INTERNATIONAL STANDARD

ISO/IEC 9592-3

Second edition
1997-11-15

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

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

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

Partie 3: Spécification du codage mode texte en clair du fichier d'archive

Reference number
ISO/IEC 9592-3:1997(E)

Contents

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 9592-3 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 9592-3:1989), which has been technically revised. It also incorporates Amendment 1:1992.

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*

Annex A of this part of ISO/IEC 9592 is for information only.

Introduction

The clear-text encoding of the PHIGS archive file provides a representation of the archive file syntax that is easy to type, edit, and read. It allows an archive file to be edited with any standard text editor, using the internal character code of the host computer system. The primary objectives are:

a) HUMAN EDITABLE: The clear-text encoding should be able to be hand-edited or, if desired, hand-constructed.

b) HUMAN-FRIENDLY: The clear-text encoding should be easy and natural for people to read and edit. Although what is easiest and most natural is a subjective judgement that varies among users, contributing factors such as ease of recognition, ease of remembering, avoidance of ambiguity, and prevention of mistyping have all been considered.

c) MACHINE-READABLE: The clear-text encoding should be able to be parsed by software.

d) USABLE IN A WIDE VARIETY OF EDITORS: The clear-text encoding should not have any features that make it difficult to edit in normal text editors.

e) INTERCHANGEABLE BETWEEN DIVERSE SYSTEMS: The clear-text encoding should be encoded in such a way as to maximize the set of systems which can utilize it. No assumptions should be made as to word size or arithmetic modes used to interpret the archive file.

f) USES STANDARDIZED ABBREVIATIONS: Where language encoding of other graphics standards have established standard abbreviations, or where common practice in the data processing and graphics industries has established well-known abbreviations, these abbreviations are used. In accordance with the principle of "least astonishment", this approach should minimize the time needed to learn to use this encoding.

This part of ISO/IEC 9592 draws extensively for its model of an archive file format on ISO 8632. The set of characters needed to implement the clear-text encoding is a subset of those included in national versions of ISO 646. Any character set that can be mapped to and from that subset may be used to implement the encoding.

Information technology – Computer graphics and image processing –

Programmer's Hierarchical Interactive Graphics System (PHIGS) – Part 3: Specification for clear-text encoding of archive file

1 Scope

This part of ISO/IEC 9592 specifies a clear-text encoding of the PHIGS archive file. For each of the archive file elements specified in ISO/IEC 9592-2, a clear text encoding is specified. This part of ISO/IEC 9592 specifies the overall format of the archive file and the means by which comments may be interspersed in the archive file.

This encoding of the PHIGS archive file allows archive files to be created and maintained in a form which is simple to type, easy to edit and convenient to read.

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*

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 archive file descriptor: A group of elements that describe the functional capabilities needed to process the archive file.

3.2 archive file generation: The process that produces a PHIGS archive file.

3.3 archive file retrieval: The process that reads a PHIGS archive file, retrieves the contents, and transfers the result to the PHIGS centralized structure store.

4 Clear-text encoding format

4.1 Notational conventions

a) Unbracketed strings are terminals of this grammar which appear exactly, subject to the variations on case and null characters given in 4.2.2.

b) Bracketed strings are either non-terminals (with further productions given), character symbol names (such as COMMA), or parameters of the PHIGS archive file element in the form <x:y>.

c) The following metasymbols define productions, grouping, and repetition.

::=	→	''becomes'' or ''is realized as''
<...>*	→	*star closure (0 or more occurrences)
<...>+	→	plus closure (1 or more occurrences)
<...>o	→	optional (exactly 0 or 1 occurrences)
<x:y>	→	parameter type x with meaning y
x | y | z	→	exactly one of the items x, y and z (any of x, y and z may be multiple non-terminals if there is no ambiguity)
<x|y|z>	→	exactly one of x or y or z, brackets delineate the scope
{...}	→	a comment (not part of the production)
<...>(n)	→	exactly n occurrences, n=0,1,2,...
<x,..,x>	→	a list of occurences of x each separated by <SEP>

The meaning of the last notation can be expressed as follows:

<A,..,A> ::= <A> < <SEP> <A> >*

d) SPACES are used for readability in the grammar description; SPACES in the actual archive file are indicated through the separator productions given in 4.3.3.

e) The metasymbols used in describing the grammar do not appear in the actual archive file.

4.2 Archive file format

4.2.1 Introduction

A clear-text encoding of a PHIGS archive file consists of a stream of characters forming a series of elements, each of which starts with an element name and ends with one of the element delimiters, either the SLASH character (also known as SLANT or SOLIDUS) or the SEMICOLON character. These characters do not act as element delimiters when occurring within the bounds of a string parameter, as defined below.

The order of elements within a clear-text encoding of a PHIGS archive file is specified by ISO/IEC 9592-2. This specifies a formal grammar over the following eight symbols:

BEGIN ARCHIVE FILE
END ARCHIVE FILE
BEGIN STRUCTURE
END STRUCTURE
ARCHIVE FILE VERSION
ARCHIVE FILE DESCRIPTION
STRUCTURE ELEMENT
EXTERNAL ELEMENT

Each of these symbols is treated as a non-terminal in the formal grammar that follows. Taken together, the formal

grammar of ISO/IEC 9592-2 and this part of ISO/IEC 9592 provide a formal grammar for a PHIGS archive file over the ISO/IEC 646 character set.

4.2.2 Character repertoire

In order that the metasymbols used in describing the grammar do not appear in the actual archive file, the character repertoire of the clear-text encoding will be limited to those characters enumerated below, except for string parameters, which, at the minimum, support the full ISO/IEC 646 character set and, optionally may include characters which shift into other character sets. Each string is assumed to start in the ISO/IEC 646 character set.

Upper-case characters:
"A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M",
"N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z"

- Lower-case characters:
"a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l", "m",
"n", "o", "p", "q", "r", "s", "t", "u", "v", "w", "x", "y", "z"

Digits:
"0", "1", "2", "3", "4", "5", "6", "7", "8","9"

" " (SPACE character)
"+" (PLUS character)
"-" (MINUS character)
"#" (NUMBER SIGN)
";" (SEMICOLON character)
"/" (SLASH, SLANT, or SOLIDUS character)
"(" (LEFT or OPEN PARENTHESIS character)
")" (RIGHT or CLOSE PARENTHESIS character)
"," (COMMA character)
"." (DECIMAL POINT or PERIOD character)
"'" (APOSTROPHE or SINGLE QUOTE character)
"""" (DOUBLE QUOTE character)
"_" (UNDERSCORE character)
"$" (DOLLAR SIGN or CURRENCY symbol)
"%" (PERCENT SIGN character)

Lower-case characters are considered to be the same as upper-case characters when occurring outside of string parameters. Any combination of lower-case and upper-case characters may be used within an element or enumerated parameter name.

The UNDERSCORE and DOLLAR SIGN symbols are defined as "null characters" within this encoding. They may appear anywhere within an archive file, and are mandated to have no effect on parsing (outside of string parameters). They are available for the generator or editor of an archive file to use in enhancing readability of tokens. For example, the following are all equivalent:

linetype, LINETYPE, LineType, line_type, $LINETYPE, L_I_N_E$T_Y_P_E.

The following are all equivalent:

123456, $123456, 123_456, $123_456, $12$34$56.

Those control characters that are format effectors (BACKSPACE, CARRIAGE RETURN, LINEFEED, NEWLINE, HORIZONTAL TAB, VERTICAL TAB, and FORMFEED) are permitted in the archive file, but are treated as SPACE characters (that is, as soft delimiters) by the archive file interpreter whenever they occur outside of a string parameter. They may be used to assist in formatting the archive file to improve its readability. A PHIGS archive file written in the clear-text encoding is considered not to be a conforming interchange if it includes characters other than those listed in

the repertoire and the format effectors (outside of string parameters). Implementation-dependent extensions which require use of characters other than the above should be embedded in the string parameters of the GSE or APPLICATION DATA elements, or in comments.

The code set of the characters is not fixed by this part of ISO/IEC 9592. In order to accomplish the objective of editability, it is permitted to encode the clear-text encoding using the character set codes native to the system. It is presumed that standard conversion facilities can be used in translating clear-text PHIGS archive files from one system's character set codes to another, consistent with the treatment of other text files being transferred between systems. The ISO/IEC 646 codes should be used to encode clear-text archive files for transport between diverse systems.

Null characters or format effectors outside of text strings which do not exist in the target system's encoding may be dropped in such translation, and lower-case letters translated to upper case as necessary, without altering the information content of the archive file. Likewise, the two statement delimiter characters are interchangeable and may be changed in such a translation without affecting the information content of the archive file. The two string delimiter characters are interchangeable, but any translation shall correctly handle the possible occurrence of either string delimiter character within the string parameter.

4.2.3 Separators

4.2.3.1 Element separators

<TERM> ::= <OPTSEP> <SLASH | SEMICOLON> <OPTSEP>

The SEMICOLON and SLASH characters may be used to delimit elements in a clear text archive file. These elements do not, however, terminate an element when they occur within a string parameter, as described in 4.2.4.3.

The elements of the archive file are not terminated by the ends of records, as indicated by control characters such as CR (carriage return) or LF (linefeed). Multiple elements may exist on one line, and any element may extend over multiple lines.

4.2.3.2 Parameter separators

The following productions are used in the clear-text encoding for parameter separators:

<SEPCHAR> ::= <SPACE | CARRIAGE RETURN | LINEFEED | HORIZONTAL TAB | VERTICAL TAB | FORMFEED>
<SOFTSEP> ::= <SEPCHAR>+
<OPTSEP> ::= <SEPCHAR>*
<HARDSEP> ::= <OPTSEP> <COMMA> <OPTSEP>
<SEP> ::= <SOFTSEP> | <HARDSEP>

Most commands require a SOFTSEP after the element name (e.g., at least one space). This permits element names to be formed from a mixture of alpha and numeric characters.

The separator between parameters is usually a SEP. This format permits omission of parameters. (Two consecutive COMMAs indicate an omitted parameter.)

