
**Information technology — Computer
graphics — Programmer's Hierarchical
Interactive Graphics System (PHIGS)
language bindings —**

**Part 4:
C**

**AMENDMENT 2:
Incorporation of PHIGS amendments**

*Technologies de l'information — Infographie — Interfaces langage avec
PHIGS —*

Partie 4: C

AMENDEMENT 2: Incorporation des amendements de PHIGS



Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Amendment 2 to ISO/IEC 9593-4:1991 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 24, *Computer graphics and image processing*.

© ISO/IEC 1998

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher

ISO/IEC Copyright Office • Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in Switzerland

Information technology - Computer graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings -

Part 4:

C

Amendment 2: Incorporation of PHIGS amendments

Page 2

2 Normative references

The number of the first reference is changed to ISO/IEC 9592-1:1996.

Page 3

3 The C language binding of PHIGS

3.1 Conformance

The following text replaces the content of 3.1:

“This binding incorporates the rules of conformance defined in the PHIGS Standard (ISO/IEC 9592) for PHIGS implementations, with those additional requirements specifically defined for C language implementations of PHIGS.

The following criteria are established for determining conformance of an implementation to this binding:

- a) In order to conform to the *Basic PHIGS profile*, an implementation of the C binding of PHIGS shall implement at least the functionality defined for the *Basic PHIGS profile* as specified in ISO/IEC

9592-1. It shall make visible all of the declarations in the C binding specified in clauses 5, 6, and 7 of this part of ISO/IEC 9593.

b) In order to conform to the *PHIGS PLUS profile*, an implementation of the C binding of PHIGS shall implement at least the functionality defined for the *PHIGS PLUS profile* as specified in ISO/IEC 9592-1. It shall make visible all of the declarations in the C binding specified in clauses 5 through 10 of this part of ISO/IEC 9593.

c) In order to conform to the *Full PHIGS profile*, an implementation of the C binding of PHIGS shall implement all of the functionality defined for the *Full PHIGS profile* as specified in ISO/IEC 9592-1. It shall make visible all of the declarations in the C binding specified in this part of ISO/IEC 9593.

d) The syntax of the function names shall be precisely as specified in this part of ISO/IEC 9593 and the parameters shall be of the data types stated in this part of ISO/IEC 9593.

Pages 10 to 13

4 Tables

4.2 Table of abbreviations

The following abbreviations are added alphabetically to Table 1

“

Table 1 - Abbreviations ordered alphabetically

Word or Phrase	Abbreviation
acknowledgement	ack
address	addr
associate	assoc
association	assoc
attribute	attr
automatic	auto
background	backg
boundary	bound
buffer	buf
centre	ctr
channel	chan
complete	comp
composition	compos
conditions	conds
configuration	config
continuity	cont
correlate, correlation	corr
define	def
destination	dest
disassociate	disassoc
disposition	dispos

Table 1 - Abbreviations ordered alphabetically (Continued)

Word or Phrase	Abbreviation
distance	dist
frequency	freq
group	grp
groups	grps
heuristics	heur
include	incl
incompatible	incompat
incomplete	incomp
indicator	indic
instance	inst
logical input device	lid
manipulation	manip
measure	meas
non-atomic	na
non-retained	nr
non-retained data	nrd
optimization	opt
parameterization	param
parameters	params
picture	pict
posting	post
process, processing	proc
reference	ref
registered	reg
render, rendering	rend
resource	res
sampling	sampl
source	src
target	targ
texture	textr
transfer	trans
traverse, traversal	trav
trigger	trig
uncompressed	uncomp
undefine	undef
weight	wt
which	NULL

”

Pages 14 to 20

4.3 Function names

4.3.1 List ordered alphabetically by bound name

The following function names are merged alphabetically by bound name in Table 2

“
Table 2 - Function names ordered by bound name

C Name	PHIGS and PHIGS PLUS Name
passoc_image_res (...)	ASSOCIATE IMAGE RESOURCE
passoc_trav_res (...)	ASSOCIATE TRAVERSAL RESOURCE
pattach_lid_to_light_src (...)	ATTACH LOGICAL INPUT DEVICE TO LIGHT SOURCE
pattach_lid_to_view (...)	ATTACH LOGICAL INPUT DEVICE TO VIEW
pcircle (...)	CIRCLE
pcircle3 (...)	CIRCLE 3
pcircular_arc (...)	CIRCULAR ARC
pcircular_arc3 (...)	CIRCULAR ARC 3
pcircular_arc_close (...)	CIRCULAR ARC CLOSE
pcircular_arc_close3 (...)	CIRCULAR ARC CLOSE 3
pclear_targ (...)	CLEAR TARGET
pclose_di_struct (...)	CLOSE DIRECT INTERPRETATION STRUCTURE
pcond_exec_struct (...)	CONDITIONAL EXECUTE STRUCTURE
pcond_inst_struct (...)	CONDITIONAL INSTANCE STRUCTURE
pcond_return (...)	CONDITIONAL RETURN
pcond_skip_elements (...)	CONDITIONAL SKIP ELEMENTS
pcond_skip_to_label (...)	CONDITIONAL SKIP TO LABEL
pcopy_elem_struct (...)	COPY ELEMENT FROM STRUCTURE
pcopy_elem_range_struct (...)	COPY ELEMENT RANGE FROM STRUCTURE
pcopy_elems_between_labels_struct (...)	COPY ELEMENTS BETWEEN LABELS FROM STRUCTURE
pcopy_targ (...)	COPY TARGET
pcreate_composite_measure (...)	CREATE COMPOSITE MEASURE
pcreate_mipmap_texture (...)	CREATE MIPMAP TEXTURE
pcreate_set_measure (...)	CREATE SET MEASURE
pcreate_targ (...)	CREATE TARGET
pdefine_choice (...)	DEFINE CHOICE
pdefine_composite (...)	DEFINE COMPOSITE
pdefine_linetype (...)	DEFINE LINETYPE
pdefine_locator (...)	DEFINE LOCATOR
pdefine_marker_type (...)	DEFINE MARKER TYPE
pdefine_pick (...)	DEFINE PICK
pdefine_post_grp (...)	DEFINE POSTING GROUP
pdefine_set (...)	DEFINE SET
pdefine_string (...)	DEFINE STRING
pdefine_stroke (...)	DEFINE STROKE

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pdefine_valuator (...)	DEFINE VALUATOR
pdestroy_composite_measure (...)	DESTROY COMPOSITE MEASURE
pdestroy_set_measure (...)	DESTROY SET MEASURE
pdestroy_targ (...)	DESTROY TARGET
pdetach_lid_from_light_src (...)	DETACH LOGICAL INPUT DEVICE FROM LIGHT SOURCE
pdetach_lid_from_view (...)	DETACH LOGICAL INPUT DEVICE FROM VIEW
pdisable_di_pick (...)	DISABLE DIRECT INTERPRETATION PICK
pdisassoc_image_res (...)	DISASSOCIATE IMAGE RESOURCE
pdisassoc_trav_res (...)	DISASSOCIATE TRAVERSAL RESOURCE
pellipse (...)	ELLIPSE
pellipse3 (...)	ELLIPSE 3
pelliptical_arc (...)	ELLIPTICAL ARC
pelliptical_arc3 (...)	ELLIPTICAL ARC 3
pelliptical_arc_close (...)	ELLIPTICAL ARC CLOSE
pelliptical_arc_close3 (...)	ELLIPTICAL ARC CLOSE 3
penable_di_pick (...)	ENABLE DIRECT INTERPRETATION PICK
pend_watch_on_elem_range (...)	END WATCH ON ELEMENT RANGE
pfill_circle (...)	FILL CIRCLE
pfill_circle3 (...)	FILL CIRCLE 3
pfill_ellipse (...)	FILL ELLIPSE
pfill_ellipse3 (...)	FILL ELLIPSE 3
pget_composite (...)	GET COMPOSITE
pget_composite3 (...)	GET COMPOSITE 3
pget_set (...)	GET SET
pget_set3 (...)	GET SET 3
pinit_composite (...)	INITIALIZE COMPOSITE
pinit_composite3 (...)	INITIALIZE COMPOSITE 3
pinit_di_pick (...)	INITIALIZE DIRECT INTERPRETATION PICK
pinit_di_pick3 (...)	INITIALIZE DIRECT INTERPRETATION PICK 3
pinit_set (...)	INITIALIZE SET
pinit_set3 (...)	INITIALIZE SET 3
pinq_alpha_fac (...)	INQUIRE ALPHA FACILITIES
pinq_appl_filter (...)	INQUIRE APPLICATION FILTER
pinq_assoc_image_res (...)	INQUIRE ASSOCIATED IMAGE RESOURCES
pinq_back_clip_indicator (...)	INQUIRE BACK CLIPPING INDICATOR
pinq_back_plane_dist (...)	INQUIRE BACK PLANE DISTANCE
pinq_choice_fac (...)	INQUIRE CHOICE FACILITIES
pinq_composite_fac (...)	INQUIRE COMPOSITE FACILITIES
pinq_composite_meas_def (...)	INQUIRE COMPOSITE MEASURE DEFINITION
pinq_composite_st (...)	INQUIRE COMPOSITE DEVICE STATE
pinq_composite_st3 (...)	INQUIRE COMPOSITE DEVICE STATE 3
pinq_cond_trav_fac (...)	INQUIRE CONDITIONAL TRAVERSAL FACILITIES
pinq_config_setting_fac (...)	INQUIRE CONFIGURATION SETTING FACILITIES
pinq_dc_clip_regions (...)	INQUIRE DEVICE COORDINATE CLIP REGIONS
pinq_dc_clip_regions3 (...)	INQUIRE DEVICE COORDINATE CLIP REGIONS 3

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pinq_dc_clip_regions_fac(...)	INQUIRE DEVICE COORDINATE CLIP REGIONS FACILITIES
pinq_def_composite_data (...)	INQUIRE DEFAULT COMPOSITE DEVICE DATA
pinq_def_composite_data3 (...)	INQUIRE DEFAULT COMPOSITE DEVICE DATA 3
pinq_def_di_pick_data (...)	INQUIRE DEFAULT DIRECT INTERPRETATION PICK DATA
pinq_def_di_pick_data3 (...)	INQUIRE DEFAULT DIRECT INTERPRETATION PICK DATA 3
pinq_def_set_data (...)	INQUIRE DEFAULT SET DEVICE DATA
pinq_def_set_data3 (...)	INQUIRE DEFAULT SET DEVICE DATA 3
pinq_def_targ_disp (...)	INQUIRE DEFAULT TARGET DISPOSITION
pinq_di_mode (...)	INQUIRE DIRECT INTERPRETATION MODE
pinq_di_pick_corr_point (...)	INQUIRE DIRECT INTERPRETATION PICK CORRELATION POINT
pinq_di_pick_corr_point3 (...)	INQUIRE DIRECT INTERPRETATION PICK CORRELATION POINT 3
pinq_di_pick_set_status (...)	INQUIRE DIRECT INTERPRETATION PICK SET STATUS
pinq_di_pick_st (...)	INQUIRE DIRECT INTERPRETATION PICK STATE
pinq_di_pick_st3 (...)	INQUIRE DIRECT INTERPRETATION PICK STATE 3
pinq_disp_targ (...)	INQUIRE DISPLAY TARGET
pinq_di_trav_fac(...)	INQUIRE DIRECT INTERPRETATION TRAVERSAL FACILITIES
pinq_dyns_post_grps (...)	INQUIRE DYNAMICS OF POSTING GROUPS
pinq_dyns_ws_attrs_texture (...)	INQUIRE DYNAMICS OF WORKSTATION ATTRIBUTES TEXTURE
pinq_edge_rep_full (...)	INQUIRE EDGE REPRESENTATION FULL
pinq_ext_pat_fac(...)	INQUIRE EXTENDED PATTERN FACILITIES
pinq_ext_pat_rep (...)	INQUIRE EXTENDED PATTERN REPRESENTATION
pinq_front_clip_indicator (...)	INQUIRE FRONT CLIPPING INDICATOR
pinq_front_plane_dist (...)	INQUIRE FRONT PLANE DISTANCE
pinq_grps_posted (...)	INQUIRE SET OF GROUPS TO WHICH POSTED
pinq_highl_fac(...)	INQUIRE HIGHLIGHTING FACILITIES
pinq_highl_rep (...)	INQUIRE HIGHLIGHTING REPRESENTATION
pinq_image_res (...)	INQUIRE IMAGE RESOURCE
pinq_image_res_fac(...)	INQUIRE IMAGE RESOURCE FACILITIES
pinq_lid_attached_to_light_src (...)	INQUIRE LOGICAL INPUT DEVICE ATTACHED TO LIGHT SOURCE
pinq_lid_attached_to_view (...)	INQUIRE LOGICAL INPUT DEVICE ATTACHED TO VIEW
pinq_lid_def (...)	INQUIRE LOGICAL INPUT DEVICE DEFINITION
pinq_line_rep_full (...)	INQUIRE POLYLINE REPRESENTATION FULL
pinq_linetype_def (...)	INQUIRE LINETYPE DEFINITION
pinq_linetype_def_fac(...)	INQUIRE LINETYPE DEFINITION FACILITIES
pinq_linetype_def_support (...)	INQUIRE LINETYPE DEFINITION SUPPORT
pinq_list_def_appl_filters (...)	INQUIRE LIST OF DEFINED APPLICATION FILTERS
pinq_list_highl_inds (...)	INQUIRE LIST OF HIGHLIGHTING INDICES

