINDIV
Choice Prompt and Echo Types	1 = PCHOICE_DEF	
	2 = PCHOICE_PR_ECHO	
	3 = PCHOICE_STRING_PR	
	4 = PCHOICE_STRING_IN	
	5 = PCHOICE_STRUCT	
Choice Status	0 = PIN_STATUS_NONE	0 = PNONE
	1 = PIN_STATUS_OK	1 = POK
	2 = PIN_STATUS_NO_IN	2 = PNCHOI
Clipping Indicator	0 = PIND_NO_CLIP	0 = PNCLIP
	1 = PIND_CLIP	1 = PCLIP
Color Available	0 = PMONOC	0 = PNCLIP
	1 = PCOLOR	1 = PCLIP
Color Model	1 = PMODEL_RGB	1 = PRGB
	2 = PMODEL_CIELUV	2 = PCIE
	3 = PMODEL_HSV	3 = PHSV
Composition Type	0 = PTYPE_PRECONCAT	0 = PCPRE
	1 = PTYPE_POSTCONCAT	1 = PCPOST
	2 = PTYPE_REPLACE	2 = PCREPL
Conflict Resolution	0 = PRES_MAINTAIN	0 = PCRMNT
	1 = PRES_ABANDON	1 = PCRABA
	2 = PRES_UPD	2 = PCRUPD
Control Flag	0 = PFLAG_COND	0 = PCONDI
	1 = PFLAG_ALWAYS	1 = PALWAY
Current/Requested		0 = PCURVL
		1 = PRQSVL
Default Parameters of Open PHIGS	-1 = PDEF_MEM_SIZE	
	0 = PDEF_ERR_FILE	

Data Type	C Binding	FORTRAN Binding
Deferral Mode	0 = PDEFER_ASAP	0 = PASAP
	1 = PDEFER_BNIG	1 = PBNIG
	2 = PDEFER_BNIL	2 = PBNIL
	3 = PDEFER_ASTI	3 = PASTI
	4 = PDEFER_WAIT	4 = PWAITD
Device Coordinate Units	0 = PDC_METRES	0 = PMETRE
	1 = PDC_OTHER	1 = POTHU
Display Surface Empty	0 = PSURF_NOT_EMPTY	0 = PNEMPT
	1 = PSURF_EMPTY	1 = PEMPTY
Dynamic Modification	0 = PDYN_IRG	0 = PIRG
	1 = PDYN_IMM	1 = PIMM
	2 = PDYN_CBS	2 = PCBS
Echo Switch	0 = PSWITCH_NO_ECHO	0 = PREQU
	1 = PSWITCH_ECHO	1 = PSAMPL
		2 = PEVENT
Edge Flag	0 = PEDGE_OFF	0 = POFF
	1 = PEDGE_ON	1 = PON
Edge Type	1 = PLINE_SOLID	1 = PLSOLI
	2 = PLINE_DASH	2 = PLDASH
	3 = PLINE_DOT	3 = PLDOT
	4 = PLINE_DASH_DOT	4 = PLDASD
Edit Mode	0 = PEDIT_INSERT	0 = PINSRT
	1 = PEDIT_REPLACE	1 = PREPLC
Element Enumeration	0 = PFIRST_PHIGS_ELEM	
	1 = PLAST_PHIGS_ELEM	

Data Type	C Binding	FORTRAN Binding
Element Type	0=PELEM_ALL	0=PEALL
	1=PELEM_NIL	1=PENIL
	2=PELEM_POLYLINE3	2=PEPL3
	3=PELEM_POLYLINE	3=PEPL
	4=PELEM_POLYMARKER3	4=PEPM3
	5=PELEM_POLYMARKER	5=PEPM
	6=PELEM_TEXT3	6=PETX3
	7=PELEM_TEXT	7=PETX
	8=PELEM_ANNO_TEXT_REL3	8=PEATR3
	9=PELEM_ANNO_TEXT_REL	9=PEATR
	10=PELEM_FILL_AREA3	10=PEFA3
Element Type	11=PELEM_FILL_AREA	11=PEFA
	12=PELEM_FILL_AREA_SET3	12=PEFAS3
	13=PELEM_FILL_AREA_SET	13=PEFAS
	14=PELEM_CELL_ARRAY3	14=PECA3
	15=PELEM_CELL_ARRAY	15=PECA
	16=PELEM_GDP3	16=PEGDP3
	17=PELEM_GDP	17=PEGDP
	18=PELEM_LINE_IND	18=PEPLI
	19=PELEM_MARKER_IND	19=PEPMI
	20=PELEM_TEXT_IND	20=PETXI

