INTERNATIONAL STANDARD

ISO/IEC 9592-1

Second edition
1997-11-15

Information technology — Computer graphics and image processing — Programmer's Hierarchical Interactive Graphics System (PHIGS) —

Part 1:
Functional description

Technologies de l'information — Infographie et traitement de l'image — Interface de programmation du système graphique hiérarchisé (PHIGS) —

Partie 1: Description fonctionnelle

Reference number
ISO/IEC 9592-1:1997(E)

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Contents

1 Scope 1
2 Normative references 2
3 Definitions 3
4 PHIGS functional overview 19
4.1 About this part of ISO/IEC 9592 19
4.1.1 Specification and conformance 19
4.1.2 Registration 19
4.1.3 Notational conventions 19
4.2 Overview and concepts 19
4.2.1 Overview 19
4.2.2 Concepts 20
4.2.3 PHIGS profiles 24
4.3 Structures and traversal 25
4.3.1 Structure elements and structures 25
4.3.2 Structure networks 27
4.3.3 Structure traversal and display 29
4.3.3.1 Posting 29
4.3.3.2 Traversal 29
4.3.3.3 Posting groups 30
4.3.3.4 Priority 33
4.3.3.5 Direct interpretation structure 35
4.3.3.6 Traversing structure networks 38
4.3.4 Conditional traversal of structures 38
4.3.5 Structure editing 41
4.3.6 Structure element moving and copying 42
4.3.7 Manipulation of structures in CSS 42
4.3.8 CSS search and inquiry 43
4.3.9 Structure archival and retrieval 44
4.3.10 Generalized Structure Elements (GSE) 45
4.3.11 Application data 46
4.3.12 Direct interpretation structure picture definition 46
4.3.13 Stability of the non-retained picture 47
4.4 Output primitives 48
4.4.1 Structure elements and output primitives 48
4.4.2 Marker class 49
4.4.2.1 Polymarker 49
4.4.3 Curve class 49
4.4.3.1 Polyline 49
4.4.3.2 Polyline set 49
4.4.3.3 Non-uniform B-spline curve 49
4.4.3.4 Non-uniform B-spline curve with colour 50

© ISO/IEC 1997

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

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

4.4.3.5 Circle 50
4.4.3.6 Circular arc 50
4.4.3.7 Ellipse 51
4.4.3.8 Elliptical arc....51
4.4.4 Text class 51
4.4.4.1 Text....51
4.4.4.2 Annotation text 51
4.4.5 Surface class 51
4.4.5.1 Facets 51
4.4.5.2 With-data primitives 52
4.4.5.3 Clipping surfaces 53
4.4.5.4 Modelling clip....53
4.4.5.5 Fill area 53
4.4.5.6 Fill area set....53
4.4.5.7 Fill area set with data 54
4.4.5.8 Set of fill area sets with data....54
4.4.5.9 Triangle set with data 54
4.4.5.10 Triangle strip with data....55
4.4.5.11 Quadrilateral mesh with data....55
4.4.5.12 Fill circle....56
4.4.5.13 Circular arc close 56
4.4.5.14 Fill ellipse 56
4.4.5.15 Elliptical arc close 56
4.4.5.16 Non-uniform B-spline surface 56
4.4.5.17 Surface trimming 57
4.4.5.18 Non-uniform B-spline surface with data 58
4.4.5.19 Cell array 60
4.4.5.20 Generalized drawing primitive 61
4.5 Output primitive attributes....61
4.5.1 Types of attributes 61
4.5.1.1 Main classification 61
4.5.1.2 Geometric attributes 62
4.5.1.3 Non-geometric attributes 62
4.5.1.4 Viewing attributes....64
4.5.1.5 Identification attributes 64
4.5.1.6 Rendering attributes....64
4.5.1.7 Colour 65
4.5.2 General colour specification 65
4.5.3 Polymarker attributes....67
4.5.4 Curve attributes....69
4.5.4.1 Polyline attributes 69
4.5.4.2 Polyline set attributes....70
4.5.4.3 Non-uniform B-spline curve attributes....71
4.5.4.4 Non-uniform B-spline curve with colour attributes 72
4.5.5 Text attributes 72
4.5.5.1 Text primitive attributes 72
4.5.5.2 Annotation text attributes 79
4.5.5.3 Text extent and concatenation 80
4.5.6 Surface attributes 82
4.5.6.1 Interior attributes 82
4.5.6.2 Edge attributes 84
4.5.6.3 Individual edge control for surface primitives 85

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

4.5.6.4 Data mapping attributes 85
4.5.6.5 Fill area set with data attributes 86
4.5.6.6 Set of fill area set with data attributes 86
4.5.6.7 Triangle set with data attributes 86
4.5.6.8 Triangle strip with data attributes 86
4.5.6.9 Quadrilateral mesh with data attributes 86
4.5.6.10 Non-uniform B-spline surface attributes 86
4.5.6.11 Non-uniform B-spline surface with data attributes 89
4.5.6.12 Reflectance properties 89
4.5.6.13 Alpha 90
4.5.7 Generalized drawing primitive attributes 91
4.5.8 Implicitly specified attributes 91
4.5.8.1 General 91
4.5.8.2 Facet normal 91
4.5.8.3 Facet orientation 93
4.5.8.4 Reflectance normal 93
4.5.8.5 Intrinsic colour 93
4.5.8.6 Intrinsic alpha 94
4.5.9 Name set attribute 95
4.5.10 Highlighting 96
4.5.11 Minimal simulations 96
4.5.12 Degenerate primitives 96
4.5.13 Stability 96
4.6 The rendering pipeline 97
4.6.1 General 97
4.6.1.1 Primitives affected by the rendering pipeline 97
4.6.1.2 The effect of the interior style on lighting and shading 97
4.6.1.3 Aspects and attributes used in the rendering pipeline 98
4.6.2 Alpha mapping 98
4.6.3 Data mapping 98
4.6.4 Texture mapping 103
4.6.4.1 General 103
4.6.4.2 Specification 104
4.6.4.3 Representation 104
4.6.4.4 Parameterization 105
4.6.4.5 Composition 108
4.6.4.6 Sampling 112
4.6.4.7 Binding 113
4.6.4.8 Mapping 114
4.6.4.9 Mapping controls 114
4.6.5 Lighting 115
4.6.5.1 Reflectance calculation 115
4.6.5.2 Light sources 116
4.6.5.3 Workstation light sources 116
4.6.6 Shading 117
4.6.6.1 General 117
4.6.6.2 Interpolation 117
4.6.6.3 Colour interpolation 117
4.6.6.4 Data interpolation 118
4.6.6.5 Normal-vector interpretation 118
4.6.6.6 Polyline set shading 118
4.6.6.7 Interior shading 119

Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

4.6.7 The rendering colour model 123
4.6.8 Depth cueing 124
4.6.9 Alpha application 125
4.6.10 Colour mapping 125
4.7 Workstations 127
4.7.1 Workstation characteristics 127
4.7.2 Workstation configuration 128
4.7.3 Workstation selection 129
4.7.4 Controlling picture changes 129
4.7.4.1 Introduction 129
4.7.4.2 Retained picture changes 130
4.7.4.3 Non-retained picture updating 136
4.7.4.4 Selective retained picture updating 136
4.7.5 Device coordinate clip regions 137
4.7.6 Clearing the display surface 138
4.7.7 Sending messages to a workstation 138
4.7.8 Hidden line / hidden surface removal 138
4.7.9 Image resources 139
4.7.10 Target manipulation 140
4.7.11 Traversal resources 142
4.8 Coordinate systems and transformations 143
4.8.1 Coordinate system handedness 143
4.8.2 Modelling transformations and clipping 143
4.8.3 Modelling utility functions 145
4.8.4 Viewing 145
4.8.5 Viewing construction functions 148
4.8.6 Workstation transformation 154
4.8.7 Transformation of locator input 155
4.8.8 Transformation of stroke input 156
4.8.9 Input utility functions 157
4.9 Graphical input 159
4.9.1 Introduction to logical input devices 159
4.9.2 Logical input device model 160
4.9.3 Operating modes of logical input devices 161
4.9.4 Measures of each input class 162
4.9.5 Input queue and current event report 165
4.9.6 Initialization of input devices 165
4.9.7 Locator and stroke input using 2D input 167
4.9.8 Pick input using directly structured elements 168
4.9.9 Definition of logical input devices 169
4.9.10 Local input processing 169
4.9.10.1 General 169
4.9.10.2 Local viewing operations 170
4.9.10.3 Local lighting operations 174
4.10 PHIGS metafile interface 177
4.11 PHIGS states 178
4.12 Inquiry functions 179
4.13 Error handling 180
4.14 Special interfaces between PHIGS and application program 182
4.15 Limitations 182
4.15.1 General 182
4.15.2 Non-planar geometry and data 182

AVA Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

4.15.3 Relationship of shading method to geometry 182
4.15.4 Normal-vector interpolation 183
4.15.5 Effects of transformations 183
4.15.6 Approximation criteria and data splines 183
4.16 Minimum support criteria 184
4.16.1 Concepts of minimum support 184
4.16.2 Explicitly defined and required 184
4.16.3 Explicitly defined and non-required 184
4.16.4 Conceptually defined and non-required 185
5 PHIGS Functional Specification 186
5.1 Notational conventions 186
5.1.1 Function heading and parameter list 186
5.1.2 Functions generating structure elements 186
5.2 Control functions 188
5.3 Output primitive functions 202
5.3.1 General 202
5.3.2 Function definitions 203
5.4 Attribute specification functions 216
5.4.1 Bundled attribute selection 216
5.4.1.1 General 216
5.4.1.2 Function definitions 216
5.4.2 Individually selected bundled attributes 218
5.4.2.1 General 218
5.4.2.2 Function definitions 218
5.4.3 Individual attributes 229
5.4.3.1 General 229
5.4.3.2 Function definitions 230
5.4.4 Workstation attribute table definition 236
5.4.4.1 General 236
5.4.4.2 Function definitions 236
5.4.5 Workstation filter definition 244
5.4.6 Colour model control 245
5.4.7 HLHSR attributes 245
5.4.8 Attribute utility functions 246
5.5 Transformation and clipping functions 247
5.5.1 Modelling transformations and clipping 247
5.5.1.1 General 247
5.5.1.2 Function definitions 247
5.5.2 View operations 249
5.5.3 Workstation transformation 250
5.5.4 Utility functions to support modelling 251
5.5.4.1 General 251
5.5.4.2 Function definitions 252
5.5.5 Utility functions to support viewing 255
5.5.6 View construction functions 256
5.5.7 Workstation utility functions 261
5.6 Structure content functions 261
5.7 Structure manipulation functions 272
5.8 Structure display functions 274
5.9 Structure archiving functions 278
5.10 Input functions 282
5.10.1 Pick support functions 282

Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

5.10.2 Local input device definition functions 285
5.10.2.1 General....285
5.10.2.2 Function definitions 286
5.10.3 Initialization of input devices 291
5.10.3.1 General....291
5.10.3.2 Function definitions 292
5.10.4 Setting the mode of input devices....295
5.10.5 Request input functions 298
5.10.5.1 General....298
5.10.5.2 Function definitions 298
5.10.6 Sample input functions 301
5.10.6.1 General....301
5.10.6.2 Function definitions 301
5.10.7 Event input functions....303
5.10.7.1 General....303
5.10.7.2 Function definitions 304
5.10.8 Local input operation functions....307
5.10.9 Direct interpretation picking functions....309
5.10.9.1 General....309
5.10.9.2 Function definitions 309
5.11 Metafile functions 310
5.12 Inquiry functions....312
5.12.1 Introduction 312
5.12.2 Inquiry functions for operating state values 313
5.12.3 Inquiry functions for PHIGS description table....313
5.12.4 Inquiry functions for PHIGS state list 314
5.12.5 Inquiry functions for workstation state list....314
5.12.6 Inquiry functions for workstation description table 315
5.12.6.1 General....315
5.12.6.2 Function definitions 316
5.12.7 Inquiry function for structure state list 317
5.12.8 Inquiry functions for structure content 317
5.12.9 Inquiry functions for error state list....321
5.13 Error control functions....322
5.14 Special interface function 323
6 PHIGS data structures 324
6.1 Notation and data types 324
6.1.1 Notation 324
6.1.2 Simple datatypes 324
6.1.3 Compound datatypes 325
6.1.4 Integer selection types 332
6.1.5 Enumeration types 342
6.1.6 Data records 348
6.1.7 Combination of datatypes 357
6.1.8 Variant datatypes 357
6.1.9 Language bindings 357
6.1.10 Coordinate systems 357
6.1.11 Permitted values 358
6.1.12 Alphabetic list of datatypes 359
6.2 Operating state table 363
6.3 PHIGS description table 363
6.4 PHIGS traversal state list....367

AVA Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

6.5 PHIGS state list....368
6.6 Workstation state list....369
6.7 Workstation description table....384
6.8 Structure state list....402
6.9 PHIGS error state list....402
6.10 Alphabetic list of entries....402
A Function Lists....424
A.1 Introduction....424
A.1 Alphabetic list of functions....424
B Error list....434
B.1 Error list ordered by error number....434
B.1.1 Implementation dependent....434
B.1.2 States....434
B.1.3 Workstations....434
B.1.4 Output attributes....435
B.1.5 Transformations and viewing....436
B.1.6 Structures....437
B.1.7 Input....437
B.1.8 Metafiles....437
B.1.9 Escape....438
B.1.10 Archive / retrieve....438
B.1.11 Miscellaneous....438
B.1.12 Output primitives....438
B.1.13 Texture output primitives....438
B.1.14 System....439
B.1.15 Reserved errors....439
C Interfaces....440
C.1 Introduction....440
C.2 Language binding....440
C.3 Implementation....440
D Allowable differences in PHIGS implementations....442
D.1 Introduction....442
D.2 Global differences....442
D.3 Workstation dependent differences....443
E HLHSR considerations....448
F Relationship of CGM and PHIGS....449
F.1 Introduction....449
F.2 Scope....449
F.3 Overview of the differences between PHIGS and CGM....449
F.4 Mapping concepts....449
F.4.1 Principles....449
F.4.2 Workstations....450
F.4.3 Picture generation....450
F.4.4 Picture input....450
F.4.5 Coordinates and clipping....451
F.4.6 Workstation transformation....451
G Colour models....452
G.1 Introduction....452
G.2 RGB colour model....452
G.3 CIELUV colour model....453
G.3.1 CIE XYZ colour space....453
G.3.2 CIE 1931 (Y,x,y) space....453

Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

G.3.3 The CIE 1976 (L*u*v*) CIELUV uniform colour space....456
G.3.4 Colour differences....457
G.4 HSV colour model....457
G.5 HLS colour model....458
G.6 Conversion between colour models....459
G.6.1 CIE XYZ reference model....459
G.6.2 Conversion between CIELUV and CIE XYZ models....459
G.6.3 Conversion between RGB and CIE XYZ models....460
G.6.3.1 Derivation of conversion factors....460
G.6.3.2 Conversion from RGB to CIE XYZ....460
G.6.3.3 Conversion from CIE XYZ to RGB....461
G.6.3.4 Representation of black....461
G.6.3.4 Example conversion....461
H Suggested reflectance formulae....462
H.1 Variable definitions and their sources....462
H.1 Reflection formulae....462
I Attributes applying to each primitive....464
J Inquiry error indicators and allowable states....477
K Suggested depth cueing formulae....484
K.1 Linear colour interpretation....484
K.1 Definitions....484
K.1 Formulae....484
L Bibliography....485
M Index of usage of datatypes....486
N International standardized profiles....508
O Configurability....509