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pinq_list_post_grps (...)	INQUIRE POSTED STRUCTURES FROM POSTING GROUP
pinq_list_texture_inds (...)	INQUIRE LIST OF TEXTURE INDICES
pinq_lists_avail_lids (...)	INQUIRE LISTS OF AVAILABLE LOGICAL INPUT DEVICES
pinq_loc_facns (...)	INQUIRE LOCATOR FACILITIES
pinq_marker_type_def (...)	INQUIRE MARKER TYPE DEFINITION
pinq_marker_type_def_facns (...)	INQUIRE MARKER TYPE DEFINITION FACILITIES
pinq_marker_type_def_support (...)	INQUIRE MARKER TYPE DEFINITION SUPPORT
pinq_mipmap_facns (...)	INQUIRE MIPMAP FACILITIES
pinq_num_avail_na_in (...)	INQUIRE NUMBER OF AVAILABLE NON-ATOMIC LOGICAL INPUT DEVICES
pinq_num_def_linetypes (...)	INQUIRE NUMBER OF DEFINED LINETYPES
pinq_num_def_marker_types (...)	INQUIRE NUMBER OF DEFINED MARKER TYPES
pinq_num_pred_appl_filters (...)	INQUIRE NUMBER OF PREDEFINED APPLICATION FILTERS
pinq_pick_facns (...)	INQUIRE PICK FACILITIES
pinq_pick_mapping_facns (...)	INQUIRE PICK MAPPING FACILITIES
pinq_pick_mapping_state (...)	INQUIRE PICK MAPPING STATE
pinq_pict_status (...)	INQUIRE PICTURE STATUS
pinq_posted_di_struct (...)	INQUIRE POSTED DIRECT INTERPRETATION STRUCTURE
pinq_posted_structs_from_post_grp (...)	INQUIRE POSTED STRUCTURES FROM POSTING GROUP
pinq_post_grp (...)	INQUIRE POSTING GROUP
pinq_post_grp_facns (...)	INQUIRE POSTING GROUP FACILITIES
pinq_pred_appl_filter (...)	INQUIRE PREDEFINED APPLICATION FILTER
pinq_pred_assoc_targ_trav_res (...)	INQUIRE PREDEFINED ASSOCIATION OF TARGET WITH TRAVERSAL RESOURCES
pinq_pred_assoc_trav_res_targ (...)	INQUIRE PREDEFINED ASSOCIATION OF TRAVERSAL RESOURCE WITH TARGET
pinq_pred_composite_meas_def (...)	INQUIRE PREDEFINED COMPOSITE MEASURE DEFINITION
pinq_pred_edge_rep_full (...)	INQUIRE PREDEFINED EDGE REPRESENTATION FULL
pinq_pred_ext_pat_rep (...)	INQUIRE PREDEFINED EXTENDED PATTERN REPRESENTATION
pinq_pred_highl_rep (...)	INQUIRE PREDEFINED HIGHLIGHTING REPRESENTATION
pinq_pred_image_res (...)	INQUIRE PREDEFINED IMAGE RESOURCE
pinq_pred_line_rep_full (...)	INQUIRE PREDEFINED POLYLINE REPRESENTATION FULL
pinq_pred_linetype_def (...)	INQUIRE PREDEFINED LINETYPE DEFINITION
pinq_pred_marker_type_def (...)	INQUIRE PREDEFINED MARKER TYPE DEFINITION
pinq_pred_post_grp (...)	INQUIRE PREDEFINED POSTING GROUP
pinq_pred_set_meas_def (...)	INQUIRE PREDEFINED SET MEASURE DEFINITION
pinq_pred_texture_rep (...)	INQUIRE PREDEFINED TEXTURE REPRESENTATION

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pinq_proj_ref_point (...)	INQUIRE PROJECTION REFERENCE POINT
pinq_proj_type (...)	INQUIRE PROJECTION TYPE
pinq_proj_vp (...)	INQUIRE PROJECTION VIEWPORT
pinq_proj_vp3 (...)	INQUIRE PROJECTION VIEWPORT 3
pinq_rend_targ (...)	INQUIRE RENDERING TARGET
pinq_set_fac3 (...)	INQUIRE SET FACILITIES
pinq_set_meas_def (...)	INQUIRE SET MEASURE DEFINITION
pinq_set_st (...)	INQUIRE SET DEVICE STATE
pinq_set_st3 (...)	INQUIRE SET DEVICE STATE 3
pinq_string_fac3 (...)	INQUIRE STRING FACILITIES
pinq_stroke_fac3 (...)	INQUIRE STROKE FACILITIES
pinq_targ_assoc_trav_res (...)	INQUIRE TARGET ASSOCIATED WITH TRAVERSAL RESOURCE
pinq_targ_dispos (...)	INQUIRE TARGET DISPOSITION
pinq_targ_fac3 (...)	INQUIRE TARGET FACILITIES
pinq_targ_manip_mode (...)	INQUIRE TARGET MANIPULATION MODE
pinq_targ_st (...)	INQUIRE TARGET STATE
pinq_texture_fac3 (...)	INQUIRE TEXTURE FACILITIES
pinq_texture_map_fac3 (...)	INQUIRE TEXTURE MAPPING FACILITIES
pinq_texture_rep (...)	INQUIRE TEXTURE REPRESENTATION
pinq_transparency_mode (...)	INQUIRE TRANSPARENCY MODE
pinq_transparency_thresholds (...)	INQUIRE TRANSPARENCY THRESHOLDS
pinq_trav_res_assoc_targ (...)	INQUIRE TRAVERSAL RESOURCES ASSOCIATED WITH TARGET
pinq_trav_res_fac3 (...)	INQUIRE TRAVERSAL RESOURCE FACILITIES
pinq_val_fac3 (...)	INQUIRE VALUATOR FACILITIES
pinq_view_plane_dist (...)	INQUIRE VIEW PLANE DISTANCE
pinq_view_plane_norm (...)	INQUIRE VIEW PLANE NORMAL
pinq_view_ref_point (...)	INQUIRE VIEW REFERENCE POINT
pinq_view_ref_point3 (...)	INQUIRE VIEW REFERENCE POINT 3
pinq_view_status (...)	INQUIRE VIEW STATUS
pinq_view_up_vec (...)	INQUIRE VIEW UP VECTOR
pinq_view_up_vec3 (...)	INQUIRE VIEW UP VECTOR 3
pinq_view_win_limits (...)	INQUIRE VIEW WINDOW LIMITS
pinq_watch_elem_range (...)	INQUIRE WATCH ON ELEMENT RANGE
pinq_watch_range_st (...)	INQUIRE WATCH RANGE STATE
pinq_wss_di_struct_posted (...)	INQUIRE SET OF WORKSTATIONS TO WHICH DIRECT INTERPRETATION STRUCTURE POSTED
pinq_ws_st_table_highl (...)	INQUIRE WORKSTATION STATE TABLE LENGTHS HIGHLIGHTING
pinq_ws_st_table_texture (...)	INQUIRE WORKSTATION STATE TABLE LENGTHS TEXTURE
pinq_xy_clip_indicator (...)	INQUIRE X-Y CLIPPING INDICATOR
pinst_struct (...)	INSTANCE STRUCTURE
pmanip_trav_res (...)	MANIPULATE TRAVERSAL RESOURCE
pmap_dc_point_to_pick_paths (...)	MAP DEVICE COORDINATE POINT TO PICK PATHS

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pmap_dc_to_wc (...)	MAP DEVICE COORDINATES TO WORLD COORDINATES
pmap_dc_to_wsc (...)	MAP DEVICE COORDINATES TO WINDOW SYSTEM COORDINATES
pmap_wsc_to_dc (...)	MAP WINDOW SYSTEM COORDINATES TO DEVICE COORDINATES
pmark_multi_pass_compl (...)	MARK MULTI-PASS COMPLETION
pmark_multi_pass_start (...)	MARK MULTI-PASS START
pmark_pass_compl (...)	MARK PASS COMPLETION
pmark_pass_start (...)	MARK PASS START
pmove_elem_struct (...)	MOVE ELEMENT FROM STRUCTURE
pmove_elem_range_struct (...)	MOVE ELEMENT RANGE FROM STRUCTURE
pmove_elems_between_labels_struct (...)	MOVE ELEMENTS BETWEEN LABELS FROM STRUCTURE
popen_di_struct (...)	OPEN DIRECT INTERPRETATION STRUCTURE
ppop_st (...)	POP STATE
ppost_di_struct (...)	POST DIRECT INTERPRETATION STRUCTURE
ppost_struct_to_grp (...)	POST STRUCTURE TO GROUP
ppush_st (...)	PUSH STATE
predraw_all_from_grp_on_targ (	REDRAW ALL STRUCTURES FROM POSTING GROUP ON TARGET
predraw_all_structs_from_grp (...)	REDRAW ALL STRUCTURES FROM POSTING GROUP
predraw_all_structs_on_targ (...)	REDRAW ALL STRUCTURES ON TARGET
prenew_di_state (...)	RENEW DIRECT INTERPRETATION STATE
preq_composite (...)	REQUEST COMPOSITE
preq_composite3 (...)	REQUEST COMPOSITE 3
preq_set (...)	REQUEST SET
preq_set3 (...)	REQUEST SET 3
preset_all_trav_res (...)	RESET ALL TRAVERSAL RESOURCES
pret_num_passes_req (...)	RETRIEVE NUMBER OF PASSES REQUIRED
pret_window_system_colr (...)	RETRIEVE WINDOW SYSTEM COLOUR
psample_composite (...)	SAMPLE COMPOSITE
psample_composite3 (...)	SAMPLE COMPOSITE 3
psample_set (...)	SAMPLE SET
psample_set3 (...)	SAMPLE SET 3
pset_active_textures (...)	SET ACTIVE TEXTURES
pset_alpha_data_sel_ind (...)	SET ALPHA DATA SELECTION INDEX
pset_alpha_src_sel (...)	SET ALPHA SOURCE SELECTOR
pset_appl_filter (...)	SET APPLICATION FILTER
pset_appl_int (...)	SET APPLICATION INTEGER
pset_appl_real (...)	SET APPLICATION REAL
pset_back_active_textures (...)	SET BACK ACTIVE TEXTURES
pset_back_clip_indicator (...)	SET BACK CLIPPING INDICATOR
pset_back_plane_dist (...)	SET BACK PLANE DISTANCE
pset_back_transparency (...)	SET BACK TRANSPARENCY
pset_composite_mode (...)	SET COMPOSITE MODE

