



JT File Format Reference Version 8.1 Rev-D

© 2009 Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. ' Il rig(ts reserve .
JT Reference) first e ition* JT File Format version 8.1.

+ , T&- . * ' Il information containe (erein is t(e /ro/ert# of Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. +o /art of t(is /!0lication 1%(et(er in (ar co/# or electronic form2 ma# 0e re/ro !ce) store in a retrieval s#stem) or transmitt) in an# form or 0# an# means) electronic) mec(anical) / (otoco/#ing) recor ing) or ot(er%ise) %it(o!t t(e /rior %ritten /ermission of Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. Please note t(at t(e content in t(is g!i e is /rotecte !n er co/#rig(t la% even if it is not istri0!te %it(soft%are t(at incl! es an en !ser license agreement. Siemens) JT) Parasoli an Transforming t(e /rocess of innovation are registre tra emar3s or tra emar3s of Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. in t(e 4nite States an for ot(er co!ntries. , /en 6" is a registre tra emar3 or tra emar3 of S6&. ' Il ot(er tra emar3s are t(e /ro/ert# of t(eir res/ective o%ners.

T(is /!0lication an t(e information (erein are f!rnis(e ' S &S) are f!rnis(e for informational !se on!#) are s!0ect to c(ange %it(o!t notice) an s(o!l not 0e constr!e as a commitment 0# Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. Siemens Pro!ct "ifec#cle \$ anagement Soft%are &nc. . 8PR . SS"9 D&S- " ' & \$ S ' +D ' SS4 \$. S + , R . SP , +S& : & " & T9 , R " & ' : & " & T9 F , R ' +9 . RR , RS , R & + ' - - 4R ' - & . S T ; ' T \$ ' 9 ' PP . ' R & + T ; . & + F , R \$ ' T & , + ' " - , + T . + T - , + T ' & + . D & + T ; & S 64 & D .) \$ ' < . S + , = ' RR ' + T9 , F ' +9 < & + D 1 . 8PR . SS) & \$ P " & . D) , R ST ' T4T , R92 = & T ; R . SP . - T T , T ; & S P4 : " & - ' T & , +) ' +D . 8PR . SS"9 D&S- " ' & \$ S ' +9 ' +D ' " " = ' RR ' + T & . S , F \$. R - ; ' + T ' : & " & T9) F & T + . SS F , R P ' RT & - 4 " ' R P4RP , S . S) ' +D + , + & + FR & + 6 . \$. + T , F T ; & RD-P ' RT9 R & 6 ; TS.

T(is oc!ment is s!0ect to all 4nite States government la%s) reg!lations) or ers or ot(er restrictions regar ing e>/ort from t(e 4nite States of services) commo ities) soft%are) tec(nolog# or erivatives t(ereof) as s!c(la%s) reg!lations) or ers) or ot(er restrictions ma# 0e enacte) amen e or mo ifie from

time to time. +ot%it(stan ing an#(ing to t(e contrar# in t(is oc!ment) 9o! %ill not irectl# or in irectl#) se/aratel# or as /art of a s#stem) e>/ort or ree>/ort an# Siemens Pro !ct "ifec#cle \$anagement Soft%are &nc. services) commo it#) soft%are) tec(nolog# or erivatives t(ereof or /ermit t(e !se 0# or s(i/ment of same to* li2 a national or resi ent of - !0a) &ran) +ort(<orea) S! an) S#ria) or an# ot(er co!ntr# em0argoe or restricte 0# t(e 4nite States? lii2 an#one or an# entit# on t(e 4.S. Treas!r# De/artment's "ist of S/eciall# Designate +ationals an : loc3e Persons) "ist of S/eciall# Designate Terrorists or "ist of S/eciall# Designate +arcotics Traffic3ers) or t(e 4.S. -ommerce De/artment's Denie Parties "ist or t(e 4.S. -ommerce De/artment's .ntit# "ist? or liii2 an# co!ntr# or estination for %(ic(t(e 4nite States government or a 4nite States governmental agenc# reA!ires an e>/ort license or ot(er a//roval for e>/ort %it(o!t first (aving o0taine s!c(license or ot(er a//roval. 9o! ac3no%le ge an agree t(at) !nless a vali ate e>/ort license is o0taine from t(e 4nite States De/artment of -ommerce or ot(er a//lica0le a!t(orit# %(ere reA!ire) 9o! %ill not !se t(e Siemens Pro !ct "ifec#cle \$anagement Soft%are &nc. services) commo ities) soft%are) tec(nolog# or erivatives t(ereof in t(e esign) evelo/ment) /ro !ction) stoc3/iling or !se of n!clear %ea/ons) missiles) or c(emical or 0iological %ea/ons. 9o! agree to in emnif# an (ol Siemens Pro !ct "ifec#cle \$anagement Soft%are &nc. (armless from an against all claims) losses) amages an e>/enses arising o!t of or res!lting from 9o!r fail!re to com/l# %it(t(e /rovisions set fort(in t(is Section.

Contents

2.1	What's New in This Revision	15
4.1	Terms	18
4.2	Coordinate Systems	20

a e

h ev

C.2.1.1.2.E	"ig(t Set 'ttri0!te .lement.....	90
C.2.1.1.2.F	&finite "ig(t 'ttri0!te .lement.....	90
C.2.1.1.2.C	Point "ig(t 'ttri0!te .lement.....	9B
C.2.1.1.2.8	"inst#le 'ttri0!te .lement.....	9F
C.2.1.1.2.9	Pointst#le 'ttri0!te .lement.....	9C
C.2.1.1.2.10	Geometric Transform 'ttri0!te .lement.....	99
C.2.1.1.2.11	S(a er .ffects 'ttri0!te .lement.....	100
C.2.1.1.2.12	Verte> S(a er 'ttri0!te .lement.....	10B
C.2.1.1.2.1B	Fragment S(a er 'ttri0!te .lement.....	109
C.2.1.2	Pro/ert# 'tom .lements.....	110
C.2.1.2.1	:ase Pro/ert# 'tom .lement.....	110
C.2.1.2.2	String Pro/ert# 'tom .lement.....	111
C.2.1.2.B	&nteger Pro/ert# 'tom .lement.....	111
C.2.1.2.D	Floating Point Pro/ert# 'tom .lement.....	112
C.2.1.2.E	JT ,0lect Reference Pro/ert# 'tom .lement.....	11B
C.2.1.2.F	Date Pro/ert# 'tom .lement.....	11B
C.2.1.2.C	"ate "oa e Pro/ert# 'tom .lement.....	11E
C.2.1.B	Pro/ert# Ta0le.....	11F
C.2.1.B.1	+o e Pro/ert# Ta0le.....	11C
C.2.2	S(a/e " , D Segment.....	11C
C.2.2.1	S(a/e " , D .lement.....	118
C.2.2.1.1	Verte> S(a/e " , D .lement.....	118
C.2.2.1.2	Tri-Stri/ Set S(a/e " , D .lement.....	120
C.2.2.1.B	Pol#line Set S(a/e " , D .lement.....	120
C.2.2.1.D	Point Set S(a/e " , D .lement.....	121
C.2.2.1.E	Pol#gon Set S(a/e " , D .lement.....	122
C.2.2.1.F	+!ll S(a/e " , D .lement.....	12B
C.2.2.2	Primitive Set S(a/e .lement.....	12D
C.2.2.B	=ire ;arness Set S(a/e .lement.....	1BB
C.2.B	JT :-Re/ Segment.....	1D8
C.2.B.1	JT :-Re/ .lement.....	1D9
C.2.D	8T :-Re/ Segment.....	1CF
C.2.D.1	8T :-Re/ .lement.....	1CF
C.2.E	=ireframe Segment.....	1C8
C.2.E.1	=ireframe Re/ .lement.....	1C8
C.2.F	\$eta Data Segment.....	180
C.2.F.1	Pro/ert# Pro># \$eta Data .lement.....	181
C.2.F.2	P\$& \$anager \$eta Data .lement.....	18D
C.2.F.2.1	P\$& .ntities.....	18F
C.2.F.2.1.1	P\$& Dimension .ntities.....	18C
C.2.F.2.1.2	P\$& +ote .ntities.....	19F
C.2.F.2.1.B	P\$& Dat!m Feat!re S#m0ol .ntities.....	19C
C.2.F.2.1.D	P\$& Dat!m Target .ntities.....	198
C.2.F.2.1.E	P\$& Feat!re -ontrol Frame .ntities.....	198
C.2.F.2.1.F	P\$& "ine =el .ntities.....	199
C.2.F.2.1.C	P\$& S/ot =el .ntities.....	200
C.2.F.2.1.8	P\$& S!rface Finis(.ntities.....	202
C.2.F.2.1.9	P\$& \$eas!rement Point .ntities.....	20B
C.2.F.2.1.10	P\$& "ocator .ntities.....	20D

C.2.F.2.1.11	P \$ & Reference 6eometr# .ntities.....	20D
C.2.F.2.1.12	P \$ & Design 6ro!/ .ntities	20E
C.2.F.2.1.1B	P \$ & -oor inate S#stem .ntities	208
C.2.F.2.2	P \$ & 'ssociations	209
C.2.F.2.B	P \$ & 4ser 'ttri0!tes	212
C.2.F.2.D	P \$ & String Ta0le.....	21B
C.2.F.2.E	P \$ & \$o el Vie%s.....	21D
C.2.F.2.F	6eneric P \$ & .ntities.....	21C
C.2.F.2.C	P \$ & - ' D Tag Data.....	221
C.2.C	P \$ & Data Segment.....	222

8.1	Common Com\$ression Data Co""e#tion !ormats	22(
8.1.1	&ntB2 - om/resse Data Pac3et	22B
8.1.1.1	&ntB2 Pro0a0ilit# - onte>ts	22F
8.1.1.2	&ntB2 Pro0a0ilit# - onte>t Ta0le .ntr#.....	229
8.1.2	FloatFD - om/resse Data Pac3et	2B0
8.1.2.1	FloatFD Pro0a0ilit# - onte>ts.....	2B2
8.1.2.1.1	FloatFD Pro0a0ilit# - onte>t Ta0le .ntr#.....	2BB
8.1.B	Verte> :ase S(a/e - om/resse Re/ Data	2BD
1.1.1.1	"ossless - om/resse Ra% Verte> Data.....	2BE
8.1.B.1	Te>t!re -oor :in ing	2BF
8.1.B.2	"ossless - om/resse Ra% Verte> Data.....	2BF
8.1.B.B	"oss# G!antiHe Ra% Verte> Data.....	2BC
8.1.B.B.1	G!antiHe Verte> -oor 'rra#	2B8
8.1.B.B.2	G!antiHe Verte> +ormal 'rra#	2B9
8.1.B.B.B	G!antiHe Verte> Te>t!re -oor 'rra#	2D1
8.1.B.B.D	G!antiHe Verte> -olor 'rra#	2D2
8.1.D	Point G!antiHer Data.....	2DD
8.1.E	Te>t!re G!antiHer Data.....	2DE
8.1.F	-olor G!antiHer Data	2DE
8.1.C	4niform G!antiHer Data	2DC
8.1.8	- om/resse .ntit# "ist for +on-Trivial <not Vector	2D8
8.1.9	- om/resse - ontrol Point = eig(ts Data	2E1
8.1.10	- om/resse - !rve Data	2E2
8.1.10.1	+on-Trivial <not Vector +4R: S - !rve &n ices	2EE
8.1.10.2	+4R: S - !rve - ontrol Point = eig(ts.....	2EE
8.1.10.B	+4R: S - !rve - ontrol Points.....	2EF
8.1.11	- om/resse - ' D Tag Data.....	2EF
8.1.11.1	- om/resse - ' D Tag T#/e-2 Data	2E8
8.2)n#odin *" orithms.....	25'
8.2.1	4niform Data G!antiHation.....	2E9
8.2.2	:itlengt(- , D. -	2F0
8.2.B	; !ffman - , D. -	2F1
8.2.B.1	.>am/le	2F1
8.2.D	'rit(metic - , D. -	2FB

8.2.D.1>am/le	2FD
8.2.E	Deering +ormal - , D . -	2F8
8.(+ , - . Com\$ression.....	2%0
' .1	, ate/ , oadin Data.....	2%0
' .2	. it !ie"ds	2%1
' .(Reserved !ie"d	2%1
' .4	0 etadata Conventions.....	2%1
9.D.1	- ' D Pro/erties.....	2C1
9.D.1.1	ReA!ire Pro/erties.....	2C2
9.D.1.2	, /tional Pro/erties.....	2CB
9.D.2	Tessellation Pro/erties.....	2CB
9.D.B	\$ iscellaneo!s Pro/erties	2CD
' .5	, S1 *ttri2&te *##&m&"ation Semanti#s	2%5
' .6	, S1 3art Str&#t&re.....	2%5
' .%	Ran e , 4 D Node *"ternative Re\$ Se"e#tion.....	2%6

, /

Tables

Table 1* Basic Data Types.....	2F
Table 2* Composite Data Types.....	2F
Table B* Segment Types.....	BB
Table D* Object Types.....	BF
Table E* Primitive Set Primitive Data Elements.....	12C
Table F* Primitive Set Parameters Data Fields & Interrelation.....	128
Table C* Common Properties <e> and Types (eir Value) Including Formats.....	220
Table 8* 'D' Properties - Conventions.....	2C2
Table 9* 'D', Optional Properties Units.....	2CB
Table 10* Object Types & Identifiers.....	2C8
Table 11* Semantic Value - Class (a) Parameter Values.....	2C9

Figures

Figure 1* File Structure.....	29
Figure 2* File Header Data Collection.....	B0
Figure B* T, - Segment Data Collection.....	B1
Figure D* T, - Units Data Collection.....	B2
Figure E* Data Segment Data Collection.....	BB
Figure F* Segment Header Data Collection.....	BB
Figure C* Data Data Collection.....	BE
Figure 8* Element Header Data Collection.....	BE
Figure 9* Element Header "L": Data Collection.....	BC
Figure 10* "S6 Segment Data Collection.....	B9
Figure 11* Base + Element Data Collection.....	D0
Figure 12* Base + Data Data Collection.....	D1
Figure 1B* Partition + Element Data Collection.....	D2
Figure 1D* Vertex - Point Range Data Collection.....	DB
Figure 1E* Group + Element Data Collection.....	DD
Figure 1F* Group + Data Data Collection.....	DE
Figure 1C* Instance + Element Data Collection.....	DF
Figure 18* Part + Element Data Collection.....	DF
Figure 19* Set Data + Element Data Collection.....	DC
Figure 20* Set Data + Data Data Collection.....	D8
Figure 21* ", D + Element Data Collection.....	D8
Figure 22* ", D + Data Data Collection.....	D9
Figure 2B* Range ", D + Element Data Collection.....	D9
Figure 2D* Set (+ Element Data Collection.....	E1
Figure 2E* Base Set (+ Element Data Collection.....	E2
Figure 2F* Base Set (+ Data Data Collection.....	EB
Figure 2C* Vertex - Point Range Data Collection.....	EE
Figure 28* Vertex Set (+ Element Data Collection.....	EF
Figure 29* Vertex Set (+ Data Data Collection.....	EF
Figure B0* Gradientation Parameters Data Collection.....	EC
Figure B1* Tri-Stri/ Set Set (+ Element Data Collection.....	E8
Figure B2* Polygon Set Set (+ Element Data Collection.....	E9
Figure BB* Point Set Set (+ Element Data Collection.....	E9
Figure BD* Polygon Set Set (+ Element Data Collection.....	F0

Fig!re BE* + 4 " " S(a/e +o e .lement ata collection.....	F1
Fig!re BF* Primitive Set S(a/e +o e .lement ata collection.....	F2
Fig!re BC* Primitive Set G!antiHation Parameters ata collection	FB
Fig!re B8* = ire ;arness Set S(a/e +o e .lement ata collection.....	FD
Fig!re B9* :ase 'ttri0!te .lement ata collection.....	FE
Fig!re D0* :ase 'ttri0!te Data ata collection.....	FE
Fig!re D1* \$aterial 'ttri0!te .lement ata collection	FC
Fig!re D2* Te>t!re &image 'ttri0!te .lement ata collection	C1
Fig!re DB* Te>t!re Vers-1 Data ata collection.....	C2
Fig!re DD* Vers-1 &image Format Descri/tion ata collection	CB
Fig!re DE* Vers-1 Te>t!re .nvironment ata collection	CE
Fig!re DF* Te>t!re Vers-2 Data ata collection.....	C8
Fig!re DC* Vers-2 Te>t!re .nvironment ata collection	80
Fig!re D8* Te>t!re -oor 6eneration Parameters ata collection.....	8B
Fig!re D9* &nline Te>t!re &image Data ata collection	8D
Fig!re E0* Vers-2 &image Format Descri/tion ata collection	8E
Fig!re E1* Dra% St#le 'ttri0!te .lement ata collection.....	88
Fig!re E2* "ig(t Set 'ttri0!te .lement ata collection.....	90
Fig!re EB* &nfinite "ig(t 'ttri0!te .lement ata collection	91
Fig!re ED* :ase "ig(t Data ata collection	92
Fig!re EE* Point "ig(t 'ttri0!te .lement ata collection.....	9D
Fig!re EF* S/rea 'ngle val!e %it(res/ect to t(e lig(t cone	9E
Fig!re EC* 'tten!ation -oefficients ata collection.....	9F
Fig!re E8* "inest#le 'ttri0!te .lement ata collection.....	9C
Fig!re E9* Pointst#le 'ttri0!te .lement ata collection.....	98
Fig!re F0* 6ometric Transform 'ttri0!te .lement ata collection	99
Fig!re F1* S(a er .ffects 'ttri0!te .lement ata collection	101
Fig!re F2* Verte>S(a er 'ttri0!te .lement ata collection.....	10B
Fig!re FB* :ase S(a er Data ata collection.....	10D
Fig!re FD* S(a er Parameter ata collection.....	10F
Fig!re FE* Fragment S(a er 'ttri0!te .lement ata collection	109
Fig!re FF* :ase Pro/ert# 'tom .lement ata collection.....	110
Fig!re FC* :ase Pro/ert# 'tom Data ata collection	110
Fig!re F8* String Pro/ert# 'tom .lement ata collection.....	111
Fig!re F9* &ninteger Pro/ert# 'tom .lement ata collection.....	112
Fig!re C0* Floating Point Pro/ert# 'tom .lement ata collection.....	112
Fig!re C1* JT ,0ject Reference Pro/ert# 'tom .lement ata collection.....	11B
Fig!re C2* Date Pro/ert# 'tom .lement ata collection.....	11D
Fig!re CB* "ate "oa e Pro/ert# 'tom .lement ata collection	11E
Fig!re CD* Pro/ert# Ta0le ata collection	11F
Fig!re CE* +o e Pro/ert# Ta0le ata collection.....	11C
Fig!re CF* S(a/e " , D Segment ata collection	118
Fig!re CC* Verte>S(a/e " , D .lement ata collection	118
Fig!re C8* Verte>S(a/e " , D Data ata collection.....	119
Fig!re C9* Tri-Stri/ Set S(a/e " , D .lement ata collection	120
Fig!re 80* Pol#line Set S(a/e " , D .lement ata collection.....	121
Fig!re 81* Point Set S(a/e " , D .lement ata collection.....	122
Fig!re 82* Pol#gon Set S(a/e " , D .lement ata collection.....	12B
Fig!re 8B* + !ll S(a/e " , D .lement ata collection	12D
Fig!re 8D* Primitive Set S(a/e .lement ata collection	12E