Since the enclosing APOSTROPHE or DOUBLE QUOTE character sufficiently delineates string parameters, and the statement delimiter SLASH also sets off the data on either side of it, the separators between these characters and adjacent parameters or element names are optional (OPTSEP).

SEPCHAR characters are not permitted within a name (element or enumerated type), or within the representation of a numeric parameter. Any place where a SEPCHAR is permitted (other than inside a string parameter), an arbitrary number of SEPCHARs may be used.

4.2.3.3 Comments in the archive file

Comments may be included in a clear-text archive file to enhance its readability and usefulness. Some uses of comments might be to document hand-cdited changes to the archive file, or as "notes to one's self" made while reading an archive file. To includc other forms of nongraphical information in an archive file, it is suggested that the EXTERNAL clement be used.

Comments are encoded as a series of printing characters and <SEPCHAR>s surrounded by "%" (PERCENT SIGN) characters. The text of the comment may not include this comment delimiter character.

Comments may be included any place that a separator may be used, and are equivalent to a <SOFTSEP>; they may be replaced by a SPACE character in parsing, without affecting the meaning of the archive file.

4.2.4 Encoding of parameter types

4.2.4.1 Integer-bound types

Integers, integer coordinates and indices are all bound to signed integers, indicated in the encoding as I.

<I>	::=	<decimal integer> | <based integer>
<decimal integer>	::=	<sign>o <digit>+
<sign>	::=	<PLUS SIGN> | <MINUS SIGN>
<digit>	::=	0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<based integer>	::=	<sign>o <base> <NUMBER SIGN> <extended digit>+
<base>	::=	2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16
<extended digit>	::=	<digit> | A | B | C | D | E | F | a | b | c | d | e | f

The null characters are permitted within numbers, but are not shown in the productions for simplicity.

A decimal integer has an optional sign and at least one digit. If the sign appears, it immediately precedes the number with no intervening SPACE (or other <SEPCHAR>) characters allowed.

A based integer has an optional sign, a base (an unsigned integer in the range 2..16 inclusive, represented in base 10), a "#", and a string of one or more extended digits. If the sign appears, it immediately precedes the number with no intervening SPACE (or other <SEPCHAR>) characters allowed. The extended digits used shall be valid for the base named or the archive file is not conforming; e.g., for base 8 the digits "8", "9", etc. are not valid, for base 2 only the digits "0" and "1" are valid, and so forth. Case is not significant for the extended digits.

If the sign is omitted for either form, the number is considered non-negative.

Both the base and the <extended digit>+ are interpreted as unsigned numbers, and the final result negated if a MINUS SIGN preceded the number. No assumptions should be made as to the word size of thc archivc file retrieval process, or whether the underlying arithmetic is one's complement, two's complement, or sign-magnitude. For example, -1 would be encoded in hexadecimal as -16#1, -16#0001, etc. rather than 16#FFFF. Of course, archive files may be created utilizing prior knowledge of thc intended target machine, but any such assumptions will limit the portability of archive files and arc discouraged.

The PHIGS functional data types "C", "G2", "G3", "GS", "AI", "PI", "EI", "FN" and "WI" are represented by integers in this encoding.

Some examples are:

0, 007, -5, +123_456

The following are equivalent:

65535, 16#FFFF, 16#ffff, 8#177777, 2#1111_1111_1111_1111

The following are equivalent:

-32_768, -16#8000, -8#100000, -2#10000000_00000000

Interpretation of numerically bound parameters will be "free field", that is, there is an implied radix point to the right of the rightmost digit, and neither leading nor trailing spaces are significant. Leading zeroes are not significant.

4.2.4.2 Real-bound types

Reals and real coordinates are bound to real numbers, indicated in the encoding as R. These are written as either explicit-point or scaled_real numbers (or decimal integers, where appropriate).

```
<R>                      ::=  <explicit_point_number> | <scaled_real_number> | <decimal integer>
<explicit_point_number>  ::=  <sign>o
                              <
                              < <digit> + <PERIOD> <digit>* >
                              |
                              < <digit>* <PERIOD> <digit>+ >
                              >
<scaled_real_number>     ::=  <body> < E | e > <exponent>
<body>                   ::=  <explicit_point_number> | <decimal integer>
<exponent>               ::=  <decimal integer>
```

The interpretation of the scaled_real_number is the same as standard scientific notation (similar to FORTRAN "E" format), where the number represented by <body> is multiplied by 10 taken to the power <exponent>.

There shall be at least one digit in an explicit_point_number and in the body of a scaled_real_number, which in the case of a single-digit number may appear on either side of the radix point. It is recommended but not required that there be at least one digit before the radix point, for numbers with only a fractional part. Zero may be encoded as "0.", ".0", "0.0", "0", etc, although the second form is not recommended.

In the case of a scaled_real_number number (one where an "E" or "e" appears), at least one digit shall appear in the <exponent>. No SPACE or other <SEPCHAR> characters shall be included between the <body> and the "E" or "e", or between the "E" or "e" and the <exponent>.

The interpretation of parameters bound to this data type will be "free field", that is, if there is an explicit radix point, it sets the radix point of the internal representation, and neither leading nor trailing spaces or zeroes are significant. If the radix point is omitted, it is implied to be at the right of the rightmost digit of the explicit_point_number or of the <body> of the scaled_real_number. Thus, decimal I-format numbers may appear in a conforming archive file for parameters bound to real numbers when there is no fractional part.

For real numbers in all formats, the only base of representation that shall be utilized is base 10.

If the <sign> ("+" or "-") is omitted, the number is assumed to be non-negative. If the sign is present, it immediately precedes the body of the number, with no SPACE (or other <SEPCHAR>) characters allowed between it and the leftmost digit or radix point of the body of the number.

COMMA, SPACE and other <SEPCHAR> characters are not allowed within a number, but <NULLCHAR> characters may be included (and have no effect on parsing).

Examples:

3.14159
7.853982E-7
271828e-5
42
-.04321 (not recommended form)
-0.043_21
42E2
$5321.46

4.2.4.3 String-bound types

String parameters are represented as character strings immediately surrounded by a matched pair of either APOSTROPHE (SINGLE QUOTE) or DOUBLE QUOTE characters.

If an APOSTROPHE is needed in a string delimited with APOSTROPHE characters, it is represented by two adjacent APOSTROPHE characters at that position in the string. Likewise, if a DOUBLE QUOTE character is needed in a string delimited with DOUBLE QUOTE characters, it is represented by two adjacent DOUBLE QUOTE characters. For example, the following are equivalent:

"Murphy's Law: ""If it can go wrong, it will.""";
'Murphy''s Law: "If it can go wrong, it will."';

DATA RECORD (D) data type is represented as a string in this encoding.

STRING parameters are indicated in the encoding as S.

4.2.4.4 Enumerated types

Enumerated types are bound to names, just as element names are. Where an implementation wishes to support private enumerated type values, these shall be encoded as the letters "PRIV" followed by a string of <alpha | digit | null_character>*.

4.2.4.5 Derived types

In addition to the I, R, and S parameter formats, the following abbreviations are used as shorthand for the productions shown.

<(> ::= <LEFT PAREN> <OPTSEP>

<)> ::= <OPTSEP> <RIGHT PAREN>

CELLLIST ::= <I:CELL,..,I:CELL>0

CELLROW ::= <SEP> <CELLLIST> | <SEP> <(> <CELLLIST> <)>

COLRCURVE ::= <(> <I:ORDER> <SEP> <RLIST:KNOTS> <SEP>
<RATIONAL | NONRATIONAL> <SEP> <I:COLOUR TYPE> <SEP>
<COLRVLIST:CONTROL_POINTS> <)>

COLRSURF ::= <(> <I:U_ORDER> <SEP> <I:V_ORDER> <SEP>
<RLIST:U_KNOTS> <SEP> <RLIST:V_KNOTS> <SEP>
NONRATIONAL <SEP>
<I:COLOUR TYPE> <SEP> <COLRVLISTS:CONTROL_POINTS> <)>
{Each COLRVLIST contains control points along the *u* dimension.}

COLRSURFH ::= <(> <I:U_ORDER> <SEP> <I:V_ORDER> <SEP>
<RLIST:U_KNOTS> <SEP> <RLIST:V_KNOTS> <SEP>
RATIONAL <SEP>
<I:COLOUR TYPE> <SEP> <COLRVLISTS:CONTROL_POINTS> <)>
{Each COLRVLIST contains control points along the *u* dimension.}