AVA Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

Figure **1** - Layer model of PHIGS 20
Figure **2** - Direct interpretation structure interpretation 21
Figure **3** - CSS structure interpretation 21
Figure **4** - A hierarchical structure network 28
Figure **5** - A structure network with two posted structure networks 30
Figure **6** - PHIGS structure element processing architecture 31
Figure **7** - PHIGS traversal process types 32
Figure **8** - Posting group background styles 34
Figure **9** - Posting group with TRANSPARENT background style 35
Figure **10** - Effects of direct interpretation mode 37
Figure **13** - Data that may be associated with a facet 52
Figure **12** - Examples of area clipping 53
Figure **11** - Area inside a polygon 54
Figure **14** - A triangle set with data primitive 55
Figure **15** - A triangle strip with data primitive 55
Figure **16** - A quadrilateral mesh primitive 56
Figure **17** - Parameter space mapping rules for trimming curves 58
Figure **18** - Examples of valid trimming loops 58
Figure **19** - Examples of invalid trimming loops 59
Figure **20** - Mapping of CELL ARRAY 3 60
Figure **21** - Interpretation of a general colour 65
Figure **22** - Selection of the XXX COLOUR attribute 67
Figure **23** - Font description coordinate system 73
Figure **24** - Effects of changes in the geometric text attribute CHARACTER HEIGHT 73
Figure **25** - Effects of changes in CHARACTER EXPANSION FACTOR 73
Figure **26** - Effects of changes in CHARACTER SPACING 74
Figure **27** - Effects of changes in the geometric text attribute TEXT PATH 74
Figure **28** - Effects of changes in CHARACTER UP VECTOR 75
Figure **29** - Effects of changes in horizontal TEXT ALIGNMENT 75
Figure **30** - Effects of changes in vertical TEXT ALIGNMENT 76
Figure **31** - Effects of combined changes in text attributes 77
Figure **32** - Effects of different transformations on text in STROKE precision 79
Figure **33** - Replies to INQUIRE TEXT EXTENT with different text attributes and LEFT, RIGHT alignment 80
Figure **34** - Replies to INQUIRE TEXT EXTENT with different text attributes and UP, DOWN alignment 81
Figure **35** - Replication of the pattern box with fill areas 83
Figure **36** - Projection of the PATTERN REFERENCE POINT AND VECTORS onto the plane of the area 84
Figure **37** - The conceptual rendering pipeline 97
Figure **38** - UNIFORM data mapping of data values to colours 100
Figure **39** - NON-UNIFORM mapping of data values to colour 101
Figure **40** - UNIFORM mapping of data values to colour lists 102
Figure **41** - BI-VALUE NON-UNIFORM data mapping 103
Figure **42** - Texture parameterization operations 108
Figure **43** - A conceptual rendering pipeline for shading method 1 (NONE) 119
Figure **44** - A conceptual rendering pipeline for shading method 2 (COLOUR) 120
Figure **45** - A conceptual rendering pipeline for shading method 3 (DATA) 121
Figure **46** - A conceptual rendering pipeline for shading method 4 (DATA AND DOT) 122
Figure **47** - A conceptual rendering pipeline for shading method 5

Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

(DATA AND NORMAL) 123
Figure **48** - Depth cue scaling....124
Figure **49** - Display update part 1: is correct retained picture required....133
Figure **50** - Display update part 2: check modification mode 134
Figure **51** - The transformation pipeline....145
Figure **52** - Interpretation of the view index on workstations 146
Figure **53** - Viewing model for view representation 147
Figure **54** - Formation of V axis in VRC system....149
Figure **55** - Parallel projection view volume 151
Figure **56** - Perspective projection view volume....152
Figure **57** - Data flow chart for locator input....157
Figure 58 - Relationship of device coordinates to window system coordinates ...158
Figure **59** - The relationship between the measure and trigger for different operating modes 162
Figure **60** - Possible usage of metafile....178
Figure **61** - The RGB colour model....452
Figure **62** - The CIE 1931 x y chromaticity diagram 454
Figure **63** - Range of colours displayable by typical devices....455
Figure **64** - CIE 1976 u' v' uniform chromaticity diagram....456
Figure **65** - The HSV colour model....458
Figure **66** - The HLS colour model 459

AVA Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

Table **1** - Output primitive structure elements....25
Table **2** - Bundled attribute selection structure elements....26
Table **3** - Individually selected bundled attribute structure elements....26
Table **4** - Geometric attribute structure elements....26
Table **5** - Identification attribute structure elements....26
Table **6** - Viewing attribute structure elements....26
Table **7** - Rendering attribute structure elements....27
Table **8** - Transformation and clipping structure elements....27
Table **9** - Control structure elements....27
Table **10** - Editing structure element....27
Table **11** - Generalized structure element....27
Table **12** - Application data element....27
Table **13** - Test methods and the associated test conditions....39
Table **14** - Test conditions of each type....39
Table **15** - Output primitive classes....48
Table **16** - Colour types....66
Table **17** - Normal alignments....76
Table **18** - Initial values of geometric text attributes....76
Table **19** - Initial values of implicitly specified geometric text aspects....76
Table **20** - Sources of intrinsic colour....93
Table **21** - Sources of intrinsic alpha....94
Table **22** - Intrinsic colour source selectors....99
Table **23** - COMPOSITE REPLACE....110
Table **24** - COMPOSITE MODULATE....110
Table **25** - COMPOSITE BLEND ENVIRONMENT COLOUR....111
Table **26** - COMPOSITE DECAL....111
Table **27** - COMPOSITE DECAL BACKGROUND....112
Table **28** - Interpolation dependent sample frequencies....115
Table **29** - Defined reflectance models....116
Table **30** - Rendering colour models....124
Table **31** - Workstation categories....128
Table **32** - Light source parameterizations....175
Table **33** - Minimum required support for the PHIGS profiles....185
Table **34** - Entries in state lists and description tables....403

Documents
Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

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.

International Standard ISO/IEC 9592-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 24, *Computer graphics and image processing*.

This second edition cancels and replaces the first edition (ISO/IEC 9592-1:1989) and ISO/IEC 9592-4:1992, which have been technically revised. It also incorporates ISO/IEC 9592-1:1989/Amd.1:1992, ISO/IEC 9592-1:1989/Cor.1:1993, ISO/IEC 9592-1:1989/Cor.2:1994 and ISO/IEC 9592-4:1992/Cor.1:1994.

ISO/IEC 9592 consists of the following parts, under the general title *Information technology — Computer graphics and image processing — Programmer's Hierarchical Interactive Graphics System (PHIGS)*:

— *Part 1: Functional description*

— *Part 2: Archive file format*

— *Part 3: Specification for clear-text encoding of archive file*

Annexes A, B, D, I, J, M and N form an integral part of this part of ISO/IEC 9592. Annexes C, E, F, G, H, K, L and O are for information only.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Introduction

The Programmer's Hierarchical Interactive Graphics System (PHIGS) provides a set of functions for the definition, manipulation and display of geometrically related objects. It provides support for simulating the effects of lighting, shading and other properties that are important for the display of multi-dimensional data. This support is provided by output primitives specified as curves and surfaces; output primitives containing both geometric and non-geometric data in their definition; attributes that control the application of lighting and shading; and a generalized mechanism for colour specification. Support is provided for controlling the application of textures to objects and for interactively applying changes to light source definitions.

This International Standard enables graphical (and application) data to be stored in a hierarchical data store or to be interpreted directly. Information in the hierarchical data store can be inserted, modified and deleted with the provided functions. Information for direct interpretation can only be inserted.

The main motivations for standardization are to improve portability of programs and to define a basic methodology. Portability is achieved by providing all the capabilities necessary in a device independent way. If required, it provides applications with more direct control over the capabilities of the available hardware. The following principles are used in specifying PHIGS:

a) Design: the three goals are consistency of approach, compatibility with related standards and orthogonal functionality where possible.

b) Functionality: the goals are completeness with the minimal set of functions. Organization of functions should be such as to achieve compact programs. All the functions necessary for application programs to use a dynamic hierarchical graphics system should be included. Redundant functions are only supported where their availability enables application programs to improve performance or where some collection of capabilities is frequently used. Richness should be provided by utilities and toolkits on top of PHIGS.

c) Error Handling: error conditions should be minimized, and their impact well defined.

d) Device independence: the workstation concept in PHIGS deals with all device dependent functions. It allows an application program to address facilities of different graphics input and output devices with minimal changes to the application program. Where necessary, PHIGS allows an application program to address specific capabilities of graphics input and output devices.

e) Implementation: it should be possible to support PHIGS functions using most languages on most operating systems. A PHIGS implementation should be capable of executing without consuming undue amounts of computer resources.

This international standard has a strong relationship to the principles and functionality of ISO/IEC 9592-1:1989 (known informally as PHIGS-89) and incorporates the functionality of ISO/IEC 9592-4:1992 (known informally as PHIGS PLUS). However, it has been updated to provide additional functionality of relevance to modern hardware and applications.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Information technology – Computer graphics and image processing – Programmer's Hierarchical Interactive Graphics System (PHIGS) – Part 1: Functional description

1 Scope

This part of ISO/IEC 9592 specifies a set of functions for computer graphics programming, the Programmer's Hierarchical Interactive Graphics System (PHIGS). PHIGS is a graphics system for application programs that produce computer generated pictures on output devices. It supports operator input and interactions by supplying basic functions for graphical input and hierarchical picture definition. Picture definitions can be retained centrally where they may be edited by an application. Alternatively, graphical data may be processed without first storing it. To assist in this processing, explicit control over resources used to encapsulate the results of these processing operations is provided.

Basic application requirements in the areas of lighting and shading are provided through primitives and functions for controlling the rendering of 3D objects. Utilization of raster images in the generation of PHIGS pictures is provided.

Pictures are displayed on output devices which may have associated input devices. Several input devices can be used simultaneously. The application program is allowed to adapt its behaviour to make best use of their capabilities.

Graphical output can be constrained to particular views. Views can be specified parametrically and automatic processing of input operations can be used to control viewing. New input devices can be defined from the capabilities available.

Functions are specified for archiving picture definitions to file. In addition an interface to the Computer Graphics Metafile (ISO/IEC 8632) is described.

This part of ISO/IEC 9592 defines a language independent nucleus of a graphics system for integration into a programming language. PHIGS is embedded in a language layer obeying the particular conventions of the language. Such language bindings are specified in ISO/IEC 9593.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9592. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9592 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 646:1991, *Information technology - ISO 7-bit coded character set for information interchange.*

ISO/IEC 2022:1994, *Information technology - Character code structure and extension techniques.*

ISO 6093:1985, *Information processing - Representation of numerical values in character strings for information interchange.*

ISO/IEC 8632:1992, *Information technology - Computer graphics - Metafile for the storage and transfer of picture description information*

- *Part 1 : Functional description*
- *Part 2 : Character encoding*
- *Part 3 : Binary encoding*
- *Part 4 : Clear text encoding*

ISO/IEC 9593 (all parts), *Information technology - Computer graphics - Programmer's Hierarchical Interactive Graphics System (PHIGS) language bindings.*

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3 Definitions

For the purposes of this part of ISO/IEC 9592 the following definitions apply.

NOTE

As far as possible, graphics terminology which is commonly accepted and consistent with other graphics Standards is used.

3.1 acknowledgement: Output to the operator of a logical input device indicating that the input report generated by the firing of a trigger has either been accepted or rejected.

3.2 acknowledgement process: A process which notifies an operator of the status of an input report. Such input reports are generated by firing of an input report generation trigger.

3.3 addressable point: Any point of a device that can be addressed.

3.4 alpha: A value which specifies the proportion by which an object obscures the objects behind it. This information is often used to achieve transparency effects.

3.5 alpha blending: The operation in the rendering pipeline which uses the alpha data to mix the colours of the primitive being rendered with the colours already existing in the displayed picture.

3.6 alpha mapping: The action of associating the alpha information provided by the alpha source with the primitive being rendered.

3.7 alpha source: The location where alpha values are obtained.

3.8 ambient light source: A light source that contributes to the reflectance calculation independently of the orientation or position of the facet being illuminated or the location of the viewer's eye.

3.9 ambient reflection coefficient: The fraction of ambient light reflected from a facet.

3.10 ancestor structure: A parent structure or the ancestor of a parent structure.

3.11 annotation style: An aspect of annotation indicating how relationships between an annotation primitive and a reference point are displayed.

3.12 annotation text: An output primitive consisting of a character string which is always drawn parallel to the x-y plane of the display space and is unaffected by modelling and viewing transformations. Its position is determined by a reference point defined in modelling coordinate space and an offset in normalized projection coordinates.

3.13 application data structure element: A structure element for storing data used by an application program, the nature of which is not specified in this standard.

3.14 archive file: A mechanism for the storage and transportation of graphical data, represented by PHIGS structures and their contents.

3.15 aspect ratio: The ratio of lengths along the principal axes of an object.

3.16 aspect source flag (ASF): A flag indicating whether a particular workstation dependent aspect of an output primitive is selected from an attribute bundle, or as an individual attribute selection.

3.17 aspects of output primitives: The appearance of output primitives is controlled by the values of a set of characteristics called aspects, examples of which are the height of a character or the linetype of a polyline. Geometric aspects are workstation independent and are controlled by the corresponding attributes. For non-geometric aspects, the mapping between a particular aspect and its controlling attribute is defined by an associated aspect source flag (ASF). If the ASF is set to BUNDLED this aspect of the output primitive is controlled by the bundle index attribute. If the ASF is set to INDIVIDUAL then the aspect is controlled by the corresponding attribute.

3.18 atomic device: A logical input device providing a value derived from the values delivered by a one of the atomic input classes.

3.19 atomic input class: The characterization of the functionality of a simple logical input device. There are six classes of atomic logical input device: locator, stroke, valuator, choice, pick, and string.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.20 attenuation coefficient: A coefficient that determines the decrease in intensity of light as a function of the distance between a light source and an illuminated object.

3.21 attribute structure element: A structure element which, on traversal, updates the relevant attribute entry in the traversal state list.

3.22 attribute: Attributes control the aspects and properties of output primitives. There are five types of attributes: geometric, non-geometric, viewing, rendering and identification. The geometric and non-geometric attributes control the values of aspects of output primitives.

3.23 back facing: A back-facing facet has a facet normal that, when transformed to NPC, has a negative Z component. See also front facing.

3.24 back plane: A plane parallel to the view plane whose location is specified as an N coordinate value in the view reference coordinate system. Output primitives behind the back plane lie outside the view volume.

3.25 base target: A fixed target used for absolute target addressing.

3.26 break action: An implementation dependent and workstation dependent mechanism enabling the operator to interrupt an input operation.

3.27 bundle index: An attribute of an output primitive which is an index into a bundle table; which defines the workstation dependent aspects of the output primitive.

3.28 bundle table: A workstation dependent table specifying aspects of one or more output primitives. PHIGS has polyline, polymarker, text, interior, reflectance, parametric surface, data mapping, and edge bundle tables.

3.29 bundle table entry: A single entry in a bundle table. Each entry contains one value for each aspect which applies to the corresponding output primitive. This set is workstation dependent.

3.30 cell array: An output primitive consisting of a parallelogram of equal sized cells, each of which is a parallelogram and has a single colour.

3.31 centralized structure store (CSS): The conceptual workstation independent storage area for structures. Structures may reference other structures to produce structure networks within the centralized structure store. Structures may be posted to workstations for display.

3.32 character: A recognizable abstract graphic symbol which is independent of any specific design.

3.33 character base vector: An aspect of text which defines the direction of the baseline of a character. It is a two-dimensional vector in the text plane specified in the text structure element.

3.34 character body: The rectangle defining the horizontal and vertical limits of an individual character.

3.35 character expansion factor: An aspect of text which specifies the deviation of character width from the defined nominal value of a given font on a given workstation.

3.36 character height: An aspect of text which specifies the height of an upper case character.

3.37 character set: A registered interpretation for entries in the character code table (see ISO 2022).

3.38 character spacing: An aspect of text which specifies the fraction of the character height aspect to be added between adjacent character bodies in a string.

3.39 character up vector: An aspect of text which defines the principal up direction of the text string. It is a two-dimensional vector in the text plane specified in the text structure element.

3.40 character width: An aspect of text which specifies the width of a character. The actual width depends on the width to height ratio specified by the designer of the font to which the character belongs.

3.41 child structure: A structure specified in a structure reference.

3.42 choice device: A logical input device providing a non-negative integer defining one of a set of alternatives.

3.43 CIE: Abbreviation for the Commission Internationale de l'Eclairage. Used to refer to the CIE universal colour definition system used as a colour model.