Fig!re 8E* "ossless -om/resse Primitive Set Data ata collection.....	12F
Fig!re 8F* "oss# G!antiHe Primitive Set Data ata collection.....	129
Fig!re 8C* -om/resse /arams1 ata collection.....	1B1
Fig!re 88* = ire ;arness Set S(a/e .lement ata collection.....	1BB
Fig!re 89* = ire ;arness Set ata collection.....	1BE
Fig!re 90* .ntit# -o!nts ata collection.....	1BF
Fig!re 91* To/ological .ntities ata collection.....	1BC
Fig!re 92* ;arness ata collection.....	1B8
Fig!re 9B* : !n le ata collection.....	1B9
Fig!re 9D* = ire ata collection.....	1D1
Fig!re 9E* = ire Segment ata collection.....	1DB
Fig!re 9F* :ranc(+o e ata collection.....	1DB
Fig!re 9C* 6eometric ata collection.....	1DD
Fig!re 98* : !n le S/ine - !rve ata collection.....	1DE
Fig!re 99* +4R: S 8 9L - !rve ata collection.....	1DF
Fig!re 100* .ntit# Tag -o!nters ata collection.....	1D8
Fig!re 101* JT : -Re/ Segment ata collection.....	1D9
Fig!re 102* JT : -Re/ .lement ata collection.....	1E0
Fig!re 10B* To/ological .ntit# -o!nts ata collection.....	1E2
Fig!re 10D* 6eometric .ntit# -o!nts ata collection.....	1EB
Fig!re 10E* To/olog# Data ata collection.....	1ED
Fig!re 10F* Regions To/olog# Data ata collection.....	1EE
Fig!re 10C* S(ells To/olog# Data ata collection.....	1EF
Fig!re 108* Trim "oo/ e>am/le in /arameter S/ace - , ne Face %it(2 ; oles.....	1EC
Fig!re 109* Faces To/olog# Data ata collection.....	1E8
Fig!re 110* "oo/s To/olog# Data ata collection.....	1E9
Fig!re 111* -o . ges To/olog# Data ata collection.....	1F0
Fig!re 112* . ges To/olog# Data ata collection.....	1F1
Fig!re 11B* Vertices To/olog# Data ata collection.....	1F2
Fig!re 11D* 6eometric Data ata collection.....	1FB
Fig!re 11E* S!rfaces 6eometric Data ata collection.....	1FE
Fig!re 11F* +on-Trivial <not Vector +4R: S S!rface &n ices ata collection.....	1FF
Fig!re 11C* +4R: S S!rface Degree ata collection.....	1FC
Fig!re 118* +4R: S S!rface -ontrol Point -o!nts ata collection.....	1FC
Fig!re 119* +4R: S S!rface -ontrol Point = eig(ts ata collection.....	1F8
Fig!re 120* +4R: S S!rface -ontrol Points ata collection.....	1F8
Fig!re 121* +4R: S S!rface <not Vectors ata collection.....	1F9
Fig!re 122* P - S - !rves 6eometric Data ata collection.....	1C0
Fig!re 12B* Trivial P - S - !rves ata collection.....	1C1
Fig!re 12D* \$ - S - !rves 6eometric Data ata collection.....	1CD
Fig!re 12E* Point 6eometric Data ata collection.....	1CD
Fig!re 12F* To/ological .ntit# Tag -o!nters ata collection.....	1CE
Fig!re 12C* : -Re/ - 'D Tag Data ata collection.....	1CF
Fig!re 128* 8T : -Re/ .lement ata collection.....	1CC
Fig!re 129* = ireframe Segment ata collection.....	1C8
Fig!re 1B0* = ireframe Re/ .lement ata collection.....	1C9
Fig!re 1B1* = ireframe \$ - S - !rves 6eometric Data ata collection.....	180
Fig!re 1B2* \$ eta Data Segment ata collection.....	181
Fig!re 1BB* Pro/ert# Pro># \$ eta Data .lement ata collection.....	182
Fig!re 1BD* Date Pro/ert# Val!e ata collection.....	18D

Fig!re 1BE* P \$ & \$ anager \$ eta Data . lement ata collection	18E
Fig!re 1BF* P \$ & . ntities ata collection	18C
Fig!re 1BC* P \$ & Dimension . ntities ata collection	18C
Fig!re 1B8* P \$ & 2D Data ata collection	188
Fig!re 1B9* P \$ & : ase Data ata collection	189
Fig!re 1D0* 2D-Reference Frame ata collection	190
Fig!re 1D1* 2D Te>t Data ata collection	191
Fig!re 1D2* Te>t : o> ata collection	192
Fig!re 1DB* - onstr!cting Te>t Pol#lines from ata arra#s	19B
Fig!re 1DD* Te>t Pol#line Data ata collection	19D
Fig!re 1DE* - onstr!cting +on-Te>t Pol#lines from /ac3e 2D ata arra#s	19E
Fig!re 1DF* +on-Te>t Pol#line Data ata collection	19E
Fig!re 1DC* P \$ & +ote . ntities ata collection	19C
Fig!re 1D8* P \$ & Dat!m Feat!re S#m0ol . ntities ata collection	198
Fig!re 1D9* P \$ & Dat!m Target . ntities ata collection	198
Fig!re 1E0* P \$ & Feat!re - ontr!l Frame . ntities ata collection	199
Fig!re 1E1* P \$ & "ine = el . ntities ata collection	199
Fig!re 1E2* P \$ & S/ot = el . ntities ata collection	200
Fig!re 1EB* P \$ & BD Data ata collection	201
Fig!re 1ED* P \$ & S!rface Finis(. ntities ata collection	202
Fig!re 1EE* P \$ & \$ eas!rement Point . ntities ata collection	20B
Fig!re 1EF* P \$ & "ocator . ntities ata collection	20D
Fig!re 1EC* P \$ & Reference 6eometr# . ntities ata collection	20E
Fig!re 1E8* P \$ & Design 6ro!/ . ntities ata collection	20F
Fig!re 1E9* Design 6ro!/ ' ttri0!te ata collection	20C
Fig!re 1F0* P \$ & - oor inate S#stem . ntities ata collection	208
Fig!re 1F1* P \$ & ' ssociations ata collection	210
Fig!re 1F2* P \$ & 4ser ' ttri0!tes ata collection	21B
Fig!re 1FB* P \$ & String Ta0le ata collection	21D
Fig!re 1FD* P \$ & \$ o el Vie%\$ ata collection	21E
Fig!re 1FE* 6eneric P \$ & . ntities ata collection	21C
Fig!re 1FF* P \$ & Pro/ert# ata collection	219
Fig!re 1FC* P \$ & Pro/ert# ' tom ata collection	221
Fig!re 1F8* P \$ & - ' D Tag Data ata collection	222
Fig!re 1F9* &ntB2 - om/resse Data Pac3et ata collection	22E
Fig!re 1C0* &ntB2 Pro0a0ilit# - onte>ts ata collection	22C
Fig!re 1C1* &ntB2 Pro0a0ilit# - onte>t Ta0le . ntr# ata collection	229
Fig!re 1C2* FloatFD - om/resse Data Pac3et ata collection	2B1
Fig!re 1CB* FloatFD Pro0a0ilit# - onte>ts ata collection	2BB
Fig!re 1CD* FloatFD Pro0a0ilit# - onte>t Ta0le . ntr# ata collection	2BB
Fig!re 1CE* Verte> : ase S(a/e - om/resse Re/ Data ata collection	2BE
Fig!re 1CF* "ossless - om/resse Ra% Verte> Data ata collection	2BF
Fig!re 1CC* "oss# G!antiHe Ra% Verte> Data ata collection	2B8
Fig!re 1C8* G!antiHe Verte> - oor ' rra# ata collection	2BF
Fig!re 1F9* G!antiHe Verte> - oor ' rra# ata collection	2BF

Fig!re 18E* 4 niform G!antiHer Data ata collection	2DC
Fig!re 18F* -om/resse . ntit# "ist for +on-Trivial <not Vector ata collection	2D9
Fig!re 18C* -om/resse - onrol Point = eig(ts Data ata collection	2E2
Fig!re 188* -om/resse - !rve Data ata collection	2EB
Fig!re 189* +on-Trivial <not Vector +4R: S - !rve &n ices ata collection	2EE
Fig!re 190* +4R: S - !rve - onrol Point = eig(ts ata collection.....	2EE
Fig!re 191* +4R: S - !rve - onrol Points ata collection	2EF
Fig!re 192* -om/resse - 'D Tag Data ata collection.....	2EC
Fig!re 19B* -om/resse - 'D Tag T#/e-2 Data ata collection	2E8
Fig!re 19D* ; !ffman Tree.....	2F2
Fig!re 19E* S/(ere ivi e into eig(t octants an octant ivi e into si> se>tants %it(eac(se>tant assigne an i entif#ing t(ree 0it co e.....	2F9
Fig!re 19F* JT Format - onvention for \$ o eling eac(Part in "S6	2CF

1 Siemens JT Data Format Reference Intellectual Property License Terms

The general idea of using an interchange format for electronic documents is in the public domain. 'n#one is free to devise a set of binary data structures and operators that define an interchange format for electronic documents. ;o#ever) Siemens Pro 'ict "ifec#cle \$anagement Soft#are &nc. o#ns t(e co/#rig(t for t(e /artic!lar data structures and operators) t(e JTM Data Format Reference and t(e %ritten s/ecification constituting t(e interchange format called t(e JT Data Format. T(!s) t(ese elements of t(e JT Data Format may not be copied without Siemens's permission.

Siemens will enforce its copyright. Siemens's intention is to maintain the integrity of t(e JT Data Format standard enabling t(e /!lic to isting!is(oet#een t(e JT Data Format and ot(er interchange formats for electronic documents. ;o#ever) Siemens desires to promote t(e use of t(e JT Data Format for information interchange among diverse /ro !cts and a//ications. 'ccor ing!#) Siemens gives an#one co/#rig(t /ermission) s!o#ect to t(e conditions stated below)*

- Pre/are and distribute files whose content conforms solely to t(e JT Data Format.
- Write and distribute software applications that /ro !ce discreetly /re/resente in t(e JT Data Format. Write and distribute software applications that accept in/!t in t(e form of t(e JT Data Format and is/la#) /rint) or ot(er%ise inter/ret t(e contents
- -o/# Siemens's co/#rig(t) list of data structures and operators in t(e %ritten s/ecification to t(e extent necessary to use t(e JT Data Format for t(e /r/oses above.
- For avoidance of doubt t(e /ermissions granted in t(e /rece ing sentences do not include t(e re/ing) %riting or distribution of files whose content contains o!t/!t in t(e JT Data Format and an# ot(er data in an# ot(er format and do not include t(e rig(t to incorporate) integrate) or combine t(e JT Data Format) structure) or schema into an# ot(er data format) structure) or schema.

The conditions of sale (copyright /ermission are*

- 'n#one who uses t(e co/#rig(t) list of data structures and operators) as stated above must include an appropriate copyright notice.

This limited right to use t(e co/#rig(t) list of data structures and operators does not include t(e right to copy t(is document) ot(er co/#rig(t) material from Siemens) or t(e software in an# of Siemens's /ro !cts that use t(e JT Data Format) in whole or in /art) nor does it include t(e right to use an# Siemens /atents e>cept as may be /ermitted by an official Siemens JT Data Format Reference Patent - larification +otice.

Siemens (JT) Parasoli) and Transforming t(e /rocess of innovation and all ot(er trademarks) service marks) and logos use of Siemens (the I\$ar3sK2 are t(e registered trademarks or trademarks of Siemens Pro 'ict "ifec#cle \$anagement Soft#are &nc. in t(e 4nite States and ot(er countries. +ot(ing in t(is 0oo3 is intended to grant no! an# right or license to use t(e \$ar3s for an# /r/ose.

Scope

This reference defines the semantics of the JT Version 8.1 file format.

The JT format is an industry standard (ISO 15926-2) file format for capturing and representing data that enables collaboration, validation and visualization throughout the enterprise. JT format is the de facto standard for BD visualization format in the automotive industry and the single most dominant BD visualization format in the aerospace industry. It is a standard for the representation of 3D models.

The JT format is a streamable and contains best-in-class compression for compact and efficient representation. The JT format is designed to

' long %it(t(e /!re s#ntactical e#inition of t(e JT Format) t(ere is also series of conventions %(ic(alt(o!g(not reA!ire to (ave a reference com/liant JT file) (ave 0ecome common/lace %(in JT format translators. T(ese conventions (ave 0een oc!mente in t(e I:est PracticesK section of t(is JT format reference.

T(is JT format reference oes not s/ecificall# a res# im/lementation of) nor e#ine) a r!n-time arc(itect!re for vie%ing an s#or /rocessing JT ata. T(is is 0eca!se alt(o!g(t(e JT format is closel# aligne %(a r!n-time ata re/resentation for fast an efficient loa ing!nloa ing of ata) no interaction 0e(avior is e#ine %(in t(e format itself) eit(er in t(e form of s/ecific vie%er controls) vie%/ort information) animation 0e(avior or ot(er event-0ase interactivit#. T(is e>cl!sion of interaction 0e(avior from t(e JT format ma3es t(e format more easil# re!sa0le for dissimilar a//lication intero/eration an also facilitates incremental !/ ate) %it(o!t losing o%nstream a!t(ore ata) as t(e original - ' D asset revises.

#.1 \$%at&s 'e(in T%is Revision

Revision D

Follo%ing is a s!mmar# of t(e !/ ates fo!n in t(is “*JT File Format Reference Version 8.1 Rev-D*” oc!ment %(res/ect to t(e /rocee ing “*JT File Format Reference Version 8.1 Rev-C*” oc!ment.

- Section [8.1.1 &ntB2 -om/resse Data Pac3et](#)
' itional information /rovi e for -o ec Io!t of 0an K sit!ation. +e% &B2 S#m0ol -o!nt varia0le a e on /age 22D
- Section [8.1.1.1 &ntB2 Pro0a0ilit# -onte>ts](#)
- (anges to +!m0er Val!e : its) +!m0er Reserve Fiel c(ange to +e>t -onte>t : its /age 22F
Fig!re 1C0 (as 0een !/ ate to reflect c(anges) /age 22E
- Section [8.1.1.1.1 &ntB2 Pro0a0ilit# -onte>t Ta0le .ntr#](#)
- (anges to +!m0er Reserve Fiel : its) no% calle +!m0er +e>t -onte>t : its) /age 228.
- Section [8.1.2 FloatFD -om/resse Data Pac3et](#)
+e% information a e for o!t of 0an sit!ation) /age 228
- Section [8.1.B.2 "oss# G!antiHe Ra% Verte> Data](#)
' itional information a e on Ra% Verte> Data /age 2BE
- Section [8.1.8 -om/resse .ntit# "ist for +on-Trivial <not Vector](#)
+e% information on internal 3not occ!rrence a e /age 2DC) ne% information a e to ta0le on 3not vector t#/es /age 2DC Q 2D8. +e% 3not vector e>am/les a e /age 2D8

Revision -

Follo%ing is a s!mmar# of t(e !/ ates fo!n in t(is “*JT File Format Reference Version 8.1 Rev-C*” oc!ment %(res/ect to t(e /rocee ing “*JT File Format Reference Version 8.1 Rev-B*” oc!ment.

- I[Appendix D Parasolid XT Format Reference](#)K (as 0een a e /er reA!ests of t(e &S, T - 18D5S - D BD Vis!aliation assessment committee.

Revision :

Following is a brief summary of the updates in this "JT File Format Reference Version 8.1 Rev-B" document (relative to the preceding "JT File Format Reference Version 8.1" document).

- Section [C.1.B.1 Segment ; ea er](#)
Segment t#/e I3 PMI Data" are to [Table B* Segment T#/es](#) and new section [C.2.C P\\$& Data Segment](#) as are to document t(e t#/e.
- Section [C.1.B.2.1 .lement ; ea er](#)
Unknown Graph Node Object in [Table D* .Object ; ase T#/es](#) as !/ ate to (ave an *Object Base Type* value of I2EEK.
- Section [C.2.1.1.1.B 6ro!/ +o e .lement](#)
&n [Figure 1E* 6ro!/ +o e .lement ata collection](#)) t(e incorrect name ata collection ICompleteK
%as rename to IGroup Node DataK.
- Section [C.2.1.1.2.B.1.1 Vers-1 &image Format Description](#)
-orrecte ata t#/e for *Width* ata fiel to Oe -(2 5 Width an *Height* ata fiel to Oe -(2 5 6ei ht.
These corrections resolve t(e iss!e of an !n oc!mente &B2 a//earing at t(e en of t(e [Vers-1 &image Format Description](#) ata collection. T(e e>tra &B2 %as !e to t(e s(ort rea s ca!se 0# t(e *Width* an *Height* fiel s Oeing misla0ele as ata t#/e &1F.
- Section [C.2.B.1.B.B Faces To/olog# Data](#)
' e [Figure 108* Trim "oo/ e>am/le in /arameter S/ace - ,ne Face %it\(2 ; oles](#) to f!rt(er clarif#
/ro/er trim loo/ efinition an orientation for Oot(Ianti-(oleK trim loo/s an I(oleK trim loo/s.
- Section [8.1.1 &ntB2 -om/resse Data Pac3et](#)
Reference to section (ere esca/e s#m0ol val!e is oc!mente %as a e .
- Section [8.1.1.2 &ntB2 Pro0a0ilit# -onte>t Ta0le .ntr#](#)
Description of esca/e s#m0ol val!e are to 7(28N&m2er Sym2o" .its9 5 Sym2o" ata fiel
escri/tion. &m/rove escri/tion for ata fiel 7(28N&m2er 4##&rren#e Co&nt .its9 5 4##&rren#e
Co&nt an ata fiel 7(28N&m2er :a"&e .its9 5 *sso#iated :a"&e.
- Section [8.1.2 FloatFD -om/resse Data Pac3et](#)
Description of esca/e s#m0ol val!e %as a e .
- Section [8.1.2.1 FloatFD Pro0a0ilit# -onte>ts](#)
Valid values for ata fiel -(2 5 3ro2a2i"ity Conte;t Ta2"e Co&nt %as correcte .
- Section [8.1.2.1.1 FloatFD Pro0a0ilit# -onte>t Ta0le .ntr#](#)
-orrecte ata t#/e for *Associated Value* ata fiel to Oe !64 5 *sso#iated :a"&e.

