

INTERNATIONAL STANDARD

ISO/IEC
9593-1

First edition
1990-06-01

AMENDMENT 1
1995-04-15

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

Part 1:
FORTRAN

AMENDMENT 1

*Systèmes de traitement de l'information — Infographie — Interfaces langage
entre un programme d'application et son support graphique*

Partie 1: FORTRAN

AMENDEMENT 1



Reference number
ISO/IEC 9593-1:1990/Amd 1:1995(E)
Documents
AVA Document Sharing Hub

Contents

1. Scope	1
2. Normative references	1
3. Principles	2
3.1. Specification	2
3.2. Mapping of PHIGS function names to FORTRAN subroutine names	2
3.3. Parameters	2
3.4. The FORTRAN subset	2
3.5. Error handling	3
4. Generating FORTRAN subroutine names	4
5. Data types	6
6. Enumeration types	10
7. List of the PHIGS function names	10
8. PHIGS errors specific to the FORTRAN binding	10
9. The PHIGS function interface	10
10. Utility functions not defined in PHIGS	10
11. PHIGS PLUS Enumeration types	11
12. List of the PHIGS PLUS function names	16
12.1. List of functions ordered alphabetically by bound name	16
12.2. List of functions ordered alphabetically by PHIGS PLUS function name	18
13. The PHIGS PLUS function interface	20
13.1. General principles	20
13.2. Output primitive functions	20
13.3. Attribute specification functions	33
13.3.1. Bundled attribute selection	33
13.3.2. Individual attribute selection	34
13.3.3. Aspect source flag setting	46
13.3.4. Workstation attribute table definition	47
13.4. Inquiry functions	55
13.4.1. Inquiry functions for workstation state list	55
13.4.2. Inquiry functions for workstation description table	63
14. Utility functions not defined in PHIGS and PHIGS PLUS	111
Annex A FORTRAN Examples(informative)	118
Annex B Function Lists(normative)	124
Annex C Memory Management(normative)	130

© ISO/IEC 1995

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

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 1 to International Standard ISO/IEC 9593-1:1990 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Introduction

Replace the Introduction of ISO/IEC 9593-1 with the following text:

ISO/IEC 9592-1:1989, provides a set of functions for the display and modification of 2D or 3D graphical data. Part 1 is extended by Part 4 (PHIGS PLUS) to incorporate the effects of lighting, shading and other properties that are important for the display of surfaces and multidimensional data.

ISO/IEC 9592-1 and ISO/IEC 9592-4 are specified in a language independent manner and must be embedded in language dependent layers (language bindings) for use with particular programming languages.

The purpose of this part of ISO/IEC 9593 is to define the FORTRAN language binding for ISO/IEC 9592-1 and ISO/IEC 9592-4.

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

Part 1: FORTRAN

AMENDMENT 1

1 Scope

Replace clause 1:

The "Programmer's Hierarchical Interactive Graphics System" (PHIGS), ISO/IEC 9592-1:1989, and ISO/IEC 9592-4:1992, specify a language independent nucleus of a graphics system. For integration into a programming language, PHIGS PLUS is embedded in a language dependent layer obeying the particular conventions of that language. This part of ISO/IEC 9593 specifies the FORTRAN language dependent layer.

2 Normative references

Add the following reference to clause 2:

ISO/IEC 9592-4:1992, *Information processing systems - Computer graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) Part 4 - Plus Lumière Und Surfaces (PHIGS PLUS)*.

3 Principles

3.1 Specification

Replace subclause 3.1, Specification, of ISO/IEC 9593-1 with the following text:

This part of ISO/IEC 9593 defines the PHIGS and PHIGS PLUS language binding interface for FORTRAN 77, as described in ISO 1539: 1980. With some minor modifications, application programs can be transported between full FORTRAN77 and FORTRAN77 Subset PHIGS and PHIGS PLUS installations.

This binding incorporates the rules of conformance defined in the PHIGS (ISO/IEC 9592-1) and PHIGS PLUS (ISO/IEC 9592-4) Standard for PHIGS and PHIGS PLUS implementations, with those additional requirements specifically defined for FORTRAN language implementations defined in this part of ISO/IEC 9593. The following criteria are established for determining conformance of an implementation to this binding:

In order to conform, an implementation of the FORTRAN binding of PHIGS shall implement those functions specified in ISO/IEC 9592-1. The implementation shall make visible all of the declarations in the FORTRAN binding specified in clause 5 to 10 in this part of ISO/IEC 9593.

In order to conform, an implementation of the FORTRAN binding of PHIGS PLUS shall implement those functions specified in ISO/IEC 9592-1 and also those functions specified in ISO/IEC 9592-4. The implementation shall make visible all of the declarations in the FORTRAN binding specified in clause 11 to 14 and in clause 5 to 10, as modified by clause 11 to 14, in this part of ISO/IEC 9593.

Thus, for example, 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.

A PHIGS FORTRAN application should run without modification under a PHIGS PLUS FORTRAN binding implementation.

3.2 Mapping of PHIGS function names to FORTRAN subroutine names

Append the following paragraph to subclause 3.2,

However, two abbreviations are changed for PHIGS PLUS; MAPPING becomes M and WORKSTATION becomes W, due to the FORTRAN subroutine naming restrictions.

3.3 Parameters

No change to ISO/IEC 9593-1.

3.4 The FORTRAN subset

No change to ISO/IEC 9593-1.

3.5 Error handling

No change to ISO/IEC 9593-1.

4 Generating FORTRAN subroutine names

Add the following, alphabetically, to table 2 of clause 4:

Table 2 - Reduce compound terms for uniqueness

REPRESENTATION PLUS	-> P
---------------------	------

Add the following, alphabetically, to table 3 of clause 4:

Table 3 - Deletions

GEOMETRIC	NON-UNIFORM	WITH
-----------	-------------	------

Add the following, alphabetically, to table 4 to clause 4:

Table 4 - Abbreviations

PHIGS PLUS word abbreviation		remarks
APPROXIMATION	A	SET DEPTH CUE INDEX:DP
B-SPLINE	BS	
BACK	B	
CHARACTERISTICS	C	
COORDINATE	CD	
CRITERIA	C	
CUE	C	
CULLING	C	
CURVE	C	
DATA	D	
DEPTH	D	FOR PHIGS PLUS
DIRECT	D	
DISTINGUISHING	D	
DYNAMICS	DC	
FACET	F	
LIGHT	L	
MAPPING	M	
MESH	M	
METHOD	M	
PARAMETRIC	P	
PLACEMENT	P	FOR PHIGS PLUS
PLUS	P	
PROPERTIES	P	
QUADRILATERAL	Q	
REFLECTANCE	RF	
RENDERING	R	
SHADING	S	
SPLINE	S	
STRIP	ST	
SURFACE	S	
SOURCE	S	
TRIMMING	T	
TRIANGLE	T	
WORKSTATION	W	

5 Data types

Append the following data type definitions to clause 5:

COLRV colour value

INTEGER containing colour index when CTYPE is INDIRECT

INTEGER NCC when CTYPE is not INDIRECT

REAL₁,REAL₂,REAL₃,...REAL_n containing $C_1, C_2, C_3, \dots, C_n$ ($n=NCC$) when CTYPE is not INDIRECT

COLRVH homogeneous colour value

REAL₁,REAL₂,REAL₃,...REAL_n and **REAL** containing $WC_1, WC_2, WC_3, \dots, WC_n$ and W ($n=NCC$)

GCOLOR general colour

a compound data type containing colour type and colour value of **COLRV**

NORM normal vector

REAL,REAL,REAL containing X-,Y- and Z- values

P2H two-dimensional homogeneous point

REAL,REAL,REAL containing WX-,WY- and W- values (or WU-,WV- and W- of a trimming curve control point) W value is ignored in case of non-rational type

P3H three-dimensional homogeneous point

REAL,REAL,REAL,REAL containing WX-,WY-,WZ- and W- values
W value is ignored in case of non-rational type

A(P3) array of coordinates of points

REAL(*),REAL(*),REAL(*) containing X-,Y- and Z- values(*=number of columns by number of rows)

L(L(P3{COLRV})) list of vertex data lists

REAL PXA(*),PYA(*),PZA(*) *=last value of array of end indices for point lists(NP)

INTEGER VCOLI(*) *=NP, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NP,
when CTYPE is not INDIRECT

{COLRV}{,NORM}{,L(R)} facet data

INTEGER FCOLI when CTYPE is INDIRECT

REAL FCOLR(*) *=number of components of colour value(NCC), when CTYPE is not INDIRECT

REAL FNXA,FNYA,FNZA facet normal data

REAL FDLEN length of application-specific data

REAL FDATA(FDLEN) facet application-specific data

L(L(E)) list of edge flags lists

INTEGER EDATA(*) *=last value of array of end indices for point lists

L(L(P2{COLRV}{,NORM}{,L(R)}))

REAL PXA(*),PYA(*) *=last value of array of end indices for point lists(NP)

INTEGER VCOLI(*) *=NP, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NP, when CTYPE

is not INDIRECT

REAL VNXA(*),VN YA(*) vertex normal data(*=NP)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NP)

L(L(P3{,COLRV}{,NORM}{,L(R)}))

REAL PXA(*),PYA(*),PZA(*) *=last value of array of end indices for point lists(NP)

INTEGER VCOLI(*) *=NP, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NP, when CTYPE is not INDIRECT

REAL VNXA(*),VN YA(*),VN ZA(*) vertex normal data(*=NP)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NP)

L(L(L(I))) list of lists of vertex indices lists

REAL VIND(*) *=last value of array of end vertex indices for each fill area

L(L(L(E))) list of lists of edge flags lists

INTEGER EDATA(*) *=last value of array of end vertex indices for each fill area

L({COLRV}{,NORM}{,L(R)}) list of facet data

INTEGER FCOLI(*) *=number of fill area sets(N) in set of fill area sets 3 with data or set of fill area sets with data, number of triangles(N) at following 4 functions, triangle set 3 with data, triangle set with data, triangle strip 3 with data and triangle strip with data, when CTYPE is INDIRECT

REAL FCOLR(*) *=number of components of colour value(NCC) × NFS, when CTYPE is not INDIRECT

REAL FNXA(*),FN YA(*),FN ZA(*) facet normal data(*=N)

INTEGER FDLEN length of application-specific data

REAL FDATA(*) facet application-specific data(*=FDLEN × N)

L(P3{,COLRV}{,NORM}{,L(R)}) list of vertex data

REAL PXA(*),PYA(*),PZA(*) *=number of points(NP)

INTEGER VCOLI(*) *=NP, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NP, when CTYPE is not INDIRECT

REAL VNXA(*),VN YA(*),VN ZA(*) vertex normal data(*=NP)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NP)