COLRV ::= <I:COLOUR_INDEX> | <COORDLIST:COLOUR_COORDINATES>

```
COLRVH          ::=   <COORDLIST:COLOUR_COORDINATES>

COLRVLIST       ::=   <(> <COLRV,..,COLRV>o <)>

COLRVLISTH      ::=   <(> <COLRVH,..,COLRVH>o <)>

COLRVLISTS      ::=   <(> <COLRVLIST,..,COLRVLIST>o  <)>

COLRVLISTSH     ::=   <(> <COLRVLISTH,..,COLRVLISTH>o  <)>

COLRVROWS       ::=   COLRVLISTS

CONDMASK        ::=    2 #<0 | 1> (32)

CONDTEST        ::=   <(> <I:TEST_METHOD> <TMDR:DATA_RECORD> <CONDMASK> <)>

CONDTSTLST      ::=   <(>  <CONDTEST,..,CONDTEST>o  <)>

COORD           ::=   <I> | <R> { coordinate data }

COORDLIST       ::=   <(> <COORD,...,COORD> <)>

CURVEDR         ::=   <(> < <I:COUNT> | <R:APPROXIMATION_VALUE> |
                      <S:TYPE_DEPENDENT_DATA> >o  <)>

DATAMAPDR       ::=   <(>
                      <
                      <DATAMAPDR1> | <DATAMAPDR2> | <DATAMAPDR3> | <DATAMAPDR4> |
                      <DATAMAPDR5> | <S:METHOD_DEPENDENT_DATA>
                      >o
                      <)>

DATAMAPDR1      ::=   <SRCSELECLIST:SELECTOR>

DATAMAPDR2      ::=   <SRCSELECLIST:SELECTOR> <SEP>
                      <I:INDEX> <SEP> <R:LOWER_LIMIT> <SEP> <R:UPPER_LIMIT> <SEP>
                      <I:COLOUR_TYPE> <SEP> <COLRVLIST:COLOUR_LIST>

DATAMAPDR3      ::=   <SRCSELECLIST:SELECTOR> <SEP>
                      <I:INDEX> <SEP> <RLIST:RANGE_BOUNDARIES> <SEP>
                      <I:COLOUR_TYPE> <SEP> <COLRVLIST:COLOUR_LIST>

DATAMAPDR4      ::=   <SRCSELECLIST:SELECTOR> <SEP>
                      <I:INDEX_1> <SEP> <I:INDEX_2> <SEP>
                      <R:Ra_LOWER_LIMIT> <SEP> <R:Ra_UPPER_LIMIT> <SEP>
                      <R:Rb_LOWER_LIMIT> <SEP> <R:Rb_UPPER_LIMIT> <SEP>
                      <I:COLOUR_TYPE> <SEP> <COLRVLISTS:COLOUR_LISTS>

DATAMAPDR5      ::=   <SRCSELECLIST:SELECTOR> <SEP>
```

```
                    <I:INDEX_1> <SEP> <I:INDEX_2> <SEP>
                    <RLIST:Ra_RANGE_BOUNDARIES> <SEP>
                    <RLISTS:Rb_RANGE_BOUNDARIES> <SEP>
                    <I:COLOUR_TYPE> <SEP> <COLRVLISTS:COLOUR_LISTS>

DATASRFLST    ::=   <(> <DATASURF,..,DATASURF> <)>

DATASURF      ::=   <(> <I:U_ORDER> <SEP> <I:V_ORDER> <SEP>
                    <RLIST:U_KNOTS> <SEP> <RLIST:V_KNOTS> <SEP>
                    <RATIONAL | NONRATIONAL> <SEP> <RLISTSLIST:CONTROL_POINTS> <)>
                    {Each RLIST is a single control point. Each RLISTS contains
                    control points along the u dimension.}

EDGEDTFLAG    ::=   <NONE | EDGE_VISIBILITY_FLAGS>

EDGEFLAG      ::=   <OFF | ON>

EDGEFLAG2     ::=   <(> <EDGEFLAG> <SEP> <EDGEFLAG> <)>

EDGEFLAG3     ::=   <(> <EDGEFLAG> <SEP> <EDGEFLAG> <SEP> <EDGEFLAG> <)>

EFLIST        ::=   <(> <EDGEFLAG,..,EDGEFLAG>o <)>

EFLISTS       ::=   <(> <EFLIST,..,EFLIST>o <)>

EFLISTSLIST   ::=   <(> <EFLISTS,..,EFLISTS>o <)>

EF2ROW        ::=   <(> <EDGEFLAG2,..,EDGEFLAG2>o <)>

EF3LIST       ::=   <(> <EDGEFLAG3,..,EDGEFLAG3>o <)>

FACETDATA     ::=   <(> <COLRV>o
                    < <SEP> <V3:UNIT_NORMAL> >o
                    < <SEP> <RLIST:DATA_MAP_DATA> >o
                    <)>

FASETLIST2    ::=   < <(> <POINTLIST2> <)>
                    < <OPTSEP> <(> < POINTLIST2> <)> >*
                    >o

FASETLIST3    ::=   < <(> <POINTLIST3> <)>
                    < <OPTSEP> <(> <POINTLIST3> <)> >*
                    >o

FDFLAG        ::=   < NONE | FACET_COLR | FACET_NORMAL | FACET_DATA |
                    FACET_COLR_NORMAL | FACET_COLR_DATA |
                    FACET_NORMAL_DATA | FACET_COLR_NORMAL_DATA >

FDLIST        ::=   <(> <FACETDATA,..,FACETDATA>o <)>
```

FDROW ::= <FDLIST>

GCOLR ::= <I:COLOUR_TYPE> <SEP> <COLRV>

HIGHLDR ::= <(>
<
<HIGHLDR2> | <HIGHLDR3> | <HIGHLDR4> | <HIGHLDR5> |
<S:TYPE_DEPENDENT_DATA>
>o
<)>

HIGHLDR2 ::= <R:BLINK_RATE>

HIGHLDR3 ::= <I:COLOUR_INDEX>

HIGHLDR4 ::= <GCOLR:COLOUR>

HIGHLDR5 ::= <GCOLR:BLINKING_COLOUR>

INTLIST ::= <(> <I,..,I>o <)>

INTLISTS ::= <(> <INTLIST,..,INTLIST>o <) >

INTLISTSLIST ::= <(> <INTLISTS,..,INTLISTS>o <)>

INT3 ::= <(> <I> <SEP> <I> <SEP> <I> <)>

INT3LIST ::= <(> <INT3,..,INT3>o <)>

INT3LISTS ::= <(> <INT3LIST,..,INT3LIST>o <)>

MATRIX2D ::= <ROW2D> <SEP> <ROW2D> <SEP> <ROW2D>

MATRIX3D ::= <ROW3D> <SEP> <ROW3D> <SEP> <ROW3D> <SEP> <ROW3D>

PARSURFDR ::= <(>
< <PARSURFDR3> | <PARSURFDR4> | <S:TYPE_DEPENDENT_DATA> >o
<)>

PARSURFDR3 ::= <UNIFORM | NONUNIFORM> <SEP> <I:U_COUNT> <SEP> <I:V_COUNT>

PARSURFDR4 ::= <P3:ORIGIN_POINT> <SEP> <V3:DIRECTION_VECTOR> <SEP>
<RLIST:PARAMETERS>

POINTLIST2 ::= <P2:POINT,..,P2:POINT>o

POINTLIST3 ::= <P3:POINT,..,P3:POINT>o

POINTLIST4 ::= <P4:POINT,..,P4:POINT>o

```
POINT2          ::=   <COORD> <SEP> <COORD>

POINT3          ::=   <COORD> <SEP> <COORD> <SEP> <COORD>

POINT4          ::=   <COORD> <SEP> <COORD> <SEP> <COORD> <SEP> <COORD>

P2              ::=   <POINT2> | <(> <POINT2> <)>

P3              ::=   <POINT3> | <(> <POINT3> <)>

P4              ::=   <POINT4> | <(> <POINT4> <)>

REFLDR          ::=   <(> < <REFLDR1> | <S:METHOD_DEPENDENT_DATA> >o <)>

REFLDR1         ::=   <R:AMBIENT_REFLECTION_COEFFICIENT> <SEP>
                      <R:DIFFUSE_REFLECTION_COEFFICIENT> <SEP>
                      <R:SPECULAR_REFLECTION_COEFFICIENT> <SEP>
                      <GCOLR:SPECULAR_COLOUR> <SEP>
                      <R:SPECULAR_EXPONENT>

RELATION        ::=   <IS_EQUAL | IS_NOT_EQUAL | IS_GREATER | IS_LESS |
                      IS_GREATER_OR_EQUAL | IS_LESS_OR_EQUAL >

RLIST           ::=   <(> <R,..,R>o <)>

RLISTS          ::=   <(> <RLIST,..,RLIST>o <)>

RLISTSLIST      ::=   <(> <RLISTS,..,RLISTS>o <)>

ROW2D           ::=   <(> <R:1ST ELEMENT IN ROW> <SEP> <R:2ND ELEMENT IN ROW> <SEP>
                      <R:LAST ELEMENT IN ROW> <)>

ROW3D           ::=   <(> <R:1ST ELEMENT IN ROW> <SEP> <R:2ND ELEMENT IN ROW> <SEP>
                      <R:3RD ELEMENT IN ROW> <SEP> <R:LAST ELEMENT IN ROW> <)>

SIZE2           ::=   <COORD> <SEP> <COORD>

SRCSELEC        ::=   <COLOUR_ASPECT | VERTEX_COLOUR | VERTEX_DATA |
                      FACET_COLOUR | FACET_DATA>

SRCSELECLIST    ::=   <(> <SRCSELEC,..,SRCSELEC>o <)>

SURFDR          ::=   <(>
                      <<I:U_COUNT> <SEP> <I:V_COUNT>
                      | <R:APPROXIMATION_VALUE>
                      | <R:U_APPROXIMATION_VALUE> <SEP> <R:V_APPROXIMATION_VALUE>
                      | <S:TYPE_DEPENDENT_DATA>>o
                      <)>

S2              ::=   <SIZE2> | <(> <SIZE2> <)>
```

TMDR ::= <(> < <TMDR3> | <TMDR4> | <TMDR5> | <TMDR6> | <TMDR7> | <TMDR8> | <TMDR9> | <TMDR10> | <TMDR11> | <S:TYPE_DEPENDENT_DATA> >0 <)>

TMDR3 ::= <COORDLIST>{six items} <SEP> <RELATION> <SEP> <R:THRESHOLD>

TMDR4 ::= <COORDLIST> {4 items} <SEP> <RELATION> <SEP> <R:THRESHOLD>

TMDR5 ::= <VTD3LISTS> <SEP> <IS_INSIDE | IS_PARTIALLY_INSIDE | IS_OUTSIDE>

TMDR6 ::= <VTD2LISTS> <SEP> <IS_INSIDE | IS_PARTIALLY_INSIDE | IS_OUTSIDE >

TMDR7 ::= <I:FILTER_TYPE> <SEP>
<I:FILTER_SELECTION> <SEP> <FILTER_FAILS | FILTER_PASSES>

TMDR8 ::= <I:INTEGER_SELECTOR> <SEP> <RELATION> <SEP> <I:VALUE>

TMDR9 ::= <R:REAL _SELECTOR> <SEP> <RELATION> <SEP> <R:VALUE>

TMDR10 ::= <ALL_ENABLED | ALL_DISABLED | NOT_ALL_DISABLED | NOT_ALL_ENABLED | ALWAYS_PASS | NEVER_PASS >

TMDR11 ::= <I:INTEGER_SELECTOR> <SEP>
<BITWISE_AND | BITWISE_OR|BITWISE_XOR> <SEP> <I:VALUE>

TRIM ::= <(> <TRIMCURVE,..,TRIMCURVE> <)>

TRIMCURVE ::= <(>
<I:APPROXIMATION_TYPE> <SEP>
<CURVEDR:DATA_RECORD>
<EDGEFLAG:EDGE_FLAG> <SEP>
<I:ORDER> <SEP>
<RLIST:KNOTS> <SEP>
<R:TMIN> <SEP> <R:TMAX> <SEP>
<
<RATIONAL <SEP> <POINTLIST3:CONTROL_POINTS>>
| <NONRATIONAL<SEP> <POINTLIST2:CONTROL_POINTS>>
>
<)>