Data Type	C Binding	FORTRAN Binding
Element Type	21=PELEM_INT_IND	21=PEII
		22=PEEDI
	23=PELEM_LINETYPE	23=PELN
	24=PELEM_LINEWIDTH	24=PELWSC
	25=PELEM_LINE_COLR_IND	25=PEPLCI
	26=PELEM_MARKER_TYPE	26=PEMK
	27=PELEM_MARKER_SIZE	27=PEMKSC
	28=PELEM_MARKER_COLR_IND	28=PEPMCI
	29=PELEM_TEXT_FONT	29=PETXFN
	30=PELEM_TEXT_PREC	30=PETXPR
Element Type	31=PELEM_CHAR_EXPAN	31=PECHXP
	32=PELEM_CHAR_SPACE	32=PECHSP
	33=PELEM_TEXT_COLR_IND	33=PETXCI
	34=PELEM_CHAR_HT	34=PECHH
	35=PELEM_CHAR_UP_VEC	35=PECHUP
	36=PELEM_TEXT_PATH	36=PETXP
	37=PELEM_TEXT_ALIGN	37=PETXAL
	38=PELEM_ANNO_CHAR_HT	38=PEATCH
	39=PELEM_ANNO_CHAR_UP_VEC,	39=PEATCU
	40=PELEM_ANNO_PATH	40=PEATP
Element Type	41=PELEM_ANNO_ALIGN	41=PEATAL
	42=PELEM_ANNO_STYLE	42=PEANST
	43=PELEM_INT_STYLE	43=PEIS
	44=PELEM_INT_STYLE_IND	44=PEISI
	45=PELEM_INT_COLR_IND	45=PEICI
	46=PELEM_EDGE_FLAG	46=PEEDFG
	47=PELEM_EDGETYPE	47=PEEDT
	48=PELEM_EDGEWIDTH	48=PEEWSC
	49=PELEM_EDGE COLR_IND	49=PEEDCI
	50=PELEM_PAT_SIZE	50=PEPA
Element Type	51=PELEM_PAT_REF_POINT_VECS	51=PEPRPV
	52=PELEM_PAT_REF_POINT	52=PEPARF
	53=PELEM_ADD_NAMES_SET	53=PEADS
	54=PELEM_REMOVE_NAMES_SET	54=PERES
	55=PELEM_INDIV_ASF	55=PEIASF
	56=PELEM_HLHSR_ID	56=PEHRID
	57=PELEM_LOCAL_MODEL_TRAN3	57=PELMT3
	58=PELEM_LOCAL_MODEL_TRAN	58=PELMT
	59=PELEM_GLOBAL_MODEL_TRAN3	59=PEGMT3
	60=PELEM_GLOBAL_MODEL_TRAN	60=PEGMT
Element Type	61=PELEM_MODEL_CLIP_VOL3	61=PEMCV3
	62=PELEM_MODEL_CLIP_VOL	62=PEMCV
	63=PELEM_MODEL_CLIP_IND	63=PEMCLI
	64=PELEM_RESTORE_MODEL_CLIP_VOL	64=PERMCV
	65=PELEM_VIEW_IND	65=PEVWI
	66=PELEM_EXEC_STRUCT	66=PEEXST
	67=PELEM_LABEL	67=PELB
	68=PELEM_APPL_DATA	68=PEAP
	69=PELEM_GSE	69=PEGSE
	70=PELEM_PICK_ID	70=PEPKID

Data Type	C Binding	FORTRAN Binding
Error File		1 = CONSOLE
		2 = AFMERROR
Error Handling Mode	0 = PERR_OFF	0 = POFF
	1 = PERR_ON	1 = PON