) References and Additional Information

R1S *JT Open Program* (<http://www.jtopen.com>) --- ' /rogram to (el/ mem0ers leverage t(e 0enefits of o/en colla0oration across t(e e>ten e enter/rise t(ro!g(t(e a o/tion of t(e JT format) a tecnolog# t(at ma3es it /ossi0le to vie% an s(are /ro !ct information t(ro!g(o!t t(e /ro !ct lifec#cle. \$ em0ers(i/ in t(e JT ,/en Program /rovi es access to t(e JT ,/en Tool3it li0rar#)

- (ic(among ot(er t(ings) /rovi es rea an %rite access to JT ata an enforces certain JT conventions to ens!re ata com/ati0ilit# %it(ot(er JT-ena0le a//lications.
- R2S *JT2Go download* [1\(tt/*55%%.jt2go.com](http://www.jt2go.com)2 --- JT26o is t(e no-c(arge BD JT vie%er from Siemens. JT26o /!ts BD ata at #o!r fingert/s 0# allo%ing an#one to o%nloa t(e no-c(arge vie%er. JT26o also allo%s an#one to em0e BD JT ata irectl# into \$icrosoft ,ffice oc!ments. JT26o offers f!! BD interactivit# on /arts) assem0lies) an even 2D ra%ings 1- 6 \$ T T&F2.
- RBS *Siemens: PLM Components: Parasolid: XT Pipeline* [1\(tt/*55%%.gs.com/resources/parasolid-5/inline.shtml](http://www.gs.com/resources/parasolid-5/inline.shtml)2 --- T(is %e0 /age /rovi es information on t(e Parasoli /recise 0o!n ar# re/resentation format 18T2 an (o% t(is 8T format fits %it(in t(e Siemens vision of seamless e>c(ange of igital /ro !ct mo els across enter/rises) 0et%een iffere#t isci/lines) !sing t(eir P" \$ a//lications of c(oice.
- RDS *OpenGL Programming Guide : the official guide to learning OpenGL Version 2) Fift(. ition)* 0# , /en6 " 'rc(itect!re Revie% :oar) Dave S(reiner) \$ason =oo) Jac3ie +ei er) an Tom Davis 1' ison =esle# 200E2 --- T(is 0oo3 gives in- e/t(e>/lanation of t(e , /en6 " S/ecification an %ill /rovi e f!rt(er insig(t into t(e significance of some of t(e ata le.g. \$aterials) Te>t!res2 t(at can e>ist in a JT file. &nformation in t(is 0oo3 ma# also serve as a gli e for (o% one co!! /rocess t(e ata containe in a JT file to /ro !ces'ren er an image on t(e screen.
- RES \$ ic(ael Deering) *Geometry Compression*) -om/!ter 6ra/(ics) Procee ings S&6 6R 'P; U9E) ' !g!st 199E) // . 1B-20.
- RFS \$ ic(ael Deering) -raig 6otsman) Stefan 6 !m(ol) Jare3 Rossignac) an 6a0riel Ta!0in) *3D Geometry Compression*) -o!rse +otes for S&6 6R 'P; 2000) J!# 2E) 2000.
- RCS *OpenGL Shading Language Specification* [1\(tt/*55%%.o/engl.org/documentation/shlang](http://www.opengl.org/documentation/shlang)2 --- , /en6 " S(a ing "ang!age 16 "S"2 as efine 0#t(e , /en6 " 'rc(itect!ral Revie% :oar) t(e governing 0o # of , /en6 ".
- R8S *Cg Toolkit Users Manual* [1\(tt/*55 evelo/er.nvia.com/object/cgUsersManual.html](http://www.nvidia.com/object/cgUsersManual.html)2 --- .>/lains ever#t(ing #o! nee to learn an !se t(e -g lang!age as %ell as t(e -g r!ntime li0rar#.
- R9S *The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics,* Ran ima Fernan o an \$ ar3 J. <ilgar) nV&D&' -or/oration) ' ison =esle# P!0lis(ing -om/an#) ' /ril 200B
- R10S <. =eiler. *Topological Structures for Geometric Modeling*) P(D t(esis) Rensselaer Pol#tec(nic &nstit!te) Tro#) + 9) 198F.
- R11S -. \$. ;offmann. *Geometric and Solid Modeling: An Introduction.* \$organ <a!fmann P!0lis(ers) &nc.) San \$ ateo) -alifornia) 1989.
- R12S *Planetmath.org - Huffman Coding* [1\(tt/*55/lanetmat\(.org/encyclo/ehuffman-coding.html](http://www.planetmath.org/encyclo/ehuffman-coding.html)2 --- T(is %e0 /age /rovi es a tec(nical overvie% of !ffman co ing %ic(is one form of ata enco ing !se %it(in t(e JT format.

- R1BS \$ ic(ael Sc(in ler) *Practical Huffman Coding*
 1([tt/*55%.com/resscons!lt.com5\(!ffman5Jenco ing2](http://55.com/resscons!lt.com5(!ffman5Jenco ing2) --- T(is %e0 /age /rovi es some co ing
 (ints for im/lementing ; !ffman co ing %(ic(is one form of ata enco ing !se %it(in t(e JT
 format.
- R1DS 6len 6. "ang on Jr.) *An Introduction to Arithmetic Coding* &: \$ Jo!nal of Researc(an
 Develo/ment) Vol!me 28) +!m0er 2) \$ arc(198D) //. 1BE-1D9.
- R1ES Pa!! 6. ;o%ar an Jeffrey# Scott Vitter) *Practical Implementation of Arithmetic Coding. Image
 and Text Compression* e . J. ' . Storer) <!%er ' ca emic P!0lis(ers) ' /ril 1992) //. 8E-112.
- R1FS Hli0.net 1([tt/*55%.Hli0.net52](http://55.com/Hli0.net52) --- T(is %e0 /age /rovi es leit(er irectl# or t(ro!g(lin3s2
 com/lete etaile information on L"&: com/ression incl! ing freA!entl# as3e A!estions)
 tec(nical oc!mentation) so!rce co e o%nl0a s) etc.

, Definitions

,.1 Terms

&t is ass!me t(at rea ers of t(is oc!ment are familiar %it(conce/ts in t(e area of com/ !ter gra/(ics an
 soli mo eling. T(e intention of t(is section is not to /rovi e com/re(ensive efinitions) 0!t is to /rovi e
 a s(ort intro !ction an clarification of t(e !sage of terms %it(in t(is oc!ment.

- ' ssem0l#
 - ' relate collection of *model* /arts) re/resente in a JT format
 logical scene gra/(as a logical gra/(Oranc(
- ' ttri0!te
 - , 0jects associate %it(no es in a *logical scene graph* an
 s/ecif#ing one of several a//earances) /ositioning) or ren ering
 c(aracteristics of a *shape*
- :o!n ar# Re/resentation
 - ' soli mo el re/resentation %(ere t(e soli vol!me is s/ecifie
 0# its s!rface 0o!n ar# !0ot(its geometric an to/ological
 0o!n aries2.
- o eTe>t
 - ' collection of ata in enco e form.
- Directe ' c#clic 6ra/(
 - ' *graph* is a set of no es) an a set of e ges connecting t(e no es
 in a tree li3e str!ct!re. ' *directed graph* is one in %(ic(ever#
 e ge (as a irection s!c(t(at e ge !!)v2) connecting no e-! %it(
 no e-v) is !fferent from e ge !v)!2. ' *Directed Acyclic Graph* is a
 irecte gra/(%it(no c#cles? %(ere a c#clic is a /at(lseA!ence of
 e ges2 from a no e to itself. So %it(a *Directed Acyclic Graph*
 t(ere is no /at(t(at can 0e follo%e %it(in t(e gra/(s!c(t(at t(e
 first no e in t(e /at(is t(e same as t(e last no e in t(e /at(.
- JT .na0le ' //lication
 - ' //lication %(ic(s!//orts rea ing an 5or %riting reference
 com/liant JT Format files.
- "evel of Detail
 - ,ne alternative gra/(ical re/resentation for some *model*

	com/onent le.g. /art2.
"ogical Scene 6ra/(– ' <i>scene graph</i> re/resenting t(e logical organiHation of a <i>model</i> . – ontains <i>shapes</i> an <i>attributes</i> re/resenting t(e <i>model's</i> /(#sical com/onents) <i>properties</i> i entif#ing ar0itar# meta ata le.g. names) semantic roles2 of t(ose com/onents) an a (ierarc(ical str!ct!re e>/ressing t(e com/onent relations(i/s.
\$i/ma/	– ' re !ce resol!tion version of a te>t!re ma/. \$i/ma/s are !se to te>t!re a geometric /rimitive %(ose screen resol!tion iffers from t(e resol!tion of t(e so!rce te>t!re ma/ originall# a//lie to t(e /rimitive.
\$o el	– Re/resentation) in JT format) of a /(#sical or virt!al /ro !ct) /art assem0l#? or collections of s!c(o0jects.
Parasoli 8T Format	– Parasoli 0o!n ar# re/resentation format
Pro !ct an \$an!fact!ring &nformation	– ollection of information create on a BD:2D - 'D \$o el to com/letel# oc!ment t(e /ro !ct %(it(res/ect to esign) man!fact!ring ins/ection) etc. T(is ma# incl! es ata s!c(as* • Dimensions !tolerances for eac(imension2 • Geometric tolerances of feat!re 1 at!ms) feat!re control frames2 • \$an!fact!ring information !s!rface finis() %el ing notations2 • &ns/ection information !3e# locations /oints2 • ' ssem0l# instr!ctions • Pro !ct information !materials) s!//liers) /art n!m0ers2
Pro/ert#	– 'n o0ject associate %(it(a logical scene gra/(no e an i entif#ing ar0itar# a//lication or enter/rise s/ecific information !meta- ata2 relate to t(at no e
G!antiHe	– onstrain somet(ing to a iscrete set of val!es) s!c(as an integer or integral m!lti/lier of a common factor) rat(er t(an a contin!o!s set of val!es) s!c(as a real n!m0er.
Scene 6ra/(– &n t(e conte>t of t(e JT format) a scene gra/(is a <i>directed acyclic graph</i> t(at arranges t(e logical an often !0!t not necessaril#2 s/atial re/resentation of a gra/(ical scene.
S(a er	– ' !ser- efina0le /rogram) e>/resse irectl# in a target assem0l# lang!age) or in (ig(-level form to 0e com/ile . ' s(a er /rogram re/laces a /ortion of t(e ot(er%ise fi>e -f!nctional!t# gra/(ics /i/eline %(it(some !ser- efine f!nction. 't /resent) (ar %are man!fact!rers (ave ma e it /ossi0le to r!n a s(a er for eac(verte> t(at is /rocesse or eac(/i>el t(at is ren ere .
Streaming	– &n t(e conte>t of t(e JT format) streaming refers to 0ot(* o "oa ing from is3 0ase me i!m onl# t(e /ortions of ata

t(at are reA!ire 0# t(e !ser to /erform t(e tas3s at (an .
T(e motivation 0eing to more efficient!# manage s#stem
memor#.

- o Transfer of ata in a stream of /ac3ets) over t(e internet on an on- eman 0asis) %(ere t(e ata is inter/rete in real-time 0# t(e a//lication as t(e ata /ac3ets arrive. T(e motivation 0eing t(at t(e !ser can 0egin !sing or interacting %it(t(e ata almost imme iatel# - no %aiting for t(e entire ata file!s2 to 0e transfere 0efore 0eginning

T(e esire en res!lt of 0ot(0eing to *deliver only the JT data that the user needs, where the user needs it, when the user needs it.* ' I7!st-in-timeK a//roac(to elivering JT format /ro !ct ata.

S(a/e - ' logical scene gra/(leaf no e containing or referencing t(e geometric s(a/e efinition ata le.g. vertices) /ol#gons) normals) etc.2 of a mo el com/onent.

Te>t!re - (annel - ' Te>t!re 4nit /!s t(e *texture environment*. &n , /en6 "O terms) Te>t!re - (annel 0asical# controls Igl ' ctiveTe>t!reK [RDS](#)

Te>t!re , 0lect - JT format meaning is t(e same as in , /en6 " [RDS](#) *IA named cache that stores texture data, such as the image array, associated mipmaps, and associated texture parameter values: width, height, border width, internal format, resolution of components, minification and magnification filters, wrapping modes, border color, and texture priority.*K

Te>t!re 4nit - JT format meaning is t(e same as in , /en6 " [RDS](#) %it(t(e connotation t(at *texture parameters* go %it(t(e Te>t!re 4nit lt(ro!g(0in ing of a te>t!re 00lect2 0!t *texture environment* lte>t!ring f!nction2 oes not.

,.# Coor*inate S!stems

T(e ata containe %it(in a JT file is efine %it(in one of t(e follo%ing coor inate s#stems. &f not ot(er%ise s/ecifie in a ata fiel !s escri/tion) it s(o!l 0e ass!me t(at t(e ata is efine in "ocal -oor inate S#stem.

- . T(e coor inate s#stem in %(ic(s(a/e geometr# is s/ecifie . &t is t(e coor inate s#stem !se to s/ecif# t(e Ira%K ata %it(no transforms a//lie .
- . "ocal coor inates transforme 0# an# transforms s/ecifie as attri0!tes at t(e no e. T(e + - S is also often referre to as \$ o el -oor inate S#stem !\$ - S2.
- . +o e coor inates transforme 0# transforms in(erite from a no el's /arent li.e. t(e coor inate s#stem at t(e root of t(e gra/(2.

- = orl coor inates transforme 0# a vie% matri>.

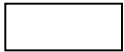
- +cron!ms an* +abbreviations

'0s	'0sol!te Val!e
: :o>	: o!n ing :o>
: -Re/	: o!n ar# Re/resentation
- ' .	- om/!ter 'i e .ngineering
-g	- for 6ra/(ics
- , D. -	- o er-Deco er
6DTT	6eometric Dimensioning an Tolerancing
6"S"	, /en6" S(a er "ang!age
6P4	6ra/(ics Processing 4nit
64&D	6lo0all# 4niA!e & entifier
; SV	; !e) Sat!ration) Val!e
; SV '	; !e) Sat!ration) Val!e) 'I/(a
" -S	"ocal -oor inate S#stem
" , D	"evel of Detail
"s0First	"east Significant : #te First
"S6	"ogical Scene 6ra/(
\$a>	\$a>im!m
\$ -S	\$o el -oor inate S#stem
\$ in	\$inim!m
\$s0First	\$ost Significant : #te First
+5'	+ot ' //lica0le
+ -S	+o e -oor inate S#stem
P -S	Parameter -oor inate S/ace
P" \$	Pro !ct "ifec#cle \$anagement
P \$ &	Pro !ct an \$an!fact!ring &nformation
R6 :	Re) 6reen) :!e
R6 : '	Re) 6reen) :!e) 'I/(a
T , -	Ta0le of -ontents.
VP -S	Vie%/oint -oor inate S#stem
4R"	4niform Reso!rce "ocator
= -S	= orl -oor inate S#stem

. 'otational Conventions

..1 Diagrams an* Fiel* Descri tions

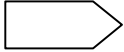
S#m0olic iagrams are !se to escri0e t(e str!ct!re of t(e JT file. T(e s#m0ols !se in t(ese iagrams (ave t(e follo%ing meaning*



Rectangles represent a data field of one of the standard data types.



Folders represent a logical collection of one or more of the standard data types. This information is provided for clarity and the basic data types that compose the groups are detailed in following sections of the document.



Rectangles with the right side corners clipped off represent information that has been composed.



Arrows convey the ordering of the information.

The format used to title the diagram symbols is dependent on the symbol type as follows:

- Diagram Rectangle: i.e. standard symbols are titled using a format of IDATAVT#* Field Name. The DATAVT# is an abbreviated symbol as defined in [F.2 Data Types](#). The field name is 1-20 characters and the DATAVT# is 1-20 characters and the field name is 1-20 characters.

-(25 Co&nt

- Diagram Folder: i.e. logical data collections symbols are titled with a collection name. The field name is 1-20 characters.

1ra\$h)"ements

- Diagram Rectangle with clipped corners: i.e. composed data fields are titled using one of the following three formats:

- Data Type followed by one of Oracle's names: i.e. Oracle's names are used to store values close Oracle's names and a colon followed by the field name. This format for titling the diagram symbol indicates that the data is composed but not enclosed. The composition is achieved by using only a portion of the total bit range of the data type to store the value. e.g. if a count value can never be larger than the value of the field, the bits are used to store all possible count values. The field name is 1-20 characters and the DATAVT# is 1-20 characters and the field name is 1-20 characters.

7 (2869 5 Co&nt

- Data Type followed by one of Oracle's names: i.e. Oracle's names are used to store values close Oracle's names and a colon followed by the field name. This format for titling the diagram symbol indicates that a vector of IDATAVT# data, i.e. *primal* values, is returned (Pre-ictor T#) close Oracle's names and a colon followed by the field name. This format for titling the diagram symbol indicates that the result of the arrangement of *residual* values is then composed and enclosed into a series of symbols using one of the three standard data types.

The following table lists the data types supported by the JT File Format.

- **Integer** - DP Q T (e Integer - DP i.e. Integer - om/resse Data Pac3et2 re/resents t(e format !se to enco e5com/ress a collection of ata into a series of Integer base s#m0ols. ' com/lete escri/tion for Integer - om/resse Data Pac3et can Oe fo!n in [8.1.1 Integer - om/resse Data Pac3et](#).
- **Float** - DP Q T (e Float - DP i.e. Float - om/resse Data Pac3et2 re/resents t(e format !se to enco e5com/ress a collection of ata into a series of Float base s#m0ols. ' com/lete escri/tion for Float - om/resse Data Pac3et can Oe fo!n in [8.1.2 Float - om/resse Data Pac3et](#).

The **Integer** - om/resse Data Pac3et t#/e is !se for com/ressingenco ing Oot(Integer and Float It(ro!g(AntiHation2 ata. = (ile t(e Float - om/resse Data Pac3et t#/e is !se for com/ressingenco ing I o!leK ata.

The **Integer** - om/resse Data Pac3et t#/e is !se for com/ressingenco ing Oot(Integer and Float It(ro!g(AntiHation2 ata. = (ile t(e Float - om/resse Data Pac3et t#/e is !se for com/ressingenco ing I o!leK ata.

: e#7 (28-nt(2CD3<, a 19 5 !irst She'''

's mentione aOove 1%it(Pre ictor T#/e algorit(m2) t(e *primal* in/!t ata val!es are +, T al%a#s %(at is enco e 5com/resse . T(is is Oeca!se t(e *primal* in/!t ata is first r!n t(ro!g(a Pre ictor T#/e algorit(m) %(ic /ro !ces an o!t/!t arra# of resi !al val!es i.e. ifferece from t(e /re icte val!e2) an t(is res!ting o!t/!t arra# of *residual* val!es is t(e ata %(ic is act!all# enco e 5com/resse . T(e JT format s!//orts several Pre ictor T#/e algorit(ms an eac(!se of Integer - DP or Float - DP s/ecifies !sing t(e aOove escriOe notation format) %(at Pre ictor T#/e algorit(m is Oeing !se on t(e ata. T(e JT format s!//orte Pre ictor T#/e algorit(ms are as follo% s Inote t(at a sam/le im/lementation of eco ing t(e /re ictor *residual* val!es Oac3 into t(e *primal* val!es can Oe fo!n in ['//en i> - *Deco ing 'lgorit\(ms Q 'n &m/lementation2*](#)

3redi#tor Ty\$e	Des#ri\$tion
"ag1	Pre icts as last val!e
"ag2	Pre icts as val!e Oefore last
Stri e1	Pre icts !sing stri e from last t%o val!es
Stri e2	Pre icts !sing stri e from val!es 2 an D Oac3
Stri/&n e>	<p>T(is is a com/lete# em/irical /re ictor. "oo3s at t(e val!es t%o Oac3 an fo!r Oac3 in t(e stream) an !ses t(e stri e Oet%een t(ese t%o val!es to /re ict t(e c!rrent val!e if an on!# if t(e stri e la#s Oet%een -8 an 8 <i>noninclusive</i>) else it /re icts t(e val!e as t(e one t%o Oac3 /!s t%o. &n /se! o-co e form t(e /re icte val!es is com/!te as follo% s*</p> <p style="text-align: center;">if1 val20ac3 - valD0ac3 Y 8 T T val20ac3 - valD0ac3 Z -8 2 iPre icte [val20ac3 \ lval20ac3 - valD0ac3?]</p>

3redi#tor Ty\$e	Des#ri\$tion
	else iPre ict e [val20ac3 \ 2?
Ram/	Pre ict val!e IiK as val!es Ii\K in e>
8or1	Pre ict as last) 0!t !se 8 ,R instea of s!0tract to com/!te resi !al
8or2	Pre ict as val!e 0efore last) 0!t !se 8 ,R instea of s!0tract to com/!te resi !al
+ 4 " "	+o /re iction a//lie

- B. IData T#/e * Fiel +ameK . T(is format for titling t(e iagram s#m0ol in icates t(at t(e ata is 0ot(com/resse an enco e . T(e DataVT#/e is an a00reviate ata t#/e s#m0ol as efine in [F.2 Data T#/es](#) an !s!all# re/resent a vector!arra# of ata. ;o% t(e ata is com/resse an enco e into t(e Data T#/e is in icate 0# a - , D. - t#/e an ot(er information store 0efore t(e /artic!lar ata in t(e file. &n t(e e>am/le 0elo% t(e DataVT#/e is IVec4B2K an Fiel V+ame is I-o eTe>t.K

: e#7 (2 5 CodeTe;t

+ote t(at for some JT file [Segment T#/es](#) t(ere is L"&: com/ression also a//lie to all 0#tes of element ata store in t(e segment. T(is L"&: com/ression a//lie to all t(e segment!s ata is not in icate in t(e iagrams t(ro!g(t(e !se of Irectangle 0o> %it(cli//e rig(t si e cornersK. &nstea) one m!st e>amine information store %it(t(e first !ement in t(e file segment to etermine if L"&: com/ression is a//lie to all ata in t(e segment. ' com/lete escri/tion of t(e JT format ata com/ression an enco ing can 0e fo!n in [C.1.B Data Segment](#) an [8 Data -om/ression](#).