TRIMLIST ::= <(> <TRIM,..,TRIM> <)>

VTDFLAG ::= < COORD | COORD_COLR | COORD_NORMAL
| COORD_DATA | COORD_COLR_NORMAL | COORD_COLR_DATA
| COORD_NORMAL_DATA | COORD_COLR_NORMAL_DATA >

VTDFLAGP ::= < COORD | COORD_COLR >

VTDPL3 ::= <(> <P3:POINT> <<SEP> <COLRV>>0 <)>

VTDPL3LIST ::= <(> <VTDPL3,..,VTDPL3>o <)>

VTDPL3LISTS ::= <(> <VTDPL3LIST,..,VTDPL3LIST>o <)>

VTD2 ::= <(> <P2:POINT> <<SEP> <COLRV>>o
<<SEP> <V3:UNIT_NORMAL>>o
<<SEP> <RLIST:DATA_MAP_DATA>>o
<)>

VTD2LIST ::= <(> <VTD2,..,VTD2>o <)>

VTD2LISTS ::= <(> <VTD2LIST,..,VTD2LIST>o <)>

VTD3 ""= <(> <P3:POINT>
<<SEP> <COLRV>>o
<<SEP> <V3:UNIT_NORMAL>>o
<<SEP> <RLIST:DATA_MAPDATA>>o
<)>

VTD3LIST ::= <(> <VTD3,..,VTD3>o <)>

VTD3LISTS ::= <(> <VTD3LIST,..,VTD3LIST>o <)>

V2 ::= <P2>

V3 ::= <P3>

Note: Each COLRVLIST contains control points along the *u* dimension.

Note: Each RLIST is a single control point. Each RLISTS contain control points along the *u* dimension.

4.2.5 Forming archiving file element names

4.2.5.1 Terms deleted

FACTOR FROM SCALE FACTOR SET (verb usage) TO OF WITH
3 (if unambiguous)

4.2.5.2 Words added

None

4.2.5.3 Words used unabbreviated

3 (if ambiguous otherwise)

ADD	ALPHA	AND	ARC	AREA	ARRAY
BACK	BASE	BOTTOM	BUNDLED	CELL	CIRCLE
CIRCULAR	CLOSE	CUE	CURVE	DATA	DEPTH
DOWN	EDGE	EDGETYPE	EDGEWIDTH	ELEMENTS	ELLIPSE
ELLIPTICAL	EMPTY	END	EXECUTE	FILL	FLAG
FLAGS	FONT	FRONT	HALF	HEIGHT	HLHSR
HOLLOW	INDEX	INSTANCE	LABEL	LEFT	LIGHT
LINETYPE	LINEWIDTH	MARKER	MESH	MODE	MODEL
NO	NORMAL	OFF	OFFSET	ON	PATH
PICK	PLANE	PLUS	POP	PUSH	REAL
REMOVE	REPLACE	RESOURCE	RIGHT	SET (noun usage)	SETS
SIZE	SKIP	SOLID	SOURCE	STATE	STRING
STRIP	STROKE	STYLE	TESTS	TEXT	TEXTURE
TEXTURESTOP	TYPE	UP	VERSION	VIEW	

4.2.5.4 Abbreviations

ACTIVE	→	ACTIV
ADAPTABILITY	→	ADAPT
ALIGNMENT	→	ALIGN
ANNOTATION	→	ANNO
APPLICATION	→	APPL
APPROXIMATION	→	APPROX
BEGIN	→	BEG
CAPITAL	→	CAP
CENTRE	→	CTR
CHARACTER	→	CHAR
CHARACTERISTICS	→	CHAR
CLIPPING	→	CLIP
COLOUR	→	COLR
CONDITION	→	COND
CONDITIONAL	→	COND
CONTINUITY	→	CONT
CORRECTION	→	CORR
CRITERIA	→	CRIT
CULLING	→	CULL
DESCRIPTION	→	DESC
DISTINGUISHING	→	DIST
EXPANSION	→	EXPAN
FREQUENCY	→	FREQ
GLOBAL	→	G
HEURISTICS	→	HEUR
HIGHLIGHTING	→	HIGH
HORIZONTAL	→	HOR
IDENTIFIER	→	ID
INDICATOR	→	IND
INDIVIDUAL	→	INDIV

INTEGER	→	INT
INTERIOR	→	INT
LOCAL	→	L
MAPPING	→	MAP
MAXIMUM	→	MAX
METHOD	→	METH
MINIMUM	→	MIN
MODELLING	→	MODEL
NAMES	→	NAME
NORMAL	→	NORM
OPTIMIZATION	→	OPT
PARAMETRIC	→	PAR
PATTERN	→	PAT
PERSPECTIVE	→	PERSP
POINT	→	PT
POLYLINE	→	LINE
POLYMARKER	→	MARKER
POSTCONCATENATE	→	POSTCONCAT
PRECISION	→	PREC
PRECONCATENATE	→	PRECONCAT
PROPERTIES	→	PROPS
QUADRILATERAL	→	QUAD
REFERENCE	→	REF
REFLECTANCE	→	REFL
RELATIVE	→	REL
RENDERING	→	REND
RESTORE	→	RES
RETURN	→	RET
SAMPLE	→	SAMP
SAMPLING	→	SAMP
SELECTION	→	SELEC
SELECTOR	→	SELECT
SHADING	→	SHAD
SOURCE	→	SRC
SPACING	→	SPACE
STRUCTURE	→	STRUCT
SURFACE	→	SURF
TRANSFORMATION	→	TRAN
TRANSPARENCY	→	TRANSP
TRIANGLE	→	TRI
VECTOR	→	VEC
VECTORS	→	VEC
VERTEX	→	VT
VERTICAL	→	VERT
VOLUME	→	VOL

4.2.5.5 Abbreviating compound terms

ARCHIVE FILE	→	ARF
ASPECT SOURCE FLAG	→	ASF

CELL ARRAY	→	CELLARRAY
DATA RECORD	→	DR
FILL AREA	→	FILLAREA
FILL AREA SET	→	FILLAREASET
GENERALIZED DRAWING PRIMITIVE	→	GDP
GENERALIZED STRUCTURE ELEMENT	→	GSE
NON-UNIFORM B-SPLINE	→	NURB

4.2.5.6 Sentinel character sequence

In order to distinguish PHIGS archive file elements the sentinel character sequence ''ARF_'' shall precede each element identifier.

4.2.5.7 The derived archive file element names

ADD NAMES TO SET	→	ARF_ADDNAMESET
ANNOTATION TEXT RELATIVE	→	ARF_ANNOTEXTREL
ANNOTATION TEXT RELATIVE 3	→	ARF_ANNOTEXTREL3
APPLICATION DATA	→	ARF_APPLDATA
ARCHIVE FILE DESCRIPTION	→	ARF_ARFDESC
ARCHIVE FILE VERSION	→	ARF_ARFVERSION
BEGIN ARCHIVE FILE	→	ARF_BEGARF
BEGIN STRUCTURE	→	ARF_BEGSTRUCT
CELL ARRAY	→	ARF_CELLARRAY
CELL ARRAY 3	→	ARF_CELLARRAY3
CELL ARRAY 3 PLUS	→	ARF_CELLARRAY3PLUS
CIRCLE	→	ARF_CIRCLE
CIRCLE 3	→	ARF_CIRCLE3
CIRCULAR ARC	→	ARF_CIRCULARARC
CIRCULAR ARC 3	→	ARF_CIRCULARARC3
CIRCULAR ARC CLOSE	→	ARF_CIRCULARARCCLOSE
CIRCULAR ARC CLOSE 3	→	ARF_CIRCULARARCCLOSE3
CONDITIONAL EXECUTE STRUCTURE	→	ARF_CONDEXECUTESTRUCT
CONDITIONAL INSTANCE STRUCTURE	→	ARF_CONDINSTANCESTRUCT
CONDITIONAL RETURN	→	ARF_CONDRET
CONDITIONAL SKIP ELEMENTS	→	ARF_CONDSKIPELEMENTS
CONDITIONAL SKIP TO LABEL	→	ARF_CONDSKIPLABEL
ELLIPSE	→	ARF_ELLIPSE
ELLIPSE 3	→	ARF_ELLIPSE3
ELLIPTICAL ARC	→	ARF_ELLIPTICALARC
ELLIPTICAL ARC 3	→	ARF_ELLIPTICALARC3
ELLIPTICAL ARC CLOSE	→	ARF_ELLIPTICALARCCLOSE
ELLIPTICAL ARC CLOSE 3	→	ARF_ELLIPTICALARCCLOSE3
END ARCHIVE FILE	→	ARF_ENDARF
END STRUCTURE	→	ARF_ENDSTRUCT
EXECUTE STRUCTURE	→	ARF_EXECUTESTRUCT
FILL AREA	→	ARF_FILLAREA
FILL AREA SET	→	ARF_FILLAREASET
FILL AREA SET WITH DATA	→	ARF_FILLAREASETDATA
FILL AREA SET 3	→	ARF_FILLAREASET3