3.44 CIELUV: Abbreviation for the CIE 1976 (L*u*v*) colour space.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.45 circular arc close: An output primitive consisting of a single filled area bounded by a circular arc in either a pie shape or with a chord joining the ends of the arc.

3.46 clipping: Removing parts of output primitives which lie outside a specified volume. The exact effect of clipping some output primitives may be implementation or workstation dependent.

3.47 colour index: An index used to access an entry in a colour table.

3.48 colour mapping: The conversion of direct colours in the rendering pipeline to other colours before they are displayed on the workstation.

3.49 colour model: Characterization of a colour space in terms of explicit parameters.

3.50 colour space: Geometric representation of colours in space, usually of three dimensions.

3.51 colour spline: The parametric curve or surface in colour space (or homogeneous colour space) defining the colour distribution over an output primitive.

3.52 colour system: A colour coordinate system.

3.53 colour table: A workstation dependent table, in which the entries specify the values defining a particular colour.

3.54 composite device: A logical input device whose measure consists of a list of values delivered by one or more different associated measure processes.

3.55 composite modelling transformation: A transformation applied to output primitives produced during structure traversal. It is defined as the concatenation of the local modelling transformation and global modelling transformations such that the local modelling transformation is the first transformation to be applied.

3.56 concentration exponent: A parameter of a spot light source that specifies the relative decrease of light as the angle of the light diverges from the centreline of the light source's cone of influence.

3.57 conditional traversal: The traversal of parts of a structure network dependent on the results of specific tests.

3.58 cone of influence: A conceptual cone that represents the influence of light from a spot light source. The cone of influence is defined by the light source's position, direction and spread angle.

3.59 conflict resolution flag: During the process of structure archiving from the CSS or structure retrieval from an archive, naming conflicts may occur between structures on the archive file and structures in the CSS. The conflict resolution flag indicates how these conflicts will be resolved.

3.60 connection identifier: An implementation specific means of defining the connection to one or more physical entities which constitute a single workstation.

3.61 control structure element: A structure element which references or defines the return from another structure to produce a structure network. The control structure elements are execute structure, push state and pop state, instance structure and return. Conditional control structure elements are defined that allow elements or structures to be skipped.

3.62 coordinate system: The metric used to specify graphical output and positional input. The coordinate systems are modelling coordinates, world coordinates, view reference coordinates, normalized projection coordinates, and device coordinates. All coordinate systems are cartesian.

3.63 curve: A class of output primitive which generates a set of curves. The output primitives in the class are polyline, polyline set with colour, and non-uniform B-spline curve.

3.64 data mapping: The conversion of application-specific data or colour to intrinsic colour.

3.65 data record: A compound data type, the content of which is defined by the context within which it is used. For example, the content of the data records used in the input device initialization functions may vary depending upon the particular prompt and echo type specified in the invocation of the initialization function.

3.66 deferral mode: The deferral mode for a workstation is part of the display update state and specifies when changes to posted structure networks and the workstation state list are to be reflected in the displayed image.

3.67 depth cue table: The table that indicates whether or not depth cueing should be performed and how.

3.68 depth cueing: An effect in which the colours of points on an output primitive are combined with a specified depth cue colour. The degree of combination is dependent on the depth (Z in normalized projection coordinates) of the points.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.69 descendant structure: A child structure or the descendant of a child structure.

3.70 description table: A data structure whose entries specify the capabilities of an implementation (PHIGS description table) or a type of workstation (generic workstation description table) or an instance of a type of workstation (workstation description table).

3.71 device coordinates (DC): A device dependent coordinate system. In PHIGS, DC units are metres on a device capable of producing a precisely scaled image, and appropriate workstation dependent units otherwise.

3.72 device driver: The device dependent part of a PHIGS implementation that supports a physical graphics device. The device driver generates device dependent output and handles device dependent interaction.

3.73 device space: The space defined by the addressable points of a display device.

3.74 diffuse reflection: An approximation of the light reflected equally in all directions from a facet.

3.75 diffuse reflection coefficient: The fraction of light from non-ambient light sources that is diffusely reflecting from a facet.

3.76 direct colour specification: A non-indexed method of specifying colour where the components of the colour i.e. coordinates in colour space, are specified together with the colour model in which those components are expressed.

3.77 direct interpretation pick: An input operation which allows picking of primitives inserted into the direct interpretation structure.

3.78 direct interpretation: Interpretation of structure elements which have not been stored in the centralized structure store.

3.79 direct interpretation structure: A conceptual workstation-independent destination to which all directly interpreted structure elements are sent. Structure elements inserted into this structure, when interpreted, affect each of the workstations to which the direct interpretation structure is posted.

3.80 directional light source: A light source that contributes to the reflectance calculation dependent on the orientation of the facet being illuminated but independent of the facet's position.

3.81 directly interpreted structure element: A structure element which is passed to the direct interpretation structure for direct interpretation at the posted workstations.

3.82 display device: A graphics device on which images can be represented. A display device is one possible component of a workstation.

3.83 display priority: The priority assigned to a structure network when it is posted. It is used to discriminate between output primitives in the same posting group when they are mapped to the same display space location.

3.84 display space:

(1) That portion of device space corresponding to the volume available for displaying images.

(2) The working space of an input device.

3.85 display surface: The physical area on a display device onto which PHIGS images may be placed.

3.86 display target: The target assigned to be the display surface.

3.87 display update state: That which determines how and when the display surface is modified to reflect changes in the centralized structure store and the workstation state list. An application selects the display update state to take into account the capabilities of a workstation and the requirements of the application program. The display update state consists of the two workstation dependent aspects of deferral mode and modification mode.

3.88 echo: The immediate notification to the operator of the current measure of a logical input device.

3.89 echo area; echo volume: An area or volume, defined in device coordinates, which may be used for the display of a prompt or echo.

3.90 echo type: A parameter of device initialization which selects the echo technique for a particular logical input device.

3.91 edge flag: An aspect of a surface primitive which enables or disables the display of edges.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.92 edge visibility flag: An indicator that is part of the specification of some output primitives such as fill area set with data, that controls whether an individual edge is visible.

3.93 edge: The boundaries of closed regions defined in the surface class of output primitives. It is also used to define the boundaries of a posting group.

3.94 edgetype: An aspect of a surface primitive which indicates the style of the edges.

3.95 edgewidth scale factor: An aspect of a surface primitive which indicates the relative width of the image of an edge. The edgewidth scale factor is applied to a workstation dependent nominal value.

3.96 edit mode: Determines whether a new structure element will replace the structure element at the element pointer or will be inserted into the open structure after the element pointer.

3.97 editing structure element: A structure element which can be used to assist the editing of structures.

3.98 element pointer: A pointer used during structure editing, the value of which identifies the position in the open structure at which element deletion and creation will occur.

3.99 element position: A number associated with a structure element which indicates the element's position within a structure.

3.100 element reference list: A list of references which define the hierarchy within one branch of a structure network. Each reference consists of a structure identifier and an element position within that structure If the list contains n pairs, then the first n-1 pairs identify structure reference elements. The jth such element references the structure named in the (j+1)st list element. The deepest element of the list may identify any type of structure element.

3.101 element type: The identifying classification of a structure element. The possible types are output primitive, attribute, control, editing, application data and generalized structure element.

3.102 elliptical arc close: An output primitive consisting of a single filled area bounded by an elliptical arc in either a pie shape or with a chord joining the ends of the arc.

3.103 empty interior style: One possible representation of the interior of a surface output primitive. If the edges are not displayed, the image of a fill area set with interior style empty is invisible. The image of a fill area output primitive with interior style empty is always invisible.

3.104 error state list: The data holding information about the most recent error condition.

3.105 escape: A function which provides access to implementation dependent or device dependent features not concerned with the generation of graphical output.

3.106 event mode: An operating mode for a logical input device in which asynchronous input is placed on the event queue as an event report when a trigger fires.

3.107 event queue: A time-ordered collection of event reports.

3.108 event report: An entry in the event queue which consists of a logical input value and identification of the logical input device responsible.

3.109 exclusion set: The portion of a filter which defines those name set members which are not eligible for a certain operation.

3.110 execute structure: To suspend the traversal of the current structure keeping its state intact and to traverse the structure specified. Traversal of the current structure continues once the execution is completed.

3.111 eye point: A point in world coordinates that transforms to infinite positive Z in normalized projection coordinates. This point is used in the reflectance calculation for determining viewing-position-dependent effects of lighting.

3.112 facet: A planar interior segment of a surface primitive. Each facet of an output primitive is defined by a subset of the primitive's set of vertices. The subset is dependent on the individual primitive type and in the case of parametric surfaces on the approximation of the surface. Facets have an orientation in NPC described as back-facing or front-facing.

3.113 facet culling: The process of removing front-facing or back-facing facets of surface primitives.

3.114 facet data: Intrinsic colour data or a normal vector specified with a surface primitive.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.115 facet normal: A normal vector associated with a facet of a surface primitive. Facet normals are used to determine the orientation of a facet and in some cases for determining the reflectance normal.

3.116 fill area: An output primitive consisting of a single polygon.

3.117 fill area set: An output primitive consisting of a set of coplanar polygons with or without edges.

3.118 fill area set with data: An output primitive consisting of a set of coplanar polygons. It is similar to the fill area set output primitive. The corresponding structure element may include other information such as colours or normals that are conditionally used to colour, light and shade the output primitive.

3.119 fill circle: An output primitive consisting of a filled circular area.

3.120 fill ellipse: An output primitive consisting of a filled elliptical area.

3.121 filter: The combination of the inclusion set and the exclusion set which identify output primitives eligible or ineligible for a certain operation. PHIGS supports filters for picking, highlighting, invisibility, and incremental spatial search.

3.122 font: A set of character representations all of which share certain visual characteristics.

3.123 front facing: A front-facing facet has a facet normal that, when transformed to NPC, has a positive or zero Z component.

3.124 front plane: A plane parallel to the view plane which is specified as an N coordinate value in the view reference coordinate system. Output primitives in front of the front plane lie outside the view volume.

3.125 general colour: A data type that allows both the direct and indirect specification of colour. General colour specifies a colour type together with a type-dependent colour value. The colour type can either indicate a colour model, in which case the colour values are coordinates in the colour space corresponding to that model, or it can indicate that the colour is being specified indirectly, in which case the single colour value is an index into the workstation-dependent colour table.

3.126 generalized drawing primitive (GDP): An output primitive which accesses implementation dependent and workstation dependent geometric capabilities. It may use the attributes and aspects specified for other output primitive classes.

3.127 generalized structure element (GSE): A structure element which is used to access implementation dependent, workstation dependent, or device dependent features during structure traversal. It is a structure element which accesses attribute or control functionality, but does not create an output primitive.

3.128 geometric attributes: A class of attribute bound to primitives when they are created which completes the geometric definitions of primitives.

3.129 geometry spline: The parametric curve or surface defining the geometry of a parametric output primitive.

3.130 global modelling transformation: A component of the composite modelling transformation. When traversal of a structure begins, it is set to the current composite modelling transformation of the parent structure, or if it is the posted structure, the default value in the PHIGS description table.

3.131 hatch interior style: One possible representation of the interior of a surface output primitive. The interior is filled with a pattern of parallel or crossing hatch lines, or both, selected from the workstation's hatch table.

3.132 hatch table: The table of the hatch values defined on a workstation.

3.133 hidden line/hidden surface removal (HLHSR): The removal of those parts of output primitives which are obscured by other output primitives.

3.134 highlighting: Emphasizing an output primitive by modifying its visual attributes in some workstation dependent manner.

3.135 highlighting filter: A filter consisting of two name sets, the highlighting inclusion set and the highlighting exclusion set, used to identify output primitives which are eligible for highlighting.

3.136 HLS: An abbreviation for the Hue, Lightness and Saturation colour model.

3.137 hollow interior style: One possible representation of the interior of a surface output primitive. The image is the boundary line only, including any boundaries created by clipping.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.138 HSV: An abbreviation for the Hue, Saturation and Value colour model.

3.139 identification attributes: A class of attribute used to name output primitives and determines visibility, pickability and highlighting of primitives, and is used to identify a primitive, or group of primitives, when a primitive is picked. Identification attributes apply to all output primitive classes.

3.140 image: The appearance of objects after rendering.

3.141 image resource: An external abstract data structure representing an array of colour definitions. The format of such data structures need not be known to the application program but can be referenced by PHIGS in an implementation-dependent manner. Examples of image resources include window system bitmaps, video signals, and scanned images.

3.142 implicit regeneration: The complete recreation of the contents of a display surface such that it is visually correct. This may occur when changes to the posted structure networks, posting group priority, display priority, or other entries in the workstation state list invalidate the displayed image. Such a regeneration is not explicitly requested by the application program.

3.143 inclusion set: The portion of a filter which defines those name set members eligible for a certain operation.

3.144 indirect colour specification: A method of specifying colour via an index into a workstation dependent colour table.

3.145 inheritance: The mechanism by which child structures obtain initial attribute settings from their ancestor structures.

3.146 input class: An abstraction of one or more physical input devices to logical input devices which delivers logical input values to the application program. There are eight classes of logical input device: locator, stroke, valuator, choice, pick, string, set and composite. The first six are atomic input classes. The classes set and composite are derived from these atomic input classes.

3.147 input mode: One of the three possible methods of obtaining data from a logical input device: REQUEST, SAMPLE, or EVENT.

3.148 input report: An instance of a data type returned by a logical input device resulting from the firing of a trigger. Two types of input reports are recognized: event report and request report.

3.149 inquiry function: A mechanism for communicating to the application program data contained in a state list or description table.

3.150 instance structure: To include the elements of the structure specified as part of the structure being traversed.

3.151 interior: The set of points which lie inside an area. PHIGS provides a rule for determining whether or not a given point is inside an area.

3.152 interior style: An aspect which indicates the style used to fill the interior of a facet.

3.153 interpretation: The second phase of structure traversal which occurs after stepping. Stepping is done by PHIGS for structures in the CSS and by the application for the direct interpretation structure. The interpretation phase is always performed by PHIGS.

3.154 intrinsic alpha: The alpha value obtained from the alpha source at the beginning of the rendering pipeline during alpha mapping. This value may be modified as it progresses through the rendering pipeline until the final value is used during alpha blending.

3.155 intrinsic colour data: Colour or application-specific data associated with output primitives and specified in the output primitive's structure element. Intrinsic colour data, when specified, is conditionally used to determine the intrinsic colour of an output primitive.

3.156 intrinsic colour: The colour or colours of an output primitive that are independent of lighting, depth cueing and colour mapping.

3.157 invisibility: The state of an output primitive being hidden from view even when lying within the display surface and not occluded by other output primitives.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.158 invisibility filter: A filter consisting of two name sets, invisibility inclusion set and the invisibility exclusion set, used to identify output primitives eligible for invisibility.

3.159 isoparametric curve: A curve on a parametric surface produced by evaluating the surface over the range of one of its independent variables while holding its other independent variable constant.

3.160 isotropic mapping: A transformation which preserves aspect ratio.

3.161 knot vector: A non-decreasing sequence of real numbers that is part of the definition of non-uniform B-splines. This vector consists of values of the independent variables and is used in computing the B-spline basis polynomials.

3.162 label: A structure element, consisting of an identifier, which can be used as a place-marker to facilitate structure editing.

3.163 language binding: The expression of a functional specification in the syntax of a particular programming language.

3.164 light source: A simulated source of light. Light sources in PHIGS can be positional, directional or spot.

3.165 light source direction: A unit vector that defines the orientation of oriented light sources.

3.166 light source state: A field in the traversal state list that selects which light sources in a workstation light source table are active.

3.167 light source table: The table of the light sources defined on a workstation.

3.168 lighting: The effect of light sources on an output primitive's appearance. This is achieved by a reflectance calculation.

3.169 linetype: An aspect which indicates the style of the image of a curve, such as solid, dashed or dotted.

3.170 linewidth scale factor: An aspect which indicates the relative width of the image of a curve. The linewidth scale factor is applied to a workstation dependent nominal value.