L(P2{,COLRV}{,NORM}{,L(R)}) list of vertex data

REAL PXA(*),PYA(*) *=number of points(NP)

INTEGER VCOLI(*) *=NP, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NP, when CTYPE is not INDIRECT

REAL VNXA(*),VN YA(*),VN ZA(*) vertex normal data(*=NP)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NP)

A(COLRV) colour array

INTEGER COLIA(*) *=number of columns(DIMX) × number of rows(DIMY), when CTYPE is INDIRECT

REAL COLRA(*) *=number of components of colour value(NCC) × DIMX × DIMY, when CTYPE is not INDIRECT

A({COLRV},{,NORM},{,L(R)}) array of facet data

INTEGER FCOLI(*) *=number of columns minus 1(NC-1) × number of rows minus 1(NR-1),
when CTYPE is INDIRECT

REAL FCOLR(*) *=number of components of colour value(NCC) × NC-1 × NR-1,
when CTYPE is not INDIRECT

REAL FNXA(*),FNXA(*),FNZA(*) facet normal data(*= NC-1 × NR-1)

INTEGER FDLEN length of application-specific data

REAL FDATA(*) facet application-specific data(*=FDLEN × NC-1 × NR-1)

A(P3{,COLRV},{,NORM},{,L(R)}) array of vertex data

REAL PXA(*),PYA(*),PZA(*) *=number of columns(NC) × number of rows(NR)

INTEGER VCOLI(*) *=NC × NR, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NC × NR,
when CTYPE is not INDIRECT

REAL VNXA(*),VNXA(*),VNZA(*) vertex normal data(*= NC × NR)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NC × NR)

A(P2{,COLRV},{,NORM},{,L(R)}) array of vertex data

REAL PXA(*),PYA(*) *=number of columns(NC) × number of rows(NR)

INTEGER VCOLI(*) *=NC × NR, when CTYPE is INDIRECT

REAL VCOLR(*) *=number of components of colour value(NCC) × NC × NR,
when CTYPE is not INDIRECT

REAL VNXA(*),VNXA(*),VNZA(*) vertex normal data(*= NC × NR)

INTEGER VDLEN length of application-specific data

REAL VDATA(*) vertex application-specific data(*=VDLEN × NC × NR)

A(2×E) array of edge data

INTEGER EDATA(*) *=2 × number of columns(NC) × number of rows(NR)

L(P3H)|L(P3) list of curve control points

REAL PXA(*),PYA(*),PZA(*),PWA(*) *=number of control points, when CTYPE is PRAT

REAL PXA(*),PYA(*),PZA(*) *=number of control points, when CTYPE is PNRAT

L(COLRVH)|L(COLRV) list of colour spline control points

REAL CSCR(*) *={number of components of colour value(NCC)+1} ×
number of colour spline control points(NCSCP), when CTYPE is PRAT

REAL CSCR(*) *=number of components of colour value(NCC) ×
number of colour spline control points(NCSCP), when CTYPE is PNRAT

A(P3H)|A(P3) array of surface control points

REAL PXA(*),PYA(*),PZA(*),PWA(*) *=u number of control points dimension(UNCP) ×
v number of control points dimension(VNCP), when CTYPE is PRAT

REAL PXA(*),PYA(*),PZA(*) *=u number of control points dimension(UNCP) ×
v number of control points dimension(VNCP), when CTYPE is PNRAT

L(L(TRIMCURVE)) list of trimming loop definitions lists

INTEGER TACRI(*) *=number of components of list of trimming loop definitions lists(NCLTL)

INTEGER TCVF(*) trimming curve visibility flag(*=NCLTL)

INTEGER TSORD(*) trimming curve spline order(*=NCLTL)

INTEGER TNKA(*) number of spline knots(*=NCLTL)

REAL TKNOTS(*) *=last value of array of number of spline knots in array(TNKA)

REAL TPARL(2,*) trimming curve parameter range limits(*=NCLTL)

INTEGER TRTYPE(*) trimming curve spline rationality(*=NCLTL)

INTEGER NTCCP(*) number of trimming curve spline control points(*=NCLTL)
REAL TPWXA(*),TPWYA(*),TPWWA(*) *=last value of number of trimming curve spline control points, when CRTYPE is PRAT
REAL TPWXA(*),TPWYA(*) *=last value of number of trimming curve spline control points, when CRTYPE is PNRAT

L(P2H)|L(P2) list of trimming curve control points

REAL TPWUA(*),TPWVA(*),TPWWA(*) *=number of control points, when CRTYPE is PRAT
REAL TPWUA(*),TPWVA(*) *=number of control points, when CRTYPE is PNRAT

A(COLRVH)|A(COLRV) array of colour spline control points

REAL CSCP(*) *={number of components of colour value(NCC)+1} ×
 u number of colour spline control points dimension(NUCSCP) ×
 v number of colour spline control points dimension(NVCSCP), when CRTYPE is PRAT
REAL CSCP(*) *=number of components of colour value(NCC) ×
 u number of colour spline control points dimension(NUCSCP) ×
 v number of colour spline control points dimension(NVCSCP), when CRTYPE is PNRAT

L(DATASPLINE) list of data spline (for non-uniform B-spline surface with data)

INTEGER DUSORD(*) u data spline order(*=NDS)
INTEGER DVSORD(*) v data spline order(*=NDS)
INTEGER DNUKA(*) number of u spline knots(*=NDS)
INTEGER DNVKA(*) number of v spline knots(*=NDS)
REAL DUKNTS(*) *=last value of number of u spline knots(DNUKA) × NDS
REAL DVKNTS(*) *=last value of number of v spline knots(DNVKA) × NDS
INTEGER DRTYPE(*) data spline rationality(*=NDS)
INTEGER NUDSCP(*) u number of data spline control points dimension(*=NDS)
INTEGER NVDSCP(*) v number of data spline control points dimension(*=NDS)
INTEGER DDIMS(*) data dimension(*=NDS)
REAL DSCP(*) *=

$$\sum_{i=1}^{NDS} (NUDSCP(i) \times NVDSCP(i) \times DDIMS(i))$$

DDIMS(i) = n+1 (n = d₁,d₂,d₃,...,d_n,w) when DRTYPE(i) is PRAT.

DDIMS(i) = n (n = d₁,d₂,d₃,...,d_n) when DRTYPE(i) is PNRAT.

A(DATAH)|A(DATA) control points of data spline

REAL DSCP(*) *=

$$\sum_{i=1}^{NDS} (NUDSCP(i) \times NVDSCP(i) \times DDIMS(i))$$

DDIMS(i) = n+1 (n = d₁,d₂,d₃,...,d_n,w) when DRTYPE(i) is PRAT.

DDIMS(i) = n (n = d₁,d₂,d₃,...,d_n) when DRTYPE(i) is PNRAT.

NDS=number of data spline

NUDSCP=u number of data spline control points dimension

NVDSCP=v number of data spline control points dimension

DDIMS=data dimension for spline

6 Enumeration types

No change to ISO/IEC 9593-1.

7 List of the PHIGS function names

No change to ISO/IEC 9593-1.

8 PHIGS errors specific to the FORTRAN binding

No change to ISO/IEC 9593-1.

9. The PHIGS function interface

No change to ISO/IEC 9593-1.

10 Utility functions not defined in PHIGS

No change to ISO/IEC 9593-1.

Add the following new clauses 11, 12, 13 and 14:

11 PHIGS PLUS Enumeration types

All the enumeration types of PHIGS PLUS are mapped to FORTRAN INTEGERS. The correspondence between PHIGS PLUS scalars and FORTRAN INTEGERS is shown as follows in a list of symbolic FORTRAN constants that may be included in any application program. This clause contains a mapping of PHIGS PLUS enumeration types to FORTRAN variable names.

"Polyline shading method", "interior shading method", "data mapping method", "reflectance property type", "reflectance model", "curve placement", "light source type", "colour mapping method", "curve approximation type", "surface approximation type" and "parametric surface characteristic type" are defined as INTEGER rather than enumeration types in PHIGS PLUS. Constant definitions for the explicitly defined and required values of these conceptually unbounded ranges are provided as a convenience.

Also, a numbering of all PHIGS PLUS functions is given for use in the error handling procedures.

Mnemonic FORTRAN names and their values for PHIGS PLUS ENUMERATION type values:

aspect identifier

	<i>polyline colour</i>				
INTEGER	PPLCOL,	PPMCOL,	PTXCOL,	PINCOL,	PEDCOL,
1	PPSHMD,	PISHMD,	PDMPMD,	PREFPR,	PREFM,
2	PBINSY,	PBINSI,	PBICOL,	PBISHM,	PBDMPM,
3	PBREFP,	PBREFM,	PCAPCR,	PSAPCR,	PPASUC
PARAMETER(	PPLCOL=18,	PPMCOL=19,	PTXCOL=20,	PINCOL=21,	PEDCOL=22,
1	PPSHMD=23,	PISHMD=24,	PDMPMD=25,	PREFPR=26,	PREFM=27,
2	PBINSY=28,	PBINSI=29,	PBICOL=30,	PBISHM=31,	PBDMPM=32,
3	PBREFP=33,	PBREFM=34,	PCAPCR=35,	PSAPCR=36,	PPASUC=37)

colour mapping method

	<i>true</i>	<i>pseudo</i>	<i>pseudo-n</i>
INTEGER	PTRUE,	PSUD,	PSUDN
PARAMETER(	PTRUE=1,	PSUD=2,	PSUDN=3)

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

colour type

	<i>indirect</i>
INTEGER	PINDIR
PARAMETER(	PINDIR=0)

The following colour models are also colour types.
RGB,CIELUV,HSV,HLS

culling mode

	<i>none</i>	<i>backfacing</i>	<i>frontfacing</i>
INTEGER	PNOFC,	PBKFC,	PFTFC
PARAMETER(	PNOFC=0,	PBKFC=1,	PFTFC=2)

curve approximation criteria type

	<i>workstation dependent chordal deviation in WC relative DC</i>	<i>constant between knots chordal deviation in NPC</i>	<i>chordal size in WC chordal deviation in DC</i>	<i>chordal size in NPC relative WC</i>	<i>chordal size in DC relative NPC</i>
INTEGER	PWDCA,	PCBKCA,	PCSWCA,	PCSNCA,	PCSDCA,
1	PCDWCA,	PCDNCA,	PCDDCA,	PRWCA,	PRNCA,
2	PRDCA				
PARAMETER(	PWDCA=1,	PCBKCA=2,	PCSWCA=3,	PCSNCA=4,	PCSDCA=5,
1	PCDWCA=6,	PCDNCA=7,	PCDDCA=8,	PRWCA=9,	PRNCA=10,
2	PRDCA=11)				