Clear-text encoding format		Archive file format
FILL AREA SET 3 WITH DATA	→	ARF_FILLAREASET3DATA
FILL AREA 3	→	ARF_FILLAREA3
FILL CIRCLE	→	ARF_FILLCIRCLE
FILL CIRCLE 3	→	ARF_FILLCIRCLE3
FILL ELLIPSE	→	ARF_FILLELLIPSE
FILL ELLIPSE 3	→	ARF_FILLELLIPSE3
GENERALIZED DRAWING PRIMITIVE	→	ARF_GDP
GENERALIZED DRAWING PRIMITIVE 3	→	ARF_GDP3
GENERALIZED STRUCTURE ELEMENT	→	ARF_GSE
INSTANCE STRUCTURE	→	ARF_INSTANCESTRUCT
LABEL	→	ARF_LABEL
NON-UNIFORM B-SPLINE CURVE 3	→	ARF_NURBCURVE
NON-UNIFORM B-SPLINE CURVE 3 WITH COLOUR	→	ARF_NURBCURVECOLR
NON-UNIFORM B-SPLINE SURFACE 3	→	ARF_NURBSURF
NON-UNIFORM B-SPLINE SURFACE 3 WITH DATA	→	ARF_NURBSURFDATA
POLYLINE	→	ARF_LINE
POLYLINE SET 3 WITH COLOUR	→	ARF_LINESET3COLR
POLYLINE 3	→	ARF_LINE3
POLYMARKER	→	ARF_MARKER
POLYMARKER 3	→	ARF_MARKER3
POP STATE	→	ARF_POPSTATE
PUSH STATE	→	ARF_PUSHSTATE
QUADRILATERAL MESH WITH DATA	→	ARF_QUADMESHDATA
QUADRILATERAL MESH 3 WITH DATA	→	ARF_QUADMESH3DATA
REMOVE NAMES FROM SET	→	ARF_REMOVENAMESET
RESTORE MODELLING CLIPPING VOLUME	→	ARF_RESMODELCLIPVOL
SET ACTIVE TEXTURES	→	ARF_ACTIV TEXTURES
SET ALPHA DATA SELECTION INDEX	→	ARF_ALPHADATASELECINDEX
SET ALPHA SOURCE SELECTOR	→	ARF_ALPHASOURCESELECT
SET ANNOTATION STYLE	→	ARF_ANNOSTYLE
SET ANNOTATION TEXT ALIGNMENT	→	ARF_ANNOTEXTALIGN
SET ANNOTATION TEXT CHARACTER HEIGHT	→	ARF_ANNOTEXTCHARHEIGHT
SET ANNOTATION TEXT CHARACTER UP VECTOR	→	ARF_ANNOTEXTCHARUPVEC
SET ANNOTATION TEXT PATH	→	ARF_ANNOTEXTPATH
SET APPLICATION INTEGER	→	ARF_APPLINT
SET APPLICATION REAL	→	ARF_APPLREAL
SET BACK ACTIVE TEXTURES	→	ARF_BACKACTIV TEXTURES
SET BACK DATA MAPPING INDEX	→	ARF_BACKDATAMAPINDEX
SET BACK DATA MAPPING METHOD	→	ARF_BACKDATAMAPMETH
SET BACK INTERIOR COLOUR	→	ARF_BACKINTCOLR
SET BACK INTERIOR INDEX	→	ARF_BACKINTINDEX
SET BACK INTERIOR		

Archive file format		Clear-text encoding format
SHADING METHOD	→	ARF_BACKINTSHADMETH
SET BACK INTERIOR STYLE	→	ARF_BACKINTSTYLE
SET BACK INTERIOR STYLE INDEX	→	ARF_BACKINTSTYLEINDEX
SET BACK REFLECTANCE INDEX	→	ARF_BACKREFLINDEX
SET BACK RELECTANCE MODEL	→	ARF_BACKREFLMODEL
SET BACK REFLECTANCE PROPERTIES	→	ARF_BACKREFLPROPS
SET BACK TRANSPARENCY	→	ARF_BACKTRANSP
SET CHARACTER EXPANSION FACTOR	→	ARF_CHAREXPAN
SET CHARACTER HEIGHT	→	ARF_CHARHEIGHT
SET CHARACTER SPACING	→	ARF_CHARSPACE
SET CHARACTER UP VECTOR	→	ARF_CHARUPVEC
SET COLOUR MAPPING INDEX	→	ARF_COLRMAPINDEX
SET CONDITION FLAGS	→	ARF_CONDFLAGS
SET CONDITION FLAGS FROM TESTS	→	ARF_CONDFLAGSTESTS
SET CURVE APPROXIMATION CRITERIA	→	ARF_CURVEAPPROXCRIT
SET DATA MAPPING INDEX	→	ARF_DATAMAPINDEX
SET DATA MAPPING METHOD	→	ARF_DATAMAPMETH
SET DEPTH CUE INDEX	→	ARF_DEPTHCUEINDEX
SET EDGE COLOUR	→	ARF_EDGECOLR
SET EDGE COLOUR INDEX	→	ARF_EDGECOLRINDEX
SET EDGE FLAG	→	ARF_EDGEFLAG
SET EDGE INDEX	→	ARF_EDGEINDEX
SET EDGETYPE	→	ARF_EDGETYPE
SET EDGETYPE ADAPTABILITY	→	ARF_EDGETYPEADAPT
SET EDGECAP	→	ARF_EDGECAP
SET EDGEJOIN	→	ARF_EDGEJOIN
SET EDGEMITRE LIMIT	→	ARF_EDGEMITRELIMIT
SET EDGETYPE CONTINUITY	→	ARF_EDGETYPECONT
SET EDGETYPE OFFSET	→	ARF_EDGETYPEOFFSET
SET EDGEWIDTH SCALE FACTOR	→	ARF_EDGEWIDTH
SET FACET CULLING MODE	→	ARF_FACETCULLMODE
SET FACET DISTINGUISHING MODE	→	ARF_FACETDISTMODE
SET GLOBAL TRANSFORMATION	→	ARF_GMODELTRAN
SET GLOBAL TRANSFORMATION 3	→	ARF_GMODELTRAN3
SET HIGHLIGHTING INDEX	→	ARF_HIGHINDEX
SET HIGHLIGHTING METHOD	→	ARF_HIGHMETH
SET HLHSR IDENTIFIER	→	ARF_HLHSRID
SET INDIVIDUAL ASF	→	ARF_INDIVASF
SET INTERIOR COLOUR	→	ARF_INTCOLR
SET INTERIOR COLOUR INDEX	→	ARF_INTCOLRINDEX
SET INTERIOR INDEX	→	ARF_INTINDEX

Clear-text encoding format		Archive file format
SET INTERIOR SHADING METHOD	→	ARF_INTSHADMETH
SET INTERIOR STYLE	→	ARF_INTSTYLE
SET INTERIOR STYLE INDEX	→	ARF_INTSTYLEINDEX
SET LIGHT SOURCE STATE	→	ARF_LIGHTSRCSTATE
SET LINECAP	→	ARF_LINECAP
SET LINEJOIN	→	ARF_LINEJOIN
SET LINEMITRE LIMIT	→	ARF_LINEMITRELIMIT
SET LINETYPE	→	ARF_LINETYPE
SET LINETYPE ADAPTABILITY	→	ARF_LINETYPEADAPT
SET LINETYPE CONTINUITY	→	ARF_LINETYPECONT
SET LINETYPE OFFSET	→	ARF_LINETYPEOFFSET
SET LINEWIDTH SCALE FACTOR	→	ARF_LINEWIDTH
SET LOCAL TRANSFORMATION	→	ARF_LMODELTRAN
SET LOCAL TRANSFORMATION 3	→	ARF_LMODELTRAN3
SET MARKER SIZE SCALE FACTOR	→	ARF_MARKERSIZE
SET MARKER TYPE	→	ARF_MARKERTYPE
SET MODELLING CLIPPING INDICATOR	→	ARF_MODELCLIPIND
SET MODELLING CLIPPING VOLUME	→	ARF_MODELCLIPVOL
SET MODELLING CLIPPING VOLUME 3	→	ARF_MODELCLIPVOL3
SET OF FILL AREA SETS WITH DATA	→	ARF_SETFILLAREASETSDATA
SET OF FILL AREA SETS 3 WITH DATA	→	ARF_SETFILLAREASETS3DATA
SET PARAMETRIC SURFACE CHARACTERISTICS	→	ARF_PARSURFCHAR
SET PARAMETRIC SURFACE INDEX	→	ARF_PARSURFINDEX
SET PATTERN REFERENCE POINT	→	ARF_PATREFPT
SET PATTERN REFERENCE POINT AND VECTORS	→	ARF_PATREFPTANDVEC
SET PATTERN SIZE	→	ARF_PATSIZE
SET PICK IDENTIFIER	→	ARF_PICKID
SET POLYLINE COLOUR	→	ARF_LINECOLR
SET POLYLINE COLOUR INDEX	→	ARF_LINECOLRINDEX
SET POLYLINE INDEX	→	ARF_LINEINDEX
SET POLYLINE SHADING METHOD	→	ARF_LINESHADMETH
SET POLYMARKER COLOUR	→	ARF_MARKERCOLR
SET POLYMARKER COLOUR INDEX	→	ARF_MARKERCOLRINDEX
SET POLYMARKER INDEX	→	ARF_MARKERINDEX
SET REFLECTANCE INDEX	→	ARF_REFLINDEX
SET REFLECTANCE MODEL	→	ARF_REFLMODEL
SET REFLECTANCE PROPERTIES	→	ARF_REFLPROPS
SET RENDERING COLOUR MODEL	→	ARF_RENDCOLRMODEL
SET SURFACE APPROXIMATION CRITERIA	→	ARF_SURFAPPROXCRIT
SET TEXT ALIGNMENT	→	ARF_TEXTALIGN
SET TEXT COLOUR	→	ARF_TEXTCOLR