Follo%ing eac(ata collection iagram is etaille escri/tions for eac(entr# in t(e ata iagram.

- For rectangles t(is etail incl! es t(e a00reviate ata t#/e s#m0ol) fiel name) ver0al ata escri/tion) an com/ression tec(niA!e!algorit(m % (ere a//ro/riate. &f t(e ata fiel is oc!mente as a collection of flags) t(en t(e fiel is to 0e treata as a 0it mas3 % (ere t(e 0it mas3 is forme 0# com0ining t(e flags !sing t(e 0inar# , R o/erator. .ac(0its !sage is oc!mente) an 0it , + in icates flag val!e is TR4. an 0it , FF in icates flag val!e is F' "S.. 'n# !n oc!mente 0its are reserve .
- For fol ers li.e. ata collections2) if t(e collection is not etaille !n er a s!0-section of t(e /artic!lar oc!ment section referencing t(e ata collection) t(en a comment is incl! e follo%ing t(e iagram in icating % (ere in t(e oc!ment t(e /artic!lar ata collection is etaille .

&f an arro% a//ears %it(a 0ranc(in its s(aft) t(en t(ere are t%o or more o/tions for ata to 0e store in t(e file. = (ic(ata is store %ill e/en on information /revio!sl# rea from t(e file. T(e follo%ing e>am/le s(o%s ata fiel ' follo%e 0#1 e/en ing on val!e of '2 eit(er ata fiel :) -) or D.

In cases where the same data is repeated a loop construct is used where the number of iterations appears next to the loop line. There are

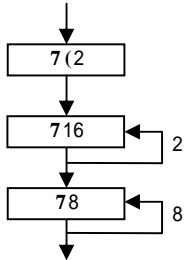
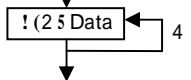
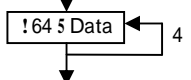
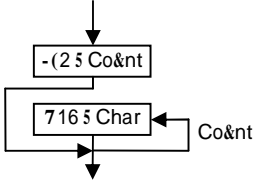
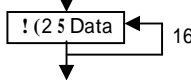
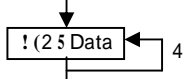
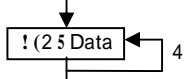
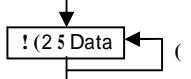
Table 15. Basic Data Types

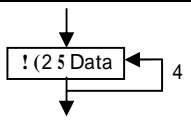
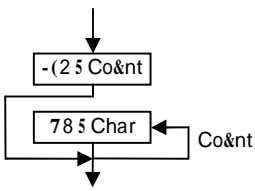
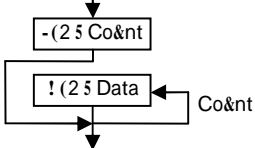
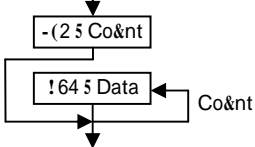
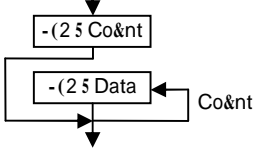
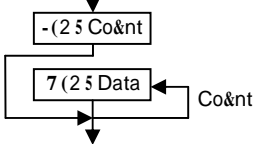
Type	Description
4 - (ar	'n !nsigne 8-0it 0#te.
48	'n !nsigne 8-0it integer val!e.
41F	'n !nsigne 1F-0it integer val!e.
4B2	'n !nsigne B2-0it integer val!e.
&1F	' signe t%o\Ns com/lement 1F-0it integer val!e.
&B2	' signe t%o\Ns com/lement B2-0it integer val!e.
FB2	'n & . . . B2-0it floating /oint n!m0er.
FFD	'n & . . . FD-0it o!0le /recision floating /oint n!m0er

Table 2* - Composite Data Types lists some composite data types which are used to represent some frequently occurring groupings of the basic data types (e.g. Vector) R6: ' color2. The composite data types are defined in this reference simply for convenience to avoid repeating in describing the JT file contents.

Table 25 Composite Data Types

Type	Description	Sym2o"i# Dia ram
: :o>FB2	The : :o>FB2 type defines a 0o!n ing 0o> !sing t%o -oor FB2 types to store the 8 9L coordinates for the 0o!n ing 0o> minimum and maximum corner points.	
-oor FB2	The -oor FB2 type defines 8) 9) L coordinate values. So a -oor FB2 is made up of three FB2 base types.	
-oor FFD	The -oor FFD type defines 8) 9) L coordinate values. So a -oor FFD is made up of three FFD base types.	

Type	Description	Symbolic Diagram
64&D	T(e 64&D t#/e is a 1F 0#te 1128-0it2 n!m0er. 64&D is store %ritten to t(e JT file !sing a fo!r-0#te %or 14B2) 2 t%o-0#te %or s 141F2) an 8 one-0#te %or s 1482 s!c(as* WBF2E0D.0-DF89-11DB-9'-0--0B-0E-.8-2--BB-01X &n t(e JT format 64&Ds are !se as !niA!e i entifiers le.g. Data Segment &D) , 0rect T#/e &D) etc.2	
; -oor FB2	T(e ; -oor FB2 t#/e efines 8) 9) L) = (omogeneo!s coor inate val!es. So an ; -oor FB2 is ma e !/ of fo!r FB2 Oase t#/es.	
; -oor FFD	T(e ; -oor FFD t#/e efines 8) 9) L) = (omogeneo!s coor inate val!es. So an ; -oor FFD is ma e !/ of fo!r FFD Oase t#/es	
\$0String	T(e \$0String t#/e starts %it(an &B2 t(at efines t(e n!m0er of c(aracters 1+ !m- (ar2 t(e string contains. T(e n!m0er of 0#tes of c(aracter ata is I2] + !m- (arK li.e. t(e strings are %ritten o!t as m!lti-0#te c(aracters % (ere eac(c(aracter is 4 1F siHe2.	
\$>DFB2	Defines a D-0#-D matri> of FB2 val!es for a total of 1F FB2 val!es. T(e val!es are store in ro% ma/or or er Irig(t most !0scri/t col!mn varies fastest) t(at is) t(e first D elements form t(e first ro% of t(e matri>.	
PlaneFB2	T(e PlaneFB2 t#/e efines a geometric Plane !sing t(e 6 general Form of t(e /lane eA!ation 1' > \ :# \ -H \ D [02. T(e PlaneFB2 t#/e is ma e !/ of fo!r FB2 Oase t#/es % (ere t(e first t(ree FB2 efine t(e /lane !nit normal vector 1') :) -2 an t(e last FB2 efines t(e negate /er/en ic!lar istance 1D2) along normal vector) from t(e origin to t(e /lane.	
G!aternion	T(e G!aternion t#/e efines a B- imensional orientation Ino translation2 in A!aternion linear com0ination form la \ 0 \ c \ 2 % (ere t(e fo!r scalar val!es la) 0) c) 2 are associate %it(t(e D imensions of a A!aternion 11 real imension) an B imaginari# imensions2. So t(e G!aternion t#/e is ma e !/ of fo!r FB2 Oase t#/es.	
R6 :	T(e R6 : t#/e efines a color com/ose of Re) 6reen) :!e com/onents eac(of % (ic(is a FB2. So a R6 : t#/e is ma e !/ of t(ree FB2 Oase t#/es. T(e Re) 6reen) :!e color val!es t#/icall# range from 0.0 to 1.0.	

Type	Description	Symbolic Diagram
RGB	The RGB type defines a color composed of Red, Green, and Blue components. Each component is a float from 0.0 to 1.0. The color values are normalized to the range [0.0, 1.0].	
String	The String type starts with a character that defines the number of characters in the string. The number of characters is a positive integer. The string is written out as a single character per byte.	
VecFB2	The VecFB2 type defines a vector array of float elements. The type starts with a character that defines the count of following float elements. So a VecFB2 is made up of one float followed by a number of floats. Note that it is valid for the float count number to be equal to 10K in indicating no following float.	
VecFFD	The VecFFD type defines a vector array of float elements. The type starts with a character that defines the count of following float elements. So a VecFFD is made up of one float followed by a number of floats. Note that it is valid for the float count number to be equal to 10K in indicating no following float.	
Vec&B2	The Vec&B2 type defines a vector array of &B2 elements. The type starts with a character that defines the count of following &B2 elements. So a Vec&B2 is made up of one &B2 followed by a number of &B2. Note that it is valid for the &B2 count number to be equal to 10K in indicating no following &B2.	
Vec4B2	The Vec4B2 type defines a vector array of 4B2 elements. The type starts with a character that defines the count of following 4B2 elements. So a Vec4B2 is made up of one 4B2 followed by a number of 4B2. Note that it is valid for the 4B2 count number to be equal to 10K in indicating no following 4B2.	

/ File Format

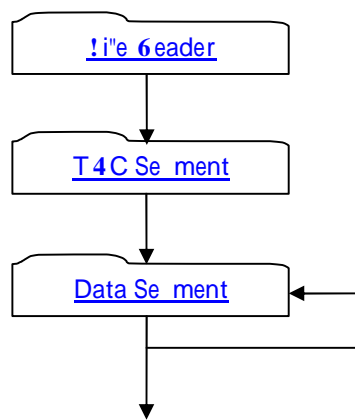
All objects represented in the JT format are assigned an object identifier (e.g. see [C.2.1.1.1.1 :ase +o_e Data](#)) or [C.2.1.1.2.1.1 :ase 'trio!te Data](#) and all references from one object to another object are represented in the JT format using the reference object's object identifier. It is the responsibility of JT format readers/writers to maintain the integrity of these object references through appropriate pointer handling as JT format data is read into memory or written out to disk. Where pointer references are used in the process of converting references on object identifiers into direct memory references an identifier is the reverse operation (i.e. replacing references based on memory pointers with object identifiers).

1.1 File Structure

A JT file is structured as a sequence of segments. The File Header is the first segment of data in the file. The File Header is followed by one or more T, - Segment and a series of other Data Segments. The one Data Segment (which must always exist to have a reference compliant JT file) is the [C.2.1 "S6 Segment](#).

The T, - Segment is located within the file using data stored in the File Header. This segment is information that locates all other Data Segments within the file. If there are no JT format compliance rules about the T, - Segment must be located within the file in practice the T, - Segment is typically located either immediately following the File Header or as shown in the following figure at the very end of the file following all other Data Segments.

Figure 15: File Structure



1.1.1 File Header

The File Header is the first segment of data in a JT file. The File Header contains information about the JT file version and the location (which is used to determine the location of the file). The exact contents of the File Header are as follows:

!i &re 25 !i"e 6eader data #o""e#tion

1C%ar 2 Version

' n 80-c(aracter version string defining t(e version of t(e file format !se to %rite t(is file. T(e Version string (as t(e follo%ing format*

Version .

= (ere is re/lace 0# t(e ma7or version n!m0er) is re/lace 0# t(e minor version n!m0er) an /rovi es ot(er !ns/ecifie information. T(e version string is /a e %it(s/aces to a lengt(of CE 'S- && c(aracters an t(en t(e final five c(aracters m!st 0e fille %it(t(e follo%ing linefee an carriage ret!rn c(aracter com0ination !s(o%n !sing c-st#le s#nta>2*

VersionRCES [U U
VersionRCFS [U^nU
VersionRCCS [U^rU
VersionRC8S [U^nU
VersionRC9S [U U

T(ese final E c(aracters !s(o%n a0ove an referre to as 'S- &&50inar# translation etection 0#tes2 can 0e !se 0# JT file rea ers to vali ate t(at t(e JT files (as not 0een corr!/te 0# 'S- && mo e FTP transfers.

So for a JT Version 8.1 file t(is string %ill loo3 as follo%s*

IVersion 8.1 JT

^n^ne

Defines the file offset or error that can be used to determine if there is a mismatch between the file offset or error and the macro line number (the loader is being run).
Valid values for :#the , r er are*

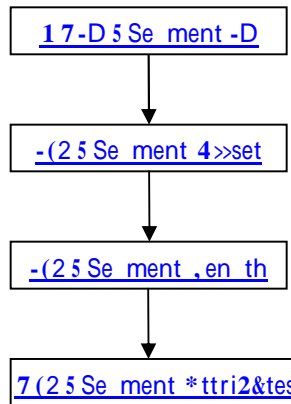
0 Q "east Significant Offset first 1" s0First2
1 Q \$ost Significant Offset first 1 \$ s0First2

I)# 2 File Attributes

All bits in this field are reserved

I)# 2 T4C

!i &re 45 T4C)ntry data #0"e#tion



5 1 ID 2 Segment ID

Segment ID is the global identifier for the segment.

1) # 2 Segment 4 offset

Segment offset defines the offset from the top of the file to start of the segment.

1) # 2 Segment length

Segment length is the total size of the segment in bytes.

1) # 2 Segment attributes

Segment attributes is a collection of segment information encoded in a single 4-byte field following bit allocation.

: bits 0 - 2B	Reserved for future use.
: bits 2D - B1	Segment table. -om/lete list of Segment table/tes can be found in Table B* Segment T#/es.

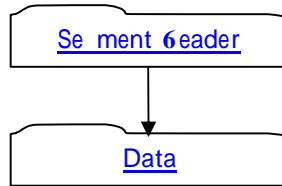
1.1.) Data Segment

All data stored in a JT file must be defined within a Data Segment. Data Segments are the basic classification of data they contain. See [Segment Table](#) field description for a complete list of the segment table.

Depending on the field combination (encoding) some Data Segment table also have a length: combination of all the Data table of information persists within the segment. = (length: combination of all the Data table of information is in the table information store within the first element in the segment. Also [Table B* Segment T#/es](#) (as a column in the Segment Table) have a length: combination of all the Data table.

All Data Segments have the same basic structure.

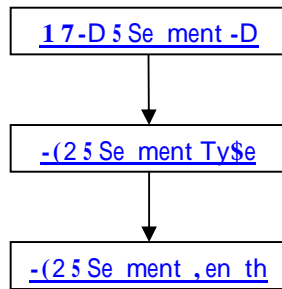
!i &re 55 Data Segment data #o"e#tion



1.1.1 Segment Header

Segment header contains information that determines (0% of the remainder of the Segment is interpreted as the header.

!i &re 65 Segment Header data #o"e#tion



5.1.2 Segment ID

Global identifier for the segment.

1.1.2 Segment Type

Segment Type defines a broad classification of the segment contents. For example a Segment Type of I1K denotes that the segment contains "Original Scene Graph (material) I2K denotes contents of a :-Re/) etc.

The complete list of segment types is as follows*

Table 5 Segment Types

Type	Data Contents	+ , - . Compression Conditionally *\$\$"ied to a" o> the Segment's) "ement Data
1	"Original Scene Graph (9 es
2	JT :-Re/	9 es
B	P \$ & Data	9 es
D	\$ eta Data	9 es

Type	Data Contents	+, -. Compression Conditionally Applied to the Segment's Element Data
F	Segment	+0
C	Segment, D0	+0
8	Segment, D1	+0
9	Segment, D2	+0
10	Segment, DB	+0
11	Segment, DD	+0
12	Segment, DE	+0
1B	Segment, DF	+0
1D	Segment, DC	+0
1E	Segment, D8	+0
1F	Segment, D9	+0
1C	8T : -Re/	9es
18	= reframe Representation	9es

Note* Segment Types C-1F all identify the contents as Segment Data (where the increasing type number is intended to convey some notion of (optional) specific segment reference. The lower type in the C-1F range the more detailed the Segment Data. Segment Type ICK is the most detailed Segment Data. For the rare case when there are more than 10 Ds, D9 and greater are all assigned Segment Type I1FK.

Note* The more generic Segment type i.e. Segment Type IFK2 is used when the Segment (as one or more of the following characteristics)

1. Not a descendant of an Segment.
2. Is reference 0 i.e. is a child of more than one Segment.
- B. Segment (as its only child-in Segment, Ds)
- D. +0%a# to determine what Segment Data Segment reference.

1) # 2 Segment Length

Segment Length is the total size of the segment in bytes. This length value includes all segment Data bytes (the Segment header bytes i.e. it is the size of the complete segment and so is equal to the length value stored in the segment's [T, - . ntr#](#).

1.1.) # Data

The interpretation of the Data section depends on the Segment Type. See [C.2 Data Segments](#) for complete description for all Data Segment that may be contained in a JT file.

Typically the Data section is Segment Type dependent (there is a common structure which often occurs within the Data section. This structure is a list or multiple lists of elements (where each element (as the same basic structure which consists of some fixed length (header information describing the type of object contained in the element) followed by some variable length object specific data.

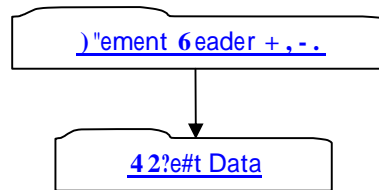
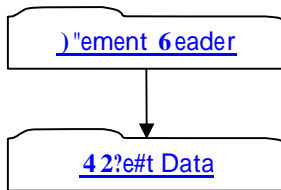
An individual data field of an element data collection (and its child data collections) may have a variable compression encoding applied to them as indicated (through compression related data values stored as part

of the /article/element's storage format. In addition another level of compression i.e. L" : compression may be conditional# applied to all objects of information store for all elements %it in a /article/segment. For all Segment t#/es s!//ort L" : compression on all Segment data as in [Table B* Segment T#/es.](#) If a /article/file segment is of type t#/e %ic(s!//orts L" : compression on all the Segment data) %et(er t(is compression is applied or not is indicated 0# data values store in the [.element ;ea er L" :](#) data collection of the first element %it in the Segment. 'n in- e/t(escription of JT file compression encoding techniques can be found in [8 Data -om/ession.](#)

!i &re %5 Data data #o"e#tion

For Segment T#/es t(at o N4T s!//ort L" : compression on all Segment Data.
 lsee [Table B* Segment T#/es.2](#)

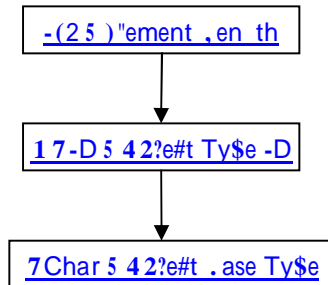
For Segment T#/es t(at s!//ort L" : compression on all Segment Data
 lsee [Table B* Segment T#/es.2](#)



/.1.)#1 Element 0ea*er

.element ;ea er contains data defining the length(in objects of the .element along %it(information escribing the object t#/e contains in the .element.