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

curve placement (of ISOPARAMETRIC CURVE)

	<i>over surface</i>	<i>between knots</i>
INTEGER	PUOSCP,	PUBKCP
PARAMETER(	PUOSCP=0,	PUBKCP=1)

curve visibility flag

Uses the POFF/PON enumeration defined for edge flag and error handling mode.

data mapping method

	<i>colour</i>	<i>single uniform</i>	<i>single non-uniform</i>	<i>BI uniform</i>	<i>BI non-uniform</i>
INTEGER	PCDM,	PSUDM,	PSNUDM,	PBUDM,	PBNUDM
PARAMETER(	PCDM=1,	PSUDM=2,	PSNUDM=3,	PBUDM=4,	PBNUDM=5)

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

depth cue mode

	<i>suppressed</i>	<i>allowed</i>
INTEGER	PSUPPR	PALLOW
PARAMETER(	PSUPPR=0,	PALLOW=1)

distinguishing mode

Uses the POFF/PON enumeration defined for edge flag and error handling mode.

edge data flag

	<i>none</i>	<i>edge visibility flag</i>
INTEGER	PENO	PEVF
PARAMETER(	PENO=0,	PEVF=1)

Not defined in PHIGS PLUS but used to indicate presence of optional data for FORTRAN binding.

element type

<i>polyline set 3 with data</i>					
INTEGER	PEPLS3,	PEFS3D,	PEFSD,	PECA3P,	PESFS3,
1	PESFSD,	PETS3D,	PETSD,	PETST3,	PETSTD,
2	PEQM3D,	PEQMD,	PEBC3,	PEBC3C,	PEBS3,
3	PEBS3D,	PEDMI,	PERFI,	PEBII,	PEBDMI,
4	PEBRFI,	PEPRSI,	PEPLC,	PEPLSM,	PEPMC,
5	PETXC,	PEFDM,	PEFCM,	PEIC,	PEISM,
6	PEDMM,	PERFP,	PERFM,	PEBIS,	PEBISI,
7	PEBIC,	PEBISM,	PEBDMM,	PEBRFP,	PEBRFM,
8	PELSS,	PEEDC,	PECAC,	PESAC,	PEPSC,
9	PERCM,	PEDPCI,	PECM1)		
PARAMETER(	PEPLS3=71,	PEFS3D=72,	PEFSD=73,	PECA3P=74,	PESFS3=75,
1	PESFSD=76,	PETS3D=77,	PETSD=78,	PETST3=79,	PETSTD=80,
2	PEQM3D=81,	PEQMD=82,	PEBC3=83,	PEBC3C=84,	PEBS3=85,
3	PEBS3D=86,	PEDMI=87,	PERFI=88,	PEBII=89,	PEBDMI=90,
4	PEBRFI=91,	PEPRSI=92,	PEPLC=93,	PEPLSM=94,	PEPMC=95,
5	PETXC=96,	PEFDM=97,	PEFCM=98,	PEIC=99,	PEISM=100,
6	PEDMM=101,	PERFP=102,	PERFM=103,	PEBIS=104,	PEBISI=105,
7	PEBIC=106,	PEBISM=107,	PEBDMM=108,	PEBRFP=109,	PEBRFM=110,
8	PELSS=111,	PEEDC=112,	PECAC=113,	PESAC=114,	PEPSC=115,
9	PERCM=116,	PEDPCI=117,	PECM1=118)		

facet data flag

	<i>no facet data</i>	<i>colour</i>	<i>normal</i>	<i>data</i>	<i>colour/normal</i>
INTEGER	PFNO,	PFC,	PFN,	PFD,	PFCN,
1	PFCD,	PFND,	PFCND		
PARAMETER(	PFNO=0,	PFC=1,	PFN=2,	PFD=3,	PFCN=4,
1	PFCD=5,	PFND=6,	PFCND=7)		

Not defined in PHIGS PLUS but used to indicate presence of optional data for FORTRAN binding.

GDP attributes

	<i>reflectance</i>	<i>parametric surface</i>
INTEGER	PRFATT,	PPSATT
PARAMETER(	PRFATT=5,	PPSATT=6)

interior shading method

	<i>none</i>	<i>colour</i>	<i>data</i>	<i>data/dot</i>	<i>data/normal</i>
INTEGER	PNOIS,	PCIS,	PDIS,	PDDIS,	PDNIS
PARAMETER(	PNOIS=1,	PCIS=2,	PDIS=3,	PDDIS=4,	PDNIS=5)

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

light source type

	<i>ambient</i>	<i>directional</i>	<i>positional</i>	<i>spot</i>
INTEGER	PAMB,	PDIRE,	PPOSI,	PSPOT
PARAMETER(	PAMB=1,	PDIRE=2,	PPOSI=3,	PSPOT=4)

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

parametric surface characteristic type

	<i>none</i>	<i>workstation dependent</i>	<i>isoparametric curve</i>	<i>level curve in MC</i>	<i>level curve in WC</i>
INTEGER	PNOPC,	PWDPC,	PICPC,	PLCMPC,	PLCWPC
PARAMETER(	PNOPC=1,	PWDPC=2,	PICPC=3,	PLCMPC=4,	PLCWPC=5)

INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.

polyline shading method

	<i>none</i>	<i>colour</i>
INTEGER	PNOPS,	PCPS
PARAMETER(	PNOPS=1,	PCPS=2)
INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.		

rationality

	<i>rational</i>	<i>non-rational</i>
INTEGER	PRAT,	PNRAT
PARAMETER(	PRAT=0,	PNRAT=1)

reflectance model

	<i>no reflectance</i>	<i>ambient</i>	<i>ambient/diffuse</i>	<i>ambient/diffuse/specular</i>
INTEGER	PNORM,	PARM,	PADRM,	PADSRM
PARAMETER(	PNORM=1,	PARM=2,	PADRM=3,	PADSRM=4)
INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.				

reflectance property type

	<i>simple reflectance</i>
INTEGER	PSRPT
PARAMETER(	PSRPT=1)
INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.	

surface approximation criteria

	<i>workstation dependent planar deviation in WC relative in DC</i>	<i>constant between knots planar deviation in NPC</i>	<i>chordal size in WC planar deviation in DC</i>	<i>chordal size in NPC relative in WC</i>	<i>chordal size in DC relative in NPC</i>
INTEGER	PWDSA,	PCBKSA,	PCSWSA,	PCSNSA,	PCSDSA,
1	PPDWSA,	PPDNSA,	PPDDSA,	PRWSA,	PRNSA,
2	PRDSA				
PARAMETER(	PWDSA=1,	PCBKSA=2,	PCSWSA=3,	PCSNSA=4,	PCSDSA=5,
1	PPDWSA=6,	PPDNSA=7,	PPDDSA=8,	PRWSA=9,	PRNSA=10,
2	PRDSA=11)				
INTEGER rather than enumeration type. Explicitly defined and required portion of conceptually unbounded range defined here.					

source selector

	<i>colour aspect</i>	<i>vertex colour</i>	<i>vertex data</i>	<i>facet colour</i>	<i>facet data</i>
INTEGER	PSCASF,	PSVC,	PSVD,	PSFC,	PSFD
PARAMETER(	PSCASF=0,	PSVC=1,	PSVD=2,	PSFC=3,	PSFD=4)

trimming curve visibility flag

Uses the POFF/PON enumeration defined for edge flag and error handling mode.

vertex data flag

	<i>coordinate only colour/data</i>	<i>colour normal/data</i>	<i>normal colour/normal/data</i>	<i>data</i>	<i>colour/normal</i>
INTEGER	PCD,	PCDC,	PCDN,	PCDD,	PCDCN,
1	PCDCD,	PCDND,	PCDCND		
PARAMETER(	PCD=0,	PCDC=1,	PCDN=2,	PCDD=3,	PCDCN=4,
1	PCDCD=5,	PCDND=6,	PCDCND=7)		

Not defined in PHIGS PLUS but used to indicate presence of optional data for FORTRAN binding.

PHIGS PLUS functions. These names are used for error handling. The names are the same as the PHIGS PLUS function names except that the sentinel character 'P' is replaced by 'E'. The same function identification used for full FORTRAN 77.

INTEGER	EPLS3C,	EFAS3D,	EFASD,	ECA3P,	ESFAS3
PARAMETER(	EPLS3C=183,	EFAS3D=184,	EFASD=185,	ECA3P=186,	ESFAS3=187)
INTEGER	ESFASD,	ETS3D,	ETSD,	ETST3D,	ETSTD
PARAMETER(	ESFASD=188,	ETS3D=189,	ETSD=190,	ETST3D=191,	ETSTD=192)
INTEGER	EQM3D,	EQMD,	EBSC3,	EBSC3C,	EBSS3
PARAMETER(	EQM3D=193,	EQMD=194,	EBSC3=195,	EBSC3C=196,	EBSS3=197)
INTEGER	EBSS3D,	ESDMI,	ESRFI,	ESBII,	ESBDMI
PARAMETER(	EBSS3D=198,	ESDMI=199,	ESRFI=200,	ESBII=201,	ESBDMI=202)
INTEGER	ESBRFI,	ESPSI,	ESPLC,	ESPLSM,	ESPMC
PARAMETER(	ESBRFI=203,	ESPSI=204,	ESPLC=205,	ESPLSM=206,	ESPMC=207)
INTEGER	ESTXC,	ESFDM,	ESFCM,	ESIC,	ESISM
PARAMETER(	ESTXC=208,	ESFDM=209,	ESFCM=210,	ESIC=211,	ESISM=212)
INTEGER	ESDMM,	ESRFP,	ESRFM,	ESBIS,	ESBISI
PARAMETER(	ESDMM=213,	ESRFP=214,	ESRFM=215,	ESBIS=216,	ESBISI=217)
INTEGER	ESBIC,	ESBISM,	ESBDMM,	ESBRFP,	ESBRFM
PARAMETER(	ESBIC=218,	ESBISM=219,	ESBDMM=220,	ESBRFP=221,	ESBRFM=222)
INTEGER	ESLSS,	ESEDC,	ESCAC,	ESSAC,	ESPSC
PARAMETER(	ESLSS=223,	ESEDC=224,	ESCAC=225,	ESSAC=226,	ESPSC=227)
INTEGER	ESRCM,	ESDPCI,	ESDMI,	ESPLP,	ESPMP
PARAMETER(	ESRCM=228,	ESDPCI=229,	ESDMI=230,	ESPLP=231,	ESPMP=232)
INTEGER	ESTXRP,	ESIP,	ESEDP,	ESDMR,	ESRFR
PARAMETER(	ESTXRP=233,	ESIP=234,	ESEDP=235,	ESDMR=236,	ESRFR=237)
INTEGER	ESPSR,	ESPAP,	ESLSR,	ESDCR,	ESCMR
PARAMETER(	ESPSR=238,	ESPAP=239,	ESLSR=240,	ESDCR=241,	ESCMR=242)

12. List of the PHIGS PLUS function names

The complete list of PHIGS PLUS function names follows.