SET TEXT COLOUR INDEX	→	ARF_TEXTCOLRINDEX
SET TEXT FONT	→	ARF_TEXTFONT
SET TEXT INDEX	→	ARF_TEXTINDEX
SET TEXT PATH	→	ARF_TEXTPATH
SET TEXT PRECISION	→	ARF_TEXTPREC
SET TEXTURE PERSPECTIVE CORRECTION	→	ARF_TEXTUREPERSPCORR
SET TEXTURE RESOURCE OPTIMIZATION HEURISTICS	→	ARF_TEXTURERESOURCEOPTHEUR
SET TEXTURE SAMPLING FREQUENCY	→	ARF_TEXTURESAMPFREQ
SET TRANSPARENCY	→	ARF_TRANSP
SET VIEW INDEX	→	ARF_VIEWINDEX
TEXT	→	ARF_TEXT
TEXT 3	→	ARF_TEXT3
TRIANGLE SET WITH DATA	→	ARF_TRISETDATA
TRIANGLE SET 3 WITH DAT	→	ARF_TRISET3DATA
TRIANGLE STRIP WITH DATA	→	ARF_TRISTRIPDATA
TRIANGLE STRIP 3 WITH DATA	→	ARF_TRISTRIP3DATA

4.3 Encoding the PHIGS archive file elements

4.3.1 Encoding delimiter elements

BEGIN ARCHIVE FILE ::= ARF_BEGARF <OPTSEP> <S:NAME> <TERM>

BEGIN STRUCTURE ::= ARF_BEGSTRUCT <SOFTSEP>
<I:STRUCTURE_IDENTIFIER> <OPTSEP> <TERM>

END ARCHIVE FILE ::= ARF_ENDARF <TERM>

END STRUCTURE ::= ARF_ENDSTRUCT <TERM>

4.3.2 Encoding archive file descriptor elements

ARCHIVE FILE DESCRIPTION ::= ARF_ARFDESC <OPTSEP>
<S:DESCRIPTION> <TERM>

ARCHIVE FILE VERSION ::= ARF_ARFVERSION <SOFTSEP>
<I:VERSION> <TERM>

4.3.3 The structure element production

STRUCTURE ELEMENT ::= <
<ADD NAMES TO SET> |
<ANNOTATION TEXT RELATIVE> |
<ANNOTATION TEXT RELATIVE 3> |
<APPLICATION DATA> |
<CELL ARRAY> |

<CELL ARRAY 3> |
<CELL ARRAY 3 PLUS> |
<CIRCLE> |
<CIRCLE 3> |
<CIRCULAR ARC> |
<CIRCULAR ARC CLOSE> |
<CIRCULAR ARC CLOSE 3> |
<CIRCULAR ARC 3> |
<CONDITIONAL EXECUTE STRUCTURE> |
<CONDITIONAL INSTANCE STRUCTURE> |
<CONDITIONAL RETURN> |
<CONDITIONAL SKIP ELEMENTS> |
<CONDITIONAL SKIP TO LABEL> |
<ELLIPSE> |
<ELLIPSE 3> |
<ELLIPTICAL ARC> |
<ELLIPTICAL ARC CLOSE> |
<ELLIPTICAL ARC CLOSE 3> |
<ELLIPTICAL ARC 3> |
<EXECUTE STRUCTURE> |
<FILL AREA> |
<FILL AREA SET> |
<FILL AREA SET WITH DATA> |
<FILL AREA SET 3> |
<FILL AREA SET 3 WITH DATA> |
<FILL AREA 3> |
<FILL CIRCLE> |
<FILL CIRCLE 3> |
<FILL ELLIPSE> |
<FILL ELLIPSE 3> |
<GENERALIZED DRAWING PRIMITIVE> |
<GENERALIZED DRAWING PRIMITIVE 3> |
<GENERALIZED STRUCTURE ELEMENT> |
<INSTANCE STRUCTURE> |
<LABEL> |
<NON-UNIFORM B-SPLINE CURVE> |
<NON-UNIFORM B-SPLINE CURVE WITH COLOUR> |
<NON-UNIFORM B-SPLINE SURFACE> |
<NON-UNIFORM B-SPLINE SURFACE WITH DATA> |
<POLYLINE> |
<POLYLINE SET 3 WITH COLOUR> |
<POLYLINE 3> |
<POLYMARKER> |
<POLYMARKER 3> |
<POP STATE> |
<PUSH STATE> |
<QUADRILATERAL MESH WITH DATA> |
<QUADRILATERAL MESH 3 WITH DATA> |
<REMOVE NAMES FROM SET> |
<RESTORE MODELLING CLIPPING VOLUME> |

<SET ACTIVE TEXTURES> |
<SET ALPHA DATA SELECTION INDEX> |
<SET ALPHA SOURCE SELECTOR> |
<SET ANNOTATION STYLE> |
<SET ANNOTATION TEXT ALIGNMENT> |
<SET ANNOTATION TEXT CHARACTER HEIGHT> |
<SET ANNOTATION TEXT CHARACTER UP VECTOR> |
<SET ANNOTATION TEXT PATH> |
<SET APPLICATION INTEGER> |
<SET APPLICATION REAL> |
<SET BACK ACTIVE TEXTURES> |
<SET BACK DATA MAPPING INDEX> |
<SET BACK DATA MAPPING METHOD> |
<SET BACK INTERIOR COLOUR> |
<SET BACK INTERIOR INDEX> |
<SET BACK INTERIOR SHADING METHOD> |
<SET BACK INTERIOR STYLE> |
<SET BACK INTERIOR STYLE INDEX> |
<SET BACK REFLECTANCE INDEX> |
<SET BACK REFLECTANCE MODEL> |
<SET BACK REFLECTANCE PROPERTIES> |
<SET BACK TRANSPARENCY> |
<SET CHARACTER EXPANSION FACTOR> |
<SET CHARACTER HEIGHT> |
<SET CHARACTER SPACING> |
<SET CHARACTER UP VECTOR> |
<SET COLOUR MAPPING INDEX> |
<SET CONDITION FLAGS> |
<SET CONDITION FLAGS FROM TESTS> |
<SET CURVE APPROXIMATION CRITERIA> |
<SET DATA MAPPING INDEX> |
<SET DATA MAPPING METHOD> |
<SET DEPTH CUE INDEX> |
<SET EDGECAP> |
<SET EDGE COLOUR> |
<SET EDGE COLOUR INDEX> |
<SET EDGE FLAG> |
<SET EDGE INDEX> |
<SET EDGETYPE> |
<SET EDGEJOIN> |
<SET EDGEMITRE LIMIT> |
<SET EDGETYPE ADAPTABILITY> |
<SET EDGETYPE CONTINUITY> |
<SET EDGETYPE OFFSET> |
<SET EDGEWIDTH SCALE FACTOR> |
<SET FACET CULLING MODE> |
<SET FACET DISTINGUISHING MODE> |
<SET GLOBAL TRANSFORMATION> |
<SET GLOBAL TRANSFORMATION 3> |
<SET HIGHLIGHTING INDEX> |

<SET HIGHLIGHTING METHOD> |
<SET HLHSR IDENTIFIER> |
<SET INDIVIDUAL ASF> |
<SET INTERIOR COLOUR> |
<SET INTERIOR COLOUR INDEX> |
<SET INTERIOR INDEX> |
<SET INTERIOR SHADING METHOD> |
<SET INTERIOR STYLE> |
<SET INTERIOR STYLE INDEX> |
<SET LIGHT SOURCE STATE> |
<SET LINECAP> |
<SET LINEJOIN> |
<SET LINEMITRE LIMIT> |
<SET LINETYPE> |
<SET LINETYPE ADAPTABILITY> |
<SET LINETYPE CONTINUITY> |
<SET LINETYPE OFFSET> |
<SET LINEWIDTH SCALE FACTOR> |
<SET LOCAL TRANSFORMATION> |
<SET LOCAL TRANSFORMATION 3> |
<SET MARKER SIZE SCALE FACTOR> |
<SET MARKER TYPE> |
<SET MODELLING CLIPPING INDICATOR> |
<SET MODELLING CLIPPING VOLUME> |
<SET MODELLING CLIPPING VOLUME 3> |
<SET OF FILL AREA SETS WITH DATA> |
<SET OF FILL AREA SETS 3 WITH DATA> |
<SET PARAMETRIC SURFACE CHARACTERISTICS> |
<SET PARAMETRIC SURFACE INDEX> |
<SET PATTERN REFERENCE POINT> |
<SET PATTERN REFERENCE POINT AND VECTORS |
<SET PATTERN SIZE> |
<SET PICK IDENTIFIER> |
<SET POLYLINE COLOUR> |
<SET POLYLINE COLOUR INDEX> |
<SET POLYLINE INDEX> |
<SET POLYLINE SHADING METHOD> |
<SET POLYMARKER COLOUR> |
<SET POLYMARKER COLOUR INDEX> |
<SET POLYMARKER INDEX> |
<SET REFLECTANCE INDEX> |
<SET REFLECTANCE MODEL> |
<SET REFLECTANCE PROPERTIES> |
<SET RENDERING COLOUR MODEL> |
<SET SURFACE APPROXIMATION CRITERIA> |
<SET TEXT ALIGNMENT> |
<SET TEXT COLOUR> |
<SET TEXT COLOUR INDEX> |
<SET TEXT FONT> |
<SET TEXT INDEX> |

<SET TEXT PATH> |
<SET TEXT PRECISION> |
<SET TEXTURE PERSPECTIVE CORRECTION> |
<SET TEXTURE RESOURCE OPTIMIZATION HEURISTICS> |
<SET TEXTURE SAMPLING FREQUENCY> |
<SET TRANSPARENCY> |
<SET VIEW INDEX> |
<TEXT> |
<TEXT 3> |
<TRIANGLE SET WITH DATA> |
<TRIANGE SET 3 WITH DATA> |
<TRIANGLE STRIP WITH DATA> |
<TRIANGE STRIP 3 WITH DATA>
>