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pset_composite_pick_filter (...)	SET COMPOSITE PICK FILTER
pset_cond_flags (...)	SET CONDITION FLAGS
pset_cond_flags_from_tests (...)	SET CONDITION FLAGS FROM TESTS
pset_dc_clip_regions (...)	SET DEVICE COORDINATE CLIP REGIONS
pset_dc_clip_regions3 (...)	SET DEVICE COORDINATE CLIP REGIONS 3
pset_depth_cue_rep_mask (...)	SET DEPTH CUE REPRESENTATION MASK
pset_di_mode (...)	SET DIRECT INTERPRETATION MODE
pset_di_pick_corr_point (...)	SET DIRECT INTERPRETATION PICK CORRELATION POINT
pset_di_pick_corr_point3 (...)	SET DIRECT INTERPRETATION PICK CORRELATION POINT 3
pset_di_pick_filter (...)	SET DIRECT INTERPRETATION PICK FILTER
pset_disp_targ (...)	SET DISPLAY TARGET
pset_edgcap (...)	SET EDGE CAP
pset_edgejoin (...)	SET EDGE JOIN
pset_edgemitre_limit (...)	SET EDGEMITRE LIMIT
pset_edge_rep_full (...)	SET EDGE REPRESENTATION FULL
pset_edge_rep_mask (...)	SET EDGE REPRESENTATION MASK
pset_edgetype_adapt (...)	SET EDGETYPE ADAPTABILITY
pset_edgetype_cont (...)	SET EDGETYPE CONTINUITY
pset_edgetype_offset (...)	SET EDGETYPE OFFSET
pset_ext_pat_rep (...)	SET EXTENDED PATTERN REPRESENTATION
pset_front_clip_indicator (...)	SET FRONT CLIPPING INDICATOR
pset_front_plane_dist (...)	SET FRONT PLANE DISTANCE
pset_highl_ind (...)	SET HIGHLIGHTING INDEX
pset_highl_method (...)	SET HIGHLIGHTING METHOD
pset_highl_rep (...)	SET HIGHLIGHTING REPRESENTATION
pset_int_rep_mask (...)	SET INTERIOR REPRESENTATION MASK
pset_linecap (...)	SET LINE CAP
pset_linejoin (...)	SET LINE JOIN
pset_linemitre_limit (...)	SET LINEMITRE LIMIT
pset_line_rep_full (...)	SET POLYLINE REPRESENTATION FULL
pset_line_rep_mask (...)	SET POLYLINE REPRESENTATION MASK
pset_linetype_adapt (...)	SET LINETYPE ADAPTABILITY
pset_linetype_cont (...)	SET LINETYPE CONTINUITY
pset_linetype_offset (...)	SET LINETYPE OFFSET
pset_marker_rep_mask (...)	SET POLYMARKER REPRESENTATION MASK
pset_pat_rep_mask (...)	SET PATTERN REPRESENTATION MASK
pset_pick_mapping_data (...)	SET PICK MAPPING DATA
pset_post_grp_backg_method (...)	SET POSTING GROUP BACKGROUND METHOD
pset_post_grp_backg_style (...)	SET POSTING GROUP BACKGROUND STYLE
pset_post_grp_border_ind (...)	SET POSTING GROUP BORDER INDEX
pset_post_grp_border_indicator (...)	SET POSTING GROUP BORDER INDICATOR
pset_post_grp_priority (...)	SET POSTING GROUP PRIORITY
pset_post_grp_status (...)	SET POSTING GROUP STATUS
pset_proj_ref_point (...)	SET PROJECTION REFERENCE POINT
pset_proj_type (...)	SET PROJECTION TYPE

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pset_proj_vp (...)	SET PROJECTION VIEWPORT
pset_proj_vp3 (...)	SET PROJECTION VIEWPORT 3
pset_refl_rep_mask (...)	SET REFLECTANCE REPRESENTATION MASK
pset_rend_targ (...)	SET RENDERING TARGET
pset_set_mode (...)	SET SET MODE
pset_set_pick_filter (...)	SET SET PICK FILTER
pset_st_visual_rep (...)	SET STATE OF VISUAL REPRESENTATION
pset_targ_dispos (...)	SET TARGET DISPOSITION
pset_targ_manip_mode (...)	SET TARGET MANIPULATION MODE
pset_targ_st_visual_rep (...)	SET TARGET STATE OF VISUAL REPRESENTATION
pset_text_rep_mask (...)	SET TEXT REPRESENTATION MASK
pset_texture_binding (...)	SET TEXTURE BINDING
pset_texture_composition (...)	SET TEXTURE COMPOSITION
pset_texture_param (...)	SET TEXTURE PARAMETRIZATION
pset_texture_perspect_corr (...)	SET TEXTURE PERSPECTIVE CORRECTION
pset_texture_rep (...)	SET TEXTURE REPRESENTATION
pset_texture_rep_mask (...)	SET TEXTURE REPRESENTATION MASK
pset_texture_res_opt_heur (...)	SET TEXTURE RESOURCE OPTIMIZATION HEURISTICS
pset_texture_sampling (...)	SET TEXTURE SAMPLING
pset_texture_sampling_freq (...)	SET TEXTURE SAMPLING FREQUENCY
pset_transparency (...)	SET TRANSPARENCY
pset_transparency_mode (...)	SET TRANSPARENCY MODE
pset_transparency_thresholds (...)	SET TRANSPARENCY THRESHOLDS
pset_view_plane_dist (...)	SET VIEW PLANE DISTANCE
pset_view_plane_norm (...)	SET VIEW PLANE NORMAL
pset_view_ref_point (...)	SET VIEW REFERENCE POINT
pset_view_ref_point3 (...)	SET VIEW REFERENCE POINT 3
pset_view_up_vec (...)	SET VIEW UP VECTOR
pset_view_up_vec3 (...)	SET VIEW UP VECTOR 3
pset_view_win_limits (...)	SET VIEW WINDOW LIMITS
pset_watch_on_elem_range (...)	SET WATCH ON ELEMENT RANGE
pset_xy_clip_indicator (...)	SET X-Y CLIPPING INDICATOR
ptrans_di_pick_set (...)	TRANSFER DIRECT INTERPRETATION PICK SET
pundefine_choice (...)	UNDEFINE CHOICE
pundefine_composite (...)	UNDEFINE COMPOSITE
pundefine_locator (...)	UNDEFINE LOCATOR
pundefine_pick (...)	UNDEFINE PICK
pundefine_post_grp (...)	UNDEFINE POSTING GROUP
pundefine_set (...)	UNDEFINE SET
pundefine_string (...)	UNDEFINE STRING
pundefine_stroke (...)	UNDEFINE STROKE
pundefine_valuator (...)	UNDEFINE VALUATOR
punpost_all_structs_from_grp (...)	UNPOST ALL STRUCTURES FROM GROUP
punpost_di_struct (...)	UNPOST DIRECT INTERPRETATION STRUCTURE
punpost_structs_from_grps (...)	UNPOST STRUCTURES FROM GROUPS

Table 2 - Function names ordered by bound name (Continued)

C Name	PHIGS and PHIGS PLUS Name
pupd_targ (...)	UPDATE TARGET
pupd_view_rep (...)	UPDATE VIEW REPRESENTATION
pws_type_create (...)	WORKSTATION TYPE CREATE
pws_type_destroy (...)	WORKSTATION TYPE DESTROY
pws_type_get (...)	WORKSTATION TYPE GET
pws_type_set (...)	WORKSTATION TYPE SET

”

*Pages 20 to 26***4.3.2 List ordered alphabetically by PHIGS and PHIGS PLUS name***The following function names are merged alphabetically by function name in Table 3:*