3.171 local input operation: An operation which is automatically applied during input generation by PHIGS. Such operations may modify workstation state list entries.

3.172 local modelling transformation: A component of the composite modelling transformation. When traversal of a structure begins, it is set to the identity transformation.

3.173 local viewing operation: A local input operation which modifies on or more values in a workstation view table entry.

3.174 locator device: A logical input device providing a position in world coordinates and an associated view index.

3.175 logical input device: Specific logical input devices may be of class locator, stroke, valuator, choice, pick, string, set and composite. They deliver logical input values to the application program.

3.176 logical input value: A measure value delivered by a logical input device.

3.177 marker: A glyph with a specified appearance which is used to identify a location on a display surface. The shape of a marker is not subject to transformation.

3.178 marker primitive class: A class of output primitive which generates a set of symbols. The only output primitive in the class is polymarker.

3.179 marker size scale factor: An aspect which indicates the relative size of the image of a marker. The marker size scale factor is applied to a workstation dependent nominal value.

3.180 marker type: An aspect which selects the type of glyph used for markers.

3.181 measure: A value which is determined by the values of one or more physical input devices and a mapping from these values to a form required by one of the logical input classes.

3.182 measure process: A process which comes into existence when a logical input device is enabled for interaction. The current state of a measure process is the measure.

3.183 message: A string sent by the application program to a workstation principally for communication with the operator.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.184 metafile: A mechanism for retaining and transporting graphical data and control information. A metafile contains a device independent description of one or more pictures. *(See ISO/IEC 8632)*

3.185 minimal simulation: The minimal required action which is necessary when a workstation does not have the capability to render an output primitive as specified by the application.

3.186 mipmap: A precomputed hierarchical format used to streamline approximate calculation of the average texture colour which corresponds to a given texture mapped pixel.

3.187 modal: A type of change mechanism which extends its effect until another change specification supersedes it. Attribute settings and control functions are modal for a given structure and its descendant structures.

3.188 modelling coordinates: The device independent three-dimensional Cartesian coordinate system in in which graphical objects are defined by the application program using the PHIGS output primitives.

3.189 modification mode: Part of the display update state. The modification mode governs what category of visual effects are to be immediately achieved, and how they are achieved, on a workstation.

3.190 name set: An output primitive attribute consisting of a set of classifications defining the eligibility of output primitives for highlighting, invisibility, picking and incremental spatial search.

3.191 nominal value: The workstation dependent base value for aspects of output primitives.

3.192 non-geometric attributes: A class of attribute bound to primitives when they are created which are used to determine the values of the attributes which will be applied to primitives at the workstation level.

3.193 non-retained picture: That portion of the picture resulting from the interpretation of structure elements inserted into the direct interpretation structure.

3.194 non-uniform B-spline curve: An output primitive consisting of a non-uniform B-spline curve. These output primitives are specified by a series of control points and parameter values that define a non-uniform B-spline curves in terms of the B-spline representation. Both rational and non-rational versions are allowed.

3.195 non-uniform B-spline surface with data: A parametric surface output primitive consisting of a non-uniform B-spline surface. The with-data primitive allows the specification of an intrinsic colour data distribution over the primitive. Both rational and non-rational versions are allowed.

3.196 non-uniform B-spline surface: A parametric surface output primitive consisting of a non-uniform B-spline surface. These output primitives are specified by a series of control points and parameter values that define non-uniform B-spline surfaces in terms of the B-spline representation. Both rational and non-rational versions are allowed.

3.197 normal vector: A unit length vector, typically indicating the orientation of a facet or object.

3.198 normalized projection coordinates (NPC): The device independent three-dimensional Cartesian coordinate system in which the composition of images is specified to the graphics system. The view clipping limits and the workstation window are specified in NPC space.

3.199 operator: Person manipulating physical input devices so as to change the measures of logical input devices and cause their triggers to fire.

3.200 output primitive: A fundamental graphical display entity representing a geometric shape. Output primitives belong to output primitive classes.

3.201 output primitive classes: A classification associated with output primitives. A set of logical attributes apply to all output primitive types in a class. The output primitive classes are marker, curve, text and surface. The output primitive, generalized drawing primitive, does not belong to any class. Some of the surface primitives are defined as with-data primitives.

3.202 output primitive structure element: A structure element which, on traversal, generates an output primitive with the relevant attributes bound from the traversal state list.

3.203 parallel transformation: The transformation of an object in which parallel lines in the object appear parallel in the resulting image without regard to relative distance or depth.

3.204 parameter range limits: Minimum and maximum parameter values, specified separately from any knot values, that limit the parameter range over which parametric curves are generated.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.205 parameter space: The coordinate system of the independent variable(s) of parametric curves and surfaces. The parameter space is one-dimensional for curves and two-dimensional for surfaces.

3.206 parametric output primitive: Output primitives defined as a mapping from a parameter space to modelling coordinates. Parametric output primitives are non-uniform B-spline curve, non-uniform B-spline surface and non-uniform B-spline surface with data.

3.207 parent structure: A structure which contains one or more structure references.

3.208 pattern interior style: One possible representation of the interior of a surface output primitive. The interior is filled in with a two dimensional pixel pattern selected from the workstation's pattern table.

3.209 pattern table: The table of the pattern values defined on a workstation.

3.210 perspective transformation: The transformation of an object in which parallel lines in the object which intersect the view plane appear to converge in the resulting image as a function of relative distance or depth.

3.211 PHIGS description table: The data which represents the static workstation independent parameters of a PHIGS system.

3.212 PHIGS state list: The data which represents the dynamic workstation independent parameters of a PHIGS system.

3.213 PHIGS traversal state list: The data which represents the dynamic workstation independent parameters of a PHIGS system during traversal.

When traversal begins, the PHIGS traversal state list values are set to the values defined in the PHIGS description table.

The beginning of traversal occurs implicitly whenever the PHIGS traversal state list is created as a result of posting or enabling the direct interpretation pick traversal process or occurs explicitly when the direct interpretation state is renewed.

3.214 physical input device: Hardware capable of generating input. A physical input device is either part of a display device or a separate part of the workstation.

3.215 pick correlation point: This is a device coordinate point which specifies a position on the display surface used to test proximity to output primitives being interpreted during direct interpretation picking.

3.216 pick device: A logical input device providing a pick path.

3.217 pick filter: A filter consisting of two name sets, the pick inclusion set and the pick exclusion set used to identify output primitives which are eligible for selection by a pick input operation.

3.218 pick identifier: A name associated with individual output primitives, and returned by a pick device as a component of the pick path.

3.219 pick path: The element reference list representing a path to a picked output primitive from its posted structure, consisting of the structure identifier, pick identifier, and element position for each level of hierarchy traversed.

3.220 pick type: A criterion used to identify when primitives are included in a direct interpretation pick measure.

3.221 picking: An operator action of identifying an output primitive using a pick device.

3.222 picture: The conceptual image produced as a result of a set of rendering operations.

The picture is comprised of the non-retained picture and the retained picture.

3.223 polyline: An output primitive consisting of a connected sequence of straight lines between specified points.

3.224 polyline set with colour: An output primitive consisting of an unconnected set of polylines. The corresponding structure element may include colour information that is conditionally used to shade the primitive.

3.225 polymarker: An output primitive consisting of a set of locations, each indicated by the same type of marker.

3.226 positional light source: A light source that contributes to the reflectance calculation dependent on the orientation and position of a facet being illuminated relative to the light source.

3.227 post: To identify a structure network for display on a particular workstation as part of a posting group.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.228 posting group: The set of structure networks whose visual effects and priority can be defined relative to other posting groups.

3.229 posting group background: The background associated with a posting group. A posting group background may have a style of transparent or opaque. When opaque, various methods are provided for defining its appearance.

3.230 posting group border: The border associated with the posting group background. If the posting group border indicator is on, the it is drawn using the edge style indicated by the posting group border index.

3.231 posting group priority: The priority assigned to a posting group when it is defined or changed. It is used to discriminate between posting groups when they are mapped to the same display space location.

3.232 posting group status: A status indicator that determines whether or not a posting group is traversed.

3.233 posting group volume: A volume defined in NPC coordinates which has Z dimensions from 0 to 1 and X, Y dimensions defined by the view clipping limits of the default view associated with the posting group. All output generated as a result of interpreting structures posted to the posting group will be clipped to the posting group volume.

3.234 profile: A subset of the PHIGS functionality. Basic, plus and full profiles are defined.

3.235 projection reference point (PRP): A point in view reference coordinate space which determines the direction of projectors when the projection type is PARALLEL, or from which all projectors emanate when the projection type is PERSPECTIVE.

3.236 projection type: The type of transformation to be used in view mapping, i.e., parallel transformation or perspective transformation.

3.237 projection viewport: A rectangular parallelepiped in normalized projection coordinate space. In the PHIGS viewing model, the view mapping transformation maps the contents of the view volume into the projection viewport.

3.238 projector: A conceptual line which passes through a point of an object and intersects the view plane.

3.239 prompt: Output to the operator indicating that a specific logical input device is available.

3.240 quadrilateral mesh with data: An output primitive in which an array of quadrilaterals is specified by a two-dimensional array of vertices.

3.241 reflectance calculation: The computation of the effect of light sources on the colour of a surface primitive's facets.

3.242 reflectance formulae: Formulae that model the light reflected by a surface primitive.

3.243 reflectance model: An aspect that selects the reflectance calculation and thereby specifies which lighting effects are to be displayed.

3.244 reflectance normal: A vector used in the reflectance calculation and indicating the orientation of a primitive at a point on the primitive. The vector is conceptually perpendicular to the surface of an object being represented by a surface primitive. It is derived from the vertex normals of the primitive, if specified, or the facet normal.

3.245 reflectance properties: An aspect of surface primitives that indicates how a primitive reflects light.

3.246 render: The process of converting an output primitive to graphical output for incorporation within a picture.

3.247 rendering attributes: A class of attributes which controls the rendering pipeline.

3.248 rendering colour model: The colour model used for performing colour interpolation during shading and depth cueing.

3.249 rendering pipeline: A sequence of operations that performs alpha mapping, data mapping, lighting, texture mapping, shading, depth cueing, alpha blending and colour mapping of output primitives. Each of these operations is considered a stage in the rendering pipeline.

3.250 rendering target: The target assigned to be the recipient of any rendering activity resulting from structure traversal.

3.251 request mode: An operating mode for a logical input device in which PHIGS waits until the operator activates the appropriate trigger where upon the measure is returned.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.252 request report: The input report generated resulting from the firing of an input report generation trigger of a logical input device operating in request mode.

3.253 retained picture: That portion of the picture resulting from traversal of posted CSS structure networks.

3.254 RGB: An abbreviation for the Red, Green and Blue colour model.

3.255 rigid-body transformation: A modelling transformation composed of at most translation, rotation, and scaling transformations, where translation transformations move every point of an object an equal distance in the same direction, the rotation transformations maintain relative angles, and the scaling transformations apply equal scaling in all coordinate dimensions.

3.256 sample mode: An operating mode for a logical input device in which the measure of a logical input device may be obtained immediately without waiting for a trigger.

3.257 set device: A logical input device whose measure consists of one or more instances of input values delivered by a single associated measure process.

3.258 set of fill area sets with data: An output primitive in which a number of possibly non-coplanar fill area sets are defined by indices into a single list of vertex data. The fill area sets are not required to form a closed or connected surface.

3.259 shading: The interpolation stage of the rendering pipeline.

3.260 solid interior style: One possible representation of the interior of a surface output primitive. The interior is filled with a uniform colour.

3.261 specular colour: A reflectance property indicating the effect of a primitive on the colour of specular reflections from that primitive.

3.262 specular exponent: A non-negative number indicating the shininess of a facet. The higher the specular exponent, the shinier the facet. A specular exponent of 0 indicates a minimum relative degree of shininess.

3.263 specular reflection: An approximation of the unequal reflection of light in different directions from a surface primitive, dependent on the relationship of the viewer to the primitive and the light source.

3.264 specular reflection coefficient: The fraction of non-ambient light contributing to specular reflection.

3.265 spot light source: A light source that contributes to the reflectance calculation dependent on the orientation and relative position of the facet being illuminated. Light from such a source is restricted to a semi-infinite cone of influence and its intensity may decrease as it deviates from the centreline of this cone. (See concentration exponent and spread angle.)

3.266 spread angle: An angle that determines the shape of the cone of influence of a spot light source. Spread angle is the angle between the centre of the cone of influence and the limit of the cone of influence measured at the position of the spot light source.

3.267 state list: A data structure whose entries specify the current values of variables relating to PHIGS as a whole (PHIGS state list) or to a specific workstation (workstation state list).

3.268 stepping: The first phase of structure traversal which steps through the structure elements. The second phase is interpretation. In traversing structures in the CSS, both the stepping and interpretation phase are performed by PHIGS. In traversing the direct interpretation structure, only the interpretation phase is performed by PHIGS; the application performs the stepping phase or defines the directly interpreted structure elements when needed from available application managed graphical data.

3.269 string: An ordered collection of character codes as described in ISO 2022.

3.270 string device: A logical input device providing a string.

3.271 stroke device: A logical input device providing a sequence of points in world coordinates, and an associated view index.

3.272 structure: A linear sequence of structure elements. Two categories of structures are supported, structures stored in the centralized structure store and direct interpretation structures. Structures may be related to each other hierarchically to form a structure network. Traversal of structures generates output primitives for display. The post operation to

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

a workstation causes the traversal process to be started. Structures may be changed by structure editing. Structure archiving is provided for long term storage and retrieval.

3.273 structure archiving: The process of storing structures and structure networks in an archive file for subsequent structure retrieval.

3.274 structure editing: The modification of a structure, including its initial creation.

3.275 structure element: The fundamental unit of data, a sequence of which is collected to form a structure.

3.276 structure identifier: A unique application specified name used to reference a structure.

3.277 structure network: A collection of structures arranged with the topology of a directed acyclic graph, where structures are thought of as nodes and structure references are thought of as connections from the parent structure to the child structure.

3.278 structure reference: The invocation of one structure from within another by an execute or instance structure element or the invocation of a structure by posting.

3.279 structure retrieval: The process of copying structures from an archive file to the centralized structure store.

3.280 structure state list: The data which represents information about each structure in the centralized structure store.

3.281 surface: A class of output primitive: fill area, fill area set, fill area set with data, set of fill area sets with data, triangle set with data, triangle strip with data, quadrilateral mesh with data, fill circle, circular arc close, fill ellipse, elliptical arc close, non-uniform B-spline surface, non-uniform B-spline surface with data and cell array primitives belong to this class.

3.282 target: One of possibly several replications of facilities which may be used as an abstract display surface.

3.283 texel: A single cell in a raster definition of a texture, or the colour and/or alpha value contained in such a cell. The term is derived by analogy to 'pixel'.

3.284 text: An output primitive consisting of a character string drawn at a specified position on a specified text plane.

3.285 text alignment: An aspect of text which specifies the mode of justification. This aspect has horizontal and vertical components.

3.286 text direction vectors: Two vectors, specified as part of the text structure element, which, together with the text position, define the text plane.

3.287 text extent: A parallelogram defining the minimum area which completely encloses the character bodies in a character string.

3.288 text font: An aspect of text which indicates certain visual properties of a character, such as typeface.

3.289 text local coordinate system: The local coordinate system for text output primitives within which all geometric text attributes are measured.

3.290 text path: An aspect of text which controls the relative positioning of successive characters in a string.

3.291 text plane: The plane in which a text output primitive appears.

3.292 text position: A point, specified as part of the text structure element, which, together with the text direction vectors, determine the text plane. It also determines, together with text alignment, the location of the character string.

3.293 text precision: An aspect of text which selects the fidelity with which the drawing of a text output primitive matches the appearance selected by an application program.

3.294 text primitive class: A class of output primitive which generates a sequence of characters. The output primitives in the class are text and annotation text.

3.295 texture: A design applied to a surface primitive interior. Such designs are commonly derived procedurally or from 2D colour images.

3.296 texture composition: The method by which a texture modifies the colour associated with an interior in the rendering pipeline.