12.1. List of functions ordered alphabetically by bound name

PBSC3	BS-C-3	NON-UNIFORM B-SPLINE CURVE 3
PBSC3C	BS-C-3-C	NON-UNIFORM B-SPLINE CURVE 3 WITH COLOUR
PBSS3	BS-S-3	NON-UNIFORM B-SPLINE SURFACE 3
PBSS3D	BS-S-3-D	NON-UNIFORM B-SPLINE SURFACE 3 WITH DATA
PCA3P	C-A-3-P	CELL ARRAY 3 PLUS
PFAS3D	F-A-S-3-D	FILL AREA SET 3 WITH DATA
PFASD	F-A-S-D	FILL AREA SET WITH DATA
PPCSC	P-C-S-C	PACK COLOUR SPLINE CURVE
PPCSS	P-C-S-S	PACK COLOUR SPLINE SURFACE
PPDSS	P-D-S-S	PACK DATA SPLINE SURFACE
PPLS3C	PL-S-3-C	POLYLINE SET 3 WITH COLOUR
PPTC	P-T-C	PACK TRIMMING CURVE
PQBSSF	Q-BS-S-F	INQUIRE B-SPLINE SURFACE FACILITIES
PQCMF	Q-C-M-F	INQUIRE COLOUR MAPPING FACILITIES
PQCMMF	Q-C-M-M-F	INQUIRE COLOUR MAPPING METHOD FACILITIES
PQCMR	Q-C-M-R	INQUIRE COLOUR MAPPING REPRESENTATION
PQCMS	Q-C-M-S	INQUIRE COLOUR MAPPING STATE
PQCVF	Q-CV-F	INQUIRE CURVE FACILITIES
PQDCF	Q-D-C-F	INQUIRE DEPTH CUE FACILITIES
PQDCMF	Q-D-C-M-F	INQUIRE DIRECT COLOUR MODEL FACILITIES
PQDCR	Q-D-C-R	INQUIRE DEPTH CUE REPRESENTATION
PQDCWP	Q-DC-W-P	INQUIRE DYNAMICS OF WORKSTATION PLUS
PQDMF	Q-D-M-F	INQUIRE DATA MAPPING FACILITIES
PQDMR	Q-D-M-R	INQUIRE DATA MAPPING REPRESENTATION
PQECMI	Q-E-C-M-I	INQUIRE LIST element OF COLOUR MAPPING INDICES
PQEDCI	Q-E-D-C-I	INQUIRE LIST element OF DEPTH CUE INDICES
PQEDMI	Q-E-D-M-I	INQUIRE LIST element OF DATA MAPPING INDICES
PQEDP	Q-ED-P	INQUIRE EDGE REPRESENTATION PLUS
PQELSI	Q-E-L-S-I	INQUIRE LIST element OF LIGHT SOURCE INDICES
PQEFSI	Q-E-P-S-I	INQUIRE LIST element OF PARAMETRIC SURFACE INDICES
PQERFI	Q-E-R-I	INQUIRE LIST element OF REFLECTANCE INDICES
PQIFP	Q-I-F-P	INQUIRE INTERIOR FACILITIES PLUS
PQIP	Q-I-P	INQUIRE INTERIOR REPRESENTATION PLUS
PQLSF	Q-L-S-F	INQUIRE LIGHT SOURCE FACILITIES
PQLSR	Q-L-S-R	INQUIRE LIGHT SOURCE REPRESENTATION
PQM3D	Q-M-3-D	QUADRILATERAL MESH 3 WITH DATA
PQMD	Q-M-D	QUADRILATERAL MESH WITH DATA
PQPAP	Q-PA-P	INQUIRE PATTERN REPRESENTATION PLUS
PQPCMR	Q-P-C-M-R	INQUIRE PREDEFINED COLOUR MAPPING REPRESENTATION
PQPCR	Q-P-D-C-R	INQUIRE PREDEFINED DEPTH CUE REPRESENTATION
PQPDMR	Q-P-D-M-R	INQUIRE PREDEFINED DATA MAPPING REPRESENTATION
PQPEDP	Q-P-ED-P	INQUIRE PREDEFINED EDGE REPRESENTATION PLUS
PQPIP	Q-P-I-P	INQUIRE PREDEFINED INTERIOR REPRESENTATION PLUS
PQPLFP	Q-PL-F-P	INQUIRE POLYLINE FACILITIES PLUS
PQPLP	Q-PL-P	INQUIRE POLYLINE REPRESENTATION PLUS
PQPLSR	Q-P-L-S-R	INQUIRE PREDEFINED LIGHT SOURCE REPRESENTATION
PQPMP	Q-PM-P	INQUIRE POLYMARKER REPRESENTATION PLUS
PQPPAP	Q-P-PA-P	INQUIRE PREDEFINED PATTERN REPRESENTATION PLUS
PQPPLP	Q-P-PL-P	INQUIRE PREDEFINED POLYLINE REPRESENTATION PLUS
PQPPMP	Q-P-PM-P	INQUIRE PREDEFINED POLYMARKER REPRESENTATION PLUS
PQPPSR	Q-P-P-S-R	INQUIRE PREDEFINED PARAMETRIC SURFACE REPRESENTATION
PQPRFR	Q-P-RF-R	INQUIRE PREDEFINED REFLECTANCE REPRESENTATION
PQPSF	Q-P-S-F	INQUIRE PARAMETRIC SURFACE FACILITIES
PQPSR	Q-P-S-R	INQUIRE PARAMETRIC SURFACE REPRESENTATION
PQPTXP	Q-P-TX-P	INQUIRE PREDEFINED TEXT REPRESENTATION PLUS
PQRCMF	Q-R-C-M-F	INQUIRE RENDERING COLOUR MODEL FACILITIES
PQRFF	Q-RF-F	INQUIRE REFLECTANCE FACILITIES
PQRFR	Q-RF-R	INQUIRE REFLECTANCE REPRESENTATION

PQTCF	Q-T-C-F	INQUIRE TRIMMING CURVE FACILITIES
PQTXP	Q-TX-P	INQUIRE TEXT REPRESENTATION PLUS
PQWSLP	Q-W-S-L-P	INQUIRE WORKSTATION STATE TABLE LENGTHS PLUS
PSBDMI	S-B-D-M-I	SET BACK DATA MAPPING INDEX
PSBDMM	S-B-D-M-M	SET BACK DATA MAPPING METHOD
PSBIC	S-B-I-C	SET BACK INTERIOR COLOUR
PSBII	S-B-I-I	SET BACK INTERIOR INDEX
PSBIS	S-B-I-S	SET BACK INTERIOR STYLE
PSBISI	S-B-I-S-I	SET BACK INTERIOR STYLE INDEX
PSBISM	S-B-I-S-M	SET BACK INTERIOR SHADING METHOD
PSBRFI	S-B-RF-I	SET BACK REFLECTANCE INDEX
PSBRFM	S-B-RF-M	SET BACK REFLECTANCE MODEL
PSBRFP	S-B-RF-P	SET BACK REFLECTANCE PROPERTIES
PSCAC	S-C-A-C	SET CURVE APPROXIMATION CRITERIA
PSCMI	S-C-M-I	SET COLOUR MAPPING INDEX
PSCMR	S-C-M-R	SET COLOUR MAPPING REPRESENTATION
PSDCI	S-D-C-I	SET DEPTH CUE INDEX
PSDCR	S-D-C-R	SET DEPTH CUE REPRESENTATION
PSDMI	S-D-M-I	SET DATA MAPPING INDEX
PSDMM	S-D-M-M	SET DATA MAPPING METHOD
PSDMR	S-D-M-R	SET DATA MAPPING REPRESENTATION
PSEDC	S-ED-C	SET EDGE COLOUR
PSEDP	S-ED-P	SET EDGE REPRESENTATION PLUS
PSFAS3	S-F-A-S-3	SET OF FILL AREA SETS 3 WITH DATA
PSFASD	S-F-A-S-D	SET OF FILL AREA SETS WITH DATA
PSFCM	S-F-C-M	SET FACET CULLING MODE
PSFDM	S-F-D-M	SET FACET DISTINGUISHING MODE
PSIC	S-I-C	SET INTERIOR COLOUR
PSIP	S-I-P	SET INTERIOR REPRESENTATION PLUS
PSISM	S-I-S-M	SET INTERIOR SHADING METHOD
PSLSR	S-L-S-R	SET LIGHT SOURCE REPRESENTATION
PSLSS	S-L-S-S	SET LIGHT SOURCE STATE
PSPAP	S-PA-P	SET PATTERN REPRESENTATION PLUS
PSPLC	S-PL-C	SET POLYLINE COLOUR
PSPLP	S-PL-P	SET POLYLINE REPRESENTATION PLUS
PSPLSM	S-PL-S-M	SET POLYLINE SHADING METHOD
PSPMC	S-PM-C	SET POLYMARKER COLOUR
PSPMP	S-PM-P	SET POLYMARKER REPRESENTATION PLUS
PSPSC	S-P-S-C	SET PARAMETRIC SURFACE CHARACTERISTICS
PSPSI	S-P-S-I	SET PARAMETRIC SURFACE INDEX
PSPSR	S-P-S-R	SET PARAMETRIC SURFACE REPRESENTATION
PSRCM	S-R-C-M	SET RENDERING COLOUR MODEL
PSRFI	S-RF-I	SET REFLECTANCE INDEX
PSRFM	S-RF-M	SET REFLECTANCE MODEL
PSRFP	S-RF-P	SET REFLECTANCE PROPERTIES
PSRFR	S-RF-R	SET REFLECTANCE REPRESENTATION
PSSAC	S-S-A-C	SET SURFACE APPROXIMATION CRITERIA
PSTXC	S-TX-C	SET TEXT COLOUR
PSTXRP	S-TX-R-P	SET TEXT REPRESENTATION PLUS
PT3D	T-S-3-D	TRIANGLE SET 3 WITH DATA
PTSD	T-S-D	TRIANGLE SET WITH DATA
PTST3D	T-ST-3-D	TRIANGLE STRIP 3 WITH DATA
PTSTD	T-ST-D	TRIANGLE STRIP WITH DATA
PUCSC	U-C-S-C	UNPACK COLOUR SPLINE CURVE
PUCSS	U-C-S-S	UNPACK COLOUR SPLINE SURFACE
PUDSS	U-D-S-S	UNPACK DATA SPLINE SURFACE
PUTC	U-T-C	UNPACK TRIMMING CURVE