“

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name

PHIGS and PHIGS PLUS Name	C Name
ASSOCIATE IMAGE RESOURCE	passoc_image_res (...)
ASSOCIATE TRAVERSAL RESOURCE	passoc_trav_res (...)
ATTACH LOGICAL INPUT DEVICE TO LIGHT SOURCE	pattach_lid_to_light_src (...)
ATTACH LOGICAL INPUT DEVICE TO VIEW	pattach_lid_to_view (...)
CIRCLE	pcircle (...)
CIRCLE 3	pcircle3 (...)
CIRCULAR ARC	pcircular_arc (...)
CIRCULAR ARC 3	pcircular_arc3 (...)
CIRCULAR ARC CLOSE	pcircular_arc_close (...)
CIRCULAR ARC CLOSE 3	pcircular_arc_close3 (...)
CLEAR TARGET	pclear_targ (...)
CLOSE DIRECT INTERPRETATION STRUCTURE	pclose_di_struct (...)
CONDITIONAL EXECUTE STRUCTURE	pcond_exec_struct (...)
CONDITIONAL INSTANCE STRUCTURE	pcond_inst_struct (...)
CONDITIONAL RETURN	pcond_return (...)
CONDITIONAL SKIP ELEMENTS	pcond_skip_elements (...)
CONDITIONAL SKIP TO LABEL	pcond_skip_to_label (...)
COPY ELEMENT FROM STRUCTURE	pcopy_elem_struct (...)
COPY ELEMENT RANGE FROM STRUCTURE	pcopy_elem_range_struct (...)
COPY ELEMENTS BETWEEN LABELS FROM STRUCTURE	pcopy_elems_between_labels_struct (...)
COPY TARGET	pcopy_targ (...)
CREATE COMPOSITE MEASURE	pcreate_composite_measure (...)
CREATE MIPMAP TEXTURE	pcreate_mipmap_texture (...)
CREATE SET MEASURE	pcreate_set_measure (...)
CREATE TARGET	pcreate_targ (...)
DEFINE CHOICE	pdefine_choice (...)
DEFINE COMPOSITE	pdefine_composite (...)
DEFINE LINETYPE	pdefine_linetype (...)
DEFINE LOCATOR	pdefine_locator (...)
DEFINE MARKER TYPE	pdefine_marker_type (...)
DEFINE PICK	pdefine_pick (...)
DEFINE POSTING GROUP	pdefine_post_grp (...)
DEFINE SET	pdefine_set (...)
DEFINE STRING	pdefine_string (...)
DEFINE STROKE	pdefine_stroke (...)
DEFINE VALUATOR	pdefine_valuator (...)
DESTROY COMPOSITE MEASURE	pdestroy_composite_measure (...)
DESTROY SET MEASURE	pdestroy_set_measure (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
DESTROY TARGET	pdestroy_targ (...)
DETACH LOGICAL INPUT DEVICE FROM LIGHT SOURCE	pdetach_lid_from_light_src (...)
DETACH LOGICAL INPUT DEVICE FROM VIEW	pdetach_lid_from_view (...)
DISABLE DIRECT INTERPRETATION PICK	pdisable_di_pick (...)
DISASSOCIATE IMAGE RESOURCE	pdisassoc_image_res (...)
DISASSOCIATE TRAVERSAL RESOURCE	pdisassoc_trav_res (...)
ELLIPSE	pellipse (...)
ELLIPSE 3	pellipse3 (...)
ELLIPTICAL ARC	pelliptical_arc (...)
ELLIPTICAL ARC 3	pelliptical_arc3 (...)
ELLIPTICAL ARC CLOSE	pelliptical_arc_close (...)
ELLIPTICAL ARC CLOSE 3	pelliptical_arc_close3 (...)
ENABLE DIRECT INTERPRETATION PICK	penable_di_pick (...)
END WATCH ON ELEMENT RANGE	pend_watch_on_elem_range (...)
FILL CIRCLE	pfill_circle (...)
FILL CIRCLE 3	pfill_circle3 (...)
FILL ELLIPSE	pfill_ellipse (...)
FILL ELLIPSE 3	pfill_ellipse3 (...)
GET COMPOSITE	pget_composite (...)
GET COMPOSITE 3	pget_composite3 (...)
GET SET	pget_set (...)
GET SET 3	pget_set3 (...)
INITIALIZE COMPOSITE	pinit_composite (...)
INITIALIZE COMPOSITE 3	pinit_composite3 (...)
INITIALIZE DIRECT INTERPRETATION PICK	pinit_di_pick (...)
INITIALIZE DIRECT INTERPRETATION PICK 3	pinit_di_pick3 (...)
INITIALIZE SET	pinit_set (...)
INITIALIZE SET 3	pinit_set3 (...)
INQUIRE ALPHA FACILITIES	pinq_alpha_facs (...)
INQUIRE APPLICATION FILTER	pinq_appl_filter (...)
INQUIRE ASSOCIATED IMAGE RESOURCES	pinq_assoc_image_res (...)
INQUIRE BACK CLIPPING INDICATOR	pinq_back_clip_indicator (...)
INQUIRE BACK PLANE DISTANCE	pinq_back_plane_dist (...)
INQUIRE CHOICE FACILITIES	pinq_choice_facs (...)
INQUIRE COMPOSITE DEVICE STATE	pinq_composite_st (...)
INQUIRE COMPOSITE DEVICE STATE 3	pinq_composite_st3 (...)
INQUIRE COMPOSITE FACILITIES	pinq_composite_facs (...)
INQUIRE COMPOSITE MEASURE DEFINITION	pinq_composite_meas_def (...)
INQUIRE CONDITIONAL TRAVERSAL FACILITIES	pinq_cond_trav_facs (...)
INQUIRE CONFIGURATION SETTING FACILITIES	pinq_config_setting_facs (...)
INQUIRE DEFAULT COMPOSITE DEVICE DATA	pinq_def_composite_data (...)
INQUIRE DEFAULT COMPOSITE DEVICE DATA 3	pinq_def_composite_data3 (...)
INQUIRE DEFAULT DIRECT INTERPRETATION PICK DATA	pinq_def_di_pick_data (...)
INQUIRE DEFAULT DIRECT INTERPRETATION PICK DATA 3	pinq_def_di_pick_data3 (...)
INQUIRE DEFAULT SET DEVICE DATA	pinq_def_set_data (...)
INQUIRE DEFAULT SET DEVICE DATA 3	pinq_def_set_data3 (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
INQUIRE DEFAULT TARGET DISPOSITION	pinq_def_targ_disp (...)
INQUIRE DEVICE COORDINATE CLIP REGIONS	pinq_dc_clip_regions (...)
INQUIRE DEVICE COORDINATE CLIP REGIONS 3	pinq_dc_clip_regions3 (...)
INQUIRE DEVICE COORDINATE CLIP REGIONS FACILITIES	pinq_dc_clip_regions_facs(...)
INQUIRE DIRECT INTERPRETATION MODE	pinq_di_mode (...)
INQUIRE DIRECT INTERPRETATION PICK CORRELATION POINT	pinq_di_pick_corr_point (...)
INQUIRE DIRECT INTERPRETATION PICK CORRELATION POINT 3	pinq_di_pick_corr_point3 (...)
INQUIRE DIRECT INTERPRETATION PICK SET STATUS	pinq_di_pick_set_status (...)
INQUIRE DIRECT INTERPRETATION PICK STATE	pinq_di_pick_st (...)
INQUIRE DIRECT INTERPRETATION PICK STATE 3	pinq_di_pick_st3 (...)
INQUIRE DIRECT INTERPRETATION TRAVERSAL FACILITIES	pinq_di_trav_facs (...)
INQUIRE DISPLAY TARGET	pinq_disp_targ (...)
INQUIRE DYNAMICS OF POSTING GROUPS	pinq_dyns_post_grps (...)
INQUIRE DYNAMICS OF WORKSTATION ATTRIBUTES TEXTURE	pinq_dyns_ws_attrs_texture (...)
INQUIRE EDGE REPRESENTATION FULL	pinq_edge_rep_full (...)
INQUIRE EXTENDED PATTERN FACILITIES	pinq_ext_pat_facs (...)
INQUIRE EXTENDED PATTERN REPRESENTATION	pinq_ext_pat_rep (...)
INQUIRE FRONT CLIPPING INDICATOR	pinq_front_clip_indicator (...)
INQUIRE FRONT PLANE DISTANCE	pinq_front_plane_dist (...)
INQUIRE HIGHLIGHTING FACILITIES	pinq_highl_facs (...)
INQUIRE HIGHLIGHTING REPRESENTATION	pinq_highl_rep (...)
INQUIRE IMAGE RESOURCE	pinq_image_res (...)
INQUIRE IMAGE RESOURCE FACILITIES	pinq_image_res_facs (...)
INQUIRE LINETYPE DEFINITION	pinq_linetype_def (...)
INQUIRE LINETYPE DEFINITION FACILITIES	pinq_linetype_def_facs (...)
INQUIRE LINETYPE DEFINITION SUPPORT	pinq_linetype_def_support (...)
INQUIRE LIST OF DEFINED APPLICATION FILTERS	pinq_list_def_appl_filters (...)
INQUIRE LIST OF HIGHLIGHTING INDICES	pinq_list_highl_inds (...)
INQUIRE LIST OF POSTING GROUPS	pinq_list_post_grps (...)
INQUIRE LIST OF TEXTURE INDICES	pinq_list_texture_inds (...)
INQUIRE LISTS OF AVAILABLE LOGICAL INPUT DEVICES	pinq_lists_avail_lids (...)
INQUIRE LOCATOR FACILITIES	pinq_loc_facs (...)
INQUIRE LOGICAL INPUT DEVICE ATTACHED TO LIGHT SOURCE	pinq_lid_attached_to_light_src (...)
INQUIRE LOGICAL INPUT DEVICE ATTACHED TO VIEW	pinq_lid_attached_to_view (...)
INQUIRE LOGICAL INPUT DEVICE DEFINITION	pinq_lid_def (...)
INQUIRE MARKER TYPE DEFINITION	pinq_marker_type_def (...)
INQUIRE MARKER TYPE DEFINITION FACILITIES	pinq_marker_type_def_facs (...)
INQUIRE MARKER TYPE DEFINITION SUPPORT	pinq_marker_type_def_support (...)
INQUIRE MIPMAP FACILITIES	pinq_mipmap_facs (...)
INQUIRE NUMBER OF AVAILABLE NON-ATOMIC LOGICAL INPUT DEVICES	pinq_num_avail_na_in (...)
INQUIRE NUMBER OF DEFINED LINETYPES	pinq_num_def_linetypes (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
INQUIRE NUMBER OF DEFINED MARKER TYPES	pinq_num_def_marker_types (...)
INQUIRE NUMBER OF PREDEFINED APPLICATION FILTERS	pinq_num_pred_appl_filters (...)
INQUIRE PICK FACILITIES	pinq_pick_facs (...)
INQUIRE PICK MAPPING FACILITIES	pinq_pick_mapping_facs (...)
INQUIRE PICK MAPPING STATE	pinq_pick_mapping_state (...)
INQUIRE PICTURE STATUS	pinq_pict_status (...)
INQUIRE POLYLINE REPRESENTATION FULL	pinq_line_rep_full (...)
INQUIRE POSTED DIRECT INTERPRETATION STRUCTURE	pinq_posted_di_struct (...)
INQUIRE POSTED STRUCTURES FROM POSTING GROUP	pinq_posted_structs_from_post_grp (...)
INQUIRE POSTING GROUP	pinq_post_grp (...)
INQUIRE POSTING GROUP FACILITIES	pinq_post_grp_facs (...)
INQUIRE PREDEFINED APPLICATION FILTER	pinq_pred_appl_filter (...)
INQUIRE PREDEFINED ASSOCIATION OF TARGET WITH TRAVERSAL RESOURCES	pinq_pred_assoc_targ_trav_res (...)
INQUIRE PREDEFINED ASSOCIATION OF TRAVERSAL RESOURCE WITH TARGET	pinq_pred_assoc_trav_res_targ (...)
INQUIRE PREDEFINED COMPOSITE MEASURE DEFINITION	pinq_pred_composite_meas_def (...)
INQUIRE PREDEFINED EDGE REPRESENTATION FULL	pinq_pred_edge_rep_full (...)
INQUIRE PREDEFINED EXTENDED PATTERN REPRESENTATION	pinq_pred_ext_pat_rep (...)
INQUIRE PREDEFINED HIGHLIGHTING REPRESENTATION	pinq_pred_highl_rep (...)
INQUIRE PREDEFINED IMAGE RESOURCE	pinq_pred_image_res (...)
INQUIRE PREDEFINED LINETYPE DEFINITION	pinq_pred_linetype_def (...)
INQUIRE PREDEFINED MARKER TYPE DEFINITION	pinq_pred_marker_type_def (...)
INQUIRE PREDEFINED POLYLINE REPRESENTATION FULL	pinq_pred_line_rep_full (...)
INQUIRE PREDEFINED POSTING GROUP	pinq_pred_post_grp (...)
INQUIRE PREDEFINED SET MEASURE DEFINITION	pinq_pred_set_meas_def (...)
INQUIRE PREDEFINED TEXTURE REPRESENTATION	pinq_pred_texture_rep (...)
INQUIRE PROJECTION REFERENCE POINT	pinq_proj_ref_point (...)
INQUIRE PROJECTION TYPE	pinq_proj_type (...)
INQUIRE PROJECTION VIEWPORT	pinq_proj_vp (...)
INQUIRE PROJECTION VIEWPORT 3	pinq_proj_vp3 (...)
INQUIRE RENDERING TARGET	pinq_rend_targ (...)
INQUIRE SET DEVICE STATE	pinq_set_st (...)
INQUIRE SET DEVICE STATE 3	pinq_set_st3 (...)
INQUIRE SET FACILITIES	pinq_set_facs (...)
INQUIRE SET MEASURE DEFINITION	pinq_set_meas_def (...)
INQUIRE SET OF GROUPS TO WHICH POSTED	pinq_grps_posted (...)
INQUIRE SET OF WORKSTATIONS TO WHICH DIRECT INTERPRETATION STRUCTURE POSTED	pinq_wss_di_struct_posted (...)
INQUIRE STRING FACILITIES	pinq_string_facs (...)
INQUIRE STROKE FACILITIES	pinq_stroke_facs (...)
INQUIRE TARGET ASSOCIATED WITH TRAVERSAL RESOURCE	pinq_targ_assoc_trav_res (...)
INQUIRE TARGET DISPOSITION	pinq_targ_dispos (...)
INQUIRE TARGET FACILITIES	pinq_targ_facs (...)
INQUIRE TARGET MANIPULATION MODE	pinq_targ_manip_mode (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
INQUIRE TARGET STATE	pinq_targ_st (...)
INQUIRE TEXTURE FACILITIES	pinq_texture_facs (...)
INQUIRE TEXTURE MAPPING FACILITIES	pinq_texture_map_facs (...)
INQUIRE TEXTURE REPRESENTATION	pinq_texture_rep (...)
INQUIRE TRANSPARENCY MODE	pinq_transparency_mode (...)
INQUIRE TRANSPARENCY THRESHOLDS	pinq_transparency_thresholds (...)
INQUIRE TRAVERSAL RESOURCE FACILITIES	pinq_trav_res_facs (...)
INQUIRE TRAVERSAL RESOURCES ASSOCIATED WITH TARGET	pinq_trav_res_assoc_targ (...)
INQUIRE VALUATOR FACILITIES	pinq_val_facs (...)
INQUIRE VIEW PLANE DISTANCE	pinq_view_plane_dist (...)
INQUIRE VIEW PLANE NORMAL	pinq_view_plane_norm (...)
INQUIRE VIEW REFERENCE POINT	pinq_view_ref_point (...)
INQUIRE VIEW REFERENCE POINT 3	pinq_view_ref_point3 (...)
INQUIRE VIEW STATUS	pinq_view_status (...)
INQUIRE VIEW UP VECTOR	pinq_view_up_vec (...)
INQUIRE VIEW UP VECTOR 3	pinq_view_up_vec3 (...)
INQUIRE VIEW WINDOW LIMITS	pinq_view_win_limits (...)
INQUIRE WATCH ON ELEMENT RANGE	pinq_watch_elem_range (...)
INQUIRE WATCH RANGE STATE	pinq_watch_range_st (...)
INQUIRE WORKSTATION STATE TABLE LENGTHS HIGHLIGHTING	pinq_ws_st_table_highl (...)
INQUIRE WORKSTATION STATE TABLE LENGTHS TEXTURE	pinq_ws_st_table_texture (...)
INQUIRE X-Y CLIPPING INDICATOR	pinq_xy_clip_indicator (...)
INSTANCE STRUCTURE	pinst_struct (...)
MANIPULATE TRAVERSAL RESOURCE	pmanip_trav_res (...)
MAP DEVICE COORDINATE POINT TO PICK PATHS	pmap_dc_point_to_pick_paths (...)
MAP DEVICE COORDINATES TO WINDOW SYSTEM COORDINATES	pmap_dc_to_wsc (...)
MAP DEVICE COORDINATES TO WORLD COORDINATES	pmap_dc_to_wc (...)
MAP WINDOW SYSTEM COORDINATES TO DEVICE COORDINATES	pmap_wsc_to_dc (...)
MARK MULTI-PASS COMPLETION	pmark_multi_pass_compl (...)
MARK MULTI-PASS START	pmark_multi_pass_start (...)
MARK PASS COMPLETION	pmark_pass_compl (...)
MARK PASS START	pmark_pass_start (...)
MOVE ELEMENT FROM STRUCTURE	pmove_elem_struct (...)
MOVE ELEMENT RANGE FROM STRUCTURE	pmove_elem_range_struct (...)
MOVE ELEMENTS BETWEEN LABELS FROM STRUCTURE	pmove_elems_between_labels_struct (...)
OPEN DIRECT INTERPRETATION STRUCTURE	popen_di_struct (...)
POP STATE	ppop_st (...)
POST DIRECT INTERPRETATION STRUCTURE	ppost_di_struct (...)
POST DIRECT INTERPRETATION STRUCTURE TO POSTING GROUP	ppost_di_struct_to_grp (...)
POST STRUCTURE TO GROUP	ppost_struct_to_grp (...)
PUSH STATE	ppush_st (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
REDRAW ALL STRUCTURES FROM POSTING GROUP	predraw_all_structs_from_grp (...)
REDRAW ALL STRUCTURES FROM POSTING GROUP ON TARGET	predraw_all_from_grp_on_targ (...)
REDRAW ALL STRUCTURES ON TARGET	predraw_all_structs_on_targ (...)
RENEW DIRECT INTERPRETATION STATE	prenew_di_state (...)
REQUEST COMPOSITE	preq_composite (...)
REQUEST COMPOSITE 3	preq_composite3 (...)
REQUEST SET	preq_set (...)
REQUEST SET 3	preq_set3 (...)
RESET ALL TRAVERSAL RESOURCES	preset_all_trav_res (...)
RETRIEVE NUMBER OF PASSES REQUIRED	pret_num_passes_req (...)
RETRIEVE WINDOW SYSTEM COLOUR	pret_window_system_colr (...)
SAMPLE COMPOSITE	psample_composite (...)
SAMPLE COMPOSITE 3	psample_composite3 (...)
SAMPLE SET	psample_set (...)
SAMPLE SET 3	psample_set3 (...)
SET ACTIVE TEXTURES	pset_active_textures (...)
SET ALPHA DATA SELECTION INDEX	pset_alpha_data_sel_ind (...)
SET ALPHA SOURCE SELECTOR	pset_alpha_src_sel (...)
SET APPLICATION FILTER	pset_appl_filter (...)
SET APPLICATION INTEGER	pset_appl_int (...)
SET APPLICATION REAL	pset_appl_real (...)
SET BACK ACTIVE TEXTURES	pset_back_active_textures (...)
SET BACK CLIPPING INDICATOR	pset_back_clip_indicator (...)
SET BACK PLANE DISTANCE	pset_back_plane_dist (...)
SET BACK TRANSPARENCY	pset_back_transparency (...)
SET COMPOSITE MODE	pset_composite_mode (...)
SET COMPOSITE PICK FILTER	pset_composite_pick_filter (...)
SET CONDITION FLAGS	pset_cond_flags (...)
SET CONDITION FLAGS FROM TESTS	pset_cond_flags_from_tests (...)
SET DEPTH CUE REPRESENTATION MASK	pset_depth_cue_rep_mask (...)
SET DEVICE COORDINATE CLIP REGIONS	pset_dc_clip_regions (...)
SET DEVICE COORDINATE CLIP REGIONS 3	pset_dc_clip_regions3 (...)
SET DIRECT INTERPRETATION MODE	pset_di_mode (...)
SET DIRECT INTERPRETATION PICK CORRELATION POINT	pset_di_pick_corr_point (...)
SET DIRECT INTERPRETATION PICK CORRELATION POINT 3	pset_di_pick_corr_point3 (...)
SET DIRECT INTERPRETATION PICK FILTER	pset_di_pick_filter (...)
SET DISPLAY TARGET	pset_disp_targ (...)
SET EDGE CAP	pset_edgcap (...)
SET EDGE JOIN	pset_edgjoin (...)
SET EDGEMITRE LIMIT	pset_edgemitre_limit (...)
SET EDGE REPRESENTATION FULL	pset_edge_rep_full (...)
SET EDGE REPRESENTATION MASK	pset_edge_rep_mask (...)
SET EDGETYPE ADAPTABILITY	pset_edgetype_adapt (...)
SET EDGETYPE CONTINUITY	pset_edgetype_cont (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
SET EDGETYPE OFFSET	pset_edgetype_offset (...)
SET EXTENDED PATTERN REPRESENTATION	pset_ext_pat_rep (...)
SET FRONT CLIPPING INDICATOR	pset_front_clip_indicator (...)
SET FRONT PLANE DISTANCE	pset_front_plane_dist (...)
SET HIGHLIGHTING INDEX	pset_highl_ind (...)
SET HIGHLIGHTING METHOD	pset_highl_method (...)
SET HIGHLIGHTING REPRESENTATION	pset_highl_rep (...)
SET INTERIOR REPRESENTATION MASK	pset_int_rep_mask (...)
SET LINECAP	pset_linecap (...)
SET LINEJOIN	pset_linejoin (...)
SET LINEMITRE LIMIT	pset_linemitre_limit (...)
SET LINETYPE ADAPTABILITY	pset_linetype_adapt (...)
SET LINETYPE CONTINUITY	pset_linetype_cont (...)
SET LINETYPE OFFSET	pset_linetype_offset (...)
SET PATTERN REPRESENTATION MASK	pset_pat_rep_mask (...)
SET PICK MAPPING DATA	pset_pick_mapping_data (...)
SET POLYLINE REPRESENTATION FULL	pset_line_rep_full (...)
SET POLYLINE REPRESENTATION MASK	pset_line_rep_mask (...)
SET POLYMARKER REPRESENTATION MASK	pset_marker_rep_mask (...)
SET POSTING GROUP BACKGROUND METHOD	pset_post_grp_backg_method (...)
SET POSTING GROUP BACKGROUND STYLE	pset_post_grp_backg_style (...)
SET POSTING GROUP BORDER INDEX	pset_post_grp_border_ind (...)
SET POSTING GROUP BORDER INDICATOR	pset_post_grp_border_indicator (...)
SET POSTING GROUP PRIORITY	pset_post_grp_priority (...)
SET POSTING GROUP STATUS	pset_post_grp_status (...)
SET PROJECTION REFERENCE POINT	pset_proj_ref_point (...)
SET PROJECTION TYPE	pset_proj_type (...)
SET PROJECTION VIEWPORT	pset_proj_vp (...)
SET PROJECTION VIEWPORT 3	pset_proj_vp3 (...)
SET REFLECTANCE REPRESENTATION MASK	pset_refl_rep_mask (...)
SET RENDERING TARGET	pset_rend_targ (...)
SET SET MODE	pset_set_mode (...)
SET SET PICK FILTER	pset_set_pick_filter (...)
SET STATE OF VISUAL REPRESENTATION	pset_st_visual_rep (...)
SET TARGET DISPOSITION	pset_targ_dispos (...)
SET TARGET MANIPULATION MODE	pset_targ_manip_mode (...)
SET TARGET STATE OF VISUAL REPRESENTATION	pset_targ_st_visual_rep (...)
SET TEXT REPRESENTATION MASK	pset_text_rep_mask (...)
SET TEXTURE BINDING	pset_texture_binding (...)
SET TEXTURE COMPOSITION	pset_texture_composition (...)
SET TEXTURE PARAMETRIZATION	pset_texture_param (...)
SET TEXTURE PERSPECTIVE CORRECTION	pset_texture_perspect_corr (...)
SET TEXTURE REPRESENTATION	pset_texture_rep (...)
SET TEXTURE REPRESENTATION MASK	pset_texture_rep_mask (...)
SET TEXTURE RESOURCE OPTIMIZATION HEURISTICS	pset_texture_res_opt_heur (...)
SET TEXTURE SAMPLING	pset_texture_sampling (...)