!i &re 85)"ement 6eader data #o"e#tion



l)#2 Element "engt%

.element "engt(is the total length(in objects of the element , Object Data.

51ID 2 4b7ect T! e ID

, Object T#/e &D is the global# identifier for the object t#/e. ' complete list of the assigned 64&D for all object t#/es store in a JT file can be found in ['//en i> '* , Object T#/e & identifiers.](#)

1C%ar 2 4b7ect 3ase T! e

, 07ect : ase T#/e identifies t(e 0ase o07ect t#/e. T(is is !sef!! %(en an !n3no%n element t#/e is enco!ntere an t(!s t(e 0est t(e loa er can o is to rea t(e 3no%n , 07ect : ase T#/e ata 0#tes !0ase t#/e o07ect ata is al%a#s %ritten first2 an t(en s3i/ lrea /ass2 t(e 0#tes of !n3no%n ata !sing 3no%le ge of n!m0er of 0#tes encom/assing t(e , 07ect : ase T#/e ata an t(e !n3no%n t#/es "engt(fiel . &f t(e , 07ect : ase T#/e is !n3no%n t(en t(e loa er s(o!! sim/l# s3i/ lrea /ass2 . lement "engt(n!m0er of 0#tes.

Vali , 07ect : ase T#/es incl! e t(e follo%ing*

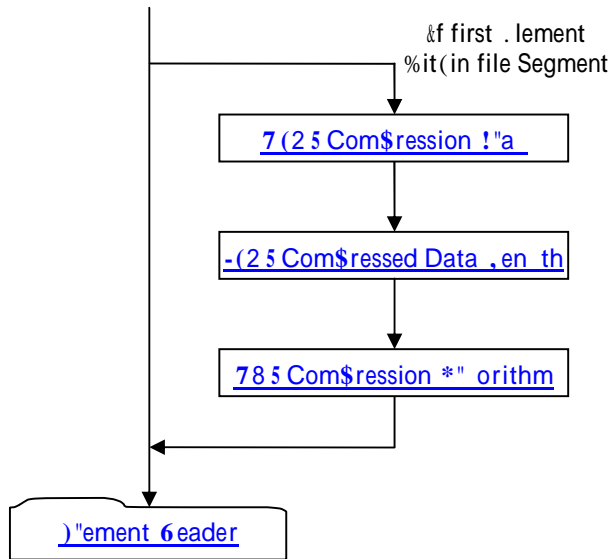
Table 45 42#e#t . ase Ty\$es

42#e#t . ase Ty\$e	Description	42#e#t . ase Ty\$e's Data Format
2EE	4n3no%n 6ra/(+o e , 07ect	none
0	: ase 6ra/(+o e , 07ect	C.2.1.1.1.1 : ase +o e Data
1	6ro!/ 6ra/(+o e , 07ect	C.2.1.1.1.B.16ro!/ +o e Data
2	S(a/e 6ra/(+o e , 07ect	C.2.1.1.1.10.1.1 : ase S(a/e Data
B	: ase 'ttri0!te , 07ect	C.2.1.1.2.1.1 : ase 'ttri0!te Data
D	S(a/e " , D	none
E	: ase Pro/ert# , 07ect	C.2.1.2.1.1 : ase Pro/ert# 'tom Data
F	JT , 07ect Reference , 07ect	C.2.1.2.EJT , 07ect Reference Pro/ert# 'tom . lement %it(o!t t(e . lement ; ea er L"&: ata collection.
8	JT "ate "oa e Pro/ert# , 07ect	C.2.1.2.C "ate "oa e Pro/ert# 'tom . lement %it(o!t t(e . lement ; ea er L"&: ata collection.
9	Jt : ase Inone2	none

/.1.) .#.# 6lement 0ea*er 8"l3

. lement ; ea er L"&: ata collection is t(e format of . lement ; ea er ata !se 0# all . lements %it(in Segment T#/es t(at s!//ort L"&: com/ression on all ata in t(e Segment. See [Table B* Segment T#/es](#) for information on %(et(er a /artic!lar Segment T#/e s!//orts L"&: com/ression on all ata in t(e Segment.

!i &re '5)"ement 6eader +, -. data #o""e#tion



-om/lete escri/ption for .lement ;ea er can Oe fo!n in [C.1.B.2.1 .lement ;ea er](#). +ote t(at if [-om/resson Flag](#) in icates t(at L"&: com/resson is , + for all element ata in t(e Segment) t(en t(e [.lement ;ea er](#) ata collection is also com/resse accor ingl#.

1)# 2 Com resson Flag

-om/resson Flag is a flag in icating %(et(er L"&: com/resson is , +5 , FF for all ata elements in t(e file Segment. Vali val!es incl! e t(e follo%ing*

[2	- L"&: com/resson is , +
_[2	- L"&: com/resson is , FF.

l)# 2 Com resse* Data "engt%

-om/resse Data "engt(s/ecifies t(e com/resse ata lengt(in n!m0er of 0#tes. +ote t(at ata fiel [-om/resson 'lgorit\(m](#) is incl! e in t(is co!nt.

18 2 Com resson +lgorit%m

-om/resson 'lgorit(m s/ecifies t(e com/resson algorit(m a//lie to all ata in t(e Segment. Vali val!es incl! e t(e follo%ing*

[1	- +o com/resson
[2	- L"&: com/resson

/ .1.) .#.) 4b7ect Data

T(e inter/retation of t(e ,07ect Data section e/en s !/on t(e ,07ect T#/e &D store in t(e .lement ;ea er lsee [C.1.B.2.1 .lement ;ea er](#)2.

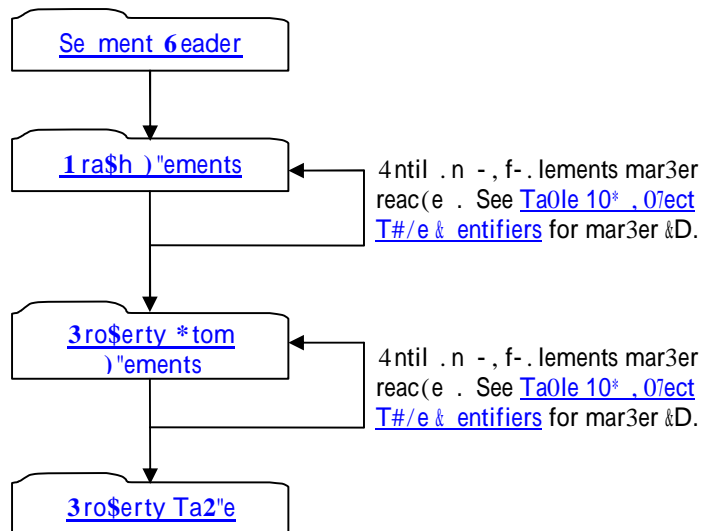
1.1 Data Segments

1.1.1 S5 Segment

S5 Segment contains a collection of objects i.e. elements connected through direct references to form a directed acyclic graph (structure i.e. tree). The S5 is the graphical description of the model and contains graphs (schemas) attributes representing the models' functional components' properties identifying arbitrary meta data (e.g. names) semantic roles of those components and a hierarchical structure expressing the component relationships. The indirect nature of the S5 references implies that there is no feedback inheritance from ancestor to descendant i.e. predecessor to successor. It is the responsibility of the loader to instantiate the graph of the S5 is maintained.

The first graph element in a S5 Segment shall always be a Partition node. The S5 Segment therefore starts with a partition on all elements so all elements in S5 Segment use the [element:ea er](#) form of element (ea er data).

Figure 105, S1 Segment data structure



om/lete description for Segment ; ea er can be found in [C.1.B.1Segment : ea er](#).

1.1.1.1 5ra % 6lements

graph elements form the backbone of the S5 directed acyclic graph (structure) and in doing so serve as the JT models' fundamental description. There are two general classifications of graph elements: nodes and attributes.

Nodes are nodes in the S5 and in general can be categorized as either an internal or leaf node. The leaf nodes are typically schemas used to represent a model's functional components and as such

either contain or reference some graphical representation or geometry. The internal nodes define the hierarchical organization of the leaf nodes forming the spatial and logical model relationships and often contain or reference information like 'triangle.elements' that is inherited from the "S6 to all descendant nodes.

'triangle.elements' represent graphical attributes (characteristics like color) or positional transformations that can be attached to a node and inherited from the "S6.

Each of these general graphical element classifications like 'triangle.elements' is split into specific concrete types based on their content and implications (behavior). The following sections describe each of these and 'triangle.element' types.

1.1.1.1 *element

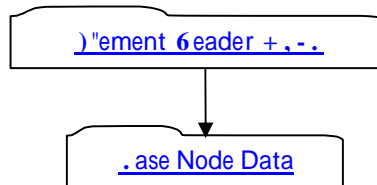
These elements represent the relationships of a model's components. The model's component hierarchy is formed via certain types of these elements containing collections of references to other elements (often in terms of reference to other collections of these elements). These elements are also the objects (either direct or indirect) of geometric properties and other information defining a model's components and representations.

1.1.1.1.1 Base *element

42?e#t Ty\$e -D5 0>10 10BE) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

Base element represents the simplest form of a node that can exist in the "S6. The base element (as no implications "S6 semantic behavior nor can it contain any children nodes.

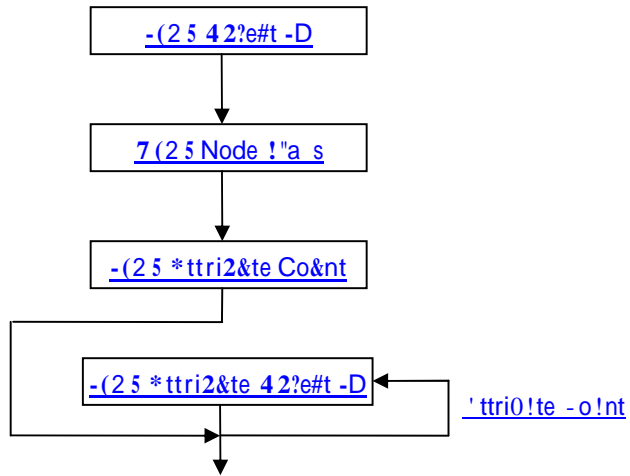
!i &re 115 .ase Node)"ement data #o"e#tion



Complete description for element ; ea er L"&: can be found in [C.1.B.2.2 .lement ; ea er L"&:](#).

1.1.1.1.1.1 Base *element Data

!i &re 125 .ase Node Data data #o"e#tion



1)# 2 4b7ect ID

, 07ect &D is t(e i entifier for t(is , 07ect. , t(er o07ects referencing t(is /artic!lar o07ect o so !sing t(e , 07ect &D.

1)# 2 ' o*e Flags

+o e Flags is a collection of flags. T(e flags are com0ine !sing t(e 0inar# , R o/erator. T(ese flags store vario!s state information of t(e no e o07ect. 'll !n oc!mente 0its are reserve .

0>0000001	- &gnore Flag [0 Q ' lgorit(ms traversing t(e "S6 str!ct!re s(o!l incl! e5/roccs t(is no e. [1 Q ' lgorit(ms traversing t(e "S6 str!ct!re s(o!l s3i/ t(e %(ole s!0gra/(roote at t(is no e. .ssential# t(e traversal s(o!l 0e /r!ne .
-----------	--

1)# 2 +ttribute Count

'ttri0!te -o!nt in icates t(e n!m0er of 'ttri0!te , 07ects reference 0# t(is +o e , 07ect. ' no e ma# (ave Hero 'ttri0!te , 07ect references.

1)# 2 +ttribute 4b7ect ID

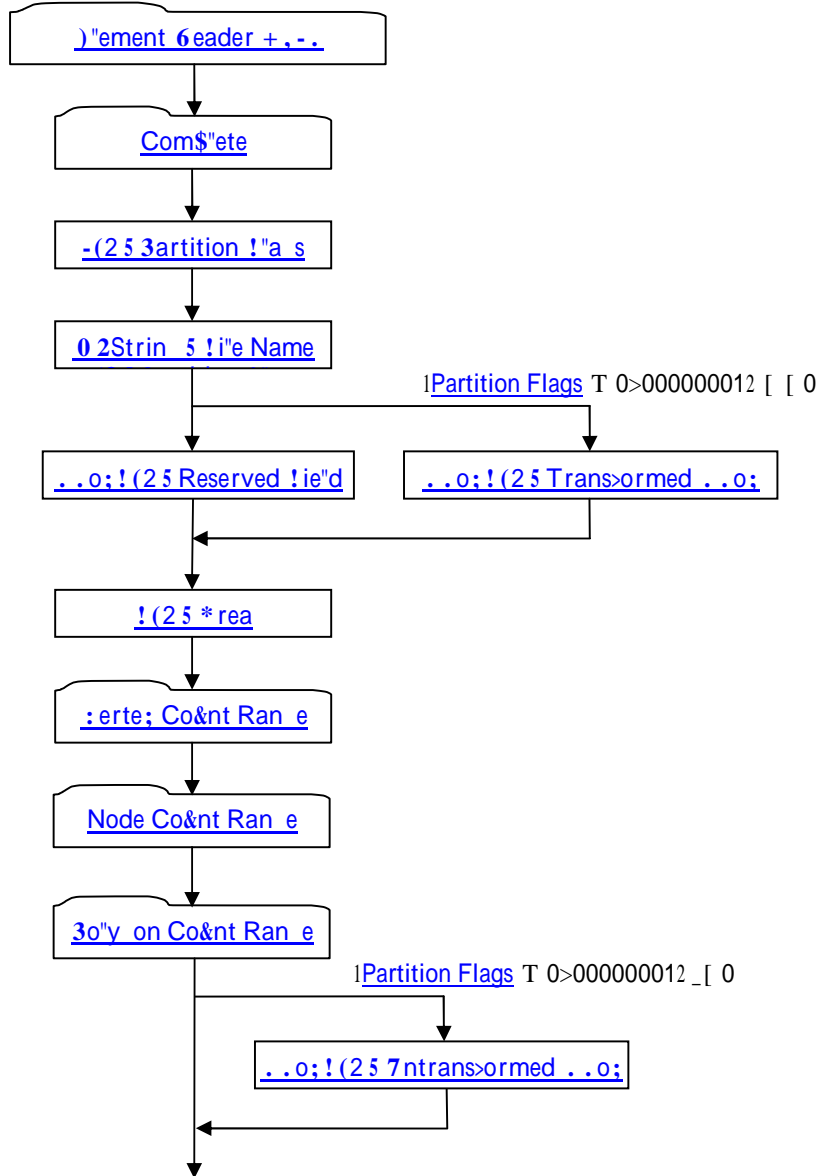
'ttri0!te , 07ect &D is t(e i entifier for a reference 'ttri0!te , 07ect.

/.#.1.1.1.#Partition ' o*e 6lement

42?e#t Ty\$e -D5 0>10 10Be) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

' Partition +o e re/resents an e>ternal JT file reference an /rovi es a means to /artition a mo el into m!lti/le /(#sical JT files le.g. se/arate JT file /er /art in an assem0l#2. = (en t(e reference JT file is o/ene) t(e Partition +o e\ s c(il ren are reall# t(e c(il ren of t(e "S6 root no e for t(e !n erl#ing JT file. 4sage of Partition +o es in "S6 also ai s in s!//orting JT file loa er'srea er IOest /racticeK of late loa ing ata li.e. can ela# o/ening an loa ing t(e e>ternall# reference JT file !ntil t(e ata is nee e 2.

!i &re 1(5 3artition Node)"ement data #o""e#tion



-om/lete escri/tion for . lement ; ea er L"&: can 0e fo!n in [C.1.B.2.2 . lement : ea er L"&:](#).

-om/lete escri/tion for 6ro!/+o e Data can 0e fo!n in [C.2.1.1.1.B.16ro!/+o e Data.](#)

l)# 2 Partition Flags

Partition Flags is a collection of flags. These flags are combined using the bitwise OR operator. These flags store various state information of the Partition object such as indicating the presence of optional data. All non-optional bits are reserved.

0x00000001	4ntransforme	00!n ing 00> is %ritten.
------------	--------------	--------------------------

9bString 2 File Name

File name is the relative path of the Partition's file location. = (ere Irelative /at(K s(o!! 0e inter/rete to mean the string contains the file name along with an additional path information that locates the Partition JT file relative to the location of the referencing JT file

33o:F) # 2 Reserve* Field*

Reserve Field is a data field reserved for future JT format extensions

33o:F) # 2 Transforme* 33o:

The Transforme : :o> is an + -S axis-aligned 00!n ing 00> and represents the transforme geometr# extents for all geometr# contained in the Partition +o e. This 00!n ing 00> information may be used as a reference of JT data to determine whether to load the data contained within the Partition node i.e. is an artifact of the 00!n ing 00> within the view frustum.

F) # 2 Area

Area is the total surface area for this node and all of its descendants. This value is stored in + -S coordinate space i.e. values scale 0# + - S scaling.

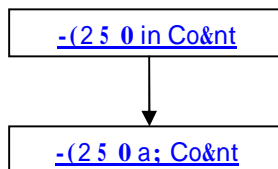
33o:F) # 1 ntransforme* 33o:

The 4ntransforme : :o> is only present if bit 0x00000001 of [Partition Flags](#) data field is set. The 4ntransforme : :o> is an + -S axis-aligned 00!n ing 00> and represents the ntransforme geometr# extents for all geometr# contained in the Partition +o e. This 00!n ing 00> information may be used as a reference of JT data to determine whether to load the data contained within the Partition node i.e. is an artifact of the 00!n ing 00> within the view frustum.

/.#.#.#.#.1 Verte: Count Range

Verte -o!nt Range is the aggregate minimum and maximum vertex count for all descendants of the Partition +o e. There is a minimum and maximum value to accommodate descendant Oranges (having " , D nodes) which encompass a range of count values within the Orange() and to accommodate nodes that can themselves generate varying representations. The minimum value represents the least vertex count that can be achieved by the Partition +o e's descendants. The maximum value represents the greatest vertex count that can be achieved by the Partition +o e's descendants.

!i &re 145 :erte; Co&nt Ran e data #o"e#tion



I)# 2 9 in Count

\$ in -o!nt is t(e least verte> co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants.

I)# 2 9a: Count

\$a> -o!nt is t(e ma>im!m verte> co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants.

/.#.1.1.1.#.# ' o*e Count Range

+o e -o!nt Range is t(e aggregate minim!m an ma>im!m co!nt of all no e escen ants of t(e Partition +o e. T(ere is a minim!m an ma>im!m val!e to accommo ate escen ant Oranc(es (aving " , D no es) %(ic(encom/ass a range of escen ant no e co!nt val!es %it(in t(e Oranc(. T(e minim!m val!e re/resents t(e least no e co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants. T(e ma>im!m val!e re/resents t(e greatest no e co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants.

T(e ata format for +o e -o!nt Range is t(e same as t(at escri0e in [C.2.1.1.2.1Verte> -o!nt Range](#).

/.#.1.1.1.#.) Pol!gon Count Range

Pol#gon -o!nt Range is t(e aggregate minim!m an ma>im!m /ol#gon co!nt for all escen ants of t(e Partition +o e. T(ere is a minim!m an ma>im!m val!e to accommo ate escen ant Oranc(es (aving " , D no es) %(ic(encom/ass a range of co!nt val!es %it(in t(e Oranc() an to accommo ate no es t(at can t(emselves generate var#ing re/resentations. T(e minim!m val!e re/resents t(e least /ol#gon co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants. T(e ma>im!m val!e re/resents t(e greatest /ol#gon co!nt t(at can 0e ac(ieve 0# t(e Partition +o e!s escen ants.