12.2 List of functions ordered alphabetically by PHIGS PLUS function name

PCA3P	C-A-3-P	CELL ARRAY 3 PLUS
PFAS3D	F-A-S-3-D	FILL AREA SET 3 WITH DATA
PFASD	F-A-S-D	FILL AREA SET WITH DATA
PQBSSF	Q-BS-S-F	INQUIRE B-SPLINE SURFACE FACILITIES
PQCMF	Q-C-M-F	INQUIRE COLOUR MAPPING FACILITIES
PQCMMF	Q-C-M-M-F	INQUIRE COLOUR MAPPING METHOD FACILITIES
PQCMR	Q-C-M-R	INQUIRE COLOUR MAPPING REPRESENTATION
PQCMS	Q-C-M-S	INQUIRE COLOUR MAPPING STATE
PQCVF	Q-CV-F	INQUIRE CURVE FACILITIES
PQDMF	Q-D-M-F	INQUIRE DATA MAPPING FACILITIES
PQDMR	Q-D-M-R	INQUIRE DATA MAPPING REPRESENTATION
PQDCF	Q-D-C-F	INQUIRE DEPTH CUE FACILITIES
PQDCR	Q-D-C-R	INQUIRE DEPTH CUE REPRESENTATION
PQDCMF	Q-D-C-M-F	INQUIRE DIRECT COLOUR MODEL FACILITIES
PQDCWP	Q-DC-W-P	INQUIRE DYNAMICS OF WORKSTATION PLUS
PQEDP	Q-ED-P	INQUIRE EDGE REPRESENTATION PLUS
PQIFP	Q-I-F-P	INQUIRE INTERIOR FACILITIES PLUS
PQIP	Q-I-P	INQUIRE INTERIOR REPRESENTATION PLUS
PQLSF	Q-L-S-F	INQUIRE LIGHT SOURCE FACILITIES
PQLSR	Q-L-S-R	INQUIRE LIGHT SOURCE REPRESENTATION
PQECMI	Q-E-C-M-I	INQUIRE LIST element OF COLOUR MAPPING INDICES
PQEDMI	Q-E-D-M-I	INQUIRE LIST element OF DATA MAPPING INDICES
PQEDCI	Q-E-D-C-I	INQUIRE LIST element OF DEPTH CUE INDICES
PQELSI	Q-E-L-S-I	INQUIRE LIST element OF LIGHT SOURCE INDICES
PQEPSI	Q-E-P-S-I	INQUIRE LIST element OF PARAMETRIC SURFACE INDICES
PQERFI	Q-E-R-I	INQUIRE LIST element OF REFLECTANCE INDICES
PQPSF	Q-P-S-F	INQUIRE PARAMETRIC SURFACE FACILITIES
PQPSR	Q-P-S-R	INQUIRE PARAMETRIC SURFACE REPRESENTATION
PQPAP	Q-PA-P	INQUIRE PATTERN REPRESENTATION PLUS
PQPLFP	Q-PL-F-P	INQUIRE POLYLINE FACILITIES PLUS
PQPLP	Q-PL-P	INQUIRE POLYLINE REPRESENTATION PLUS
PQPMF	Q-PM-P	INQUIRE POLYMARKER REPRESENTATION PLUS
PQPCMR	Q-P-C-M-R	INQUIRE PREDEFINED COLOUR MAPPING REPRESENTATION
PQPDMR	Q-P-D-M-R	INQUIRE PREDEFINED DATA MAPPING REPRESENTATION
PQPDRCR	Q-P-D-C-R	INQUIRE PREDEFINED DEPTH CUE REPRESENTATION
PQPEDP	Q-P-ED-P	INQUIRE PREDEFINED EDGE REPRESENTATION PLUS
PQPIR	Q-P-I-P	INQUIRE PREDEFINED INTERIOR REPRESENTATION PLUS
PQPLSR	Q-P-L-S-R	INQUIRE PREDEFINED LIGHT SOURCE REPRESENTATION
PQPPSR	Q-P-P-S-R	INQUIRE PREDEFINED PARAMETRIC SURFACE REPRESENTATION
PQPPAP	Q-P-PA-P	INQUIRE PREDEFINED PATTERN REPRESENTATION PLUS
PQPPLP	Q-P-PL-P	INQUIRE PREDEFINED POLYLINE REPRESENTATION PLUS
PQPPMP	Q-P-PM-P	INQUIRE PREDEFINED POLYMARKER REPRESENTATION PLUS
PQPRFR	Q-P-RF-R	INQUIRE PREDEFINED REFLECTANCE REPRESENTATION
PQPTXP	Q-P-TX-P	INQUIRE PREDEFINED TEXT REPRESENTATION PLUS
PQRFF	Q-RF-F	INQUIRE REFLECTANCE FACILITIES
PQRRF	Q-RF-R	INQUIRE REFLECTANCE REPRESENTATION
PQRCMF	Q-R-C-M-F	INQUIRE RENDERING COLOUR MODEL FACILITIES
PQTXP	Q-TX-P	INQUIRE TEXT REPRESENTATION PLUS
PQTCF	Q-T-C-F	INQUIRE TRIMMING CURVE FACILITIES
PQWSLP	Q-W-S-L-P	INQUIRE WORKSTATION STATE TABLE LENGTHS PLUS
PBSC3	BS-C-3	NON-UNIFORM B-SPLINE CURVE 3
PBSCC3	BS-C-3-C	NON-UNIFORM B-SPLINE CURVE 3 WITH COLOUR
PBSS3	BS-S-3	NON-UNIFORM B-SPLINE SURFACE 3
PBSSD3	BS-S-3-D	NON-UNIFORM B-SPLINE SURFACE 3 WITH DATA
PPCSC	P-C-S-C	PACK COLOUR SPLINE CURVE
PPCSS	P-C-S-S	PACK COLOUR SPLINE SURFACE
PPDSS	P-D-S-S	PACK DATA SPLINE SURFACE
PPTC	P-T-C	PACK TRIMMING CURVE
PPLS3C	PL-S-3-C	POLYLINE SET 3 WITH COLOUR
PQM3D	Q-M-3-D	QUADRILATERAL MESH 3 WITH DATA
PQMD	Q-M-D	QUADRILATERAL MESH WITH DATA
PSBDMI	S-B-D-M-I	SET BACK DATA MAPPING INDEX

PSBDM	S-B-D-M-M	SET BACK DATA MAPPING METHOD
PSBIC	S-B-I-C	SET BACK INTERIOR COLOUR
PSBII	S-B-I-I	SET BACK INTERIOR INDEX
PSBISM	S-B-I-S-M	SET BACK INTERIOR SHADING METHOD
PSBIS	S-B-I-S	SET BACK INTERIOR STYLE
PSBISI	S-B-I-S-I	SET BACK INTERIOR STYLE INDEX
PSBRFI	S-B-RF-I	SET BACK REFLECTANCE INDEX
PSBRFM	S-B-RF-M	SET BACK REFLECTANCE MODEL
PSBRFP	S-B-RF-P	SET BACK REFLECTANCE PROPERTIES
PSCMI	S-C-M-I	SET COLOUR MAPPING INDEX
PSCMR	S-C-M-R	SET COLOUR MAPPING REPRESENTATION
PSCAC	S-C-A-C	SET CURVE APPROXIMATION CRITERIA
PSDMI	S-D-M-I	SET DATA MAPPING INDEX
PSDMM	S-D-M-M	SET DATA MAPPING METHOD
PSDMR	S-D-M-R	SET DATA MAPPING REPRESENTATION
PSDCI	S-D-C-I	SET DEPTH CUE INDEX
PSDCR	S-D-C-R	SET DEPTH CUE REPRESENTATION
PSEDC	S-ED-C	SET EDGE COLOUR
PSERP	S-ED-P	SET EDGE REPRESENTATION PLUS
PSFCM	S-F-C-M	SET FACET CULLING MODE
PSFDM	S-F-D-M	SET FACET DISTINGUISHING MODE
PSIC	S-I-C	SET INTERIOR COLOUR
PSIP	S-I-P	SET INTERIOR REPRESENTATION PLUS
PSISM	S-I-S-M	SET INTERIOR SHADING METHOD
PSLSR	S-L-S-R	SET LIGHT SOURCE REPRESENTATION
PSLSS	S-L-S-S	SET LIGHT SOURCE STATE
PSFAS3	S-F-A-S-3	SET OF FILL AREA SETS 3 WITH DATA
PSFASD	S-F-A-S-D	SET OF FILL AREA SETS WITH DATA
PSPSC	S-P-S-C	SET PARAMETRIC SURFACE CHARACTERISTICS
PSPSI	S-P-S-I	SET PARAMETRIC SURFACE INDEX
PSPSR	S-P-S-R	SET PARAMETRIC SURFACE REPRESENTATION
PSPAP	S-PA-P	SET PATTERN REPRESENTATION PLUS
PSPLC	S-PL-C	SET POLYLINE COLOUR
PSPLP	S-PL-P	SET POLYLINE REPRESENTATION PLUS
PSPLSM	S-PL-S-M	SET POLYLINE SHADING METHOD
PSPMC	S-PM-C	SET POLYMARKER COLOUR
PSPMP	S-PM-P	SET POLYMARKER REPRESENTATION PLUS
PSRFI	S-RF-I	SET REFLECTANCE INDEX
PSRFM	S-RF-M	SET REFLECTANCE MODEL
PSRFP	S-RF-P	SET REFLECTANCE PROPERTIES
PSRFR	S-RF-R	SET REFLECTANCE REPRESENTATION
PSRCM	S-R-C-M	SET RENDERING COLOUR MODEL
PSSAC	S-S-A-C	SET SURFACE APPROXIMATION CRITERIA
PSTXC	S-TX-C	SET TEXT COLOUR
PSTXRP	S-TX-R-P	SET TEXT REPRESENTATION PLUS
PTS3D	T-S-3-D	TRIANGLE SET 3 WITH DATA
PTSD	T-S-D	TRIANGLE SET WITH DATA
PTST3D	T-ST-3-D	TRIANGLE STRIP 3 WITH DATA
PTSTD	T-ST-D	TRIANGLE STRIP WITH DATA
PUCSC	U-C-S-C	UNPACK COLOUR SPLINE CURVE
PUCSS	U-C-S-S	UNPACK COLOUR SPLINE SURFACE
PUDSS	U-D-S-S	UNPACK DATA SPLINE SURFACE
PUTC	U-T-C	UNPACK TRIMMING CURVE

13 The PHIGS PLUS function interface

13.1 General principles

For each PHIGS PLUS function the corresponding FORTRAN SUBROUTINE declaration is given. The name of the PHIGS PLUS function is listed, followed by its FORTRAN name and the corresponding parameters. After that, the list of parameters is described by type and a brief identifying phrase.