Table 3 - Function names ordered by PHIGS and PHIGS PLUS function name (Continued)

PHIGS and PHIGS PLUS Name	C Name
SET TEXTURE SAMPLING FREQUENCY	pset_texture_sampling_freq (...)
SET TRANSPARENCY	pset_transparency (...)
SET TRANSPARENCY MODE	pset_transparency_mode (...)
SET TRANSPARENCY THRESHOLDS	pset_transparency_thresholds (...)
SET VIEW PLANE DISTANCE	pset_view_plane_dist (...)
SET VIEW PLANE NORMAL	pset_view_plane_norm (...)
SET VIEW REFERENCE POINT	pset_view_ref_point (...)
SET VIEW REFERENCE POINT 3	pset_view_ref_point3 (...)
SET VIEW UP VECTOR	pset_view_up_vec (...)
SET VIEW UP VECTOR 3	pset_view_up_vec3 (...)
SET VIEW WINDOW LIMITS	pset_view_win_limits (...)
SET WATCH ON ELEMENT RANGE	pset_watch_on_elem_range (...)
SET X-Y CLIPPING INDICATOR	pset_xy_clip_indicator (...)
TRANSFER DIRECT INTERPRETATION PICK SET	ptrans_di_pick_set (...)
UNDEFINE CHOICE	pundefine_choice (...)
UNDEFINE COMPOSITE	pundefine_composite (...)
UNDEFINE LOCATOR	pundefine_locator (...)
UNDEFINE PICK	pundefine_pick (...)
UNDEFINE POSTING GROUP	pundefine_post_grp (...)
UNDEFINE SET	pundefine_set (...)
UNDEFINE STRING	pundefine_string (...)
UNDEFINE STROKE	pundefine_stroke (...)
UNDEFINE VALUATOR	pundefine_valuator (...)
UNPOST ALL STRUCTURES FROM GROUP	punpost_all_structs_from_grp (...)
UNPOST DIRECT INTERPRETATION STRUCTURE	punpost_di_struct (...)
UNPOST STRUCTURES FROM GROUPS	punpost_structs_from_grps (...)
UPDATE TARGET	pupd_targ (...)
UPDATE VIEW REPRESENTATION	pupd_view_rep (...)
WORKSTATION TYPE CREATE	pws_type_create (...)
WORKSTATION TYPE DESTROY	pws_type_destroy (...)
WORKSTATION TYPE GET	pws_type_get (...)
WORKSTATION TYPE SET	pws_type_set (...)

”

Page 27

5 C PHIGS type definitions

Change the clause heading to “C Basic PHIGS profile type definitions”.

5.1 Mapping of PHIGS data types

Change the subclause heading to “Mapping of Basic PHIGS profile data types”.

The following text replaces the text of the first paragraph:

“The Basic PHIGS profile specifies a set of abstract data types. Table 4 gives the mapping from those data types defined in the Basic PHIGS profile.”

Change the table title to “Table - 4 Mapping of Basic PHIGS profile data types to C”.

Change the left table heading to “Basic PHIGS profile data type”.

Page 27

5.2 Environmental type definitions

Append the phrase “of all profiles” to the second sentence of the first paragraph.

Page 28

5.3 Implementation dependent type definitions

Change the subclause heading to “Implementation dependent Basic PHIGS profile type definitions”.

The following text replaces the phrases of the “xxx” description concerned with the escape data records.