Data Type	C Binding	FORTRAN Binding
GDP Attributes	0 = PATTR_LINE 1 = PATTR_MARKER 2 = PATTR_TEXT 3 = PATTR_INT 4 = PATTR_EDGE	0 = PPLATT 1 = PPMATT 2 = PTXATT 3 = PINATT 4 = PEDATT
HLHSR mode	0 = OFF 1 = ON_THE_FLY	0 = OFF 1 = ON_THE_FLY
HLHSR Identifier	0 = Visualize if not hidden 1 = Visualize if hidden 2 = Visualize always 3 = Not Visualize 4 = Face-Dependent Visualization 5 = No Update 6 = Greater than 7 = Equal to 8 = Less than 9 = Not equal 10 = Less than or equal to	0 = Visualize if not hidden 1 = Visualize if hidden 2 = Visualize always 3 = Not Visualize 4 = Face-Dependent Visualization 5 = No Update 6 = Greater than 7 = Equal to 8 = Less than 9 = Not equal 10 = Less than or equal to
Input Device Class	0 = PIN_NONE 1 = PIN_LOC 2 = PIN_STROKE 3 = PIN_VAL 4 = PIN_CHOICE 5 = PIN_PICK 6 = PIN_STRING	0 = PNCLAS 1 = PLOCAT 2 = PSTROK 3 = PVALUA 4 = PCHOIC 5 = PPICK 6 = PSTRIN
Input Device Status	0 = PIN_STATUS_NONE 1 = PIN_STATUS_OK	0 = PNONE 1 = POK
Inquire Type	0 = PINQ_SET 1 = PINQ_REALIZED	0 = PSET 1 = PREALI
Interior Style	0 = PSTYLE_HOLLOW 1 = PSTYLE_SOLID 2 = PSTYLE_PAT 3 = PSTYLE_HATCH 4 = PSTYLE_EMPTY	0 = PHOLLO 1 = PSOLID 2 = PPATTR 3 = PHATCH 4 = PISEMP
Line Type	1 = PLINE_SOLID 2 = PLINE_DASH 3 = PLINE_DOT 4 = PLINE_DASH_DOT	1 = PLSOLI 2 = PLDASH 3 = PLDOT 4 = PLDASD
Locator Prompt and Echo Types	1 = PLOC_DEF 2 = PLOC_CROSS_HAIR 3 = PLOC_TRACK_CROSS 4 = PLOC_RUB_BAND 5 = PLOC_RECT 6 = PLOC_DIGIT	
Marker Type	1 = PMARKER_DOT 2 = PMARKER_PLUS 3 = PMARKER_ASTERISK 4 = PMARKER_CIRCLE 5 = PMARKER_CROSS	1 = PPOINT 2 = PPLUS 3 = PAST 4 = POMARK 5 = PXMARK
Modeling Clip Operator		1 = PMCREP 2 = PMCINT
Modification Mode	0 = PMODE_NIVE 1 = PMODE_UWOR 2 = PMODE_UQUM	0 = PNIVE 1 = PUWOR 2 = PUQUM

Data Type	C Binding	FORTRAN Binding
More Simultaneous Events	0 = PSIMULT_NO_MORE	0 = PNMORE
	1 = PSIMULT_MORE	1 = PMORE
Open Structure Status	0 = PSTRUCT_NONE	0 = PNONST
	1 = PSTRUCT_OPEN	1 = POPNST
Operating Mode	0 = POP_REQ	0 = PREQU
	1 = POP_SAMPLE	1 = PSAMPL
	2 = POP_EVENT	2 = PEVENT

Data Type
PHIGS Function Identifiers
for FORTRAN Binding

0=EOPPH,	92=EEXST,
1=ECLPH,	93=ELB,
2=EOPWK,	94=EAP,
3=ECLWK	95=EGSE,
4=ERST,	96=ESEDM,
5=EUWK,	97=ECELST,
6=ESDUS,	98=ESEP,
7=EMSG,	99=EOSEP,
8=EPL3,	100=ESEPLB,
9=EPL,	101=EDEL,
10=EPM3,	102=EDELRA,
11=EPM,	103=EDELLB
12=ETX3,	104=EEMST,
13=ETX,	105=EDST,
14=EATR3,	106=EDSN,
15=EATR,	107=EDAS,
16=EFA3,	108=ECSTID,
17=EFA,	109=ECSTRF,
18=EFAS3,	110=ECSTIR,
19=EFAS,	111=EPOST,
20=ECA3,	112=EUPOST,
21=ECA,	113=EUPAST,
22=EGDP3,	114=EOPARF,
23=EGDP,	115=ECLARF,
24=ESPLI,	116=EARST,
25=ESPMI,	117=EARSN,
26=ESTXI,	118=EARAST,
27=ESII,	119=ESCNRS,
28=ESEDI,	120=ERSID,
29=ESLN,	121=EREST,
30=ESLWSC,	122=ERESN,
31=ESPLCI,	123=ERAST,
32=ESMK,	124=EREPAN,
33=ESMKSC,	125=EREPDE,
34=ESPMCI,	126=EDSTAR,
35=ESTXFN,	127=EDSNAR,
36=ESTXPR,	128=EDASAR,
37=ESCHXP,	129=ESPKID,
38=ESCHSP,	130=ESPKFT,
39=ESTXCI,	131=EINLC3,
40=ESCHH	132=EINCL,
41=ESCHUP,	133=EINSK3,
42=ESTXP,	134=EINSK,