3.297 texture coordinate source: A value which specifies from whence the texture coordinate information is derived.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.298 texture coordinates: Data values derived from the definition of the primitive or computed during the processing of a primitive which are used to map positions on the primitive to positions on the texture. Three stages are used to obtain texture coordinates: a) the texture coordinate source (vertex data values, world coordinate vertex position, etc.) is used to choose initial source coordinates; b) reflection mapping or other non-linear projections of the source coordinates are applied as appropriate; and c) orientation/positioning converts the source coordinates into nominal texture sampling coordinates. The context makes clear which of these three stages are applicable.

3.299 texture interior style: One possible representation of the interior of a surface output primitive. Textures are applied to the interior.

3.300 texture magnification: The texture sampling mechanism used when a single texel corresponds to many pixels; that is, when the texture image is magnified in the process of being applied to the primitive. Different methods may be used to derive the sample value for that pixel.

3.301 texture mapping: The overall process by which a texture is applied to an interior.

3.302 texture minification: The texture sampling mechanism used when many texels correspond to a single pixel; that is, when the texture image is shrunk in the process of being applied to the primitive. Different methods may be used to derive the sample value for that pixel.

3.303 texture parameterization: The collected information which is used to specify how a texture is oriented to the facets on to which it is to be mapped.

3.304 texture sampling: The process by which the texture coordinates are used to derive a texture value.

3.305 texture value: The colour value obtained by texture sampling and used in the rendering pipeline in texture composition.

3.306 transformation pipeline: The ordered sequence of transformations which convert modelling coordinates to device coordinates.

3.307 transparency: An attribute which specifies the proportion of the background image which shows through a primitive.

3.308 traversal: The process of stepping through a network of structure elements and interpreting each structure element encountered.

Interpreted structure elements may generate output primitives, change the contents of a state list, or have no effect. For operations on structures in the centralized structure store, the stepping and interpretation phases are both performed by PHIGS.

For operations on the direct interpretation structure, the stepping phase is performed by the application while the interpretation phase is performed by PHIGS.

3.309 traversal resource: Any facility which may be used to assist in producing the information contained in a target or pick measure.

3.310 traversal state list: See PHIGS traversal state list.

3.311 triangle set with data: An output primitive in which a number of possibly unrelated triangular facets are defined by indices into a single list of vertex data.

3.312 triangle strip with data: An output primitive comprised of a group of adjacent triangles formed by connecting a list of vertices such that the second and third vertices of each triangle are used as the first and second vertices of the next triangle.

3.313 trigger: A physical input device or set of devices that an operator can use to indicate significant moments in time. Triggers may be used to generate input reports, indicate break actions, select portions of a measure, or edit a measure during its formulation.

3.314 trigger process: A process which notifies an associated logical input device when a trigger is fired. The trigger process only exists when a device is in EVENT mode, or in REQUEST mode with a request pending.

3.315 trimming curve: A parametric curve in the parameter space of the surface to which it applies. Trimming curves are combined to form trimming loops which limit the parameter range over which a parametric surface is evaluated.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Definitions

3.316 trimming loop: A sequence of connected and similarly oriented trimming curves that form a closed path. Trimming loops are used to limit the parameter range over which a parametric surface is evaluated.

3.317 unpost: To remove the identification of a structure within a posting group for display on a particular workstation.

3.318 valuator device: A logical input device providing a real number.

3.319 vertex colour: A general colour associated with each vertex of some output primitives. This colour is conditionally used within the rendering pipeline to colour and shade the primitive.

3.320 vertex data: Geometric, intrinsic colour data, or vertex normal data specified at vertices of certain output primitives.

3.321 vertex normal: A normal vector optionally supplied with the vertex data of some surface primitives.

3.322 view construction parameters: Parameters from which a view representation can be formed.

3.323 view index: An output primitive attribute that selects the view representation applied to output primitives and geometric attributes during structure traversal.

3.324 view mapping matrix: A 4×4 matrix which specifies the transformation of points from view reference coordinates to normalized projection coordinates.

3.325 view orientation matrix: A 4×4 matrix which specifies the conversion of points from world coordinates to view reference coordinates.

3.326 view plane: A plane parallel to the U-V plane in view reference coordinate system (VRC). It is specified as an N coordinate value in VRC.

3.327 view plane normal: A vector in world coordinates, relative to the view reference point, which defines the N coordinate axis of the view reference coordinate system.

3.328 view reference coordinate (VRC) system: A device independent three-dimensional Cartesian coordinate system in which the parameters for the view mapping transformation are specified. The axes of the view reference coordinate system are called the U, V and N axes. The position and orientation of this coordinate system, relative to world coordinates, are defined by the view reference point, the view plane normal and the view up vector. These define the view orientation matrix.

3.329 view reference point: A point in world coordinates which defines the view reference coordinate system origin.

3.330 view representation: A single entry in the view table. It consists of a composite data type that defines the view orientation matrix, the view mapping matrix, the view clipping limits and a set of clipping indicators. The view clipping limits are also used to define the posting group volume.

3.331 view table: The table of view representations and view construction parameters defined on a workstation. Its entries are referenced by the view index attribute.

3.332 view transformation input priority: That which determines the order in which view table entries are tested when selecting the inverse viewing transformation to be applied to locator and stroke input.

3.333 view up vector: A vector defined in world coordinate space. The view reference coordinate system V coordinate axis is defined as an orthogonal projection of the view up vector onto the plane through the view reference point and perpendicular to the view plane normal. Vectors which are parallel to the view up vector in world coordinate space will appear vertical in the final image.

3.334 view update method: A technique which determines how a view representation is updated from the view construction parameters.

3.335 view volume: The view volume is determined by the view window defined in the view plane, the front plane, the back plane, and projectors through the corners of the view window. If the projection type is PERSPECTIVE, the projectors all pass through the projection reference point, the view volume is therefore a frustum. If the projection type is PARALLEL, the projectors are parallel to the vector joining the projection reference point to the centre of the view window, the view volume in this case is a parallelepiped. In some application programs, the view volume reduces to the view window.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

3.336 view window: A rectangle in the view plane. Projectors passing through the corners of the view window define the left, right, bottom and top surfaces of the view volume.

3.337 viewing attributes: A class of attributes which selects the viewing and HLHSR parameters for the workstation dependent sections of the pipeline.

3.338 visually correct: The display on a workstation is visually correct when it corresponds exactly to the state of the centralized structure store and the workstation state list.
The existence of a non-retained picture on a display does not effect whether the display is visually correct.

3.339 with-data primitives: Any of the output primitives: fill area set with data, set of fill area sets with data, triangle set with data, triangle strip with data, quadrilateral mesh with data and non-uniform B-spline surface with data. In addition, some generalized drawing primitives may have this classification.

3.340 workstation: PHIGS is based on the concept of abstract graphical resources, which provide the logical interface through which the application program controls physical devices. A workstation consists of zero or one display space and zero or more logical input devices.

3.341 workstation category: A property of a workstation which indicates whether it can perform graphics input only, graphics output only, or both.

3.342 workstation description table: The data which represents capabilities for a particular workstation type.

3.343 workstation identifier: A unique application specified name used to identify a particular workstation.

3.344 workstation state list: The data which represents dynamic information for a particular workstation.

3.345 workstation table: A table on a workstation that controls output in a workstation dependent way. Each workstation has a set of bundle tables and additional tables to support the rendering of output primitives.

3.346 workstation transformation: A transformation which maps the contents of the workstation window into the workstation viewport, preserving aspect ratio in X and Y, but not necessarily in Z.

3.347 workstation type: A type or class of actual workstation, sharing common characteristics and a single workstation description table.

3.348 workstation viewport: A rectangular parallelepiped in device coordinate space. All graphics output appears within this volume.

3.349 workstation window: A rectangular parallelepiped in normalized projection coordinate space. Its contents are mapped by the workstation transformation into the workstation viewport.

3.350 world coordinates (WC): The device independent three-dimensional Cartesian coordinate system used by the application program to organize modelled objects for display. The effect of applying the composite modelling transformation to modelling coordinates is to produce world coordinates.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

4 PHIGS functional overview

4.1 About this part of ISO/IEC 9592

4.1.1 Specification and conformance

The set of functions known as PHIGS shall be as described in clauses 4, 5 and 6. A conforming implementation of PHIGS shall correctly implement all the functions described. Minimum support criteria are specified in tables 33 to 35 (4.16). A conforming PHIGS implementation shall also provide the information described in annex D. In a conforming implementation all graphical capabilities that can be addressed by PHIGS shall be used only via PHIGS.

4.1.2 Registration[1)]

For certain parameters of the functions, PHIGS defines value ranges as being reserved for registration or future standardization. The meanings of these values will be defined in the register. Subclause 6.1.4 defines the integer selection types where this applies. These procedures do not apply to values and value ranges defined as being workstation or implementation dependent; these values and ranges are not standardized.

4.1.3 Notational conventions

This part of ISO/IEC 9592 uses the following typographical conventions.

a) The names of primitive attributes appear in upper case.

b) Geometric aspects of primitives appear in upper case. Non-geometric aspects appear in lower case except when the corresponding individually specified attributes are being used, in which case they appear in upper case.

c) The names of entries in the PHIGS data structures appear in lower case surrounded by single quotation marks.

d) The names of structure elements appear in lower case surrounded by double quotation marks.

e) The names of PHIGS functions appear in upper case.

f) The values of an enumeration data type appear in upper case.

g) The names of data types appear in upper case.

4.2 Overview and concepts

4.2.1 Overview

The Programmer's Hierarchical Interactive Graphics System (PHIGS) provides a functional interface between an application program and a configuration of graphical input and output devices. The functional interface contains basic functions for dynamic interactive hierarchical graphics on a wide variety of graphics equipment. It provides facilities for the specification of curved and faceted surfaces. Lighting and other effects such as the application of textures, simulation of surface shape and reflection of the visual environment can be specified to allow realistic presentation of the geometry. There is direct support for the visualization of data that can be associated with the geometry.

The interface is at such a level of abstraction that hardware peculiarities are shielded from the application program. As a result, a simplified interface presenting uniform output primitives and uniform input devices is obtained. Some PHIGS facilities are described as being implementation dependent. This means that an implementation may determine

1) For the purpose of this part of ISO/IEC 9592 and according to the rules for the designation and operation of registration authorities in the ISO/IEC Directives, the ISO and IEC Councils have designated the following as the registration authority: National Imagery and Mapping Agency (NIMA) (Joint Interoperability Test Command), The Registration Authority for Graphical Items, Building 57305, Fort Huachuca, Arizona 85613-7020, USA.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

the effect (rather than having the effect mandated). However, such effects shall be the same on all devices in a single implementation.

Annex D describes allowable differences between PHIGS implementations.

PHIGS defines only a language independent nucleus of a graphics system. For integration into a language, PHIGS is embedded in a language dependent layer containing the language conventions, for example, parameter and name assignment. The PHIGS language bindings are defined in ISO/IEC 9593.

The layer model represented in figure 1 illustrates the role of PHIGS in a graphics system. Each layer may call the functions of the adjoining lower layers. In general the application program uses the application oriented layer, the language dependent layer, other application dependent layers, and operating system resources.

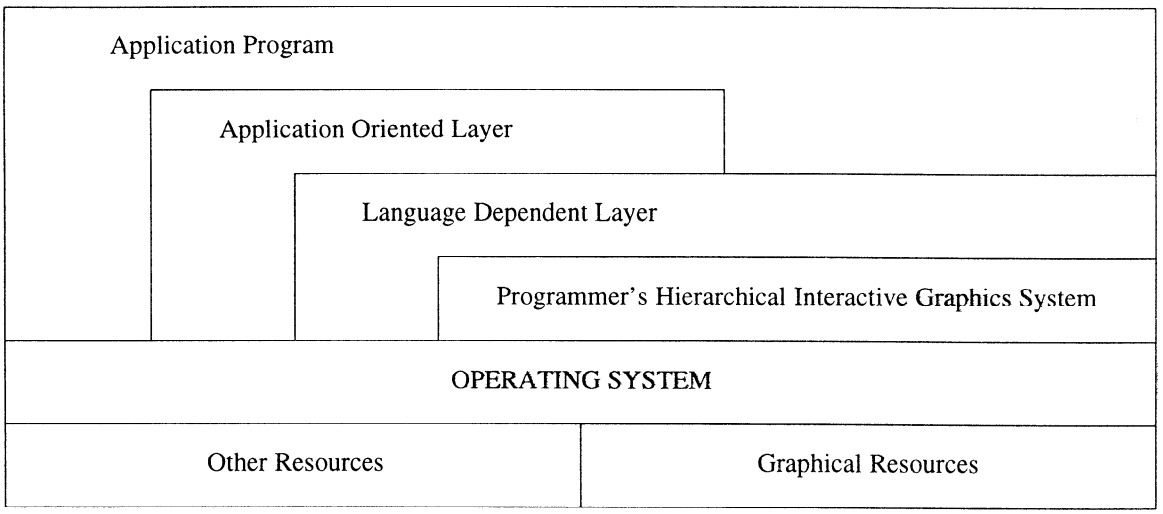

Figure 1 - Layer model of PHIGS

4.2.2 Concepts

PHIGS supports the storage and manipulation of data. The fundamental entity of data is a structure element and these are grouped together into units called structures. Graphical output is generated by the traversal of a structure which involves stepping through the individual structure elements and interpreting them. Structures are stored by the application or by PHIGS. If PHIGS is responsible for storing a particular structure, the structure is stored in the PHIGS centralized structure store (CSS), and PHIGS is responsible for both stepping through and interpreting the individual structure elements. Otherwise, the application is responsible for storing the structure and stepping through the individual structure elements presenting them to PHIGS for interpretation. In this case, it is called a direct interpretation structure. Structure elements inserted into the direct interpretation structure are not retained after interpretation. These concepts are illustrated in figures 2 and 3. Structures are organized as acyclic directed graphs called structure networks. Facilities are provided to link structures together in the CSS and to allow the application to perform such linkage.

Structure elements can be either graphical data or application specific data. Graphical data consists of information for structure management and control as well as for output definition. Functions are provided to edit structures by the insertion and removal of structure elements. Functions are also provided to support searching and inquiry of the content and topology of structure networks and the CSS in general; these functions are not available for the direct interpretation structure.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

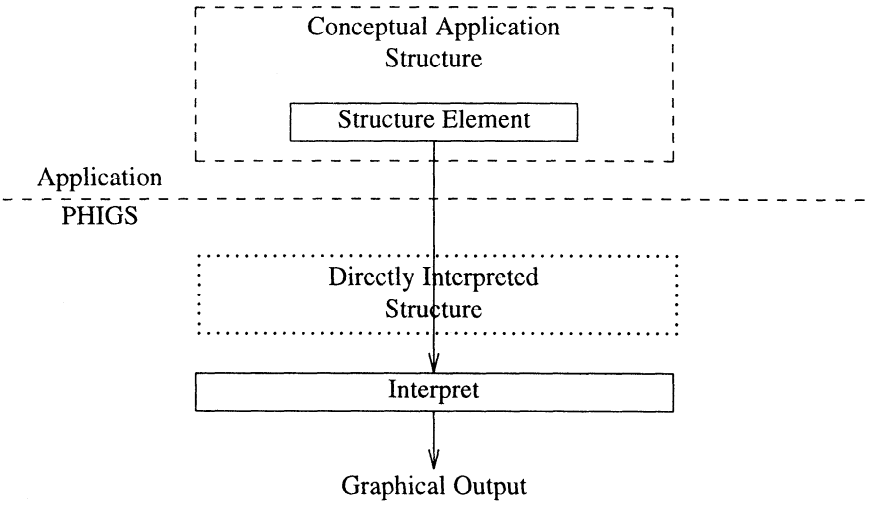

Figure 2 - Direct interpretation structure interpretation

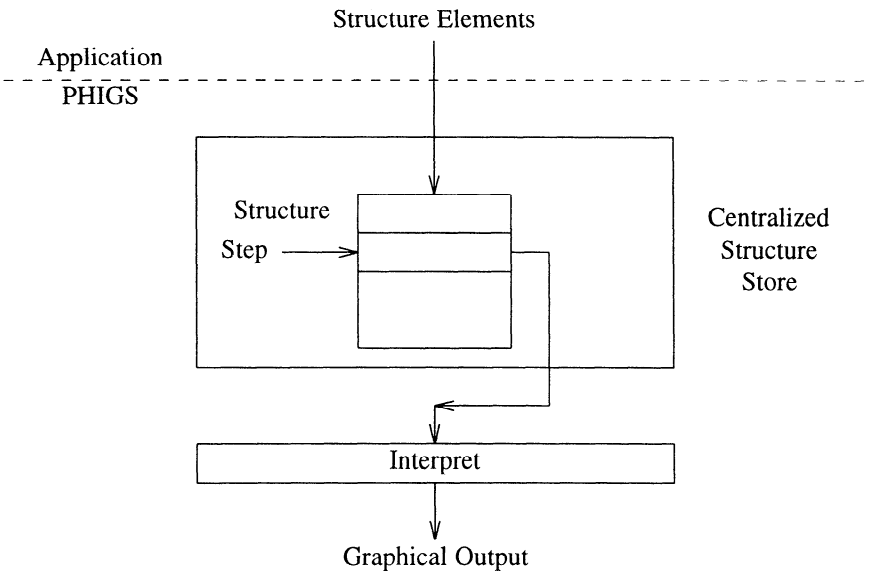

Figure 3 - CSS structure interpretation

The graphical output that is generated by PHIGS is built up from two groups of basic elements called output primitives and attributes.