‘, “escape_in” for ESCAPE input data records, “escape_out” for ESCAPE output data records, “targ_op” for target operation types, “trav_res_op” for traversal resource operations, “in_type” for new classes of input measures’

Pages 29 and 30

The following type definitions replace the Pchoice_data and Pchoice_data3 type definitions:

“

Pchoice_data CHOICE DATA RECORD

```
typedef struct {
    union Pchoice_pets {
        struct Pchoice_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pchoice_pet_r1{
            . . .      /* impl. dependent */
        } pet_r1; /* For PET 1 */
        struct Pchoice_pet_r2 {
            Pint        num_prompts; /* number of prompts*/
            Ppr_switch  *prompts;    /* array of prompts*/
            . . .      /* impl. defined*/
        } pet_r2; /* For PET 2 */
    }
}
```

```

    struct Pchoice_pet_r3 {
        Pint      num_strings;      /* number of choice strings*/
        char      **strings;        /* array of choice strings*/
        . . .                      /* impl. defined*/
    } pet_r3; /* For PET 3 */
    struct Pchoice_pet_r4 {
        Pint      num_strings;      /* number of alternatives*/
        char      **strings;        /* array of strings*/
        . . .                      /* impl. defined*/
    } pet_r4; /* For PET 4 */
    struct Pchoice_pet_r5 {
        Pint      struct_id;        /* structure identifier*/
        Pint      num_pick_ids;     /* number of alternatives*/
        Pint      *pick_ids;        /* array of pick identifiers*/
        . . .                      /* impl. defined*/
    } pet_r5; /* For PET 5 */
    . . . /* data for impl. defined pets */
} pets;

union Pchoice_measure_data {
    struct Pchoice_meas_other {
        Pint      unused;
    } meas_other; /* When no measure-specific data is required */
    . . . /* data for impl. defined measure processes */
} measure_data;

union Pchoice_trigger_data {
    struct Pchoice_trig_other {
        Pint      unused;
    } trig_other; /* When no trigger-specific data is required */
    . . . /* data for impl. defined trigger processes */
} trigger_data;

union Pchoice_ack_data {
    struct Pchoice_ack_other {
        Pint      unused;
    } ack_other; /* When no ack.-specific data is required */
    . . . /* data for impl. defined ack. processes */
} acknowledgement_data;
} Pchoice_data;

```

Pchoice_data3 CHOICE DATA RECORD 3

```

typedef struct {
    union Pchoice3_pets {
        struct Pchoice3_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pchoice3_pet_r1{
            . . .      /* impl. dependent*/
        } pet_r1; /* For PET 1 */
        struct Pchoice3_pet_r2 {
            Pint        num_prompts;      /* number of prompts*/
            Ppr_switch*prompts;          /* array of prompts*/
            . . .      /* impl. defined*/
        } pet_r2; /* For PET 2 */
        struct Pchoice3_pet_r3 {
            Pint        num_strings;      /* number of choice strings*/
            char        **strings;        /* array of choice strings*/
            . . .      /* impl. defined*/
        } pet_r3; /* For PET 3 */
        struct Pchoice3_pet_r4 {
            Pint        num_strings;      /* number of alternatives*/
            char        **strings;        /* array of strings*/
            . . .      /* impl. defined*/
        } pet_r4; /* For PET 4 */
        struct Pchoice3_pet_r5 {
            Pint        struct_id;        /* structure identifier*/
            Pint        num_pick_ids;     /* number of alternatives*/
            Pint        *pick_ids;        /* array of pick identifiers*/
            . . .      /* impl. defined*/
        } pet_r5; /* For PET 5 */
        . . .      /* data for impl. defined pets */
    } pets;
    union Pchoice3_measure_data {
        struct Pchoice_meas_other {
            Pint        unused;
        } meas_other; /* When no measure-specific data is required */
        . . .      /* data for impl. defined measure processes */
    } measure_data;
}

```

```

union Pchoice3_trigger_data {
    struct Pchoice3_trig_other {
        Pint        unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .        /* data for impl. defined trigger processes */
} trigger_data;
union Pchoice3_ack_data {
    struct Pchoice3_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .        /* data for impl. defined ack. processes */
} acknowledgement_data;

} Pchoice_data3;”

```

Pages 33 and 34

The following type definition replaces the definitions of *Ploc_data* and *Ploc_data3* respectively:

“

Ploc_data LOCATOR DATA RECORD

```

typedef struct {
    union Ploc_pets {
        struct Ploc_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Ploc_pet_r1 {
            . . .        /* impl. dependent*/
        } pet_r1;      /* Supports standard PET 1 */
        struct Ploc_pet_r2 {
            . . .        /* impl. dependent*/
        } pet_r2;      /* Supports standard PET 2 */
        struct Ploc_pet_r3 {
            . . .        /* impl. dependent*/
        } pet_r3;      /* Supports standard PET 3 */
        struct Ploc_pet_r4 {
            Pline_attrs line_attrs; /* polyline attributes*/
            . . .        /* impl. dependent*/
        } pet_r4;      /* Supports standard PET 4 */
    };
};

```

```

struct Ploc_pet_r5 {
    Pline_fill_ctrl_flag line_fill_ctrl_flag; /* ctrl. flag*/
    union Ploc_attrs {
        Pline_attrs line_attrs; /* polyline attributes*/
        Pint_attrs int_attrs; /* interior attributes*/
        struct Ploc_fill_set {
            Pint_attrs int_attrs; /* interior attributes*/
            Pedge_attrs edge_attrs; /* edge attributes*/
        } fill_set;
    } attrs;
    . . . /* data for impl. defined PET's */
} pet_r5; /* Supports standard PET 5 */
struct Ploc_pet_r6 {
    . . . /* impl. dependent*/
} pet_r6; /* Supports standard PET 6 */
. . . /* data for impl. defined PET's */
} pets;

union Ploc_measure_data {
    struct Ploc_meas_other {
        Pint unused;
    } meas_other; /* When no measure-specific data is required */
    . . . /* data for impl. defined measure processes */
} measure_data;

union Ploc_trigger_data{
    struct Ploc_trig_other {
        Pint unused;
    } trig_other; /* When no trigger-specific data is required */
    . . . /* data for impl. defined trigger processes */
} trigger_data;

union Ploc_ack_data {
    struct Ploc_ack_other {
        Pint unused;
    } ack_other; /* When no ack.-specific data is required */
    . . . /* data for impl. defined ack. processes */
} acknowledgement_data;
} Ploc_data;

```

Ploc_data3 LOCATOR DATA RECORD 3

```

typedef struct {
    union Ploc3_pets {
        struct Ploc3_pet_other {
            Pint        unused;
        } pet_other;    /* When no echo-specific data is required */
        struct Ploc3_pet_r1 {
            . . .                /* impl. dependent*/
        } pet_r1;        /* Supports standard PET 1 */
        struct Ploc3_pet_r2 {
            . . .                /* impl. dependent*/
        } pet_r2;        /* Supports standard PET 2 */
        struct Ploc3_pet_r3 {
            . . .                /* impl. dependent*/
        } pet_r3;        /* Supports standard PET 3 */
        struct Ploc3_pet_r4 {
            Pline_attrs line_attrs;    /* polyline attributes*/
            . . .                /* impl. dependent*/
        } pet_r4;        /* Supports standard PET 4 */
        struct Ploc3_pet_r5 {
            Pline_fill_ctrl_flag line_fill_ctrl_flag; /* ctrl. flag*/
            union Ploc3_attrs {
                Pline_attrs line_attrs; /* polyline attributes*/
                Pint_attrs int_attrs; /* interior attributes*/
                struct Ploc3_fill_set {
                    Pint_attrs int_attrs; /* interior attributes*/
                    Pedge_attrs edge_attrs; /* edge attributes*/
                } fill_set;
            } attrs;
            . . .                /* data for impl. defined PET's */
        } pet_r5;        /* Supports standard PET 5 */
        struct Ploc3_pet_r6 {
            . . .                /* impl. dependent*/
        } pet_r6;        /* Supports standard PET 6 */
        . . .                /* data for impl. defined PET's */
    } pets;

    union Ploc3_measure_data {
        struct Ploc3_meas_other {
            Pint        unused;
        } meas_other; /* When no measure-specific data is required */
    };
};

```

```

        . . .      /* data for impl. defined measure processes */
    } measure_data;
    union Ploc3_trigger_data {
        struct Ploc3_trig_other {
            Pint      unused;
        } trig_other; /* When no trigger-specific data is required */
        . . .      /* data for impl. defined trigger processes */
    } trigger_data;
    union Ploc3_ack_data {
        struct Ploc3_ack_other {
            Pint      unused;
        } ack_other; /* When no ack.-specific data is required */
        . . .      /* data for impl. defined ack. processes */
    } acknowledgement_data;
} Ploc_data3;”

```

Pages 35 through 38

The following type definitions replace the corresponding definitions respectively:

“

Ppick_data PICK DATA RECORD

```

typedef struct {
    union Ppick_pets {
        struct Ppick_pet_other {
            Pint      unused;
        } pet_other; /* When no echo-specific data is required */
        struct Ppick_pet_r1 {
            . . .      /* impl. dependent*/
        } pet_r1; /* For PET 1 */
        . . .      /* data for impl. defined pets */
        struct Ppick_pet_r2 {
            . . .      /* implementation dependent*/
        } pet_r2; /* For PET 2 */
        struct Ppick_pet_r3 {
            . . .      /* implementation dependent*/
        } pet_r3; /* For PET 3 */
    }
}

```

```

    struct Ppick_pet_r4 {
        Pint        highl_ind;           /* highlighting index*/
        . . .                          /* impl. dependent*/
    } pet_r4; /* For PET 4 */
    struct Ppick_pet_r5 {
        Pint        highl_ind;           /* highlighting index*/
        . . .                          /* impl. dependent*/
    } pet_r5; /* For PET 5 */
    struct Ppick_pet_r6 {
        Pint        highl_ind;           /* highlighting index*/
        . . .                          /* impl. dependent*/
    } pet_r6; /* For PET 6 */
    . . .                               /* implementation defined PET's*/
} pets;

union Ppick_measure_data {
    struct Ppick_meas_other {
        Pint        unused;
    } meas_other; /* When no measure-specific data is required */
    . . .        /* data for impl. defined measure processes */
} measure_data;

union Ppick_trigger_data {
    struct Ppick_trig_other {
        Pint        unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .        /* data for impl. defined pick processes */
} trigger_data;

union Ppick_ack_data {
    struct Ppick_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .        /* data for impl. defined ack. processes */
} acknowledgement_data;
} Ppick_data;

```

Ppick_data3 PICK DATA RECORD 3

```

typedef struct {
    union Ppick3_pets {
        struct Ppick3_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Ppick3_pet_r1 {
            . . .
        } pet_r1; /* For PET 1 */
        struct Ppick3_pet_r2 {
            . . .
        } pet_r2; /* For PET 2 */
        struct Ppick3_pet_r3 {
            . . .
        } pet_r3; /* For PET 3 */
        struct Ppick3_pet_r4 {
            Pint        highl_ind;
            . . .
        } pet_r4; /* For PET 4 */
        struct Ppick3_pet_r5 {
            Pint        highl_ind;
            . . .
        } pet_r5; /* For PET 5 */
        struct Ppick3_pet_r6 {
            Pint        highl_ind;
            . . .
        } pet_r6;
        . . . /* data for impl. defined pets */
    } pets; /* For PET 6 */
    union Ppick3_measure_data {
        struct Ppick3_meas_other {
            Pint        unused;
        } meas_other; /* When no measure-specific data is required */
        . . . /* data for impl. defined measure processes */
    } measure_data;
    union Ppick3_trigger_data {
        struct Ppick3_trig_other {
            Pint        unused;
        } trig_other; /* When no trigger-specific data is required */
    }

```

```

    . . .      /* data for impl. defined pick processes */
} trigger_data;
union Ppick3_ack_data {
    struct Ppick3_ack_other {
        Pint      unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .      /* data for impl. defined ack. processes */
} acknowledgement_data;
} Ppick_data3;

```

Pstring_data **STRING DATA RECORD**

```

typedef struct {
    Pint          in_buf_size; /* input buffer size*/
    Pint          init_pos;    /* initial editing position*/
    union Pstring_pets {
        struct Pstring_pet_other {
            Pint      unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pstring_pet_r1{
            . . .      /* impl. dependent*/
        } pet_r1; /* For PET 1 */
        . . .      /* data for impl. defined pets */
    } pets;
    union Pstring_measure_data {
        struct Pstring_meas_other {
            Pint      unused;
        } meas_other; /* When no measure-specific data is required */
        . . .      /* data for impl. defined measure processes */
    } measure_data;
    union Pstring_trigger_data {
        struct Pstring_trig_other {
            Pint      unused;
        } trig_other; /* When no trigger-specific data is required */
        . . .      /* data for impl. defined trigger processes */
    } trigger_data;
}