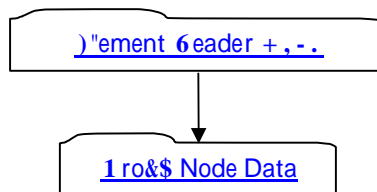
T(e ata format for Pol#gon -o!nt Range is t(e same as t(at escri0e in [C.2.1.1.2.1Verte> -o!nt Range](#).

/.#.1.1.1.) 5rou ' o*e 6lement

42?e#t Ty\$e -D5 0>10 1010) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

6ro!// +o es contain an or ere list of references to ot(er no es) calle t(e gro!//\s children. 6ro!// no es ma# contain Hero or more c(il ren? t(e c(il ren ma# 0e of an# no e t#/e. 6ro!// no es ma# not contain references to t(emselves or t(eir ancestors.

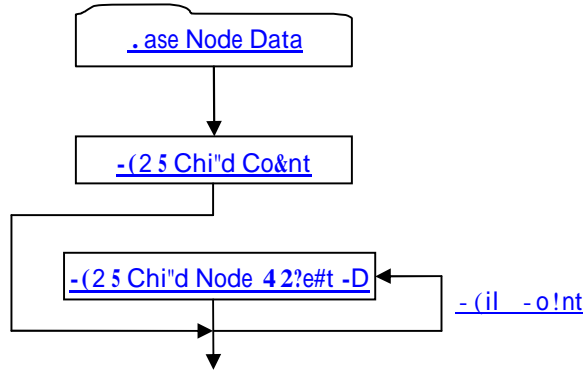
!i &re 155 1 ro&\$ Node)"ement data #o""e#tion



-om/lete escri/tion for . lement ; ea er L"&: can 0e fo!n in [C.1.B.2.2 . lement ; ea er L"&:](#).

/.#.1.1.1.).1 5rou 'o*e Data

!i &re 165 1ro&\$ Node Data data #o""e#tion



-om/lete escri/tion for :ase +o e Data can 0e fo!n in [C.2.1.1.1.1 :ase +o e Data](#).

l)# 2 C%il* Count

-(il -o!nt in icates t(e n!m0er of c(il no es for t(is 6ro!/ +o e ,07ect. ' no e ma# (ave Hero c(il ren.

l)# 2 C%il* 'o*e 4b7ect ID

-(il +o e ,07ect &D is t(e i entifier for t(e reference +o e ,07ect.

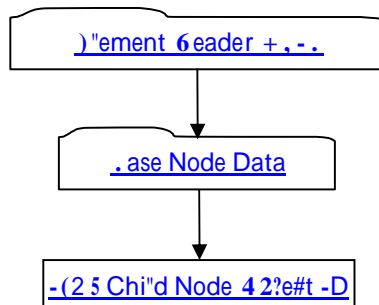
/.#.1.1.1., Instance 'o*e 6lement

42?e#t Ty\$e -D5 0>10 102a) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

'n &nstance +o e contains a single reference to anot(er no e. T(eir /!r/ose is to allo% s(aring of no es an assignment of instance-s/ecific attri0!tes for t(e instance no e. &nstance +o es ma# not contain references to t(emselves or t(eir ancestors.

For e>am/le) a 6ro!/ +o e co!! !se &nstance +o es to instance t(e same S(a/e +o e several times) a//l#ing iffereent material /ro/erties an matri> transformations to eac(instance. +ote t(at t(is co!! also 0e one 0# !sing 6ro!/ +o es instea of &nstance +o es) 0!t &nstance +o es reA!ire fe%er reso!rces.

!i &re 1%5 -nstan#e Node)"ement data #o""e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#)

-om/lete escri/tion for :ase +o e Data can Oe fo!n in [C.2.1.1.1.1 :ase +o e Data.](#)

l)# 2 C%il* ' o*e 4b7ect ID

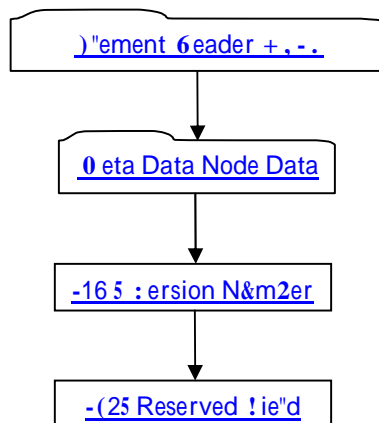
-(il +o e ,07ect &D is t(e i entifier for t(e instance +o e ,07ect.

/.#.1.1.1.-Part ' o*e 6lement

42?e#t Ty\$e -D5 0>ceBEC2DD) 0>B8f0) 0>11 1) 0>aE) 0>F) 0>0) 0>F0) 0>9C) 0>0) 0>cF) 0>e1

' Part +o e .lement re/resents t(e root no e for a /artic!lar Part %it(in a "S6 str!ct!re. .ver# !niA!e Part re/resente %it(in a "S6 str!ct!re s(o!l (ave a corres/on ing Part +o e .lement. ' Part +o e .lement t#/icall# references !!sing "ate "oa e Pro/ert# 'toms2 a itional Part s/ecific geometric ata an s/or /ro/erties Ie.g. :-Re/ ata) P \$& ata2.

!i &re 185 3art Node)"ement data #o""e#tion



-om/lete escri/tion for .lement ;ea er L"!!: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"!!](#) .

-om/lete escri/tion for \$ eta Data +o e Data can Oe fo!n in [C.2.1.1.1.F.1 \\$ eta Data +o e Data](#).

11.2 Version 'umber

Version +!m0er is t(e version i entifier for t(is no e. Version n!m0er IO>0001K is c!rrrent!# t(e on!# vali val!e for Part no es.

1) #2 Reserve * Fiel *

Reserve Fiel is a ata fiel reserve for f!t!re JT format e>/ansion

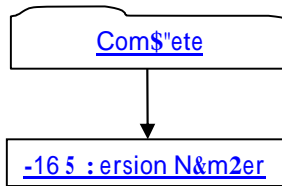
/.#.1.1.1.. 9 eta Data 'o*e 6lement

42?e#t Ty\$e -D5 0>ceBEC2DE) 0>B8f0) 0>11 1) 0>aE) 0>F) 0>0) 0>F0) 0>9C) 0>0) 0>cF) 0>e1

T(e \$ eta Data +o e .lement is a no e t#/e !se for storing references to s/ecific llate loa e K meta- ata ie.g. /ro/erties) P \$!!. T(e reference meta- ata is store in a se/arate a ressa0le segment of t(e JT File lsee [C.2.F \\$ eta Data Segment](#)) an t(!s t(e !se of t(is \$ eta Data +o e .lement is in s!//ort of t(e JT file loa er!rea er IOest /racticeK of late loa ing ata li.e. storing t(e reference meta- ata in se/arate a ressa0le segment of t(e JT file allo%s a JT file loa er!rea er to ignore t(is no e!s meta- ata on initial loa an instea late-loa t(e no e!s meta- ata !/on eman so t(at t(e associate meta- ata oes not cons!me memor# !ntil nee e 2.

!i &re 1'5 0 eta Data Node)"ement data #o"e#tion m(the le F e s e e i n " o

!i &re 205 0 eta Data Node Data data #o"e#tion



-om/lete escri/ tion for 6ro!/ +o e Data can Oe fo!n in [C.2.1.1.1.B.16ro!/ +o e Data](#).

11. 2 Version 'umber

Version +!m0er is t(e version i entifier for t(is ata. Version n!m0er IO>0001K is c!rrent!# t(e on!# vali val!e.

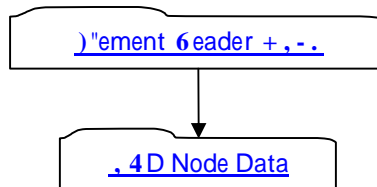
/.#.1.1.1./ "4D 'o*e 6lement

42?e#t Ty\$e -D5 0>10 102c) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

'n ", D +o e (ol s a list of alternate re/resentations. T(e list is re/resente as t(e c(il ren of a 0ase gro!/ no e) (o%ever) t(ere are no im/licit semantics associate %it(t(e or ering. Traversers of "S6 ma# a//!# semantics to t(e or ering as /art of alternative re/resentation selection.

.ac(alternative re/resentation co!! Oe a s!0-assem0!# %(ere t(e alternative re/resentation is a gro!/ no e %it(an assem0!# of c(il ren.

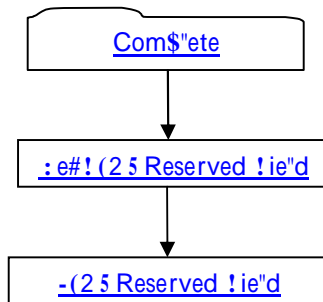
!i &re 215 , 4D Node)"ement data #o"e#tion



-om/lete escri/ tion for . lement ; ea er L"&: can Oe fo!n in [C.1.B.2.2 . lement ; ea er L"&:](#).

/.#.1.1.1./1 "4D 'o*e Data

!i &re 225 , 4D Node Data data #o"e#tion



-om/lete escri/ tion for 6ro!/ +o e Data can 0e fo!n in [C.2.1.1.1.B.16ro!/ +o e Data](#).

VecF)# 2 Reserve* Fiel*

Reserve Fiel is a vector ata fiel reserve for f!t!re JT format e>/ansion.

I)# 2 Reserve* Fiel*

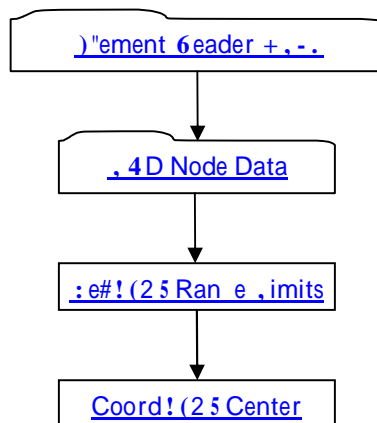
Reserve Fiel is a ata fiel reserve for f!t!re JT format e>/ansion.

/.#.1.1.1.8Range " 4D ' o*e 6lement

42?e#t Ty\$e -D5 0>10 10Dc) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

Range " , D +o es (ol a list of alternate re/resentations an t(e ranges over %(ic(t(ose re/resentations are a//ro/riate. Range "imits in cate t(e instance 0et%een a s/ecifie center /oint an t(e e#e /oint) %(it(in %(ic(t(e corres/on ing alternate re/resentation is a//ro/riate. Traversers of "S6 cons!t t(ese range limit val!es %(en ma3ing an alternative re/resentation selection.

!i &re 2(5 Ran e , 4D Node)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L": can Oe fo!n in [C.1.B.2.2 .lement ;ea er L":](#).

-om/lete escri/tion for " , D +o e Data can Oe fo!n in [C.2.1.1.1.C.1 " , D +o e Data](#)

VecF)# 2 Range "imits

Range "imits in icate t(e = -S instance Oet%een a s/ecifie center /oint an t(e e#e /oint) %it(in %(ic(t(e corres/on ing alternate re/resentation is a//ro/riate. It is not reA!ire t(at t(e co!nt of range limits is eA!ivalent to t(e n!m0er of alternative re/representations. T(ese val!es are consi ere Isoft val!esK in t(at loa ersvie%ers of JT ata are free to t(ro% t(ese val!es a#a# an com/!te ne% val!es Oase on t(eir esire " , D selection semantics.

:est /ractices s!ggest t(at "S6 traversers a//l# t(e follo%ing strateg#) at Range " , D +o es) %(en ma3ing alternative re/representation selection ecisions Oase on Range "imits* T(ese first alternate re/representation is vali %(en t(e instance Oet%een t(e center an t(e e#e /oint is less t(an or eA!al to t(e first range limit lan %(en no range limits are s/ecifie 2. T(ese secon alternate re/representation is vali %(en t(e instance is greater t(an t(e first limit an less t(an or eA!al to t(e secon limit) an so on. T(ese last alternate re/representation is vali for all instances greater t(an t(e last s/ecifie limit.

Coor*F)# 2 Center

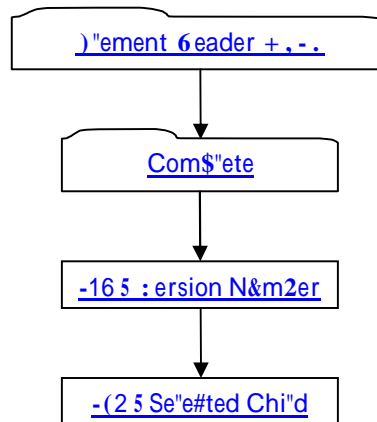
-enter s/ecifies t(e 8)9L coor inates for t(e +-S center /oint !/on %(ic(alternative re/representation selection e#e instance com/!tations are Oase . T#/icall# t(is location is t(e center of t(e ig(est- etail alternative re/representation. T(ese val!es are consi ere Isoft val!esK in t(at loa ersvie%ers of JT ata are free to t(ro% t(ese val!es a#a# an com/!te ne% val!es Oase on t(eir esire " , D selection semantics

/.#.1.1.1.;S(itc% ' o*e 6lement

42?e#t Ty\$e -D5 0>10 10fB) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

T(ese S%itc(+o e is ver# m!c(li3e a 6ro!/ +o e in t(at it contains an or ere list of references to ot(er no es) calle t(e *children* no es. T(ese ifferece is t(at a S%itc(+o e also contains a itional ata in icateing %(ic(c(il lone or none2 a "S6 traverser s(o!! /rocess\$traverse.

!i &re 245 Swit#h Node)"ement data #o""e#tion



-om/lete escri/tion for . lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 . lement :ea er L"&:](#).

-om/lete escri/tion for 6ro!/+o e Data can Oe fo!n in [C.2.1.1.1.B.16ro!/+o e Data](#).

11. 2 Version 'umber

Version +!m0er is t(e version i entifier for t(is no e. Version n!m0er IO>0001K is c!rrent!# t(e on!# vali val!e for S%itc(no es.

1)# 2 Selecte* C%il*

Selecte -(il is t(e in e> for t(e selecte c(il no e. Vali Selecte -(il val!es resi e %(in t(e follo%ing range* I-1 Y Selecte -(il Y -(il -o!ntK. =(ere I-1K in icates t(at no c(il is to Oe selecte an I-(il -o!ntK is t(e ata fiel val!e from [C.2.1.1.1.B.16ro!/+o e Data](#).

1.#.1.1.1.1< S%a e 'o*e 6lements

S(a/e +o e .lements are IleafK no es %(in t(e "S6 str!ct!re an contain or reference t(e geometric s(a/e efiniion ata ie.g. vertices) /ol#gons) normals) etc.2.

T#/icall# S(a/e +o e .lements o not irectl# contain t(e actual geometric s(a/e efiniion ata) O!t instea reference !!sing "ate "oa e Pro/ert# 'toms2 S(a/e " ,D Segments %(in t(e file for t(e actual geometric s(a/e efiniion ata. Storing t(e geometric s(a/e efiniion ata %(in se/arate in e/en ent!# a ressa0le ata segments in t(e JT file) allo% s a JT file rea er to Oe str!ct!re to s!//ort t(e IOest /racticeK of ela#ing t(e loa ing\$rea ing of associate ata !ntil it is actual!# nee e. -om/lete escri/tions for "ate "oa e Pro/ert# 'tom .lements an S(a/e " ,D Segments can Oe fo!n in [C.2.1.2.C "ate "oa e Pro/ert# 'tom .lement an C.2.2 S\(a/e " ,D Segment res/ectivel#.](#)

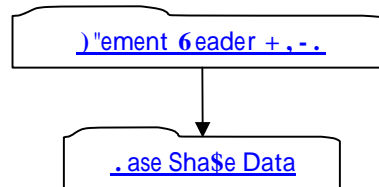
T(ere are several t#/es of S(a/e +o e .lements %(ic(t(e JT format s!//orts. T(e follo%ing s!0-sections oc!ment t(e vario!s S(a/e +o e .lement t#/es.

C.2.1.1.10.1.1 Base SHA-256 Element

42?e#t Ty\$e -D5 0>10 10E9) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

:ase S(a/e +o e .lement re/resents t(e sim/lest form of a s(a/e no e t(at can e>ist %it(in t(e "S6.

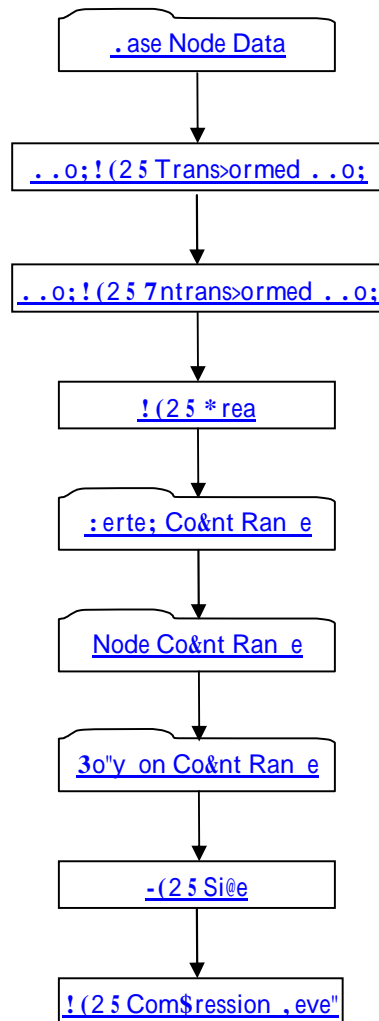
!i &re 255 .ase Sha\$e Node)"ement data #o"#e#tion



-om/lete escri/ption for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

C.2.1.1.10.1.1 Base SHA-256 Data

!i &re 265 .ase Sha\$e Data data #o"e#tion



-om/lete escri/tion for :ase +o e Data can 0e fo!n in [C.2.1.1.1.1.1 :ase +o e Data](#)

33o:F)# 2 Transforme* 33o:

T(e Transforme : :o> is an a>is-aligne +-S 0o!n ing 0o> an re/resents t(e transforme geometr# e>tents for all geometr# containe in t(e S(a/e +o e.

33o:F)# 2 1ntransforme* 33o:

T(e 4ntransforme : :o> is an a>is-aligne "-S 0o!n ing 0o> an re/resents t(e !ntransforme geometr# e>tents for all geometr# containe in t(e S(a/e +o e.

F)# 2 +rea

'area is the total surface area for this node and all of its descendants. This value is stored in +-S coordinate space i.e. values scale 0# +-S scaling².

I)# 2 Size

Size specifies the in memory length of the associated reference S(a/e ", D .lement. This size value (as no relevance to the on-disk file size of the associated reference S(a/e ", D .lement. 'value of zero indicates that the in memory size is unknown. See [C.2.2.1S\(a/e ", D .lement](#) for complete description of S(a/e ", D .lements. JT file loaders can leverage this size value during late load processing to determine if there is sufficient memory to load the S(a/e ", D .lement.