PHIGS provides a basic set of output primitives for marking points, drawing curves and displaying surfaces. PHIGS defines a set of output primitives whose definitions contain, in addition to the geometry of the primitive, information that further specifies certain characteristics of the primitive and influences its appearance on the display surface. These output primitives are called with-data primitives. The data defined for these with-data primitives is used to colour and shade the primitive, and in the case of surface primitives, to indicate the primitive's apparent orientation and control the visibility of its edges. Depending on the output primitive, the information that can be specified in addition to geometry is intrinsic colour data that indicates a colour distribution across the primitive, normal vectors that indicate the primitive's apparent orientation, and edge visibility flags that control the visibility of a surface primitive's edges. This

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

data is optionally specified and conditionally used when rendering the primitive. Some of the with-data primitives offer data compaction and other advantages.

PHIGS defines non-uniform B-spline curve and surface output primitives. These output primitives are specified by a series of control points and parameter values that define non-uniform B-spline curves and surfaces in terms of the B-spline representation. This functionality allows the exact specification of commonly required shapes including conics, conic sections such as circles, ellipses, parabolae, spheres, and spheroids without resorting to special cases. The curves and surfaces are represented by non-rational and rational splines using the mathematically robust B-spline representation. Variants of these primitives are defined that allow specification of an intrinsic colour data distribution over the primitive.

Output primitive attributes control the properties of the primitives on a device, such as their aspects; for example line-type, colour, and character height. A central concept is the workstation which is introduced both for structuring PHIGS and for realizing device independence. The concept of multiple workstations allows simultaneous output to and input from various display systems. The concept of target control allows control over the display capabilities of a workstation. Non-geometric aspects such as colour can be controlled for each workstation separately to make best use of its capabilities. Such concepts are described as being workstation dependent. This means that an implementation may specify the characteristics of these concepts on a workstation by workstation basis. All workstation dependent concepts are explicitly defined.

Output primitives and their attributes are described in 4.4 and 4.5.

A colour is specified in PHIGS as a general colour, which consists of a colour type and a type-dependent colour value. The colour type can indicate any one of a number of colour models, in which case the corresponding colour value is interpreted as a value in that colour space. Alternatively, the colour type can indicate an indirect colour, in which case the colour value is an index into the workstation colour table. Colour types are defined which allow alpha data to be incorporated with colour components.

The graphical information that is input from a device as a result of operator actions is mapped by PHIGS onto eight classes of input, each represented by a data type referred to as a logical input value. An instance of such a device representation is called a logical input device. The effect of input actions on the display surface, such as prompts echoes, is controlled by PHIGS for each logical input device individually. Input is described in 4.9. Workstation state list variables may be automatically modified based on input values under control of the application program. How input operations can control viewing and lighting is described in 4.9.10.

The two abstract concepts (abstract output and abstract input) are the building blocks of an abstract workstation. A PHIGS workstation represents a unit consisting of zero or one display surfaces and zero or more input devices, such as keyboard, tablet and lightpen. The workstation presents these devices to the application program as a configuration of abstract devices thereby shielding the hardware peculiarities. Workstations are described in 4.7.

Graphical output on a workstation is produced by traversing a structure identified for display on that workstation and interpreting the structure elements. Application specific data is generally ignored during structure traversal. Structure networks identified for display on a workstation may be grouped as a posting group. Control is provided which prioritizes the display of posting groups. Access to image information is specified in an image resource table. Uses for this imaging information include posting group backgrounds and pattern representations.

The geometric information (coordinates) stored within the graphical structure elements are processed through workstation independent and workstation dependent stages each with a defined coordinate system. The workstation independent stage of structure traversal performs a mapping from modelling coordinates to a world coordinate system during which graphical structure elements can produce output primitives.

The workstation dependent stage then performs a transformation on the geometric information contained in output primitives, attributes and logical input values (locator and stroke measures). These transformations perform mappings between four coordinate systems, namely:

a) World Coordinates (WC) used to defined a uniform coordinate system for all abstract workstations;

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

b) View Reference Coordinates (VRC), used to define a view;

c) Normalized Projection Coordinates (NPC) used to facilitate assemblies of different views;

d) Device Coordinates (DC), one coordinate system per workstation representing its display space.

Output primitives and attributes are mapped from WC to VRC by the view orientation transformation, from VRC to NPC by the view mapping transformation and finally from NPC to DC by the workstation transformation.

Regions may be identified in the world and normalized projection coordinate systems which, during the mappings described above, may result in the clipping of parts of graphical output which lie outside one or more of the regions. Modelling clip regions may be defined in modelling coordinates. These regions are then transformed to world coordinates and combined to form a composite modelling clip region which may then be applied to the graphical output. Another clip region is defined as part of a view definition to limit the generation of graphical output. A third clip region associated with the workstation transformation identifies a subset of the image for display on the workstation.

PHIGS has the concept of a rendering pipeline. Conceptually, all output primitives pass through this pipeline as they are displayed. Operations are performed in the pipeline that affect the colour and appearance of each facet of a primitive. These operations include, for most primitives, the mapping of transparency data to a primitive (alpha mapping), the mapping of application-specific data to colour (data mapping), the calculation of reflectance effects (lighting), the application of textures (texture mapping), data interpolation (shading), the blending of the resultant colour of a primitive with the colour already in the displayed picture (alpha blending), and depth cueing. Output primitive attributes and any data attached to the primitives are conditionally used in these operations. The last stage of the rendering pipeline is a colour mapping that maps the resultant colours from the earlier stages into workstation dependent colours. The application may have control over which of these stages are active and what methods are used within each stage. The rendering pipeline is described in 4.6.

Data mapping in PHIGS provides a mechanism to convert application-specific data to colour within the rendering pipeline. Data such as temperature, pressure or other scalar or multi-dimensional information can be specified for most surface primitives. Selected portions of this data may then be used during traversal to determine the intrinsic colour of the primitive, which is the colour or set of colours associated with a primitive before the effects of lighting are applied. The method to be used to convert the data to intrinsic colour is under application control. More data can be specified than are used during any single rendering of a primitive.

Explicitly specified and implicitly specified aspects control the operations of the rendering pipeline. Explicitly specified aspects are facet normals, used to determine facet orientation, and vertex normals used to determine reflectance normals in reflectance calculations. When not supplied, facet normals are implicitly specified from the geometry of the facet.

PHIGS defines the light sources and suggests their interaction with output primitives. Several types of light sources are defined: ambient, directional, positional and spot. Workstation light sources are associated with workstations and are defined as entries in the light source table in the workstation state list. (Light sources with their specification included directly in the CSS are not defined by this part of ISO/IEC 9592, but their inclusion as GSEs or GDPs is not prohibited.) Active light sources are selected from this table by the 'current light source state' in the traversal state list.

Each facet of a surface primitive has an orientation, either front facing or back facing as determined in normalized projection coordinates from the facet normal. PHIGS defines separate attributes for front and back facing facets of surface primitives and provides controls for determining when they should be applied. PHIGS also defines an attribute to control whether individual facets are removed prior to viewing based on their orientation to the viewer.

The attributes which control the appearance of parts of the picture (output primitives, prompt and echo types) on the display space are organized in a uniform manner. Two groups of attributes apply to the appearance of each output primitive: primitive attributes (that are workstation independent) and workstation attributes. Primitive attributes are specified modally and are bound to a primitive when it is created by structure element interpretation. The primitive attributes include all geometric aspects of primitives, such as character height for text and pattern size for surface primitives. In addition, the non-geometric aspects of primitives are controlled by the primitive attributes in one of two ways. Either an attribute is used to specify a group of the non-geometric aspects of the primitive by an index (or set of indices in the case of the surface primitives) which points to a workstation dependent representation (set of values) or one attribute is used to specify each of the non-geometric aspects of the primitive in a workstation independent way. The former is referred to as bundled specification and the latter is referred to as individual specification. In addition, when a

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

primitive is created by structure element interpretation, an index is bound to it which selects the view orientation and view mapping transformations which are workstation dependent.

Each primitive has associated with it a NAME SET attribute defined by structure element interpretation. This NAME SET attribute defines the eligibility of the primitive for highlighting, invisibility, picking and incremental spatial search. A filter mechanism is incorporated in PHIGS with an inclusion and exclusion set mask at each workstation to control specific rendering of the primitives.

During traversal, the current value of the NAME SET supports conditional traversal by applying a filter to the NAME SET value that determines whether traversal of a specific structure will take place.

The attribute selection is controlled by the nature of the structure network. The effect of an attribute is defined within the structure in which it occurs and in subsequent structures below it in the hierarchy. Output renderings of the same structure may therefore be different due to the inheritance of attributes.

Workstation attributes include representations for those non-geometric aspects which are workstation dependent, and are selected via bundle indices. For example, the representations (or bundles) for polyline each contain values of linetype, linewidth scale factor and colour. Workstation attributes also specify the colour and pattern tables and control the updating of the display. Workstation attributes can be reset dynamically.

Hidden lines or hidden surfaces may be removed using a workstation dependent method (see 4.5.1.4).

The attributes which control the operation of logical input devices can be specified either upon initialization or as part of input device setting, depending upon the attributes. Through initialization, an initial value, a prompt and echo technique, and an area on the device for echoing can be specified. A data record may further provide device specific attributes. Through input device setting, the operating mode may be selected and the echo may be switched on or off. The operating modes of logical input devices specify who (operator or application program) has the initiative:

e) SAMPLE input is acquired directly by the application program;

f) REQUEST input is produced by the operator in direct response to the application program;

g) EVENT input is generated asynchronously by the operator and is collected in a queue for the application program. An acknowledgement of the acceptance or rejection of logical input values produced in event or request modes is supported.

PHIGS supports the definition of set devices and composite devices which can be made to model the conceptual input devices required by the application.

PHIGS provides an interface to enable the information contained in the centralized structure store to be archived or retrieved. These PHIGS archive files provide for the storage and interchange of graphics models expressed as PHIGS structure networks.

PHIGS also provides an interface to a system for filing graphical image information for the purpose of external long term storage and exchange of graphical image information. The interface consists of a metafile output workstation, which writes to a graphics metafile, and a metafile input workstation, which reads from the metafile. In addition to the normal functions for output to workstations, a metafile workstation may accept items containing non-graphical information. Input from a metafile is controlled by read and interpret functions.

4.2.3 PHIGS profiles

PHIGS functionality is organized into three profiles supporting subsets of the full functionality:

a) The *basic profile* defines the basic set of functionality needed for PHIGS 3D operations. This basic set of functionality includes support for structures, workstations, atomic logical input devices and the basic set of output primitives and attributes.

b) The *plus profile* defines the set of functionality which adds support for lighting and shading. This profile includes all functionality in the basic profile while incorporating additional attributes and primitives to support the concept of a rendering pipeline with lighting and shading.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

c) The *full profile* defines the complete set of PHIGS functionality. In addition to the functionality incorporated in the plus profile, this profile adds functionality to define linetypes, marker types, and logical input devices, functionality to support conditional traversal, and functionality to support direct interpretation of structure elements.

While not separate profiles, 2D versions of each of these profiles can be identified by removing the 3D versions of the PHIGS functions. A PHIGS implementation conforms to a PHIGS profile if it supports at least the minimum required capabilities for that profile including all functions associated with that profile. Supporting additional functionality does not invalidate such conformance. An implementation which supports some but not all of the required functionality of a profile may not claim conformance with that profile. The relationship of PHIGS profiles to International Standardized Profiles (ISPs) is described in Annex N.

4.3 Structures and traversal

4.3.1 Structure elements and structures

PHIGS supports the storage and manipulation of data. Such data is organized into units called structures, which may be related to each other hierarchically to form structure networks. Two categories of structures are supported in PHIGS, structures in a centralized structure store (CSS) managed by PHIGS and a direct interpretation structure managed by the application. A structure is identified by a unique name, the structure identifier.

A structure consists of a sequence of structure elements, where a structure element is the fundamental entity of data. Structure elements are used to represent application specified graphics data for output primitives, attribute selections, view selections, modelling transformations and clipping, name sets, invocations of other structures, and to represent application data. In addition, generalized structure elements can be used to access additional, implementation dependent features. Label structure elements may be used to identify locations in structures to assist in editing. Structure elements cannot exist outside of structures.

The following structure elements are defined in PHIGS.

Table 1-Output primitive structure elements

annotation text relative	elliptical arc close 3	non-uniform B-spline surface 3 with data
annotation text relative 3	fill area	polyline
cell array	fill area set	polyline 3
cell array 3	fill area set with data	polyline set 3 with colour
cell array 3 plus	fill area 3	polymarker
circle	fill area set 3	polymarker 3
circle 3	fill area set 3 with data	quadrilateral mesh with data
circular arc	fill circle	quadrilateral mesh 3 with data
circular arc 3	fill circle 3	set of fill area sets with data
circular arc close	fill ellipse	set of fill area sets 3 with data
circular arc close 3	fill ellipse 3	text
ellipse	generalized drawing primitive	text 3
ellipse 3	generalized drawing primitive 3	triangle set with data
elliptical arc	non-uniform B-spline curve 3	triangle set 3 with data
elliptical arc 3	non-uniform B-spline curve 3 with colour	triangle strip with data
elliptical arc close	non-uniform B-spline surface 3	triangle strip 3 with data

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Table 2-Bundled attribute selection structure elements

set back data mapping index	set edge index	set polyline index
set back interior index	set highlighting index	set polymarker index
set back reflectance index	set interior index	set reflectance index
set data mapping index	set parametric surface index	set text index

Table 3-Individually selected bundled attribute structure elements

set back data mapping method	set edgetype	set linetype offset
set back interior colour	set edgetype adaptability	set linewidth scale factor
set back interior shading method	set edgetype continuity	set marker size scale factor
set back interior style	set edgetype offset	set marker type
set back interior style index	set edgewidth scale factor	set parametric surface characteristics
set back reflectance model	set highlighting method	set polyline colour
set back reflectance properties	set interior colour	set polyline colour index
set character expansion factor	set interior colour index	set polyline shading method
set character spacing	set interior shading method	set polymarker colour
set curve approximation criteria	set interior style	set polymarker colour index
set data mapping method	set interior style index	set reflectance model
set edge colour	set linecap	set reflectance properties
set edge colour index	set linejoin	set surface approximation criteria
set edge flag	set linemitre limit	set text colour
set edgecap	set linetype	set text colour index
set edgejoin	set linetype adaptability	set text font
set edgemitre limit	set linetype continuity	set text precision

Table 4-Geometric attribute structure elements

set annotation style	set character height	set pattern size
set annotation text alignment	set character up vector	set text alignment
set annotation text character height	set individual ASF	set text path
set annotation text character up vector	set pattern reference point	
set annotation text path	set pattern reference point and vectors	

Table 5-Identification attribute structure elements

add names to set	remove names from set	set pick identifier

Table 6-Viewing attribute structure elements

set facet culling mode	set hlhsr identifier	set view index
set facet distinguishing mode		

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Table 7-Rendering attribute structure elements

set active textures	set colour mapping index	set texture resource optimization heuristics
set alpha data selection index	set depth cue index	set texture sampling
set alpha source selector	set light source state	set transparency
set back active textures	set rendering colour model	
set back transparency	set texture perspective correction	

Table 8-Transformation and clipping structure elements

restore modelling clipping volume	set local transformation	set modelling clipping volume
set global transformation	set local transformation 3	set modelling clipping volume 3
set global transformation 3	set modelling clipping indicator	

Table 9-Control structure elements

conditional execute structure	execute structure	set application real
conditional instance structure	instance structure	set condition flags
conditional return	pop state*	set condition flags from tests
conditional skip elements	push state*	
conditional skip to label	set application integer	

*Only available for direct interpretation.