Data Type

43=ESTXAL,
44=ESATCH,
45=ESATCU,
46=ESATP,
47=ESATAL,
48=ESANS,
49=EISI,
50=ESISI,
51=ESICI,
52=ESEDFG,
53=ESEDT,
54=ESEWSC,
55=ESEDCI,
56=ESPA,
57=ESPRPV,
58=ESPARF,
59=EADS,
60=ERES,
61=ESIASF,
62=ESPLR,
63=ESPMR,
64=ESTXR,
65=ESIR,
66=ESEDR,
67=ESPAR,
68=ESCR,
69=ESHLFT,
70=ESIVFT,
71=ESCMD,
72=ESHRID,
73=ESHRM,
74=ESLMT3,
75=ESLMT,
76=ESGMT3,
77=ESGMT,
78=ESMCV3,
79=ESMCV,
80=ESMCLI,
81=ERMCV,
82=ESVWI,
83=ESVWR3,
84=ESVWR,
85=ESVTIP,
86=ESWKW3,
87=ESWKW,
88=ESWKV3,
89=ESWKV,
90=EOPST,
91=ECLST,
135=EINVL3,
136=EINVL,
137=EINCH3,
138=EINCH,
139=EINPK3,
140=EINPK,
141=EINST3,
142=EINST,
143=ESLCM,
144=ESSKM,
145=ESVLM,
146=ESCHM,
147=ESPKM,
148=ESSTM,
149=ERQLC3,
150=ERQLC,
151=ERQSK3,
152=ERQSK,
153=ERQVL,
154=ERQCH,
155=ERQPK,
156=ERQST,
157=ESMLC3,
158=ESMLC,
159=ESMSK3,
160=ESMSK,
161=ESMVL,
162=ESMCH,
163=ESMPK,
164=ESMST,
165=EWAIT,
166=EFLUSH,
167=EGTLC3,
168=EGTLC,
169=EGTSK3,
170=EGTSK,
171=EGTVL,
172=EGTCH,
173=EGTPK,
174=EGTST,
175=EWITM,
176=EGTITM,
177=ERDITM,
178=EIITM,
179=ESERHM,
180=EESC,
181=EPREC,
182=EUREC

Data Type	**C Binding**	**FORTRAN Binding**
Pick Path Order	0 = PORDER_TOP_FIRST 1 = PORDER_BOTTOM_FIRST	0 = PPOTOP 1 = PPOBOT