F)# 2 Compression Level

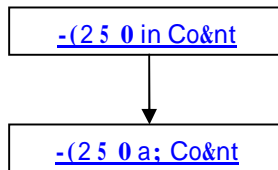
Compression level specifies the qualitative compression level applied to the associated reference S(a/e ", D .lement. See [C.2.2.1S\(a/e ", D .lement](#) for complete description of S(a/e ", D .lements. This compression level value is a qualitative representation of the compression applied to the S(a/e ", D .lement. The absolute compression derive from this qualitative level applied to the S(a/e ", D .lement is /(#sical# represented in the JT format other data store %it(out(the S(a/e +o e an the S(a/e ", D .lement e.g. [C.2.1.1.10.2.1G!antiHation Parameters](#)) and this is not necessarily in order to understand the qualitative value to absolute compression values in order to understand the data

[0.0	- I"osslessK compression level .
[0.1	- I\$ minimal# "oss#K compression level . This setting generally results in most compression ratios %it(little if an# visual difference %en compare to the same images rendered from I"osslessK compression S(a/e ", D .lement.
[0.E	- I\$ moderate "oss#K compression level . This setting results in more data loss than I\$ minimal# "oss#K and this (igger compression ratio is obtained . Some visual difference %ill likely be noticeable %en compare to the same images rendered from I"osslessK compression S(a/e ", D .lement.
[1.0	- I' ggressive "oss#K compression level . = it(this setting as much data as possible %ill be retained # resulting in (ighest compression ratio) %ile still maintaining a most# !sea0le representation of the original data. Visual differences may be evident %en compare to the same images rendered from I"osslessK compression S(a/e ", D .lement.

/.#1.1.1.1<1.1.1 Verte: Count Range

Verte> -o!nt Range is the aggregate minimum and maximum verte> count for this S(a/e +o e. There is a minimum and maximum value to accommodate s(a/e t#es that can themselves generate varying representations. The minimum value represents the least verte> count that can be achieved 0# the S(a/e +o e. The maximum value represents the greatest verte> count that can be achieved 0# the S(a/e +o e.

!i &re 2%5 :erte; Co&nt Ran e data #o"e#tion



I)# 2 9 in Count

\$ in -o!nt is t(e least verte> co!nt t(at can 0e ac(ieve 0# t(is S(a/e +o e.

I)# 2 9a: Count

\$a> -o!nt is t(e ma>im!m verte> co!nt t(at can 0e ac(ieve 0# t(is S(a/e +o e. ' val!e of I-1K in icates ma>im!m verte> co!nt is !n3no%n.

/.#.1.1.1.1<.1.1.# 'o*e Count Range

+o e -o!nt Range is t(e aggregate minim!m an ma>im!m co!nt of all no e escen ants of t(e S(a/e +o e. T(e minim!m val!e re/resents t(e least no e co!nt t(at can 0e ac(ieve 0# t(e S(a/e +o e)\s escen ants. T(e ma>im!m val!e re/resents t(e greatest no e co!nt t(at can 0e ac(ieve 0# S(a/e +o e)\s escen ants. For S(a/e +o es t(e minim!m an ma>im!m co!nt val!es s(o!l al%a#s 0e eA!al to I1K.

T(e ata format for +o e -o!nt Range is t(e same as t(at escri0e in [C.2.1.1.1.10.1.1.1Verte> -o!nt Range](#).

/.#.1.1.1.1<.1.1.) Pol!gon Count Range

Pol#gon -o!nt Range is t(e aggregate minim!m an ma>im!m /ol#gon co!nt for t(is S(a/e +o e. T(ere is a minim!m an ma>im!m val!e to accommo ate s(a/e t#/es t(at can t(emselves generate var#ng re/resentations. T(e minim!m val!e re/resents t(e least /ol#gon co!nt t(at can 0e ac(ieve 0# t(e S(a/e +o e. T(e ma>im!m val!e re/resents t(e greatest /ol#gon co!nt t(at can 0e ac(ieve 0# t(e S(a/e +o e.

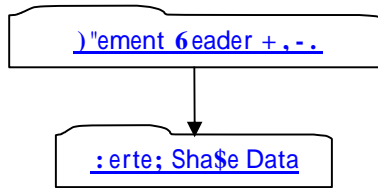
T(e ata format for Pol#gon -o!nt Range is t(e same as t(at escri0e in [C.2.1.1.1.10.1.1.1Verte> -o!nt Range](#).

/.#.1.1.1.1<.# Verte: S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>10 10Cf) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

Verte> S(a/e +o e . lement re/resents s(a/es efine 0# collections of vertices.

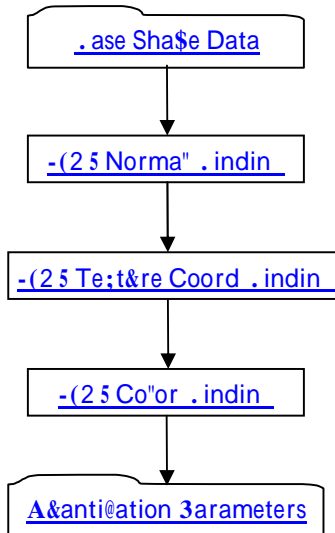
!i &re 285 :erte; Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

C.2.1.1.10.2.1 Verte: S(a/e Data

!i &re 2'5 :erte; Sha\$e Data data #o"e#tion



-om/lete escri/tion for : ase S(a/e Data can Oe fo!n in [C.2.1.1.10.1.1 : ase S\(a/e Data.](#)

l)# 2 'ormal 3in*ing

+ormal :in ing s/ecifies (o% lat %(at gran!larit#2 normal vector ata is s!/lie 1I0o!n K2 for t(e s(a/e in t(e associate s/reference S(a/e " ,D .lement. See [C.2.2.1S\(a/e " ,D .lement](#) for com/lete escri/tion of S(a/e " ,D .lements.

[0	- +one. S(a/e (as no normal ata.
[1	- Per Verte>. S(a/e (as a normal vector for ever# verte>.
[2	- Per Facet. S(a/e (as a normal vector for ever# face5/ol#gon.

[B	– Per Primitive. S(a/e (as a normal vector for each (s(a/e /rimitive le.g. a normal for each tri-stri/ in a tri-stri/ set).
-----	--

1) # 2 Te:ture Coord* 3in*ing

Te>t!re -oor :in ing s/ecifies (o% lat %(at gran!larit#2 te>t!re coor inate ata is s!//lie II0o!n K2 for t(e s(a/e in t(e associate \$reference S(a/e " ,D .lement. Vali val!es are t(e same as oc!mente for [B2* +ormal :in ing](#) ata fiel .

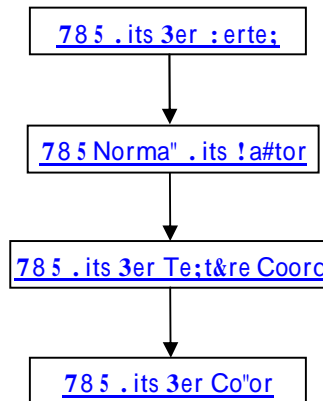
1) # 2 Color 3in*ing

-olor :in ing s/ecifies (o% lat %(at gran!larit#2 color ata is s!//lie II0o!n K2 for t(e s(a/e in t(e associate \$reference S(a/e " ,D .lement. Vali val!es are t(e same as oc!mente for [B2* +ormal :in ing](#) ata fiel .

/.#1.1.1.1<.#.1.1 >uantit=ation Parameters

G!antiHation Parameters s/ecifies for each (s(a/e ata t#e gro!/ing li.e. Verte>) +ormal) Te>t!re -oor inates) -olor2 t(e n!m0er of A!antiHation Oits !se for given A!alitative com/ression level. ' It(o!g(t(ese G!antiHation Parameters val!es are save in t(e associate \$reference S(a/e " ,D .lement) t(e# are also save (ere so t(at a JT File loa er\$rea er oes not (ave to loa t(e S(a/e " ,D .lement in or er to etermine t(e S(a/e A!antiHation level. See [C.2.2.1S\(a/e " ,D .lement](#) for com/lete escri/tion of S(a/e " ,D .lements.

!i &re (05 A&anti@ation 3arameters data #o""e#tion



182 3its Per Verte:

: its Per Verte> s/ecifies t(e n!m0er of A!antiHation Oits /er verte> coor inate com/onent. Val!e m!st 0e %it(in range R0*2DS incl!sive.

182 'ormal 3its Factor

+ormal :its Factor is a /arameter !se to calc!late t(e n!m0er of A!antiHation Oits for normal vectors. Val!e m!st 0e %it(in range R0*1BS incl!sive . T(e act!al n!m0er of A!antiHation Oits /er normal is com/!te !sing t(is factor an t(e follo%ing form!la* I : itsPer+ormal [F \ 2] +ormal :its FactorK

18 2 3its Per Te:ture Coor*

: its Per Te>t!re -oor s/ecifies t(e n!m0er of A!antiHation Oits /er te>t!re coor inate com/onent. Val!e m!st 0e %it(in range R0*2DS incl!sive.

18 2 3its Per Color

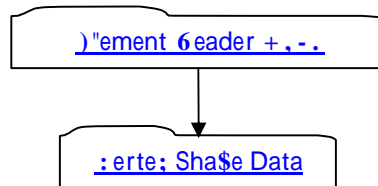
: its Per -olor s/ecifies t(e n!m0er of A!antiHation Oits /er color com/onent. Val!e m!st 0e %it(in range R0*2DS incl!sive.

/.#.1.1.1.1<.) Tri-Stri Set S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>10 10CC) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

' Tri-Stri/ Set S(a/e +o e .lement efines a collection of in e/en ent an !nconnecte triangle stri/s. .ac(stri/ constit!tes one /rimitive of t(e set an is efine 0# one list of verte> coor inates.

!i &re (15 Tri/Stri\$ Set Sha\$e Node)"ement data #o""e#tion



-om/lete escri/tion for .lement ;ea er L"&: can 0e fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

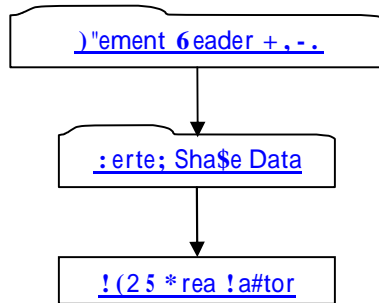
-om/lete escri/tion for Verte> S(a/e Data can 0e fo!n in [C.2.1.1.1.10.2.1Verte> S\(a/e Data.](#)

/.#.1.1.1.1<., Pol!line Set S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>10 10DF) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

' Pol#line Set S(a/e +o e .lement efines a collection of in e/en ent an !nconnecte /ol#lines. .ac(/ol#line constit!tes one /rimitive of t(e set an is efine 0# one list of verte> coor inates.

!i &re ((25 3o"y"ine Set Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

-om/lete escri/tion for Verte> S(a/e Data can Oe fo!n in [C.2.1.1.10.2.1Verte> S\(a/e Data](#).

F)# 2 +rea Factor

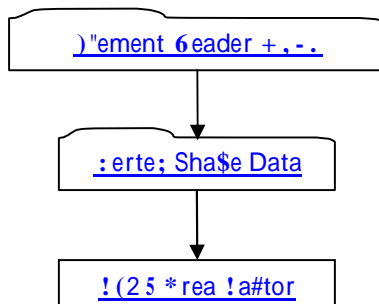
'rea Factor s/ecifies a m!lti/lier factor a//lie to a Pol#line Set com/!te s!rface area. &nJT ata vie%er a//lications t(ere ma# Oe " , D selection semantics t(at are Oase on screen coverage calc!lations. T(e so-calle Ks!rface areaK of a /ol#line is com/!te as if eac(line segment %ere a sA!are. T(is 'rea Factor t!rns eac(e ge into a narro% rectangle. Vali 'rea Factor val!es lie in t(e range 10)1S.

/.#.1.1.1.1<.- Point Set S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>981BDC1F) 0>0010) 0>0818) 0>19) 0>98) 0>08) 0>00) 0>09) 0>8B) 0>E) 0>Ea

' Point Set S(a/e +o e .lement efines a collection of in e/en ent an !nconnecte /oints. .ac(/oint constit!tes one /rimitive of t(e set an is efine 0# one verte> coor inate.

!i &re ((5 3oint Set Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

-om/lete scri/ption for Verte> S(a/e Data can 0e fo!n in [C.2.1.1.1.10.2.1Verte> S\(a/e Data](#).

F)# 2 +rea Factor

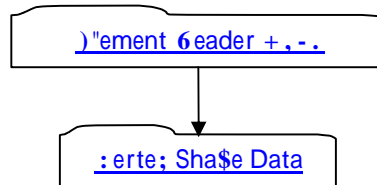
'rea Factor s/ecifies a m!ltiplier factor a//lie to t(e Point Set com/!te s!rface area. &n JT ata vie%er a//lications t(ere ma# 0e " , D selection semantics t(at are 0ase on screen coverage calc!lations. T(e com/!te Is!rface areaK of a Point Set is eA!al to t(e larger li.e. %(ic(ever is greater2 of eit(er t(e area of t(e Point Set\%s 0o!n ing 0o>) or I1.0K. 'rea Factor scales t(e res!lt of t(is Is!rface areaK com/!tation..

/.#.1.1.1.1<.. Pol!gon Set S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>10 10D8) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

' Pol#gon Set S(a/e +o e .lement efines a collection of in e/en ent an !nconnecte /ol#gons. .ac(/ol#gon constit!tes one /rimitive of t(e set an is efine 0# one list of verte> coor inates.

!i &re (45 3o"y on Set Sha\$e Node)"ement data #o"e#tion



-om/lete scri/ption for .lement ;ea er L"&: can 0e fo!n in [C.1.B.2.2 .lement ;ea er L"&:.](#)

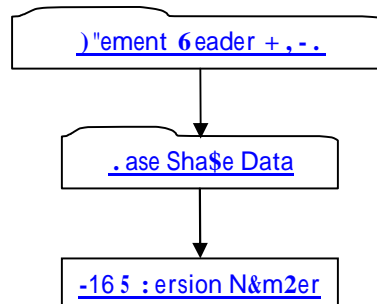
-om/lete scri/ption for Verte> S(a/e Data can 0e fo!n in [C.2.1.1.1.10.2.1Verte> S\(a/e Data](#).

/.#.1.1.1.1<./ ' 1 "" S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0> 2B9eC0F) 0> CC) 0>D289) 0>a0) 0>C) 0>00) 0>ee) 0>C9) 0>fC) 0>9D) 0>9D

' + 4 "" S(a/e +o e .lement efines a s(a/e %(ic((as no irect geometric /rimitive re/resentation li.e. it is em/t#5+ 4 ""2. + 4 "" S(a/e +o e .lements are often !se as I/ro>#5/lace(ol erK no es %(in t(e serialiHe "S6 %(en t(e act!al S(a/e " , D ata is r!n time generate li.e. not /ersiste 2.

!i &re (55 N7 , , Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

-om/lete escri/tion for :ase S(a/e Data can Oe fo!n in [C.2.1.1.1.10.1.1 :ase S\(a/e Data.](#)

11. 2 Version 'umber

Version +!m0er is t(e version i entifier for t(is no e. Version n!m0er I0>0001K is c!rrent!# t(e on!# vali val!e for +4 " " S(a/e +o e .lement.

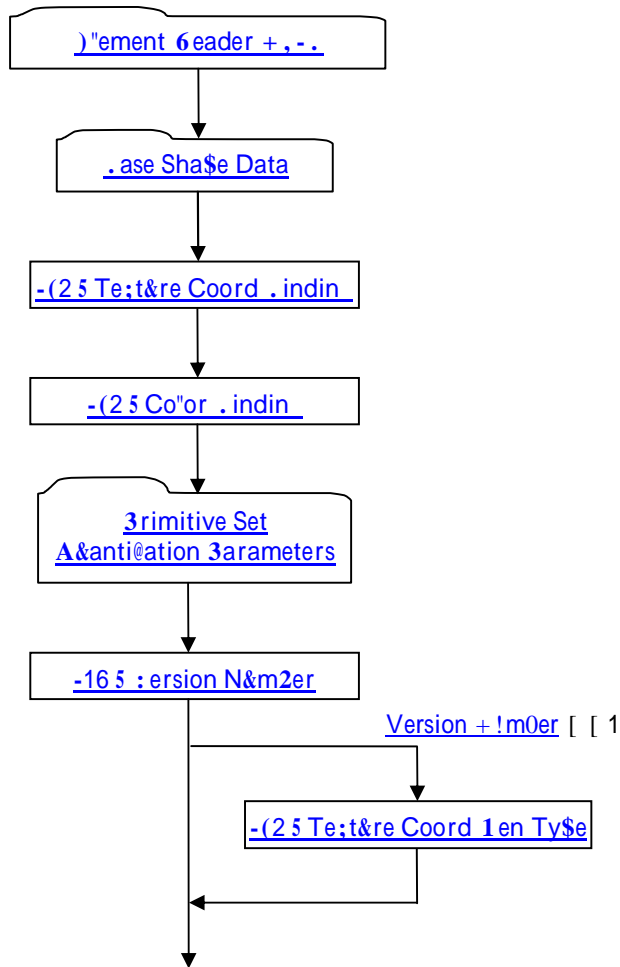
1.#.1.1.1.1<.8 Primitive Set S%a e 'o*e 6lement

42?e#t Ty\$e -D5 0>eD0BCBc1) 0>1a 9) 0>11 B) 0>9) 0>af) 0>0) 0>a0) 0>c9) 0>cC) 0>) 0>c2

' Primitive Set S(a/e +o e .lement re/resents a listset of /rimitive s(a/es le.g. 0o>) c#lin er) s/(ere) etc.2 %o\%s " ,Ds can Oe /roce !rall# generate . IProce !rall# generateK means t(at t(e ra% geometric s(a/e efinition ata le.g. vertices) /ol#gons) normals) etc2 for " ,Ds is not irect!# store ? instea some Oasic s(a/e information is store le.g. s/(ere center an ra i!s2 from %(ic(" ,Ds can Oe generate .

Primitive Set S(a/e +o e .lements act!all# o not even irect!# contain t(is Oasic s(a/e efinition ata) O!t instea reference !!sing "ate "oa e Pro/ert# 'toms2 [Primitive Set S\(a/e .lements](#) %it(in t(e file for t(e act!al Oasic s(a/e efinition ata. Storing t(e Oasic s(a/e efinition ata %it(in se/arate in e/en ent!# a ressaOle ata segments in t(e JT file) allo%a JT file rea er to Oe str!ct!re to s!//ort t(e IOest /racticeK of ela#ing t(e loa ing!rea ing of associate ata !ntil it is act!all# nee e . -om/lete escri/tions for "ate "oa e Pro/ert# 'tom .lements an [Primitive Set S\(a/e .lement](#) can Oe fo!n in [C.2.1.2.C "ate "oa e Pro/ert# 'tom .lement](#) an [C.2.2.2 Primitive Set S\(a/e .lement](#) res/ectivel#.

!i &re (65 3rimitive Set Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for . lement ;ea er L"&: can 0e fo!n in [C.1.B.2.2 . lement ;ea er L"&:](#).

-om/lete escri/tion for : ase S(a/e Data can 0e fo!n in [C.2.1.1.1.10.1.1 : ase S\(a/e Data.](#)

I)# 2 Te:t!re Coor* 3in*ing

Te>t!re -oor :in ing s/ecifies (o% lat %(at gran!larit#2 te>t!re coor inate ata is s!//lie I!0o!n K2 for t(e s(a/e in t(e associate sreference S(a/e " , D . lement. Vali val!es are as follo%\$*

[0	- +one. S(a/e (as no te>t!re coor inate ata.
[1	- Per Verte>. S(a/e (as te>t!re coor inates for ever# verte>.

I)# 2 Color 3in*ing

-olor :in ing s/ecifies (o% lat %(at gran!larit#2 color ata is s!//lie II0o!n K2 for t(e s(a/e in t(e associate sreference S(a/e " ,D .lement. Vali val!es are t(e same as oc!mente for [Text!re -oor :in ing](#) ata fiel .

11.2 Version Number

Version +!m0er is t(e version i entifier for t(is element. T(e val!e of t(is Version +!m0er in icates t(e format of ata fiel s to follo%.