Table 10-Editing structure element

label

Table 11-Generalized structure element

generalized structure element

Table 12-Application data element

application data

Although the CSS is conceptually a centralized data store, this does not stop an implementation from decentralizing the physical organization of the data storage, provided that this places no restrictions on the functionality.

4.3.2 Structure networks

Structure networks are organized as directed acyclic graphs. A structure may invoke other structures within the CSS. The direct interpretation structure may also invoke structures managed by the application. Structure invocations in the CSS are achieved by one of the following structure elements:

a) execute structure,

b) instance structure,

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

c) conditional execute structure,

d) conditional instance structure,

The directly interpreted structure elements ''push state'' and ''pop state'' are used as part of the invocation of structures managed by the application. Such invocations are known as structure references. Structure networks cannot be recursive (there should not exist a sequence of invocations from a structure which results in the invocation of that structure). However, it is not mandated that this condition be checked.

The function *instance structure* allows an application to include the set of structure elements belonging to a structure other than the one currently being traversed. During traversal, the structure elements belonging to the structure being instanced appear to be a part of the current structure being traversed.

There is a single direct interpretation structure with a name defined when it is posted to a workstation. A network is achieved by the application using the ''push state'' and ''pop state'' structure elements to save and restore the state or by referencing structures in the centralized structure store. A structure network is shown diagrammatically in figure 4. Structures are represented by circles labelled with the structure identifiers. Structure references are shown by arrows pointing to the invoked structures.

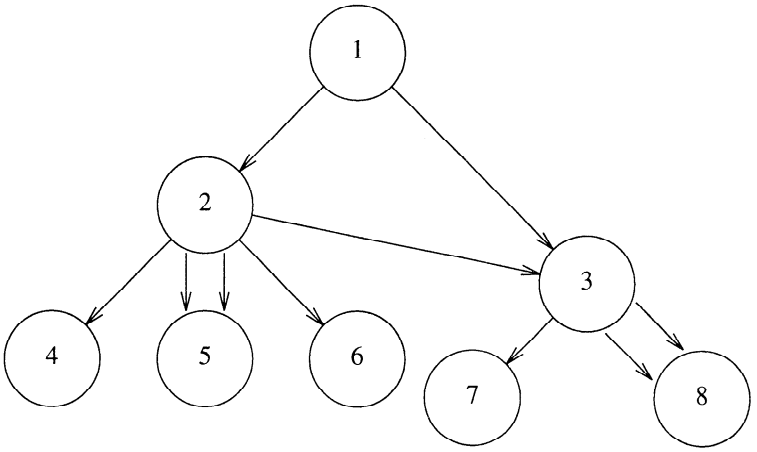

Figure 4 - A hierarchical structure network

When a structure S contains invocations of other structures, it is said to be a parent structure of those structures; correspondingly, each of the invoked structures is said to be a child structure of S. For example, structure 2 is the parent of structures 4,5, and 6; structures 2 and 3 are children of structure 1. Also, structures 1 and 2 are said to be ancestors of structures 4,5 and 6; while structures 7 and 8 are descendants of structures 1, 2 and 3.

Structure 1 can be a direct interpretation structure. Structures 2 and 3 and lower level structures in the network may also be structure invocations achieved by ''push state'' and ''pop state'' directly interpreted structure elements (see 4.3.3). The only constraint is that a structure in the centralized structure store cannot invoke a directly interpreted structure.

There is a single set of names for structure identifiers. The application is responsible for maintaining any distinctions between identifiers used for the direct interpretation structure operations and for CSS structures. However, it is allowed to use identifiers for the direct interpretation structure which are the same as CSS structure identifiers. This is not considered an error. The initial identifier associated with the direct interpretation structure is assigned when it is posted or renewed.

The identifier associated with the direct interpretation structure is used for the outermost level of the hierarchy. Inner level identifiers are specified within structure elements which change the level of the hierarchy (for example, the ''push state'' and ''pop state'' elements). The same identifiers may be used for multiple levels of the direct interpretation structure network hierarchy.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Many types of operations available to the application are allowed on structure networks within the CSS. These are structure display, structure editing, structure manipulation, search and inquiry, and structure archival and retrieval. Conceptually, all changes to the CSS which result from any of the above operations are atomic and take effect immediately. However, the actual time at which visual changes take effect depends on the capabilities of the workstation and the current display update state (see 4.7.4). Visual changes resulting from the above operations may affect the display of information resulting from the interpretation of directly interpreted structure elements.

A structure can be created in one of the following ways:

e) when a reference to a non-existent structure is inserted into a structure;

f) when the structure is opened for the first time;

g) when the structure is posted for display on a workstation;

h) when the structure is referenced in any function changing the structure identifier;

i) when a structure not existing in the CSS is retrieved from an archive;

j) when a non-existent structure is emptied;

k) when the action of pushing the traversal state references a structure identifier for the first time.

The following functions take no action if a specified structure is non-existent:

l) an unpost operation;

m) an operation to copy all elements from a non-existent structure into the currently open structure;

n) a delete structure operation.

4.3.3 Structure traversal and display

4.3.3.1 Posting

Posting causes structure networks to be displayed on a workstation.

Structure networks are grouped together into posting groups. How a posting group is displayed relative to other posting groups can be defined. For each workstation, posting group 0 is predefined. A structure is identified for display on a workstation by a posting function. Invoking the function POST STRUCTURE adds the specified structure network to posting group 0 and UNPOST STRUCTURE removes the specified structure from posting group 0. A structure network may be displayed on a workstation only if it is a member of a posting group on that workstation (see figure 5). Invoking the function POST STRUCTURE TO GROUP adds the specified structure network to the specified posting group and UNPOST STRUCTURES FROM GROUPS removes the specified structure networks from the specified posting groups. The function UNPOST ALL STRUCTURES removes all structure networks from all posting groups on the specified workstation. The function UNPOST ALL STRUCTURES FROM GROUP removes all structure networks from the specified posting group. The posting groups to which a structure network has been posted may be determined by invoking the INQUIRE SET OF GROUPS TO WHICH POSTED function.

When a non-existent structure is posted, a new empty structure is created and posted. Each workstation keeps a record of the posting groups defined for that workstation and which structure networks are contained in each posting group.

Structure display functions affect the list of posted structure networks for posting group 0 in the workstation state lists. The actual display of the structures identified in the list of posted structures is subject to display update state (see 4.7.4).

4.3.3.2 Traversal

Conceptually, a traversal process is associated with each instance of a posted structure network.

The traversal process remains in existence until the structure network is unposted from the workstation through UNPOST or DELETE functions. To display a network, the structure elements have to be extracted from the CSS and processed. Traversal of a network leads to the display of graphical output from the structure network at each workstation to which the network is posted. Traversal is divided into two phases. The first phase steps sequentially through the

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

structure elements while the second phase interprets each structure element encountered. In traversing structures in the CSS, both the stepping and interpretation phase are performed by PHIGS. In traversing the direct interpretation structure, only the interpretation phase is performed by PHIGS; the application performs the stepping phase or defines the directly interpreted structure elements when needed from available application managed graphical data.

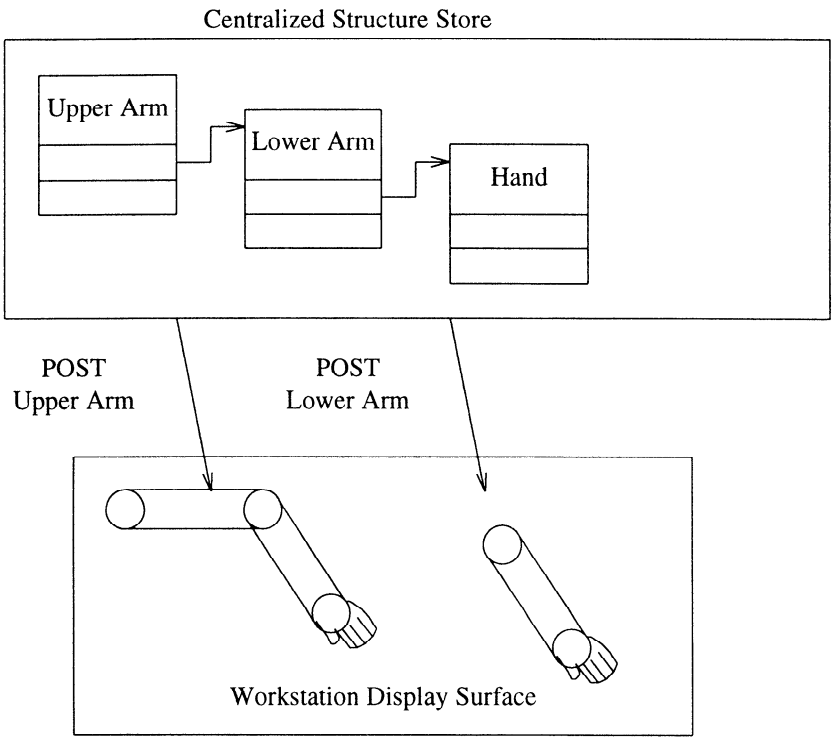

Figure 5 - A structure network with two posted structure networks

The interpretation phase of traversal is performed by the traversal process which is associated with the posting of a structure. These concepts are illustrated in figures 6 and 7.

The traversal process interprets each structure element in the structure network sequentially, starting at the first element of the top of the network. Any change to the posted structure network triggers a retraversal of the structure network, so that any changes to a posted structure network are processed immediately. Whether the resulting changes are visible at the workstation depends on the display update state of the workstation (see 4.7.4). No knowledge of how traversal interpretation takes place is required. Its effect is as described above. How the traverser chooses to achieve the effect is implementation dependent.

A PHIGS traversal state list is associated with each traversal process. Values in this state list may be accessed or modified when structure elements are interpreted. Each time a traversal is initiated, the associated PHIGS traversal state list is initialized. The initial values are taken from the PHIGS description table.

4.3.3.3 Posting groups

Each posting group on a specified workstation has a view index defined as the default. When a posted structure network in a posting group is traversed, the view index in the traversal state list is initialized to the view index defined as default for the posting group. Posting group 0 has view index 0 defined as default. Other posting groups have the associated default view index specified by the DEFINE POSTING GROUP function. If no ''set view index'' structure elements appear in a structure network in the posting group, the effect is as if ''set view index'' to the view index defined for the posting group is the first structure element of the structure network.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