4.3.4 Encoding output primitive elements

ANNOTATION TEXT RELATIVE ::= ARF_ANNOTEXTREL <SOFTSEP>
<P2:REFERENCE_POINT> <SEP> <V2:ANNOTATION_OFFSET>
<OPTSEP> <S:TEXT_STRING> <TERM>

ANNOTATION TEXT RELATIVE 3 ::= ARF_ANNOTEXTREL3 <SOFTSEP>
<P3:REFERENCE_POINT> <SEP> <V3:ANNOTATION_OFFSET>
<OPTSEP> <S:TEXT_STRING> <TERM>

CELL ARRAY ::= ARF_CELLARRAY <SOFTSEP>
<P2:P_POINT> <SEP> <P2:Q_POINT>
<SEP> <I:DX> <SEP> <I:DY>
<CELLROW>* <TERM>

CELL ARRAY 3 ::= ARF_CELLARRAY3 <SOFTSEP>
<P3:P_POINT> <SEP> <P3:Q_POINT> <SEP> <P3:R_POINT>
<SEP> <I:DX> <SEP> <I:DY>
<CELLROW>* <TERM>

CELL ARRAY 3 PLUS ::= ARF_CELLARRAY3PLUS <SOFTSEP>
<P3:P_POINT> <SEP> <P3:Q_POINT> <SEP> <P3:R_POINT> <SEP>
<I:COLOUR_TYPE> <SEP>
<I:X_DIM> <SEP> <I:Y_DIM> <SEP>
<COLRVROWS> <TERM>

CIRCLE ::= ARF_CIRCLE <SOFTSEP>
<P2:CENTRE_POINT> <SEP> <R:RADIUS> <TERM>

CIRCLE 3 ::= ARF_CIRCLE3 <SOFTSEP>
<P3:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <V3:FIRST_REFERENCE_VECTOR>

<SEP> <V3:SECOND_REFERENCE_VECTOR> <TERM>

CIRCULAR ARC ::= ARF_CIRCULARARC <SOFTSEP>
<P2:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE> <TERM>

CIRCULAR ARC CLOSE ::= ARF_CIRCULARARCCLOSE <SOFTSEP>
<P2:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE>
<SEP> <PIE | SEGMENT> <TERM>

CIRCULAR ARC CLOSE 3 ::= ARF_CIRCULARARCCLOSE3 <SOFTSEP>
<P3:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <V3:FIRST_REFERENCE_VECTOR>
<SEP> <V3:SECOND_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE>
<SEP> <PIE | SEGMENT> <TERM>

CIRCULAR ARC 3 ::= ARF_CIRCULARARC3 <SOFTSEP>
<P3:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <V3:FIRST_REFERENCE_VECTOR>
<SEP> <V3:SECOND_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE> <TERM>

ELLIPSE ::= ARF_ELLIPSE <SOFTSEP>
<P2:CENTRE_POINT>
<SEP> <V2:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V2:MINOR_AXIS_REFERENCE_VECTOR> <TERM>

ELLIPSE 3 ::= ARF_ELLIPSE3 <SOFTSEP>
<P3:CENTRE_POINT>
<SEP> <V3:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V3:MINOR_AXIS_REFERENCE_VECTOR> <TERM>

ELLIPTICAL ARC ::= ARF_ELLIPTICALARC <SOFTSEP>
<P2:CENTRE_POINT>
<SEP> <V2:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V2:MINOR_AXIS_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE> <TERM>

ELLIPTICAL ARC CLOSE ::= ARF_ELLIPTICALARCCLOSE <SOFTSEP>
<P2:CENTRE_POINT>
<SEP> <V2:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V2:MINOR_AXIS_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE>
<SEP> <PIE | SEGMENT> <TERM>

ELLIPTICAL ARC CLOSE 3 ::= ARF_ELLIPTICALARCCLOSE3 <SOFTSEP>
<P3:CENTRE_POINT>
<SEP> <V3:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V3:MINOR_AXIS_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE>
<SEP> <PIE | SEGMENT> <TERM>

ELLIPTICAL ARC 3 ::= ARF_ELLIPTICALARC3 <SOFTSEP>
<P3:CENTRE_POINT>
<SEP> <V3:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V3:MINOR_AXIS_REFERENCE_VECTOR>
<SEP> <R:START_ANGLE> <SEP> <R:END_ANGLE> <TERM>

FILL AREA ::= ARF_FILLAREA <SOFTSEP> <POINTLIST2> <TERM>

FILL AREA SET ::= ARF_FILLAREASET <SOFTSEP> <FASETLIST2> <TERM>

FILL AREA SET WITH DATA ::= ARF_FILLAREASETDATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>0
<<SEP> <FACETDATA>>0
<<SEP> <EFLISTS>>0
<SEP> <VTD2LISTS> <TERM>

FILL AREA SET 3 ::= ARF_FILLAREASET3 <SOFTSEP> <FASETLIST3> <TERM>

FILL AREA SET 3 WITH DATA ::= ARF_FILLAREASET3DATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>0
<<SEP> <FACETDATA>>0
<,SEP> <EFLISTS>>0
<SEP> <VTD3LISTS> <TERM>

FILL AREA 3 ::= ARF_FILLAREA3 <SOFTSEP> <POINTLIST3> <TERM>

FILL CIRCLE ::= ARF_FILLCIRCLE <SOFTSEP>
<P2:CENTRE_POINT> <SEP> <R:RADIUS> <TERM>

FILL CIRCLE 3 ::= ARF_FILLCIRCLE3 <SOFTSEP>
<P3:CENTRE_POINT> <SEP> <R:RADIUS>
<SEP> <V3:FIRST_REFERENCE_VECTOR>
<SEP> <V3:SECOND_REFERENCE_VECTOR> <TERM>

FILL ELLIPSE ::= ARF_FILLELLIPSE <SOFTSEP>
<P2:CENTRE_POINT>
<SEP> <V2:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V2:MINOR_AXIS_REFERENCE_VECTOR> <TERM>

FILL ELLIPSE 3 ::= ARF_FILLELLIPSE3 <SOFTSEP>
<P3:CENTRE_POINT>
<SEP> <V3:MAJOR_AXIS_REFERENCE_VECTOR>
<SEP> <V3:MINOR_AXIS_REFERENCE_VECTOR> <TERM>

GENERALIZED DRAWING PRIMITIVE ::= ARF_GDP <SOFTSEP>
<I:GDP_IDENTIFIER> <SEP> <POINTLIST2>
<<OPTSEP> | <HARDSEP>> <S:GDP_DATA_RECORD> <TERM>

GENERALIZED DRAWING PRIMITIVE 3 ::= ARF_GDP3 <SOFTSEP>
<I:GDP_IDENTIFIER> <SEP> <POINTLIST3>
<<OPTSEP> | <HARDSEP>> <S:GDP_DATA_RECORD> <TERM>

NON-UNIFORM B-SPLINE CURVE 3 ::= ARF_NURBCURVE <SOFTSEP>
<I:SPLINE_ORDER> <SEP>
<RLIST:KNOTS> <SEP>
<R:TMIN> <SEP> <R:TMAX> <SEP>
<
<RATIONAL<SEP> <POINTLIST4:CONTOL_POINTS>>
| <NONRATIONAL<SEP> <POINTLIST3:CONTROL_POINTS>>
> <TERM>

NON-UNIFORM B-SPLINE CURVE 3 WITH COLOUR ::= ARF_NURBCURVECOLR <SOFTSEP>
<I:SPLINE_ORDER> <SEP>
<RLIST:KNOTS> <SEP>
<R:TMIN> <SEP> <R:TMAX> <SEP>
<
<RATIONAL<SEP> <POINTLIST4:CONTROL_POINTS>>
| <NONRATIONAL<SEP> <POINTLIST3:CONTROL_POINTS>>
>
<<SEP> <COLRCURVE>>o <TERM>

NON-UNIFORM B-SPLINE SURFACE ::= ARF_NURBSURF <SOFTSEP>
<I:U_ORDER> <SEP> <I:V_ORDER> <SEP>
<RLIST:U_KNOTS> <SEP> <RLIST:V_KNOTS> <SEP>
<
<RATIONAL<SEP>
<POINTLIST4:U_DIRECTION_CONTROL_POINTS>
<<SEP> <POINTLIST4:U_DIRECTION_CONTROL_POINTS>>*
| <NONRATIONAL<SEP>
<POINTLIST3:U_DIRECTION_CONTROL_POINTS>
<<SEP> <POINTLIST3:U_DIRECTION_CONTROL_POINTS>>*
>
<<SEP>TRIMLOOPS<SEP> <TRIMLIST>>o <TERM>

NON-UNIFORM B-SPLINE SURFACE WITH DATA ::=
ARF_NURBSURFDATA <SOFTSEP>
<I:U_ORDER> <SEP> <I:V_ORDER> <SEP>
<RLIST:U_KNOTS> <SEP> <RLIST:V_KNOTS> <SEP>
<
<RATIONAL<SEP>
<POINTLIST4:U_DIRECTION_CONTROL_POINTS>
<<SEP> <POINTLIST4:U_DIRECTION_CONTROL_POINTS>>*
<<SEP>TRIMLOOPS<SEP> <TRIMLIST>>o
<<SEP>COLRSURFH<SEP> <COLRSURFH>>o
<<SEP>DATASURFS<SEP> <DATASRFLST>>o <TERM>
| <NONRATIONAL<SEP>
<POINTLIST3:U_DIRECTION_CONTROL_POINTS>
<<SEP> <POINTLIST3:U_DIRECTION_CONTROL_POINTS>>*
<<SEP>TRIMLOOPS<SEP> <TRIMLIST>>o
<<SEP>COLRSURF<SEP> <COLRSURF>>o
<<SEP>DATASURFS<SEP> <DATASRFLST>>o <TERM>
>

POLYLINE ::=
ARF_LINE <SOFTSEP> <POINTLIST2> <TERM>

POLYLINE SET 3 WITH COLOUR ::=
ARF_LINESET3COLR <SOFTSEP> <VTDFLAGP> <SEP>
< <I:COLOUR_TYPE> <SEP> >o <VTDPL3LISTS> <TERM>