For the mapping of ENUMERATION types see clause 6 and clause 11.

13.2 Output primitive functions

POLYLINE SET 3 WITH COLOUR

SUBROUTINE PPLS3C(VFLAG,CTYPE,NCC,NPL,IXA,PXA,PYA,PZA,VCOLI,VCOLR)

Input Parameters:

INTEGER VFLAG	data per vertex data flag (PCD,PCDC)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NPL	number of point lists
INTEGER IXA(NPL)	array of end indices for point lists
REAL PXA(IXA(NPL)),PYA(IXA(NPL)),PZA(IXA(NPL))	coordinates of points(MC)
INTEGER VCOLI(IXA(NPL))	vertex colour indices
REAL VCOLR(NCC*IXA(NPL))	vertex colour values

FILL AREA SET 3 WITH DATA

```

SUBROUTINE PFAS3D(FFLAG,EFLAG,VFLAG,CTYPE,NCC,FCOLI,FCOLR,FNXA,
*FNZA,FDLEN,FDATA,NPL,IXA,EDATA,PXA,PYA,PZA,VCOLI,VCOLR,
*VNZA,VNZA,VDLEN,VDATA)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PEN,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN,PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER FCOLI	facet colour index
REAL FCOLR(NCC)	facet colour values
REAL FNXA,FNZA	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN)	facet application-specific data
INTEGER NPL	number of point lists
INTEGER IXA(NPL)	array of end indices for point lists
INTEGER EDATA(IXA(NPL))	edge data(POFF,PON)
REAL PXA(IXA(NPL)),PYA(IXA(NPL)),PZA(IXA(NPL))	coordinates of points(MC)
INTEGER VCOLI(IXA(NPL))	vertex colour indices
REAL VCOLR(NCC*IXA(NPL))	vertex colour values
REAL VNXA(IXA(NPL)),VNZA(IXA(NPL)),VNZA(IXA(NPL))	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*IXA(NPL))	vertex application-specific data

FILL AREA SET WITH DATA

```
SUBROUTINE PFASD(FFLAG,EFLAG,VFLAG,CTYPE,NCC,FCOLI,FCOLR,FXA,
*FNXA,FNZA,FDLEN,FDATA,NPL,IXA,EDATA,PXA,PYA,VCOLI,VCOLR,VNXA,
*VNZA,VNZA,VDLEN,VDATA)
```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCID,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PEN0,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER FCOLI	facet colour index
REAL FCOLR(NCC)	facet colour values
REAL FXA,FNZA,FNZA	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN)	facet application-specific data
INTEGER NPL	number of points lists
INTEGER IXA(NPL)	array of end indices for point lists
INTEGER EDATA(IXA(NPL))	edge data(POFF,PON)
REAL PXA(IXA(NPL)),PYA(IXA(NPL))	coordinates of points(MC)
INTEGER VCOLI(IXA(NPL))	vertex colour indices
REAL VCOLR(NCC*IXA(NPL))	vertex colour values
REAL VNXA(IXA(NPL)),VNZA(IXA(NPL)),VNZA(IXA(NPL))	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*IXA(NPL))	vertex application-specific data

CELL ARRAY 3 PLUS

```
SUBROUTINE PCA3P(CPXA,CPYA,CPZA,DIMX,DIMY,ISC,ISR,DX,DY,CTYPE,
*NCC,COLIA,COLRA)
```

Input Parameters:

REAL CPXA(3),CPYA(3),CPZA(3)	cell parallelogram(P,Q,R)(MC)
INTEGER DIMX,DIMY	the number of columns and number of rows in COLIA or COLRA
INTEGER ISC,ISR	indices of start column, start row
INTEGER DX,DY	number of columns,number of rows
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER COLIA(DIMX,DIMY)	colour index array
REAL COLRA(NCC*DIMX*DIMY)	colour value array

SET OF FILL AREA SETS 3 WITH DATA

```

SUBROUTINE PSFAS3(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NFS,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,NF,NP,IXA,EDATA,PXA,PYA,PZA,VCOLI,
*VCOLR,VNXA,VNYA,VNZA,VDLEN,VDATA,VIND)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PENNO,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN,PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NFS	number of fill area sets
INTEGER FCOLI(NFS)	facet colour indices
REAL FCOLR(NCC*NFS)	facet colour values
REAL FNXA(NFS),FNYA(NFS),FNZA(NFS)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*NFS)	facet application-specific data
INTEGER NF(NFS)	array of end vertex indices in IXA for each fill area set
INTEGER NP	number of points
INTEGER IXA(NF(NFS))	array of end vertex indices in VIND for each fill area
INTEGER EDATA(IXA(NF(NFS)))	edge data(POFF,PON)
REAL PXA(NP),PYA(NP),PZA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNYA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data
INTEGER VIND(IXA(NF(NFS)))	vertex indices for each fill area

SET OF FILL AREA SETS WITH DATA

```
SUBROUTINE PSFASD(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NFS,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,NF,NP,IXA,EDATA,PXA,PYA,VCOLI,VCOLR,
*VNXA,VNYA,VNZA,VDLEN,VDATA,VIND)
```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PENO,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NFS	number of fill area sets
INTEGER FCOLI(NFS)	facet colour indices
REAL FCOLR(NCC*NFS)	facet colour values
REAL FNXA(NFS),FNYA(NFS),FNZA(NFS)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*NFS)	facet application-specific data
INTEGER NF(NFS)	array of end indices in IXA for each fill area set
INTEGER NP	number of points
INTEGER IXA(NF(NFS))	array of end vertex indices in VIND for each fill area
INTEGER EDATA(IXA(NF(NFS)))	edge data(POFF,PON)
REAL PXA(NP),PYA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNYA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data
INTEGER VIND(IXA(NF(NFS)))	vertex indices for each fill area

TRIANGLE SET 3 WITH DATA

SUBROUTINE PTS3D(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NTRI,FCOLI,FCOLR,
 *FNXA,FNYA,FNZA,FDLEN,FDATA,NP,EDATA,PXA,PYA,PZA,VCOLI,VCOLR,
 *VNXA,VNYA,VNZA,VDLEN,VDATA,VIND)

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PEN,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NTRI	number of triangles
INTEGER FCOLI(NTRI)	facet colour indices
REAL FCOLR(NCC*NTRI)	facet colour values
REAL FNXA(NTRI),FNYA(NTRI),FNZA(NTRI)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*NTRI)	facet application-specific data
INTEGER NP	number of points
INTEGER EDATA(3*NTRI)	edge data(POFF,PON)
REAL PXA(NP),PYA(NP),PZA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNYA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data
INTEGER VIND(NTRI*3)	vertex indices

TRIANGLE SET WITH DATA

```

SUBROUTINE PTSD(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NTRI,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,NP,EDATA,PXA,PYA,VCOLI,VCOLR,VNXA,
*VNZA,VNZA,VDLEN,VDATA,VIND)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFID,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PENO,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NTRI	number of triangles
INTEGER FCOLI(NTRI)	facet colour indices
REAL FCOLR(NCC*NTRI)	facet colour values
REAL FNXA(NTRI),FNYA(NTRI),FNZA(NTRI)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*NTRI)	facet application-specific data
INTEGER NP	number of points
INTEGER EDATA(3*NTRI)	edge data(POFF,PON)
REAL PXA(NP),PYA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNZA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data
INTEGER VIND(NTRI*3)	vertex indices

TRIANGLE STRIP 3 WITH DATA

```

SUBROUTINE PTST3D(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NP,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,EDATA,PXA,PYA,PZA,VCOLI,VCOLR,VNXA,
*VNZA,VNZA,VDLEN,VDATA)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PENO,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NP	number of points
INTEGER FCOLI(NP-2)	facet colour indices
REAL FCOLR(NCC*(NP-2))	facet colour values
REAL FNXA(NP-2),FNYA(NP-2),FNZA(NP-2)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*(NP-2))	facet application-specific data
INTEGER EDATA(2*NP-3)	edge data(POFF,PON)
REAL PXA(NP),PYA(NP),PZA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNZA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data

TRIANGLE STRIP WITH DATA

```

SUBROUTINE PTSTD(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NP,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,EDATA,PXA,PYA,VCOLI,VCOLR,VNXA,VNYA,
*VNZA,VDLEN,VDATA)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PENO,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN,PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NP	number of points
INTEGER FCOLI(NP-2)	facet colour indices
REAL FCOLR(NCC*(NP-2))	facet colour values
REAL FNXA(NP-2),FNYA(NP-2),FNZA(NP-2)	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*(NP-2))	facet application-specific data
INTEGER EDATA(2*NP-3)	edge data(POFF,PON)
REAL PXA(NP),PYA(NP)	coordinates of points(MC)
INTEGER VCOLI(NP)	vertex colour indices
REAL VCOLR(NCC*NP)	vertex colour values
REAL VNXA(NP),VNYA(NP),VNZA(NP)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NP)	vertex application-specific data

QUADRILATERAL MESH 3 WITH DATA