```

```

union Pstring_ack_data {
    struct Pstring_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .        /* data for impl. defined ack. processes */
} acknowledgement_data;
} Pstring_data;

```

Pstring_data3 STRING DATA RECORD 3

```

typedef struct {
    Pint            in_buf_size; /* input buffer size*/
    Pint            init_pos;    /* initial editing position*/
    union Pstring3_pets {
        struct Pstring_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pstring3_pet_r1{
            . . .        /* impl. dependent*/
        } pet_r1; /* For PET 1 */
        . . .        /* data for impl. defined pets */
    } pets;
    union Pstring3_measure_data {
        struct Pstring3_meas_other {
            Pint        unused;
        } meas_other; /* When no measure-specific data is required */
        . . .        /* data for impl. defined measure processes */
    } measure_data;
    union Pstring3_trigger_data {
        struct Pstring3_trig_other {
            Pint        unused;
        } trig_other; /* When no trigger-specific data is required */
        . . .        /* data for impl. defined trigger processes */
    } trigger_data;
    union Pstring3_ack_data {
        struct Pstring3_ack_other {
            Pint        unused;
        } ack_other; /* When no ack.-specific data is required */
    }
}

```

```

    . . .      /* data for impl. defined ack. processes */
} acknowledgement_data;

} Pstring_data3;

```

Pstroke_data **STROKE DATA RECORD**

```

typedef struct {
    Pint          buffer_size; /* input buffer size */
    Pint          init_pos;    /* initial editing position */
    Pfloat        x_interval;  /* x trigger interval */
    Pfloat        y_interval;  /* y trigger interval */
    Pfloat        time_interval; /* time trigger interval */
    union {
        struct Pstroke_pet_other {
            Pint          unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pstroke_pet_r1 {
            . . .      /* impl. dependent */
        } pet_r1; /* For std PET 1 */
        struct Pstroke_pet_r2 {
            . . .      /* impl. dependent */
        } pet_r2; /* For std PET 2 */
        struct Pstroke_pet_r3 {
            Pmarker_attrs marker_attrs; /* marker attributes */
            . . .      /* impl. dependent */
        } pet_r3; /* For std PET 3 */
        struct Pstroke_pet_r4 {
            Pline_attrs line_attrs; /* line attributes */
            . . .      /* impl. dependent */
        } pet_r4; /* For std PET 4 */
        . . .      /* data for impl. defined PET's */
        } pets;
    union Pstroke_measure_data {
        struct Pstroke_meas_other {
            Pint          unused;
        } meas_other; /* When no measure-specific data is required */
        . . .      /* data for impl. defined measure processes */
    } measure_data;
}

```

```

union Pstroke_trigger_data {
    struct Pstroke_trig_other {
        Pint        unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .        /* data for impl. defined trigger processes */
} trigger_data;
union Pstroke_ack_data {
    struct Pstroke_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .        /* data for impl. defined ack. processes */
} acknowledgement_data;
} Pstroke_data;

```

Pstroke_data3 STROKE DATA RECORD 3

```

typedef struct {
    Pint        in_buf_size; /* input buffer size*/
    Pint        init_pos;    /* initial editing position*/
    Pfloat      x_interval;  /* x trigger interval*/
    Pfloat      y_interval;  /* y trigger interval*/
    Pfloat      z_interval;  /* z trigger interval*/
    Pfloat      time_interval; /* time trigger interval*/
    union Pstroke3_pets {
        struct Pstroke3_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        struct Pstroke3_pet_r1 {
            . . .        /* impl. dependent*/
        } pet_r1; /* For std PET 1 */
        struct Pstroke3_pet_r2 {
            . . .        /* impl. dependent */
        } pet_r2; /* For std PET 2 */
        struct Pstroke3_pet_r3 {
            Pmarker_attrs marker_attrs; /* marker attributes*/
            . . .        /* impl. dependent*/
        } pet_r3; /* For std PET 3 */
    }
}

```

```

    struct Pstroke3_pet_r4 {
        Pline_attrs line_attrs;    /* line attributes*/
        . . .                      /* impl. dependent*/
    } pet_r4; /* For std PET 4 */
    . . .    /* data for impl. defined PET's */
} pets;

union Pstroke3_measure_data {
    struct Pstroke3_meas_other {
        Pint unused;
    } meas_other; /* When no measure-specific data is required */
    . . .    /* data for impl. defined measure processes */
} measure_data;

union Pstroke3_trigger_data {
    struct Pstroke3_trig_other {
        Pint unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .    /* data for impl. defined trigger processes */
} trigger_data;

union Pstroke3_ack_data {
    struct Pstroke3_ack_other {
        Pint unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .    /* data for impl. defined ack. processes */
} acknowledgement_data;

} Pstroke_data3;

```

Pval_data VALUATOR DATA RECORD

```

typedef struct {
    Pfloat low;    /* Low value of valuator range*/
    Pfloat high;   /* High value of valuator range*/
    union Pval_pets {
        struct Pval_pet_other {
            Pint unused;
        } pet_other; /* When no echo-specific data is required */
        . . .    /* data for impl. defined pets */
    } pets;
}

```

```

union Pval_measure_data {
    struct Pval_measure_other {
        Pint        unused;
    } meas_other; /* When no measure-specific data is required */
    . . .         /* data for impl. defined measure processes */
} measure_data;
union Pval_trigger_data {
    struct Pval_trig_other {
        Pint        unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .         /* data for impl. defined trigger processes */
} trigger_data;
union Pval_ack_data {
    struct Pval_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .         /* data for impl. defined ack. processes */
} acknowledgement_data;
} Pval_data;

```

Pval_data3 VALUATOR DATA RECORD 3

```

typedef struct {
    Pfloat          low;          /* Low value of valuator range*/
    Pfloat          high;         /* High value of valuator range*/
    union Pval3_pets {
        struct Pval3_pet_other {
            Pint        unused;
        } pet_other; /* When no echo-specific data is required */
        . . .         /* data for impl. defined pets */
    } pets;
    union Pval3_measure_data {
        struct Pval3_measure_other {
            Pint        unused;
        } meas_other; /* When no measure-specific data is required */
        . . .         /* data for impl. defined measure processes */
    } measure_data;
}

```

```

union Pval3_trigger_data {
    struct Pval3_trig_other {
        Pint        unused;
    } trig_other; /* When no trigger-specific data is required */
    . . .        /* data for impl. defined trigger processes */
} trigger_data;
union Pval3_ack_data {
    struct Pval3_ack_other {
        Pint        unused;
    } ack_other; /* When no ack.-specific data is required */
    . . .        /* data for impl. defined ack. processes */
} acknowledgement_data;
} Pval_data3;”

```

Page 38

5.4 Implementation independent type definitions

Change the subclause heading to “**Implementation independent Basic PHIGS profile type definitions**”.

Page 65

The following text replaces item PSTRUCT_ST_STOP in the definition of Pstruct_st:

```

“PSTRUCT_ST_STOP,
 PSTRUCT_ST_DISO”

```

Page 73

6 C PHIGS macro definitions

Change the clause heading to “**C Basic PHIGS profile macro definitions**”.

Page 83

6.3 Linetypes

6.3.1 Linetypes

The following entry is appended:

```

“
#define PLINE_DASH_DOT_DOT      (5)”

```

Page 76

6.4 Error codes

The following definition replaces error 5:

```
“
#define PE_NOT_STOP          (5)    /* Ignoring function, function requires
                                     state (PHOP,*,STOP | DISO,*) */”
```

Page 85

7 C PHIGS functions

Change the clause heading to “C Basic PHIGS profile functions”.

Page 17 (ISO/IEC 9593-4:1992/Amd. 1:1994)

8 C PHIGS PLUS type definitions

Change the subclause heading to “C PHIGS PLUS profile type definitions”.

8.1 Mapping of PHIGS PLUS data types

Change the clause heading to “Mapping of PHIGS PLUS profile data types”.

The following text replaces the text of the first paragraph:

“The PHIGS PLUS profile specifies a set of abstract data types beyond the types defined in the Basic PHIGS profile. Table 4 gives the mapping from those additional data types defined in the PHIGS PLUS profile.”

Change the table title to “Table - 4 Mapping of PHIGS PLUS profile data types to C”.

Change the left table heading to “PHIGS PLUS profile data type”.

Page 18 (ISO/IEC 9593-4:1992/Amd. 1:1994)

8.2 Modifications to PHIGS data types

Change the subclause heading to “Modifications to Basic PHIGS profile data types”.

Page 24 (ISO/IEC 9593-4:1992/Amd. 1:1994)

8.3 Implementation dependent PHIGS PLUS type definitions

Change the subclause heading to “Implementation dependent PHIGS PLUS profile type definitions”.

Page 34 (ISO/IEC 9593-4:1992/Amd. 1:1994)

8.4 Implementation independent PHIGS PLUS type definitions

Change the subclause heading to “Implementation independent PHIGS PLUS profile type definitions”.

Page 56 (ISO/IEC 9593-4:1992/Amd. 1:1994)

9 C PHIGS PLUS macro definitions

Change the clause heading to “C PHIGS PLUS profile macro definitions”.

Page 63 (ISO/IEC 9593-4:1992/Amd. 1:1994)

10 C PHIGS PLUS functions

Change the clause heading to “C PHIGS PLUS profile functions”.

Page 95 (ISO/IEC 9593-4:1992/Amd. 2:1994)

The following new clauses are inserted following clause 10.

“

11 C Full PHIGS profile type definitions

11.1 Mapping of Full PHIGS profile data types

The *Full PHIGS profile* specifies a set of abstract data types beyond the types defined in the *PHIGS PLUS profile*. Table 4 gives the mapping from those additional data types defined in the *Full PHIGS profile*.

Table 4 - Mapping of Full PHIGS profile data types to C

PHIGS data type		C binding data type
@t	location of t	*t
AP	acknowledgement process	Pack_process
BIT	bit	unsigned int:1
CF	configuration setting list	Pconfig_settings
CFT	condition flag test	Pcond_test
EP	echo process	Pecho_process
M	measure	Pmeasure
MD	marker descriptor	Pmarker_desc
MP	measure process	Pmeas_process
MS	measure set	Pmeasure_set
TA	target address	Ptarg_addr
TO	target operation	Ptarg_op
TP	trigger process	Ptrig_process
TR	traversal resource	Ptrav_res

11.2 Modifications to *Basic PHIGS profile* type definitions

Paspect ASPECT

```
typedef enum {
    /* start of PHIGS enumerations */
    ...,
    /* end of PHIGS enumerations */
    /* start of PHIGS PLUS enumerations */
    ...,
    /* end of PHIGS PLUS enumerations */
    /* start of PHIGS FULL enumerations */
    PASPECT_HIGHL_METHOD
    /* end of PHIGS FULL enumerations */
} Paspect;
```

Pcolr_rep COLOUR REPRESENTATION

```
typedef union {
    /* start of Basic PHIGS union members */
    Prgb      rgb;      /* RGB colour specification */
    Pcielv    cielv;    /* CIE L*u*v* colour specification */
    Phls      hls;      /* HLS colour specification */
    Phsv      hsv;      /* HSV colour specification */
    /* end of Basic PHIGS union members */
    /* start of Full PHIGS union members */
    Prgba     rgba;     /* RGB specification with alpha */
    Pcieluva  cieluva;  /* CIE L*u*v* colour specification with alpha */
    Phlsa     hlsa;     /* HLS colour specification with alpha */
    Phsva     hsva;     /* HSV colour specification with alpha */
    /* end of Full PHIGS union members */
    ...        /* implementation defined */
} Pcolr_rep;
```