Data Type	C Binding	FORTRAN Binding
Pick Prompt and Echo Types	1 = PPICK_DEF 2 = PPICK_GROUP_HIGHL 3 = PPICK_STRUCT_NETWORK	
Pick Status	1 = PIN_STATUS_OK 2 = PIN_STATUS_NO_IN	1 = POK 2 = PNPICK
Polyline/Fill Area Control Flag	0 = PFLAG_LINE 1 = PFLAG_FILL 2 = PFLAG_FILL_SET	0 = PPLINE 1 = PFILLA 2 = PFILAS
Polyline Line Type	1 = PLINE_SOLID 2 = PLINE_DASH 3 = PLINE_DOT 4 = PLINE_DASH_DOT	1 = PLSOLI 2 = PLDASH 3 = PLDOT 4 = PLDASD
Polyline Control Flag	0 = PFLAG_LINE 1 = PFLAG_FILL 2 = PLAG_FILL_SET	0 = PPLINE 1 = PFILLA 2 = PFILAS
Presence of Invalid Values		0 = PABSNT 1 = PPRSNT
Reference Handling Flag	0 = PFLAG_DEL 1 = PFLAG_KEEP	0 = PDELE 1 = PKEEP
Regeneration Flag	0 = PFLAG_POSTPONE 1 = PFLAG_PERFORM	0 = PPOSTP 1 = PPERFO
Relative Input Priority	0 = PPRI_HIGHER 1 = PPRI_LOWER	0 = PHIGHR 1 = PLOWER
Search Direction	0 = PDIR_BACKWARD 1 = PDIR_FORWARD	0 = PBWD 1 = PFWD
Search Status Indicator	0 = PSEARCH_STATUS_FAILURE 1 = PSEARCH_STATUS_SUCCESS	0 = PFAIL 1 = PSUCC
State of Visual Representation	0 = PVISUAL_ST_CORRECT 1 = PVISUAL_ST_DEFER 2 = PVISUAL_ST_SIMULATED	0 = PVROK 1 = PVRDRF 2 = PVRSIM
String Prompt and Echo Types	1 = PSTRING_DEF	
Stroke Prompt and Echo Types	1 = PSTROKE_DEF 2 = PSTROKE_DIGIT 3 = PSTROKE_MARKER 4 = PSTROKE_LINE	
Structure Network Source	0 = PNET_CSS 1 = PNET_AR	0 = PCSS 1 = PARCHV
Structure State Value	0 = PSTRUCT_ST_STCL 1 = PSTRUCT_ST_STOP	0 = PSTCL 1 = PSTOP
Structure Status Indicator	0 = PSTRUCT_STATUS_NON_EXISTENT 1 = PSTRUCT_STATUS_EMPTY 1 = PSTRUCT_STATUS_NOT_EMPTY	0 = PSNOEX 1 = PSEMPT 2 = PSNEMP
System State Value	0 = PSYS_ST_PHCL 1 = PSYS_ST_PHOP	0 = PPHCL 1 = PPHOP
Text Alignment Horizontal	0 = PHOR_NORM 1 = PHOR_LEFT 2 = PHOR_CTR 3 = PHOR_RIGHT	0 = PAHNOR 1 = PALEFT 2 = PACENT 3 = PARITE
Text Alignment Vertical	0 = PVERT_NORM 1 = PVERT_TOP 2 = PVERT_CAP 3 = PVERT_HALF 4 = PVERT_BASE 5 = PVERT_BOTTOM	0 = PAVNOR 1 = PATOP 2 = PACAP 3 = PAHALF 4 = PABASE 5 = PABOTT
Text Path	0 = PPATH_RIGHT 1 = PPATH_LEFT 2 = PPATH_UP 3 = PPATH_DOWN	0 = PRIGHT 1 = PLEFT 2 = PUP 3 = PDOWN
Text Precision	0 = PPREC_STRING 1 = PPREC_CHAR 2 = PPREC_STROKE	0 = PSTRP 1 = PCHARP 2 = PSTRKP
Type of Returned Values	0 = PINQ_SET 1 = PINQ_REALIZED	0 = PSET 1 = PREALI
Update State	0 = PUPD_NOT_PEND 1 = PUPD_PEND	0 = PNPEND 1 = PPEND
Valuator Prompt and Echo Types	1 = PVAL_DEF 2 = PVAL_GRAPH 3 = PVAL_DIGIT	
View Type	0 = PTYPE_PARAL 1 = PTYPE_PERSPECT	0 = PPARL 1 = PPERS

Data Type	C Binding	FORTRAN Binding
Visual State	0 = PVISUAL_ST_CORRECT 1 = PVISUAL_ST_DEFER 2 = PVISUAL_ST_SIMULATED	0 = PVROK 1 = PVRDRF 2 = PVRSIM
Workstation Category	0 = PCAT_OUT 1 = PCAT_IN 2 = PCAT_OUTIN 3 = PCAT_MO 4 = PCAT_MI	0 = POUTPT 1 = PINPUT 2 = POUTIN 3 = PMQ 4 = PMI
Workstation Classification	0 = PCLASS_VEC 1 = PCLASS_RASTER 2 = PCLASS_OTHER	0 = PVECTR 1 = PRASTR 2 = POTHWK
Workstation Dependency Indicator	0 = PWS_INDEP 1 = PWS_DEP	0 = PWKI 1 = PWKD
Workstation State Value	0 = PWS_ST_WSCL 1 = PWS_ST_WSOP	0 = PWSL 1 = PWSOP
Workstation Types	1 = 6090 2 = 5080 3 = GDDM 4 = GDF 5 = CGM 6 = X 8 = XSOFT 9 = XPEX 10 = IMAGE	1 = 6090 2 = 5080 3 = GDDM 4 = GDF 5 = CGM 6 = X 8 = XSOFT 9 = XPEX 10 = IMAGE

Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Dept. LRAS/Bldg. 003
11400 Burnet Road
Austin, TX 78758-3498
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX
AIXwindows
GDDM
IBM
RS/6000

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be the trademarks or service marks of others.

Readers' Comments — We'd Like to Hear from You

The graPHIGS Programming Interface: ISO PHIGS Quick Reference

Publication No. SC28-2705-02

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC28-2705-02

IBM

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM Corporation
Information Development
Department H6DS-905-6C006
11501 Burnet Road
Austin, TX 78758-3493

Fold and Tape | **Please do not staple** | Fold and Tape

SC28-2705-02

Cut or Fold
Along Line

Printed in U.S.A.

SC28-2705-02