```

SUBROUTINE PQM3D(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NC,NR,FCOLI,FCOLR,
*FNXA,FNYA,FNZA,FDLEN,FDATA,EDATA,PXA,PYA,PZA,VCOLI,VCOLR,VNXA,
*VNYA,VNZA,VDLEN,VDATA)

```

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFD,PFCN,PFCD,PFND,PFCND)
INTEGER EFLAG	data per edge data flag (PEN,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NC,NR	number of columns,number of rows
INTEGER FCOLI((NC-1)*(NR-1))	facet colour indices
REAL FCOLR(NCC*(NC-1)*(NR-1))	facet colour values
REAL FNXA((NC-1)*(NR-1)),FNYA((NC-1)*(NR-1)),FNZA((NC-1)*(NR-1))	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*(NC-1)*(NR-1))	facet application-specific data
INTEGER EDATA(2*NC*NR)	edge data(POFF,PON)
REAL PXA(NC*NR),PYA(NC*NR),PZA(NC*NR)	coordinates of points(MC)
INTEGER VCOLI(NC*NR)	vertex colour indices
REAL VCOLR(NCC*NC*NR)	vertex colour values
REAL VNXA(NC*NR),VNYA(NC*NR),VNZA(NC*NR)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NC*NR)	vertex application-specific data

QUADRILATERAL MESH WITH DATA

SUBROUTINE PQMD(FFLAG,EFLAG,VFLAG,CTYPE,NCC,NC,NR,FCOLI,FCOLR,
 *FNXA,FNYA,FNZA,FDLEN,FDATA,EDATA,PXA,PYA,VCOLI,VCOLR,VNXA,
 *VNYA,VNZA,VDLEN,VDATA)

Input Parameters:

INTEGER FFLAG	data per facet data flag (PFNO,PFC,PFN,PFDP,PCFN,PCFD,PFND,PCFND)
INTEGER EFLAG	data per edge data flag (PEN0,PEVF)
INTEGER VFLAG	data per vertex data flag(PCD,PCDC,PCDN, PCDD,PCDCN,PCDCD,PCDND,PCDCND)
INTEGER CTYPE	colour type
INTEGER NCC	number of components of colour value
INTEGER NC,NR	number of columns,number of rows
INTEGER FCOLI((NC-1)*(NR-1))	facet colour indices
REAL FCOLR(NCC*(NC-1)*(NR-1))	facet colour values
REAL FNXA((NC-1)*(NR-1)),FNYA((NC-1)*(NR-1)),FNZA((NC-1)*(NR-1))	facet normal data(MC)
INTEGER FDLEN	length of application-specific data list per facet
REAL FDATA(FDLEN*(NC-1)*(NR-1))	facet application-specific data
INTEGER EDATA(2*NC*NR)	edge data(POFF,PON)
REAL PXA(NC*NR),PYA(NC*NR)	coordinates of points(MC)
INTEGER VCOLI(NC*NR)	vertex colour indices
REAL VCOLR(NCC*NC*NR)	vertex colour values
REAL VNXA(NC*NR),VNZA(NC*NR),VNZA(NC*NR)	vertex normal data(MC)
INTEGER VDLEN	length of application-specific data list per vertex
REAL VDATA(VDLEN*NC*NR)	vertex application-specific data

NON-UNIFORM B-SPLINE CURVE 3

SUBROUTINE PBSC3(SORD,NKA,KNOTS,PARL,RTYPE,NCP,PXA,PYA,PZA,PWA)

Input Parameters:

INTEGER SORD	spline order
INTEGER NKA	number of knots
REAL KNOTS(NKA)	knots
REAL PARL(2)	parameter range limits
INTEGER RTYPE	rationality (PRAT,PNRAT)
INTEGER NCP	number of control points
REAL PXA(NCP),PYA(NCP),PZA(NCP),PWA(NCP)	control points(MC)

NON-UNIFORM B-SPLINE CURVE 3 WITH COLOUR

SUBROUTINE PBSC3C(SORD,NKA,KNOTS,PARL,RTYPE,NCP,PXA,PYA,PZA,PWA
*CSCLDR,CSCREC)

Input Parameters:

INTEGER SORD	spline order
INTEGER NKA	number of knots
REAL KNOTS(NKA)	knots
REAL PARL(2)	parameter range limits
INTEGER RTYPE	rationality (PRAT,PNRAT)
INTEGER NCP	number of control points
REAL PXA(NCP),PYA(NCP),PZA(NCP),PWA(NCP)	control points(MC)
INTEGER CSCLDR	number of array elements used in CSCREC
CHARACTER*80 CSCREC(CSCLDR)	colour spline curve data record ¹

NON-UNIFORM B-SPLINE SURFACE 3

SUBROUTINE PBSS3(USORD,VSORD,UNKA,VNKA,UKNOTS,VKNOTS,RTYPE,
*UNCP,VNCP,PXA,PYA,PZA,PWA,TCLDR,TCREC)

Input Parameters:

INTEGER USORD	u spline order
INTEGER VSORD	v spline order
INTEGER UNKA	number of knots for u
INTEGER VNKA	number of knots for v
REAL UKNOTS(UNKA)	u knots
REAL VKNOTS(VNKA)	v knots
INTEGER RTYPE	rationality (PRAT,PNRAT)
INTEGER UNCP	u number of control points dimension
INTEGER VNCP	v number of control points dimension
REAL PXA(UNCP*VNCP),PYA(UNCP*VNCP),PZA(UNCP*VNCP),PWA(UNCP*VNCP)	control points(MC)
INTEGER TCLDR	number of array elements used in TCREC

¹See PACK COLOUR SPLINE CURVE and UNPACK COLOUR SPLINE CURVE for a description of the colour spline curve data record.

CHARACTER*80 TCREC(TCLDR) trimming curve data record²

NON-UNIFORM B-SPLINE SURFACE 3 WITH DATA

SUBROUTINE PBSS3D(USORD,VSORD,UNKA,VNKA,UKNOTS,VKNOTS,RTYPE,
*UNCP,VNCP,PXA,PYA,PZA,PWA,TCLDR,TCREC,CSSLDR,CSSREC,DSSLDR,
*DSSREC)

Input Parameters:

INTEGER USORD	u spline order
INTEGER VSORD	v spline order
INTEGER UNKA	number of knots for u
INTEGER VNKA	number of knots for v
REAL UKNOTS(UNKA)	u knots
REAL VKNOTS(VNKA)	v knots
INTEGER RTYPE	rationality (PRAT,PNRAT)
INTEGER UNCP	u number of control points dimension
INTEGER VNCP	v number of control points dimension
REAL PXA(UNCP*VNCP),PYA(UNCP*VNCP),PZA(UNCP*VNCP),PWA(UNCP*VNCP)	control points(MC)
INTEGER TCLDR	number of array elements used in TCREC
CHARACTER*80 TCREC(TCLDR)	trimming curve data record ³
INTEGER CSSLDR	number of array elements used in CSSREC
CHARACTER*80 CSSREC(CSSLDR)	colour spline surface data record ⁴
INTEGER DSSLDR	number of array elements used in DSSREC
CHARACTER*80 DSSREC(DSSLDR)	data spline surface data record ⁵

²See PACK TRIMMING CURVE and UNPACK TRIMMING CURVE for a description of the trimming curve data record.

³See PACK TRIMMING CURVE and UNPACK TRIMMING CURVE for a description of the trimming curve data record.

⁴See PACK COLOUR SPLINE SURFACE and UNPACK COLOUR SPLINE SURFACE for a description of the colour spline surface data record.

⁵See PACK DATA SPLINE SURFACE and UNPACK DATA SPLINE SURFACE for a description of the data spline surface data record.

13.3 Attribute specification functions

13.3.1. Bundled attribute selection

SET DATA MAPPING INDEX

SUBROUTINE PSDMI(DMI)

Input Parameters:

INTEGER DMI

data mapping index

SET REFLECTANCE INDEX

SUBROUTINE PSRFI(RFI)

Input Parameters:

INTEGER RFI

reflectance index

SET BACK INTERIOR INDEX

SUBROUTINE PSBII(BII)

Input Parameters:

INTEGER BII

back interior index

SET BACK DATA MAPPING INDEX

SUBROUTINE PSBDMI(BDMI)

Input Parameters:

INTEGER BDMI

back data mapping index

SET BACK REFLECTANCE INDEX

SUBROUTINE PSBRFI(BRFI)

Input Parameters:

INTEGER BRFI

back reflectance index

SET PARAMETRIC SURFACE INDEX

SUBROUTINE PPSI(PSI)

Input Parameters:

INTEGER PSI

parametric surface index

13.3.2 Individual attribute selection**SET POLYLINE COLOUR**

SUBROUTINE PSPLC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	polyline colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET POLYLINE SHADING METHOD

SUBROUTINE PSPLSM(PLSM)

Input Parameters:

INTEGER PLSM	polyline shading method
--------------	-------------------------

SET POLYMARKER COLOUR

SUBROUTINE PSPMC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	polymarker colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET TEXT COLOUR

SUBROUTINE PSTXC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	text colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET FACET DISTINGUISHING MODE

SUBROUTINE PSFDM(DMODE)

Input Parameters:

INTEGER DMODE	distinguishing mode (POFF,PON)
---------------	--------------------------------

SET FACET CULLING MODE

SUBROUTINE PSFCM(CMODE)

Input Parameters:

INTEGER CMODE	culling mode (PNOFC,PBKFC,PFTFC)
---------------	----------------------------------

SET INTERIOR COLOUR

SUBROUTINE PSIC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	interior colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET INTERIOR SHADING METHOD

SUBROUTINE PSISM(ISM)

Input Parameters:

INTEGER ISM	interior shading method
-------------	-------------------------

SET DATA MAPPING METHOD

SUBROUTINE PSDMM(DMM,LDR,DATREC)

Input Parameters:

INTEGER DMM	data mapping method
INTEGER LDR	length of data record array
CHARACTER*80 DATREC(LDR)	data record

DATREC parameters to build DATA MAPPING METHOD DATA RECORD for

DMM = PCDDM:

The IA array contains:

- The number of source selectors
- The actual list of source selectors

The other arrays are empty.

Common element:

IL= 1+*

IA(1)= number of source selectors = NSS

IA= element 2 through NSS+1 contain source selectors

RL= 0

RA= ()

SL= 0

LSTR= ()

STR= ()

DATREC parameters to build DATA MAPPING METHOD DATA RECORD for

DMM = PSUDM:

The IA array contains:

- The number source selectors
- The actual list of source selectors
- The index of the data value
- The colour type(CT)
- The number of colour values(NCV)
- The list of colour indices. There are NCV colour indices, if CT is indirect.
- The number of components of colour value(NCC), if CT is not indirect.

The RA array contains:

- The lower range limit
- The upper range limit
- The list of colour values. There are NCV colour values, each having as many components as required by CT.

Common element:

IL= 5+*

IA(1)= number of source selectors = NSS

IA= element 2 through NSS+1 contain source selectors

IA(NSS+2)= data value index

IA(NSS+3)= colour type = CT

IA(NSS+4)= number of specified colour = NCV

RL= 2+*

RA(1)= lower range limit

RA(2)= upper range limit

SL= 0

LSTR= ()

STR= ()

Individual element:

CASE colour type=PINDDIR:

IA=element NSS+5 through NSS+4+NCV contain the colour indices

CASE colour type=PRGB or PCIE or PHSV or PHLS:

IA(NSS+5)= number of components of colour value = NCC

RA=element 3 through NCC*NCV+2 contain the colour values

example r1,g1,b1,r2,g2,b2,...

DATREC parameters to build DATA MAPPING METHOD DATA RECORD for

DMM = PSNUDM:

The IA array contains:

- The number of source selectors
- The actual list of source selectors
- The index of the data value
- The colour type(CT)
- The number of range boundaries(NRB)
- The list of colour indices. There are NRB+1 colour indices, if CT is indirect.