Pelem_data ELEMENT DATA

```
typedef union {
    /* start of Basic PHIGS element data */
    ...,
    /* end of Basic PHIGS element data */
}
```

```

/* start of PHIGS PLUS element data */
...
/* end of PHIGS PLUS element data */
/* start of Full PHIGS element data */
struct Pappl_int {
    Pint    integer_id;    /* application integer identifier    */
    Pint    value;         /* value for application integer    */
} appl_int;
struct Pappl_real {
    Pint    real_id;       /* application real identifier    */
    Pfloat  value;         /* value for application real    */
} appl_real;
struct Pcircle3 {
    Ppoint3  center_point; /* center point    */
    Pfloat   radius;       /* radius    */
    Pvec3    ref_vecs[2];  /* reference vectors    */
} circle3;
struct Pcircle {
    Ppoint   center_point; /* center point    */
    Pfloat   radius;       /* radius    */
} circle;
struct Pcircular_arc3 {
    Ppoint3  center_point; /* center point    */
    Pfloat   radius;       /* radius    */
    Pvec3    ref_vecs[2];  /* reference vectors    */
    Pfloat   start;        /* start angle    */
    Pfloat   end;          /* end angle    */
} circular_arc3;
struct Pcircular_arc {
    Ppoint   center_point; /* center point    */
    Pfloat   radius;       /* radius    */
    Pfloat   start;        /* start angle    */
    Pfloat   end;          /* end angle    */
} circular_arc;
struct Pellipse3 {
    Ppoint3  center_point; /* center point    */
    Pvec3    major_ref_vec; /* major axis reference vector    */
    Pvec3    minor_ref_vec; /* minor axis reference vector    */
} ellipse3;
struct Pellipse {
    Ppoint   center_point; /* center point    */
    Pvec     major_ref_vec; /* major axis reference vector    */
    Pvec     minor_ref_vec; /* minor axis reference vector    */
} ellipse;

```

```

struct Pelliptical_arc3 {
    Ppoint3      center_point;      /* center point          */
    Pvec3        major_ref_vec;      /* major axis reference vector */
    Pvec3        minor_ref_vec;      /* minor axis reference vector */
    Pfloat       start;              /* start angle            */
    Pfloat       end;                /* end angle              */
} elliptical_arc3;

struct Pelliptical_arc {
    Ppoint      center_point;      /* center point          */
    Pvec        major_ref_vec;      /* major axis reference vector */
    Pvec        minor_ref_vec;      /* minor axis reference vector */
    Pfloat      start;              /* start angle            */
    Pfloat      end;                /* end angle              */
} elliptical_arc;

struct Pfill_circle3 {
    Ppoint3      center_point;      /* center point          */
    Pfloat       radius;             /* radius                */
    Pvec3        ref_vecs[2];        /* reference vectors      */
} fill_circle3;

struct Pfill_circle {
    Ppoint      center_point;      /* center point          */
    Pfloat      radius;            /* radius                */
} fill_circle;

struct Pcircular_arc_close3 {
    Ppoint3      center_point;      /* center point          */
    Pfloat       radius;             /* radius                */
    Pvec3        ref_vecs[2];        /* reference vectors      */
    Pfloat       start;              /* start angle            */
    Pfloat       end;                /* end angle              */
    Pclosure     type;              /* closure type           */
} circular_arc_close3;

struct Pcircular_arc_close {
    Ppoint      center_point;      /* center point          */
    Pfloat      radius;            /* radius                */
    Pfloat      start;              /* start angle            */
    Pfloat      end;                /* end angle              */
    Pclosure     type;              /* closure type           */
} circular_arc_close;

struct Pfill_ellipse3 {
    Ppoint3      center_point;      /* center point          */
    Pvec3        major_ref_vec;      /* major axis reference vector */
    Pvec3        minor_ref_vec;      /* minor axis reference vector */
} fill_ellipse3;

struct Pfill_ellipse {
    Ppoint      center_point;      /* center point          */
    Pvec        major_ref_vec;      /* major axis reference vector */
    Pvec        minor_ref_vec;      /* minor axis reference vector */
} fill_ellipse;

```

```

struct Pelliptical_arc_close3 {
    Ppoint3      center_point;      /* center point */
    Pvec3        major_ref_vec;     /* major axis reference vector */
    Pvec3        minor_ref_vec;     /* minor axis reference vector */
    Pfloat       start;             /* start angle */
    Pfloat       end;               /* end angle */
    Pclosure     type;              /* closure type */
} elliptical_arc_close3;

struct Pelliptical_arc_close {
    Ppoint      center_point;      /* center point */
    Pvec        major_ref_vec;     /* major axis reference vector */
    Pvec        minor_ref_vec;     /* minor axis reference vector */
    Pfloat      start;             /* start angle */
    Pfloat      end;               /* end angle */
    Pclosure    type;              /* closure type */
} elliptical_arc_close;

Plinetype_adapt  linetype_adapt;  /* linetype adaptability */
Plinetype_cont   linetype_cont;   /* linetype continuity */
Phighl_method    highl_method;    /* highlighting method */
Pint_list        active_textures; /* active textures */
Pint_list        back_active_textures; /* back active textures */
Pperspect_corr   perspect_corr;   /* textr perspective correction */

struct Ptexture_res_opt_heur {
    Pint      opt_hint;          /* optimization hint */
    Pint_list usage_priorities; /* texture usage priorities */
} texture_res_opt_heur;

Pint_list      alpha_src_sel; /* alpha source selector */

struct Pcond_exec_struct {
    Pint      struct_id;          /* structure identifier */
    Ptest     test;               /* condition test */
} cond_exec_struct;

struct Pcond_inst_struct {
    Pint      struct_id;          /* structure identifier */
    Ptest     test;               /* condition test */
} cond_inst_struct;

Ptest      cond_return; /* condition test */

struct Pcond_skip_elems {
    Pint      skip_count; /* number of elements to skip */
    Ptest     test;       /* condition test */
} cond_skip_elems;

```

```

struct Pcond_skip_to_label {
    Pint      label;          /* label to which to skip    */
    Ptest     test;           /* condition test            */
} cond_skip_to_label;
/* end of Full PHIGS element data */

} Pelem_data;

```

Pelem_type ELEMENT TYPE

```

typedef enum {
    /* start of PHIGS enumerations */
    ...,
    /* end of PHIGS enumerations */
    /* start of PHIGS PLUS enumerations */
    ...,
    /* end of PHIGS PLUS enumerations */
    /* start of Full PHIGS enumerations */
    PELEM_APPL_INT,
    PELEM_APPL_REAL,
    PELEM_CIRCLE3,
    PELEM_CIRCLE,
    PELEM_CIRCULAR_ARC3,
    PELEM_CIRCULAR_ARC,
    PELEM_ELLIPSE3,
    PELEM_ELLIPSE,
    PELEM_ELLIPTICAL_ARC3,
    PELEM_ELLIPTICAL_ARC,
    PELEM_FILL_CIRCLE3,
    PELEM_FILL_CIRCLE,
    PELEM_CIRCULAR_ARC_CLOSE3,
    PELEM_CIRCULAR_ARC_CLOSE,
    PELEM_FILL_ELLIPSE3,
    PELEM_FILL_ELLIPSE,
    PELEM_ELLIPTICAL_ARC_CLOSE3,
    PELEM_ELLIPTICAL_ARC_CLOSE,
    PELEM_HIGHL_IND,
    PELEM_LINETYPE_ADAPT,
    PELEM_LINETYPE_CONT,
    PELEM_LINETYPE_OFFSET,

```

```
PELEM_HIGHL_METHOD,  
PELEM_ACTIVE_TEXTURES,  
PELEM_BACK_ACTIVE_TEXTURES,  
PELEM_TEXTURE_PERSPECT_CORR,  
PELEM_TEXTURE_SAMPLING_FREQ,  
PELEM_TEXTURE_RES_OPT_HEUR,  
PELEM_TRANSPARENCY,  
PELEM_BACK_TRANSPARENCY,  
PELEM_ALPHA_SRC_SEL,  
PELEM_ALPHA_DATA_SEL_IND,  
PELEM_INST_STRUCT,  
PELEM_COND_EXEC_STRUCT,  
PELEM_COND_INST_STRUCT,  
PELEM_COND_RETURN,  
PELEM_COND_SKIP_ELEMS,  
PELEM_COND_SKIP_TO_LABEL,  
PELEM_PUSH_ST,  
PELEM_POP_ST  
/* end of Full PHIGS enumerations */  
  
} Pelem_type;
```

Pin_class **INPUT CLASS**

```
typedef enum {  
    /* start of Basic PHIGS enumerations */  
    PIN_NONE,  
    PIN_LOC,  
    PIN_STROKE,  
    PIN_VAL,  
    PIN_CHOICE,  
    PIN_PICK,  
    PIN_STRING,  
    /* end of Basic PHIGS enumerations */  
    /* start of Full PHIGS enumerations */  
    PIN_SET,  
    PIN_COMPOSITE  
    /* end of Full PHIGS enumerations */  
} Pin_class;
```

Pint_style INTERIOR STYLE

```
typedef enum {  
    /* start of Basic PHIGS enumerations */  
    ...  
    /* end of Basic PHIGS enumerations */  
    /* start of Full PHIGS enumerations */  
    PSTYLE_TEXTURE  
    /* end of Full PHIGS PLUS enumerations */  
} Pint_style;
```

Popen_struct_status OPEN STRUCTURE STATUS

```
typedef enum {  
    /* start of Basic PHIGS enumerations */  
    PSTRUCT_NONE,  
    PSTRUCT_OPEN,  
    /* end of Basic PHIGS enumerations */  
    /* start of Full PHIGS enumerations */  
    PSTRUCT_DI_OPEN  
    /* end of Full PHIGS enumerations */  
} Popen_struct_status;
```

11.3 Modifications to *PHIGS PLUS* profile type definitions

Pcolr_rep_ptr COLOUR REPRESENTATION POINTER

```
typedef union {
    /* start of PHIGS PLUS union members */
    Prgb      *rgb;          /* pointer to RGB colour values */
    Pcielvuv  *cielvuv;      /* pointer to CIELUV pointer values */
    Phls      *hls;          /* pointer to HLS pointer values */
    Phsv      *hsv;          /* pointer to HSV pointer values */
    /* end of PHIGS PLUS union members */
    /* start of Full PHIGS union members */
    Prgba     *rgba;         /* pointer to RGBA colour values */
    Pcieluva  *cieluva;      /* pointer to CIELUVA pointer values */
    Phlsa     *hlsa;         /* pointer to HLSA pointer values */
    Phsva     *hsva;         /* pointer to HSVA pointer values */
    /* end of Full PHIGS union members */
    ...                /* implementation defined */
} Pcolr_rep_ptr;
```

Ppat_rep_plus PATTERN REPRESENTATION PLUS

```
typedef Pgcolr_array Ppat_rep_plus;
```

11.4 Implementation dependent *Full PHIGS* profile type definitions

Pbackg_method POSTING GROUP BACKGROUND METHOD

```
typedef struct {
    Pint      method;        /* background method */
    union Pbackg_method_data {
        struct Pcolr_table_0 {
            Pint      unused; /* unused */
        } colr_table_0;
        struct Pcolr_ind {
            Pint      colr_ind; /* colour index */
        } colr_ind;
        struct Pcolr {
            Pcolr_rep colr_rep; /* target address */
        } colr;
    };
};
```

```

    struct Pimage_resource {
        Pint      image_resource_id; /* image resource identifier */
    } image_resource;
    ...                               /* implementation defined */
} data;
} Pbackg_method;

```

Pclamp_method **BOUNDARY CLAMP METHOD DATA RECORD**

```

typedef struct {
    Pint      method;           /* clamp method */
    union Pclamp_method_data {
        Pint      unused;       /* used by method 1 */
        Pcolr     clamp_colr;   /* used by method 2 */
        ...       /* implementation defined */
    } data;
} Pclamp_method;

```

Pcoord_src **COORDINATE SOURCE DATA RECORD**

```

typedef struct {
    Pint      coord_src;        /* coordinate source */
    union Pcoord_src_data {     /* coordinate source data record */
        Pint      unused;       /* used by coord sources 1 - 5 */
        Pint      coord_ind;    /* used by coord source 6 */
        Pmatrix3  refl_matrix;  /* used by coord sources 7 and 8 */
        ...       /* implementation defined */
    } data;
} Pcoord_src;

```

Pdi_pick_data **DIRECT INTERPRETATION PICK DATA RECORD**

```

typedef struct {
    union Pdi_pick_pets {
        struct Pdi_pick_pet_other {
            Pint      unused;
        } pet_other; /* When no echo-specific data is required */
    }
}

```