POLYLINE 3 ::=
ARF_LINE3 <SOFTSEP> <POINTLIST3> <TERM>

POLYMARKER ::=
ARF_MARKER <SOFTSEP> <POINTLIST2> <TERM>

POLYMARKER 3 ::=
ARF_MARKER3 <SOFTSEP> <POINTLIST3> <TERM>

QUADRILATERAL MESH WITH DATA ::=
ARF_QUADMESHDATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG> <SEP>
<I:NUM_ROWS> <SEP> <I:NUM_COLUMNS>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDROW>>* <<SEP> <EF2ROW>>*
<SEP> <VTD2LISTS> <TERM>

QUADRILATERAL MESH 3 WITH DATA ::=
ARF_QUADMESH3DATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG> <SEP>
<I:NUM_ROWS> <SEP> <I:NUM_COLUMNS>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDROW>>* <<SEP> <EF2ROW>>*
<SEP> <VTD3LISTS> <TERM>

SET OF FILL AREA SETS WITH DATA ::=
ARF_SETFILLAREASETSDATA <SOFTSEP>

<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDLIST>>o <<SEP> <EFLISTSLIST>>o
<SEP> <VTD2LIST> <SEP>
<INTLISTSLIST:VERTEX_INDICES> <TERM>

SET OF FILL AREA SETS 3 WITH DATA ::= ARF_SETFILLAREASETS3DATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDLIST>>o <<SEP> <EFLISTSLIST>o
<SEP> <VTD3LIST> <SEP>
<INTLISTSLIST:VERTEX_INDICES> <TERM>

TEXT ::= ARF_TEXT <SOFTSEP>
<P2:TEXT_POSITION> <OPTSEP> <S:TEXT_STRING> <TERM>

TEXT 3 ::= ARF_TEXT3 <SOFTSEP>
<P3:TEXT_POSITION>
<SEP> <V3:FIRST_TEXT_DIRECTION_VECTOR>
<SEP> <V3:SECOND_TEXT_DIRECTION_VECTOR>
<OPTSEP> <S:TEXT_STRING> <TERM>

TRIANGLE SET WITH DATA ::= ARF_TRISETDATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDLIST>>o <<SEP> <EF3LIST>o
<SEP> <VTD2LIST> <SEP>
<INT3LISTS:VERTEX_INDICES> <TERM>

TRIANGLE SET 3 WITH DATA ::= ARF_TRISET3DATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o
<<SEP> <FDLIST>>o <<SEP> <EF3LIST>o
<SEP> <VTD3LIST> <SEP>
<INT3LISTS:VERTEX_INDICES> <TERM>

TRIANGLE STRIP WITH DATA ::= ARF_TRISTRIPDATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o
<,SEP> <FDLIST>>o <<SEP> <EFLIST>>o
<SEP> <VTD2LIST>o <TERM>

TRIANGLE STRIP 3 WITH DATA ::= ARF_TRISTRIP3DATA <SOFTSEP>
<FDFLAG> <SEP> <EDGEDTFLAG> <SEP> <VTDFLAG>
<<SEP> <I:COLOUR_TYPE>>o

<<SEP> <FDLIST>>o <<SEP> <EFLIST>>o
<SEP> <VTD3LIST> <TERM>

4.3.5 Encoding attribute elements

ADD NAMES TO SET ::= ARF_ADDNAMESET <SOFTSEP>
<I:CLASS_NAME>,...,<I:CLASS_NAME>o <TERM>

REMOVE NAMES FROM SET ::= ARF_REMOVENAMESET <SOFTSEP>
<<I:CLASS_NAME>,...,<I:CLASS_NAME>o <TERM>

SET ACTIVE TEXTURES ::= ARF_ACTIV TEXTURES <SOFTSEP>
<INTLIST:LIST_OF_ACTIVE_TEXTURES> <TERM>

SET ALPHA DATA SELECTION INDEX ::= ARF_ALPHADATASELECINDEX <SOFTSEP>
<I:ALPHA_DATA_SELECTION_INDEX> <TERM>

SET ALPHA SOURCE SELECTOR ::= ARF_ALPHASOURCESELECT <SOFTSEP>
<INTLIST:ALPHA_SOURCE_SELECTOR_LIST> <TERM>

SET ANNOTATION TEXT ALIGNMENT ::= ARF_ANNOTEXTALIGN <SOFTSEP>
< NORMHOR | LEFT | CTR | RIGHT > <SEP>
< NORMVERT | TOP | CAP | HALF | BASE | BOTTOM > <TERM>

SET ANNOTATION TEXT CHARACTER HEIGHT ::= ARF_ANNOTEXTCHARHEIGHT <SOFTSEP>
<R:CHARACTER_HEIGHT> {in TLC} <TERM>

SET ANNOTATION TEXT CHARACTER UP VECTOR ::= ARF_ANNOTEXTCHARUPVEC <SOFTSEP>
<V2:CHARACTER_UP_VECTOR> {non-zero length, in TLC}
<TERM>

SET ANNOTATION TEXT PATH ::= ARF_ANNOTEXTPATH <SOFTSEP>
< RIGHT | LEFT | UP | DOWN > <TERM>

SET ANNOTATION STYLE ::= ARF_ANNOSTYLE <SOFTSEP>
<I:ANNOTATION_STYLE> <TERM>

SET BACK ACTIVE TEXTURES ::= ARF_BACKACTIV TEXTURES <SOFTSEP>
< INTLIST:LIST_OF_BACK_ACTIVE_TEXTURES > <TERM>

SET BACK DATA MAPPING INDEX ::= ARF_BACKDATAMAPINDEX <SOFTSEP>

<I:BACK_DATA_MAP_INDEX> <TERM>

SET BACK DATA MAPPING METHOD ::= ARF_BACK_DATAMAPMETH <SOFTSEP> <I:BACK_DATA_MAP_METHOD> <SEP> <DATAMAPDR:DATA_RECORD> <TERM>

SET BACK INTERIOR COLOUR ::= ARF_BACKINTCOLR <SOFTSEP> <GCOLR> <TERM>

SET BACK INTERIOR INDEX ::= ARF_BACKINTINDEX <SOFTSEP> <I:BACK_INTERIOR_INDEX> <TERM>

SET BACK INTERIOR SHADING METHOD ::= ARF_BACKINTSHADMETH <SOFTSEP> <I:SHADING_METHOD> <TERM>

SET BACK INTERIOR STYLE ::= ARF_BACKINSTYLE <SOFTSEP> <HOLLOW | SOLID | PATTERN | HATCH | EMPTY | TEXTURE> <TERM>

SET BACK INTERIOR STYLE INDEX ::= ARF_BACKINTSTYLEINDEX <SOFTSEP> <I:INTERIOR_STYLE_INDEX> <TERM>

SET BACK REFLECTANCE MODEL ::= ARF_BACKREFLMODEL <SOFTSEP> <I:REFLECTANCE_MODEL> <TERM>

SET BACK REFLECTANCE PROPERTIES ::= ARF_BACKREFLPROPS <SOFTSEP> <I:PROPERTIES_TYPE> <SEP> <REFLDR:DATA_RECORD> <TERM>

SET BACK REFLECTANCE INDEX ::= ARF_BACKREFLINDEX <SOFTSEP> <I:BACK_REFLECTANCE_INDEX> <TERM>

SET BACK TRANSPARENCY ::= ARF_BACKTRANSP <SOFTSEP> <R:BACK_TRANSPARENCY> <TERM>

SET CHARACTER EXPANSION FACTOR ::= ARF_CHAREXPAN <SOFTSEP> <R:CHARACTER_EXPANSION_FACTOR> <TERM>

SET CHARACTER HEIGHT ::= ARF_CHARHEIGHT <SOFTSEP> <R:CHARACTER_HEIGHT> {in TLC} <TERM>

SET CHARACTER SPACING	::=	ARF_CHARSPACE <SOFTSEP> <R:CHARACTER_SPACING> {in TLC} <TERM>
SET CHARACTER UP VECTOR	::=	ARF_CHARUPVEC <SOFTSEP> <V2:CHARACTER_UP_VECTOR> {non-zero length, in TLC} <TERM>
SET COLOUR MAPPING INDEX	::=	ARF_COLRMAPINDEX <SOFTSEP> <I:COLOUR_MAP_INDEX> <TERM>
SET CURVE APPROXIMATION CRITERIA	::=	ARF_CURVEAPPROXCRIT <SOFTSEP> <I:APPROXIMATION_TYPE> <SEP> <CURVEDR:DATA_RECORD> <TERM>
SET DATA MAPPING INDEX	::=	ARF_DATAMAPINDEX <SOFTSEP> <I:DATA_MAP_INDEX> <TERM>
SET DATA MAPPING METHOD	::=	ARF_DATAMAPMETH <SOFTSEP> <I:DATA_MAP_METHOD> <SEP> <DATAMAPDR:DATA_RECORD> <TERM>
SET DEPTH CUE INDEX	::=	ARF_DEPTHCUEINDEX <SOFTSEP> <I:DEPTH_CUE_INDEX> <TERM>
SET EDGECAP	::=	ARF_EDGECAP<SOFTSEP> <BUTT | ROUND | SQUARE> <TERM>
SET EDGE COLOUR	::=	ARF_EDGECOLR <SOFTSEP> <GCOLR:EDGE_COLOUR> <TERM>
SET EDGE COLOUR INDEX	::=	ARF_EDGECOLRINDEX <SOFTSEP> <I:EDGE_COLOUR_INDEX> <TERM>
SET EDGE FLAG	::=	ARF_EDGEFLAG <SOFTSEP> <EDGEFLAG> <TERM>
SET EDGE INDEX	::=	ARF_EDGEINDEX <SOFTSEP> <I:EDGE_INDEX> <TERM>
SET EDGEJOIN	::=	ARF_EDGEJOIN<SOFTSEP> <FLAT | MITRE | ROUND | BEVEL> <TERM>