- The number of components of colour value(NCC), if CT is not indirect.

The RA array contains:

- The actual data of range boundaries
- The list of colour values. There are NRB+1 colour values, each having as many components as required by CT.

Common element:

IL= 5+NSS+*

IA(1)= number of source selectors = NSS

IA= element 2 through NSS+1 contain source selectors

IA(NSS+2)= data value index

IA(NSS+3)= colour type = CT

IA(NSS+4)= number of range boundaries = number of specified colour -1 = NRB

RL= NRB+*

RA=element 1 through NRB contain the range boundary

SL= 0

LSTR= ()

STR= ()

Individual element:

CASE colour type=PINDIR:

IA=element NSS+5 through NSS+4+NRB+1 contain the colour indices

CASE colour type=PRGB or PCIE or PHSV or PHLS:

IA(NSS+5)= number of components of colour value = NCC

RA=element NRB+1 through NCC*(NRB+1)+NRB contain the colour values

example r1,g1,b1,r2,g2,b2,...

DATREC parameters to build DATA MAPPING METHOD DATA RECORD for
DMM = PBUDM:

The IA array contains:

- The number of source selectors
- The actual list of source selectors
- The two indices of the data value
- The colour type(CT)
- The number of colour lists(NCL)
- The array of end indices of colour lists
- The list of colour indices. There are a lot of colour indices, if CT is indirect.
- The number of components of colour value(NCC), if CT is not indirect.

The RA array contains:

- The actual data of lower and upper limit of Ra range boundary
- The actual data of lower and upper limit of Rb range boundary
- The list of colour values. There are a lot of colour values, each having as many components as required by CT.

Common element:

IL= 6+NSS+NCL

IA(1)= number of source selectors = NSS

IA= element 2 through NSS+1 contain source selectors

IA(NSS+2)= data value index 1

IA(NSS+3)= data value index 2

IA(NSS+4)= colour type = CT

IA(NSS+5)= number of colour lists = NCL

IA= element NSS+6 through NSS+5+NCL contain array of end indices of colour lists

RL= 4+*

RA(1)= lower limit of Ra range

RA(2)= upper limit of Ra range

RA(3)= lower limit of Rb range

RA(4)= upper limit of Rb range

SL= 0

LSTR= ()

STR= ()

Individual element:

CASE colour type=PINDIR:

IA=element NSS+5+NCL+1 through NSS+5+NCL+IA(NSS+5+NCL) contain
the colour indices

CASE colour type=PRGB or PCIE or PHSV or PHLS:

IA(NSS+5+NCL+1)= number of components of colour value = NCC

RA=element 5 through NCC*IA(NSS+5+NCL)+4 contain the colour values

example r1,g1,b1,r2,g2,b2,...

DATREC parameters to build DATA MAPPING METHOD DATA RECORD for
DMM = PBNUDM:

The IA array contains:

- The number of source selectors
- The actual list of source selectors

- The two indices of the data value
 - The colour type(CT)
 - The number of Ra range boundaries(NRAB)
 - The array of end indices of Rb range boundaries data
 - The list of colour indices. There is a lot of colour indices (number of Rb range boundaries data + number of Ra range boundaries), if CT is indirect.
 - The number of components of colour value(NCC), if CT is not indirect.
- The RA array contains:

- The actual data of Ra range boundaries
- The actual data of Rb range boundaries
- The list of colour values. There is a lot of colour values (number of Rb range boundaries data + number of Ra range boundaries), each having as many components as required by CT.

Common element:

IL= 6+*

IA(1)= number of source selectors = NSS

IA= element 2 through NSS+1 contain source selectors

IA(NSS+2)= data value index 1

IA(NSS+3)= data value index 2

IA(NSS+4)= colour type = CT

IA(NSS+5)= number of Ra range boundaries = NRAB

IA=element NSS+6 through NSS+5+NRAB array of end indices of Rb range boundaries

RL= NRAB+IA(NSS+5+NRAB)+*

RA=element 1 through NRAB contain Ra range boundary

RA=element NRAB+1 through NRAB+IA(NSS+5+NRAB) contain Rb range boundary

SL= 0

LSTR= ()

STR= ()

Individual element:

CASE colour type=PINDIR:

IA=element NSS+5+NRAB+1 through NSS+4+NRAB+1+IA(NSS+5+NRAB)
contain colour indices

CASE colour type=PRGB or PCIE or PHSV or PHLS:

IA(NSS+5+NRAB+1)= number of components of colour value = NCC

RA=element NRAB+IA(NSS+5+NRAB)+1 through
NRAB+IA(NSS+5+NRAB)+NCC*(IA(NSS+5+NRAB))
contain the colour values

example r1,g1,b1,r2,g2,b2,...

SET REFLECTANCE PROPERTIES

SUBROUTINE PSRFP(RPTYPE,LDR,DATREC)

Input Parameters:

INTEGER RPTYPE reflectance properties type

INTEGER LDR length of data record array

CHARACTER*80 DATREC(LDR) data record

DATREC parameters to build REFLECTANCE PROPERTIES DATA RECORD for
RPTYPE = PSRPT:

The IA array contains:

- The colour type(CT)
- The number of components of colour value(NCC)

- The colour index of the specular colour if CT is indirect.

The RA array contains:

- The actual data of ambient reflection coefficient
- The actual data of diffuse reflection coefficient
- The actual data of specular reflection coefficient
- The actual data of specular exponent
- The actual data of colour values as required by CT.

Common element:

IL= 2+*

IA(1)= colour type = CT

IA(2)= number of components of colour value = NCC

RL= 4+*

RA(1)= ambient reflection coefficient

RA(2)= diffuse reflection coefficient

RA(3)= specular reflection coefficient

RA(4)= specular exponent

SL= 0

LSTR= ()

STR= ()

Individual element:

CASE colour type=PINDIR:

IA(3)= specular colour index

CASE colour type=PRGB or PCIE or PHSV or PHLS:

RA=element 5 through 4+NCC contain the colour values

SET REFLECTANCE MODEL

SUBROUTINE PSRFM(RFM)

Input Parameters:

INTEGER RFM

reflectance model

SET BACK INTERIOR STYLE

SUBROUTINE PSBIS(INTS)

Input Parameters:

INTEGER INTS

interior style

(PHOLLO,PSOLID,PPATTR,PPATCH,PISEMP)

SET BACK INTERIOR STYLE INDEX

SUBROUTINE PSBISI(ISTYLI)

Input Parameters:

INTEGER ISTYLI

interior style index

SET BACK INTERIOR COLOUR

SUBROUTINE PSBIC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	back interior colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET BACK INTERIOR SHADING METHOD

SUBROUTINE PSBISM(ISM)

Input Parameters:

INTEGER ISM	interior shading method
-------------	-------------------------

SET BACK DATA MAPPING METHOD

SUBROUTINE PSBDMM(DMM,LDR,DATREC)

Input Parameters:

INTEGER DMM	data mapping method
INTEGER LDR	length of data record array
CHARACTER*80 DATREC(LDR)	data record ⁶

SET BACK REFLECTANCE PROPERTIES

SUBROUTINE PSBRFP(RPTYPE,LDR,DATREC)

Input Parameters:

INTEGER RPTYPE	reflectance properties type
INTEGER LDR	length of data record array
CHARACTER*80 DATREC(LDR)	data record ⁷

SET BACK REFLECTANCE MODEL

SUBROUTINE PSBRFM(RFM)

Input Parameters:

INTEGER RFM	reflectance model
-------------	-------------------

⁶See SET DATA MAPPING METHOD for a description of the data mapping method data records.

⁷See SET REFLECTANCE PROPERTIES for a description of the reflectance properties data records.

SET LIGHT SOURCE STATE

SUBROUTINE PSLSS(NACTI,ACTLST,NDEAI,DEALST)

Input Parameters:

INTEGER NACTI	number of activation indices
INTEGER ACTLST(NACTI)	activation list
INTEGER NDEAI	number of deactivation indices
INTEGER DEALST(NDEAI)	deactivation list

SET EDGE COLOUR

SUBROUTINE PSEDC(CTYPE,COLI,NCC,COLR)

Input Parameters:

INTEGER CTYPE	edge colour type
INTEGER COLI	colour index
INTEGER NCC	number of components of colour value
REAL COLR(NCC)	colour values

SET CURVE APPROXIMATION CRITERIA

SUBROUTINE PSCAC(ACRI,LDR,DATREC)

Input Parameters:

INTEGER ACRI	curve approximation criteria type
INTEGER LDR	length of data record array
CHARACTER*80 DATREC(LDR)	data record

DATREC parameters to build CURVE APPROXIMATION CRITERIA DATA RECORD
for ACRI=PWDCA: No data record

DATREC parameters to build CURVE APPROXIMATION CRITERIA DATA RECORD
for ACRI =PCBKCA:

The IA array contains:

- The actual data of count

The other arrays are empty.

IL= 1

IA(1)= count

RL= 0

RA= ()

SL= 0

LSTR= ()

STR= ()

DATREC parameters to build CURVE APPROXIMATION CRITERIA DATA RECORD
for ACRI =PCSWCA or PCSNCA or PCSDCA or PCDWCA or PCDNCA or
PCDDCA or PRWCA or PRNCA or PRDCA:

The RA array contains:

- The actual data of approximation value

The other arrays are empty.

IL= 0

IA= ()

RL= 1

RA(1)= approximation value

SL= 0

LSTR= ()

STR= ()

SET SURFACE APPROXIMATION CRITERIA

SUBROUTINE PSSAC(ACRI,LDR,DATREC)

Input Parameters:

INTEGER ACRI	surface approximation criteria type
INTEGER LDR	length of data record array
CHARACTER*80 DATREC(LDR)	data record

DATREC parameters to build SURFACE APPROXIMATION CRITERIA DATA RECORD for ACRI=PWDCA: No data record

DATREC parameters to build SURFACE APPROXIMATION CRITERIA DATA RECORD for ACRI=PCBKSA:

The IA array contains:

- The actual data of u count
- The actual data of v count

The other arrays are empty.

IL= 2

IA(1)= u count

IA(2)= v count

RL= 0

RA= ()

SL= 0

LSTR= ()

STR= ()

DATREC parameters to build SURFACE APPROXIMATION CRITERIA DATA RECORD for ACRI = PCSWSA or PCSNSA or PCSDSA:

The RA array contains:

- The actual data of u approximation value
- The actual data of v approximation value

The other arrays are empty.

IL= 0

IA= ()

RL= 2

RA(1)= u approximation value

RA(2)= v approximation value

SL= 0

LSTR= ()

STR= ()

DATREC parameters to build SURFACE APPROXIMATION CRITERIA DATA RECORD for ACRI = PPDWSA or PPDNSA or PPDDSA or PRWSA or PRNSA or PRDSA:

The RA array contains:

- The actual data of approximation value

The other arrays are empty.

IL= 0

IA= ()

RL= 1

RA(1)= approximation value

SL= 0

LSTR= ()

STR= ()