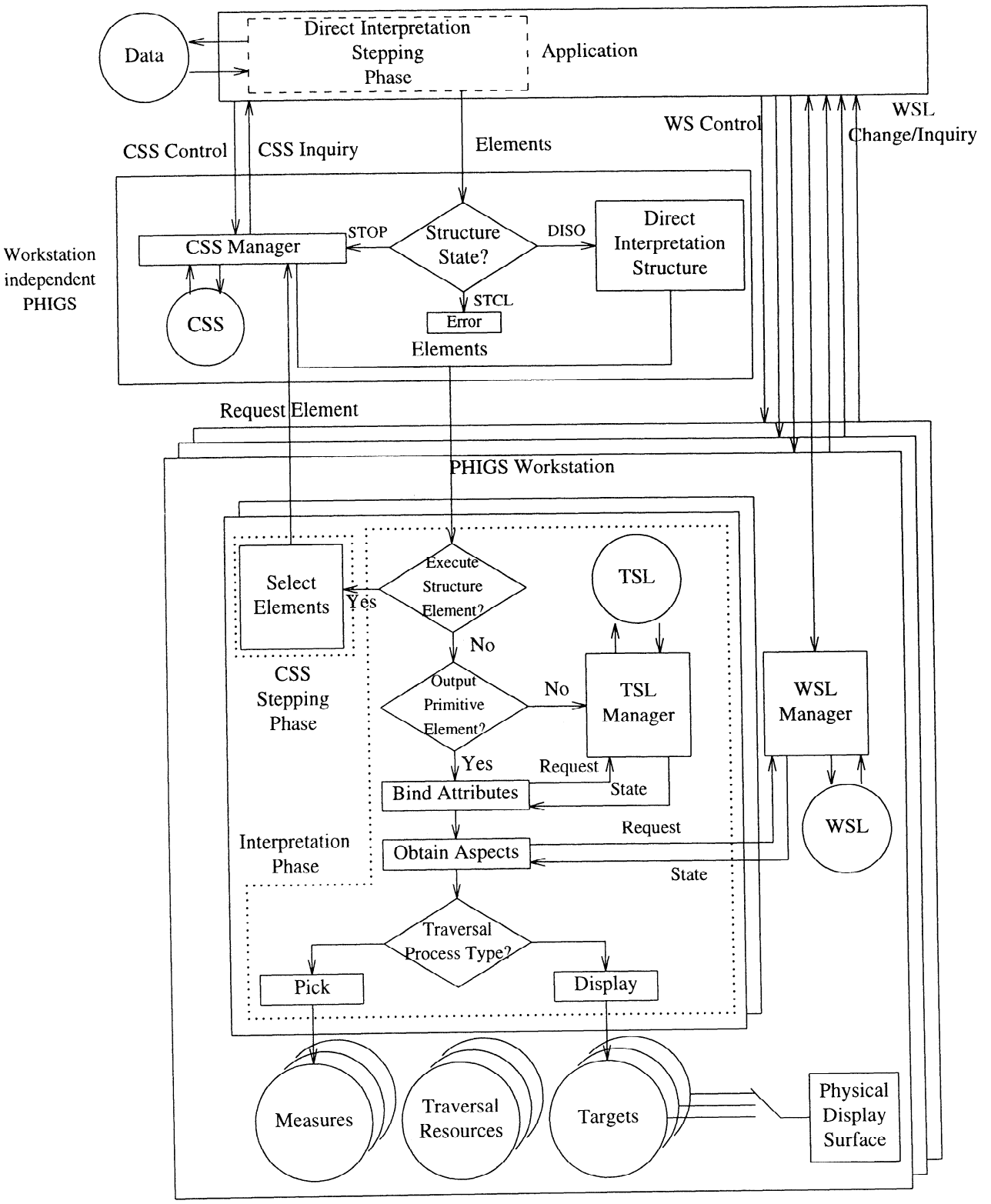

Figure 6 - PHIGS structure element processing architecture

AVA Documents Document Sharing Hub
Full document details: ISO_IEC 9592-1_1997

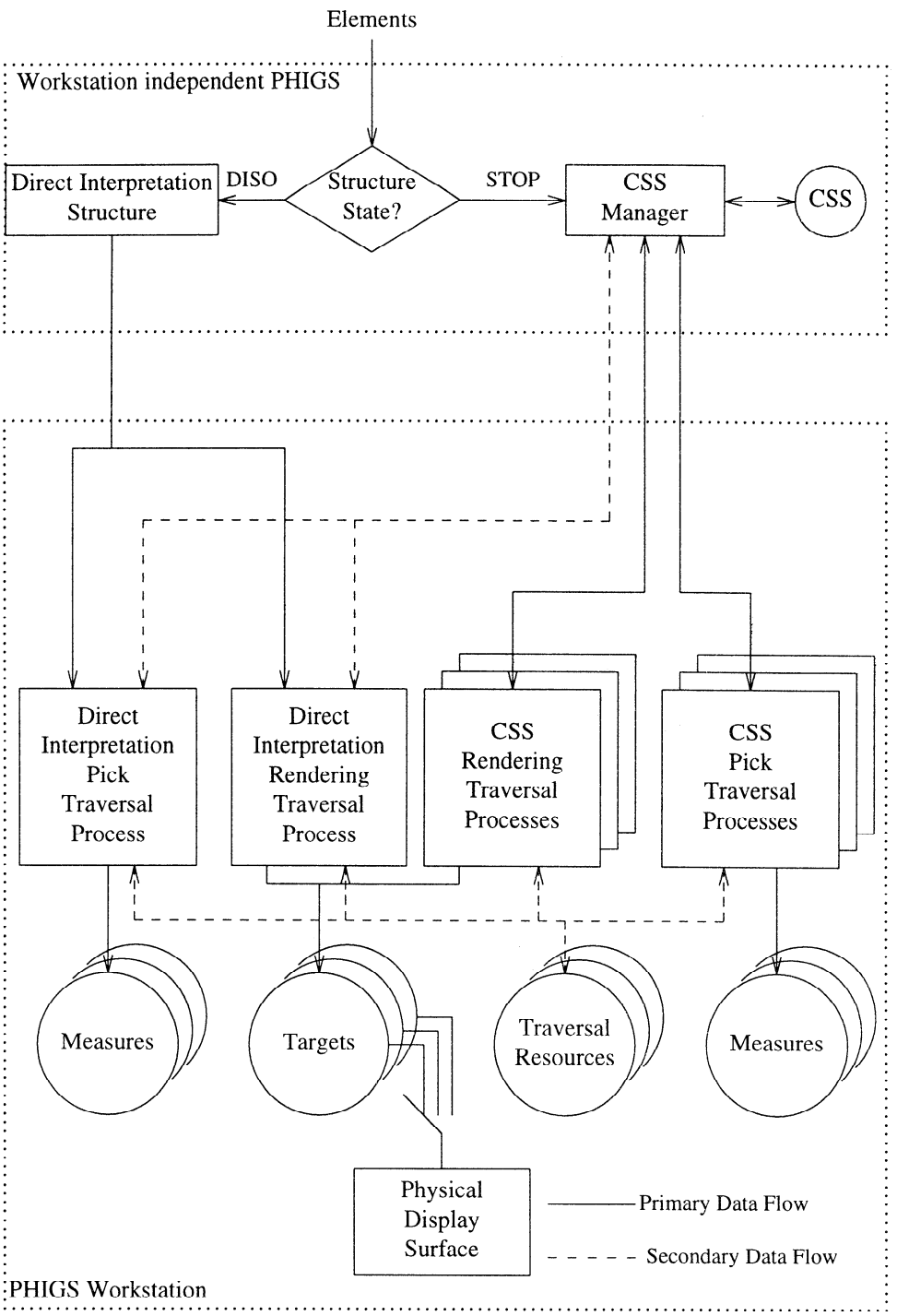

Figure 7 - PHIGS traversal process types

Each posting group has a posting group volume, a posting group priority, a posting group status, a posting group background style, a posting group background method, a posting group border indicator and a posting group border index. The view clipping limits associated with the specified default view index for the posting group at the time traversal commences together with the Z range from 0 to 1 define the posting group volume. The other items have default values

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

explicitly defined by the parameters of the DEFINE POSTING GROUP function which may be later modified.

Any ''set view index'' structure elements encountered during traversal are interpreted as specified in 4.8.4. The results of interpreting output primitive generating structure elements contained within the structures posted to the posting group are clipped to the intersection of the posting group volume and the view clipping limits of the current view.

A posting group on a specified workstation is traversed if the posting group status is set to ACTIVE and is not traversed if the posting group status is set to INACTIVE. Posting groups other than 0 are defined as INACTIVE initially. Posting group 0 is initialized to ACTIVE. The posting group status can be changed by invoking the function SET POSTING GROUP STATUS. Whether traversing the posting group affects the display on the workstation depends on the other posting group parameters.

Each posting group has a posting group background style which can be either TRANSPARENT or OPAQUE. The extent of the background is defined by the posting group volume. The background has a posting group border indicator which can be either ON or OFF. The posting group border has an edge defined by the border index, an index into the edge bundle table.

If the posting group background style is TRANSPARENT, this posting group background does not obscure posting groups of lower posting group priority. If the posting group background style is OPAQUE, it is possible for this posting group background to obscure another posting group of lower posting group priority. The graphical output of the posting group with lower posting group priority which is within the posting group volume of the posting group with higher posting group priority will be obscured by a background associated with the posting group with higher posting group priority. The background of an ACTIVE posting group is behind any output produced by traversing structure networks posted to the posting group.

The manner in which the posting group background of an OPAQUE posting group is displayed is defined by the posting group background method. Posting group background methods defined in PHIGS are:

≤0 implementation dependent

1 colour table entry 0

2 colour index

3 colour specification

4 image resource

≥5 reserved for registration or future standardization

The default posting group background method is 1. The posting group background method can be changed by invoking the function SET POSTING GROUP BACKGROUND METHOD.

The function SET POSTING GROUP BORDER INDICATOR sets the posting group border indicator and the function SET POSTING GROUP BORDER INDEX specifies the posting group border appearance using an index into the edge bundle table. If the posting group border indicator is set to ON and the posting group background style is OPAQUE, the posting group border is displayed dependent on the relative posting group priority.

Posting group 0 is initialized with posting group background style set to OPAQUE, the posting group border indicator set to OFF, the posting group border index set to 1, and the posting group background method set to 1. The values for the posting group 0 cannot be changed nor can posting group 0 be undefined.

When a posting group is no longer needed, it may be eliminated by invoking the UNDEFINE POSTING GROUP function.

4.3.3.4 Priority

A display priority is assigned to a structure network when it is added to a posting group. The priorities supported by an implementation shall be mapped uniformly across the range [0,1]. The application specifies the relative importance of structure networks within a posting group by this display priority. If several structure networks within a posting group on traversal display graphical output at the same display space location, the implementation will ensure the display of the structure network with the highest display priority within the posting group if the posting group is visible. Where two structure networks in a posting group have the same display priority, an implicit display priority based on the

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

sequence of posting is used, whereby the last posted structure network has higher display priority within the posting group.

Posting groups on a specified workstation have a precisely defined posting group priority. The posting groups are ordered from the lowest to the highest. Invoking the function DEFINE POSTING GROUP defines a posting group on a specified workstation and specifies its posting group priority relative to a previously defined posting group. If posting group priority is specified as HIGHER than the reference posting group, the new posting group is defined as having posting group priority immediately higher than the reference posting group. If posting group priority is specified as LOWER than the reference posting group, the new posting group is defined as having posting group priority immediately lower than the reference posting group. The posting group priority of a posting group can be changed by invoking the function SET POSTING GROUP PRIORITY.

If several posting groups on traversal display graphical output at the same display location, the implementation ensures the display of the posting group with the highest posting group priority.

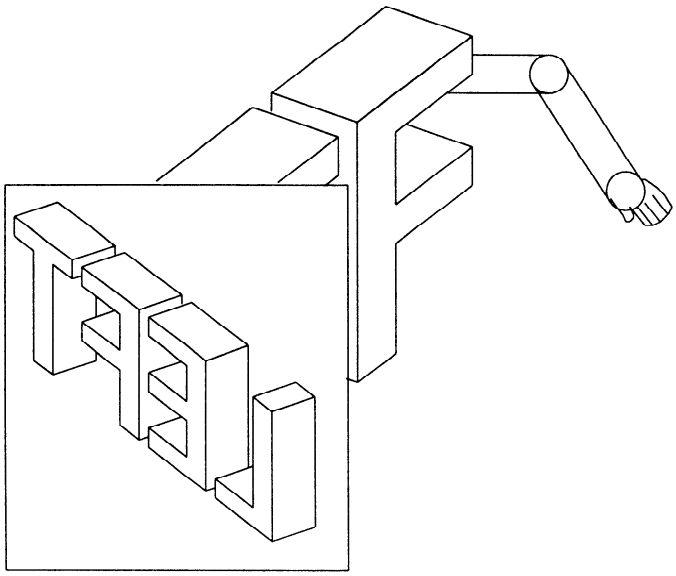

Figure 8 - Posting group background styles

Figure 8 shows three posting groups displayed on a workstation. The posting group with highest posting group priority is the letters LEFT. The display priority of the letters is with L having the highest display priority, then E and F and T has the lowest. L may obscure E, F and T. E may obscure F and T. F may obscure T. This posting group has the background style set to OPAQUE. As it is the posting group with highest posting group priority, the background obscures the other two posting groups. The posting group with the letters E and F is the posting group whose posting group priority is greater than the Upper Arm from figure 5 but lower than the priority of the one with the OPAQUE background style. Its background is transparent. The display priority of E is higher than F so E obscures F within the posting group. As it has higher posting group priority than the Upper Arm, it also obscures the Upper Arm. All three posting groups are ACTIVE. The border indicator for the LEFT posting group is set to ON and the border index has not been reset so it is defined as border index 1.

Figure 9 shows the display after the LEFT posting group has been reset to having the background style TRANSPARENT and the Upper Arm posting group has been reset to INACTIVE.

The current status of a posting group can be obtained by invoking the function INQUIRE POSTING GROUP. The structure networks in a posting group can be obtained by invoking the function INQUIRE POSTED STRUCTURES FROM POSTING GROUP. The list of defined posting groups can be obtained by invoking the function INQUIRE LIST OF POSTING GROUPS.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

Posting groups may be predefined. Information concerning a predefined posting group can be obtained by invoking the function INQUIRE PREDEFINED POSTING GROUP. Information about posting group resources can be obtained by invoking the function INQUIRE POSTING FACILITIES.

When a non-existent structure is posted, a new empty structure is created and posted. The functions UNPOST STRUCTURE and UNPOST ALL STRUCTURES are used to disassociate structure networks from a workstation. The structure display functions affect the list of posted structure networks for posting group 0 in the workstation state lists. The actual display of the structures identified in the list of posted structure networks is subject to display update state (see 4.7.4).

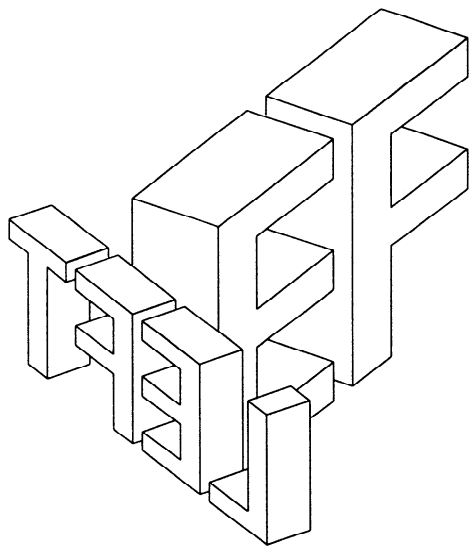

Figure 9 - Posting group with TRANSPARENT background style

4.3.3.5 Direct interpretation structure

The direct interpretation structure is associated with a posting group. The direct interpretation structure is identified for display on a workstation by the function POST DIRECT INTERPRETATION STRUCTURE which associates it with posting group 0. The function POST DIRECT INTERPRETATION STRUCTURE TO POSTING GROUP identifies it for display with the specified posting group. The posting group attributes (default view index, status, background style, background method, background data record, border indicator, border index) also apply to the direct interpretation structure. A structure identifier is associated with the interpreted items posted to this workstation. Each posting creates a rendering traversal process and initializes the corresponding traversal state list. Interpretation of directly interpreted structure elements on the specified workstation is discontinued by invoking UNPOST DIRECT INTERPRETATION STRUCTURE which also destroys the corresponding rendering traversal process and traversal state list. If a direct interpretation pick traversal process is also active (see 4.9.8), it is disabled and destroyed.

Interpreting of directly interpreted structure elements posted to workstations only occurs when the direct interpretation structure is OPEN. The functions OPEN DIRECT INTERPRETATION STRUCTURE and CLOSE DIRECT INTERPRETATION STRUCTURE can be invoked at any time. If the direct interpretation structure is opened when there are no postings of the direct interpretation structure, structure elements which are inserted will be discarded. The element pointer is set to zero at the time that the direct interpretation structure is opened.

Values in the traversal state list for the direct interpretation structure posting to a workstation can be accessed or modified when directly interpreted structure elements are interpreted. These values can be reset to the initial values by unposting and then reposting the direct interpretation structure or by invoking the function RENEW DIRECT INTERPRETATION STATE.

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997

The ''push state'' and ''pop state'' directly interpreted structure elements support the same behaviour as the saving and restoring of state which occurs when processing the ''execute structure'' structure element. This invocation type is called EXECUTE. Also supported is a behaviour which does not store the state. This invocation type is called INSTANCE. When the ''push state'' element is interpreted, the following actions occur:

a) traversal of the current level of hierarchy in the direct interpretation structure network is suspended;

b) the name of the current structure is saved along with information for returning to the next element in the current level of the hierarchy (i.e., the element pointer);

c) the structure identifier specified becomes the name of the new current level of the hierarchy and the element pointer is set to zero.

d) if the invocation type is EXECUTE, the current state of the PHIGS traversal state list is saved and the global and local transformations are updated as for ''execute structure''.

Traversal interpretation will now take place at the next level of hierarchy. When the corresponding ''pop state'' element is interpreted, the following actions occur:

e) the name of the previous level of hierarchy is restored and the element pointer is restored to the saved value.

f) if the corresponding ''push state'' was of invocation type EXECUTE, the PHIGS traversal state list values saved by the corresponding ''push state'' are restored;

g) traversal of the previous level of hierarchy in the direct interpretation structure network is resumed.

The edit mode used for editing structures is ignored when the direct interpretation structure is open. All elements are considered to have been inserted into the directly open structure. The element pointer is updated as each directly interpreted structure element is processed. The element pointer is reset whenever the RENEW DIRECT INTERPRETATION STATE function is invoked.

Posting the direct interpretation structure to a workstation will set the 'direct interpretation structure posted' in the workstation state list for the direct interpretation render traversal process to TRUE. The actual processing of directly interpreted structure elements will occur as these elements are interpreted according to the capabilities of each workstation to which the direct interpretation structure is posted. Figures 6 and 7 show how structure elements in the centralized structure store and the directly interpreted structure are processed.

For directly interpreted structures, the direct interpretation mode can influence the effect of interpreting directly interpreted structure elements as shown in figure 10. If the direct interpretation mode is set to INTERPRET_ALL, the effect is as shown in figures 6 and 7. This is the initial setting of direct interpretation mode.

Invoking the function SET DIRECT INTERPRETATION MODE can change the mode to INTERPRET_NONE or INTERPRET_STATE. These modes are provided for better control of quick updates or for partial redrawing of the non-retained picture.

If the direct interpretation mode is set to INTERPRET_NONE, no elements are interpreted. If the direct interpretation mode is set to INTERPRET_STATE, only those elements that affect the traversal state list are interpreted.

The current setting of the direct interpretation mode can be determined by invoking the function INQUIRE DIRECT INTERPRETATION MODE.

The actions of OPEN/CLOSE DIRECT INTERPRETATION STRUCTURE are independent of those of POST/UNPOST DIRECT INTERPRETATION STRUCTURE. Any structure elements inserted into the direct interpretation structure prior to the direct interpretation structure being posted to any workstation will be discarded. Traversal processes initiated by posting the direct interpretation structure on a particular workstation are suspended while the direct interpretation structure is closed. Such traversal processes will resume whenever the direct interpretation structure is opened.

The set of workstations to which the direct interpretation structure is posted can be inquired by invoking the INQUIRE SET OF WORKSTATIONS TO WHICH DIRECT INTERPRETATION STRUCTURE POSTED function. The characteristics of direct interpretation structure processing on a particular workstation can be determined by invoking the INQUIRE POSTED DIRECT INTERPRETATION STRUCTURE function.

Documents
Document Sharing Hub

Full document details: ISO_IEC 9592-1_1997