[0	- Version 0 Format
[1	- Version 1 Format

1) #2 Text!re Coor* 5en T! e

Text!re -oor 6en T#/e s/ecifies (o% te>t!re coor inates are to 0e -

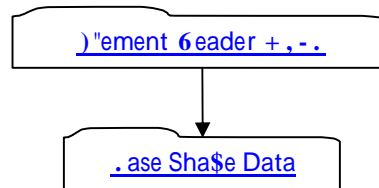
1.1.1.1.1 <; \$ ire 0 arness Set S%a e ' o * e 6 lement

42?e#t Ty\$e -D5 0>DccCaE21) 0>C28) 0>11 B) 0>9) 0>80) 0>0) 0>a0) 0>c9) 0>cC) 0>) 0>c2

' = ire ; arness Set S(a/e +o e . lement re/resents a list of %ire (arness s(a/es. = (ere a %ire (arness is define as a single man!fact!re %ire !nit consisting of several /(#sical electrical %ires all 0o!n toget(er into a 0ranc(ing str!ct!re of %ire 0!n les t(at terminate at connectors. ' = ire ; arness Set S(a/e +o e . lement is meant to /roce !rall# generate its ",Ds. IProce !rall# generateK means t(at t(e ra% geometric s(a/e efinition ata le.g. vertices) /ol#gons) normals) etc2 for ",Ds is not irectl# store ? instea some escri/tive s(a/e information is store from %(ic(",Ds can 0e generate lif esire 2 at loa time.

= ire ; arness Set S(a/e +o e . lements act!all# o not even irectl# contain t(is escri/tion s(a/e efinition ata) 0!t instea reference !!sing "ate "oa e Pro/ert# ' toms2 = ire ; arness Set S(a/e . lement %it(in t(e file for t(e act!al escri/tive s(a/e efinition ata. Storing t(e escri/tive s(a/e efinition ata %it(in se/arate in e/en ent!# a ressa0le ata segments in t(e JT file allo% a JT file rea er to 0e str!ct!re to s!//ort t(e I0est /racticeK of ela#ing t(e loa ing5rea ing of associate ata !ntil it is act!all# nee e . -om/lete escri/tions for "ate "oa e Pro/ert# ' tom . lements an = ire ; arness Set S(a/e . lement can 0e fo!n in [C.2.1.2.C "ate "oa e Pro/ert# ' tom . lement](#) an [C.2.2.B = ire ; arness Set S\(a/e . lement](#) res/ectivel#.

!i &re (85 Wire 6 arness Set Sha\$e Node)"ement data #o"e#tion



-om/lete escri/tion for . lement ; ea er L"&: can 0e fo!n in [C.1.B.2.2 . lement ; ea er L"&:](#).

-om/lete escri/tion for : ase S(a/e Data can 0e fo!n in [C.2.1.1.10.1.1 : ase S\(a/e Data](#).

1.1.1.# +ttribute 6 lements

' ttri0!te . lements le.g. color te>t!re) material) lig(ts) etc.2 are /lace in "S6 as o07ects associate %it(no es. ' ttri0!te . lements are not no es t(emselves) 0!t can 0e associate %it(an# no e.

For a//lications /ro !cing or cons!ming JT format ata) it is im/ortant t(at t(e JT format semantics of (o% attri0!tes are meant to 0e a//lie an acc!m!late o%n t(e "S6 are follo%e . &f not follo%e) t(en consistenc# 0et%een t(e a//lications in terms of BD /ositioning an ren ering of "S6 mo el ata %ill not 0e ac(ieve .

To t(at en eac(attri0!te t#/e efines its o%n a//lication an acc!m!lation semantics) 0!t in general attri0!tes at lo%er levels in t(e "S6 ta3e /rece ence an re/lace or acc!m!late %it(attri0!tes set at (ig(er levels. +o es %it(o!t associate attri0!tes in(erit t(ose of t(eir /arents. ' ttri0!tes in(erit onl# from t(eir

/arents) t(!s a no e\ s attri0!tes o not affect t(at no e\ s si0lings. T(e root of a /artition in(erits t(e attri0!tes in effect at t(e referring /artition no e.

' ttri0!tes can 0e eclare IfinalK lsee [C.2.1.1.2.1.1: ase ' ttri0!te Data2](#)) %(ic(terminates acc!m!lation of t(at attri0!te t#/e at t(at attri0!te an /ro/agates t(e acc!m!late val!es t(ere to all escen ants of t(e associate no e. Descen ants can e>/licitl# o a one-s(ot overri e of IfinalK !sing t(e attri0!te IforceK flag lsee [C.2.1.1.2.1.1: ase ' ttri0!te Data2](#)) 0!t o not 0# efa!lt. +ote t(at IforceK oes not t!rn , FF IfinalK Q it is sim/l# a one-s(ot overri e of IfinalK for t(e



,07ect &D is t(e identifier for t(is ,07ect. , t(er o07ects referencing t(is /artic!lar o07ect o so !sing t(e ,07ect &D.

18.2 State Flags

State Flags is a collection of flags. T(e flags are com0ine !sing t(e 0inar# , R o/erator an store vario!s state information for ' ttri0!te .lements? s!c(as in icating t(at t(e attri0!tes acc!m!lation is final. ' ll !n oc!mente 0its are reserve .

0>01	<p>– 'cc!m!lation Final flag. Provi es a means to terminate a /artic!lar attri0!te t#/e!s acc!m!lation at an# no e of t(e "S6 an t(ere0# force all escen ants to ave t(at val!e of t(e attri0!te. [0 Q 'cc!m!lation is to occ!r normall# [1 Q 'cc!m!lation is IfinalK</p>
0>02	<p>– 'cc!m!lation Force flag. Provi es a %a# to assign no es in "S6 attri0!tes t(at m!st not 0e overri en 0# ancestors. [0 Q 'cc!m!lation of t(is attri0!te o0e#s ancestor!s Final flag setting. [1 Q 'cc!m!lation of t(is attri0!te is force !overri es ancestor!s Final flag setting2</p>
0>0D	<p>– 'cc!m!lation &gnore Flag Provi es a %a# to in icate t(at t(e attri0!te is to 0e ignore !not acc!m!late 2. [0 Q ' ttri0!te is to 0e acc!m!late normall# !s!07ect to val!es of Force!Final flags2 [1 Q ' ttri0!te is to 0e ignore .</p>

1)# 2 Fiel* In%ibit Flags

Fiel &n(i0it Flags is a collection of flags. T(e flags are com0ine !sing t(e 0inar# , R o/erator an store t(e /er attri0!te val!e acc!m!lation flag. .ac(val!e /resent in an ' ttri0!te .lement is given a fiel n!m0er ranging from 0 to B1. &f t(e fiel !s corres/on ing 0it in &n(i0it Flags is set) t(en t(e fiel s(o!l not /artic!late in attri0!te acc!m!lation. ' ll 0its are reserve .

See eac(/artic!lar ' ttri0!te .lement le.g. \$ aterial ' ttri0!te .lement2 for a escri!tion of 0it fiel assignments for eac(attri0!te val!e.

/.#1.1.#9aterial +ttribute 6lement

42?e#t Ty\$e -D5 0>10 10B0) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

\$ aterial ' ttri0!te .lement efines t(e reflective c(aracteristics of a material. JT format "S6 traversal semantics ictate t(at material attri0!tes acc!m!late o%n t(e "S6 0# re/lacement.

T(e Fiel &n(i0it flag lsee [C.2.1.1.2.1.1:ase ' ttri0!te Data2](#) 0it assignments for t(e \$ aterial ' ttri0!te .lement ata fiel s) are as follo%#s*

!ie"d-nhi2it !"a .it	Data !ie"dBsC .it *\$\$ies To
0	'_m0ient -ommon R6 : Val!e) '_m0ient -olor
1	Diff!se -olor
2	S/ec!lar -ommon R6 : Val!e) S/ec!lar -olor
B	.mission -ommon R6 : Val!e) .mission -olor

D	: len_ing Flag) So!rce : len_ing Factor) Destination : len_ing Factor
E	.verri e Verte> -olor Flag

!i &re 415 0 ateria" * ttri2&te)"ement data #o""e#tion

Definizione di JT File Format Reference Version 8.1 Rev-D

rlmbmw n n

11.2 Data Flags

Data Flags is a collection of flags and factor data. The flags and factor data are combined using the Oinar#, R operator. The flags store information to be used for interpreting (or to read) the material data fields. The following are reserved.

The following Pattern Flags are used to optimize color data storage size to a single FB2 for the common case where the color is defined as Rc) c) 1.0S i.e. R6 : values are the same Ick value and 'l/(a is al#a#s I1.0K2.

0>0001	<p>– Pattern flag bits are used flag i.e. 'm0ient's mission's/ec!lar /attern flags2</p> <p>[0 Q Pattern bits are to be ignored .</p> <p>[1 Q Pattern bits are valid .</p>
0>0002	<p>– 'm0ient Pattern Flag</p> <p>[0 Q 'm0ient data store as for FB2.</p> <p>[1 Q 'm0ient data store as one FB2 and res!ltant color eA!als Rc) c) 1.0S</p>
0>000D	<p>– .mission Pattern Flag</p> <p>[0 Q .mission data store as for FB2.</p> <p>[1 Q .mission data store as one FB2 and res!ltant color eA!als Rc) c) 1.0S</p>
0>0008	<p>– S/ec!lar Pattern Flag</p> <p>[0 Q S/ec!lar data store as for FB2.</p> <p>[1 Q S/ec!lar data store as one FB2 and res!ltant color eA!als Rc) c) 1.0S</p>
0>0010	<p>– : len ing Flag. : len ing is a color combining operation in the graphics pipeline that allows the first color to be written to the frame buffer. &f : len ing is , + (the incoming fragment R6 : ' color values are used base on Source : len Factor2 and existing frame buffer R6 : ' color values are used base on Destination : len Factor2 to the incoming fragment R6 : ' and the current frame buffer R6 : ' to arrive at a new R6 : ' color to be written into the frame buffer. &f : len ing is , FF (the incoming fragment R6 : ' color is written directly into the frame buffer i.e. com/letel# overriding existing frame buffer R6 : ' color2. ' Additional information on (or one might leverage the : len ing Flag and : len ing Factors to render an image can be found in the references listed in section B References and Additional Information.</p> <p>[0 Q : len ing , FF.</p> <p>[1 Q : len ing , +</p>
0>0020	<p>– ,verri e Verte> -olors Flag. &f , +) (the source/destination colors are to be overridden) (the accumulated material color.</p> <p>[0 Q ,verri e , FF</p> <p>[1 Q ,verri e , +</p>
0>0C-0	<p>– Source : len Factor store in bits F Q 10 or in Oinar# notation 00000111110000002. &f : len ing Flag enable (this value indicates (or the incoming fragment i.e. the source R6 : ' color values are to be used to the destination i.e. the destination R6 : ' color values. ' Additional information on the interpretation of the : len ing Factor values and (or one might leverage them to render an image can be found in reference RDS listed in section B References and Additional Information.</p>

	<p>[0 Q &nter/ret same as , /en 6 " 1 ,D+)R4 : len ing Factor</p> <p>[1 Q &nter/ret same as , /en 6 " 1 ,D4N) : len ing Factor</p> <p>[2 Q &nter/ret same as , /en 6 " 1 ,DDSTDC4 , 4R : len ing Factor</p> <p>[B Q &nter/ret same as , /en 6 " 1 ,DSRCDC4 , 4R : len ing Factor</p> <p>[D Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSTDC4 , 4R : len ing Factor</p> <p>[E Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSRCDC4 , 4R : len ing Factor</p> <p>[F Q &nter/ret same as , /en 6 " 1 ,DSRCD* , 36* : len ing Factor</p> <p>[C Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSRCD* , 36* : len ing Factor</p> <p>[8 Q &nter/ret same as , /en 6 " 1 ,DDSTD* , 36* : len ing Factor</p> <p>[9 Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDDSTD* , 36* : len ing Factor</p> <p>[10 Q &nter/ret same as , /en 6 " 1 ,DSRCD* , 36*DS*T7R*T) : len ing Factor</p>
0>F800	<p>– Destination : len Factor lstore in 0its 11 Q 1E or in 0inar# notation 1111100000000002. 2. &f : len ing Flag ena0le) t(is val!e in icates (o% t(e c!rrent frame0!fferNs lt(e estination2 R6 : ' color val!es are to 0e !se to 0len %it(t(e incoming fragmentNs lt(e so!rce2 R6 : ' color val!es. ' itional information on t(e inter/retation of t(e : len ing Factor val!es an (o% one mig(t leverage t(em to ren er an image can 0e fo!n in reference RDS liste in section B References an ' itional &nformation.</p> <p>[0 Q &nter/ret same as , /en 6 " 1 ,D+)R4 : len ing Factor</p> <p>[1 Q &nter/ret same as , /en 6 " 1 ,D4N) : len ing Factor</p> <p>[2 Q &nter/ret same as , /en 6 " 1 ,DDSTDC4 , 4R : len ing Factor</p> <p>[B Q &nter/ret same as , /en 6 " 1 ,DSRCDC4 , 4R : len ing Factor</p> <p>[D Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSTDC4 , 4R : len ing Factor</p> <p>[E Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSRCDC4 , 4R : len ing Factor</p> <p>[F Q &nter/ret same as , /en 6 " 1 ,DSRCD* , 36* : len ing Factor</p> <p>[C Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDSRCD* , 36* : len ing Factor</p> <p>[8 Q &nter/ret same as , /en 6 " 1 ,DDSTD* , 36* : len ing Factor</p> <p>[9 Q &nter/ret same as , /en 6 " 1 ,D4N)D0 -N7SDDSTD* , 36* : len ing Factor</p> <p>[10 Q &nter/ret same as , /en 6 " 1 ,DSRCD* , 36*DS*T7R*T) : len ing Factor</p>

F)# 2 +mbient Common R53 Value

' m0ient -ommon R6 : Val!e is t(e assigne val!e for t(e Re) 6reen) an :!le com/onents of t(e am0ient color li.e. Re) 6reen) an :!le are all eA!al to t(is same val!e? R [6 [: [val!e2. ' Iso t(e ' l/(a com/onent is al%a#s ass!me to 0e eA!al to I1.0K. ' m0ient -ommon R6 : Val!e is onl# /resent if [' m0ient Pattern Flag](#) eA!als 1.

R53+ 2 +mbient Color

' m0ient -olor s/ecifies t(e am0ient re) green) 0!le) al/(a color val!es of t(e material. ' m0ient -olor is onl# /resent if [' m0ient Pattern Flag](#) eA!als 0.

R53+ 2 Diffuse Color

Diff!se -olor s/ecifies t(e iff!se re) green) 0!le) al/(a color val!es of t(e material.

F)# 2 S ecular Common R53 Value

S/ec!lar -ommon R6 : Val!e is t(e assigne val!e for t(e Re) 6reen) an :!le com/onents of t(e s/ec!lar color li.e. Re) 6reen) an :!le are all eA!al to t(is same val!e? R [6 [: [val!e2. ' Iso t(e ' l/(a com/onent is al%a#s ass!me to 0e eA!al to I1.0K. S/ec!lar -ommon R6 : Val!e is onl# /resent if [S/ec!lar Pattern Flag](#) eA!als 1.

R53+2 S ecular Color

S/ec!lar -olor s/ecifies t(e s/ec!lar re) green) 0!e) al/(a color val!es of t(e material. S/ec!lar -olor is on!# /resent if [S/ec!lar Pattern Flag](#) eA!als 0.

F)# 2 6mission Common R53 Value

.mission -ommon R6 : Val!e is t(e assigne val!e for t(e Re) 6reen) an :!e com/onents of t(e emissive color li.e. Re) 6reen) an :!e are all eA!al to t(is same val!e? R [6 [: [val!e2. ' Iso t(e ' l/(a com/onent is al#a#s ass!me to 0e eA!al to I1.0K. .mission -ommon R6 : Val!e is on!# /resent if [.mission Pattern Flag](#) eA!als 1.

R53+2 6mission Color

.mission -olor s/ecifies t(e emissive re) green) 0!e) al/(a color val!es of t(e material. .mission -olor is on!# /resent if [.mission Pattern Flag](#) eA!als 0.

F)# 2 S%ininess

S(ininess is t(e e>/onent associate %it(s/ec!lar reflection an (ig(lig(ting. S(ininess controls t(e egree %it(%(ic(t(e s/ec!lar (ig(lig(t eca#s. , n!# val!es in t(e range R1)128\$ are vali .

/.#.1.1.#.) Te>ture Image +ttribute 6lement

42?e#t Ty\$e -D5 0>10 10CB) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

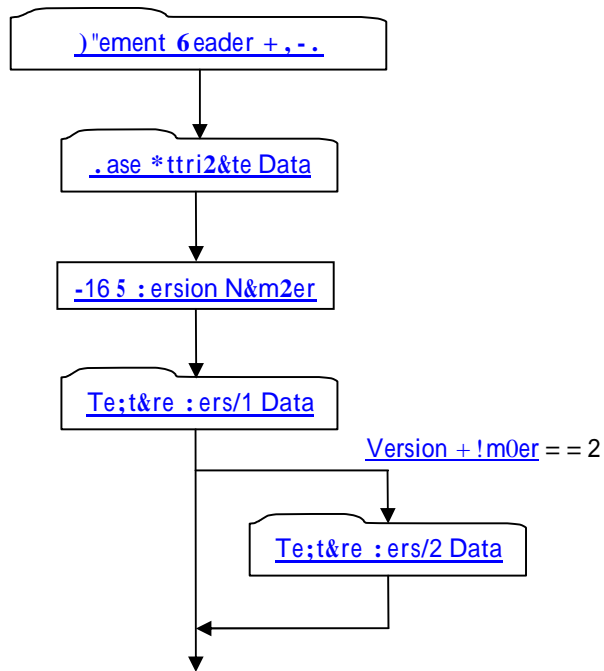
Te>tre &image 'ttri0!te .lement efines a te>tre image an its ma//ing environment. JT format "S6 traversal semantics ictate t(at te>tre image attri0!tes acc!m!late o%n t(e "S6 0# re/lacement.

+ote t(at a itional information on t(e inter/retation of t(e vario!s Te>tre &image 'ttri0!te .lement ata fiel s can 0e fo!n in t(e ,/en6 " references liste in section [B References an ' itional &nformation.](#)

T(e Fiel &n(i0it flag lsee [C.2.1.1.2.1.1: ase 'ttri0!te Data?](#) 0it assignments for t(e Te>tre &image 'ttri0!te .lement ata fiel s) are as follo%S*

!ie"d-nhi2it !"a .it	Data !ie'dBsC .it *\$\$ies To
0	Te>tre T#/e) &image Te>el Data) \$i/ma/ &image Te>el Data) .>ternal Storage +ame) S(are &image Flag
1	:or er \$o e) :or er -olor
2	\$i/ma/ \$inification Filter) \$i/ma/ \$agnification Filter
B	S-Dimen =ra/ \$o e) T-Dimen =ra/ \$o e) R-Dimen =ra/ \$o e
D	:len T#/e) :len -olor
E	Te>tre Transform
F	Te> -oor 6en \$o e) Te> -oor Reference Plane) .nvironment \$a//ing Flag
8	&nternal -om/ression "evel

!i &re 425 Te;t&re -ma e *ttri2&te)"ement data #o"e#tion



- om/lete escri/tion for .lement ; ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ; ea er L"&:](#).
- om/lete escri/tion for : ase ' ttri0!te Data can Oe fo!n in [C.2.1.1.2.1.1 : ase ' ttri0!te Data.](#)

11.2 Version Number

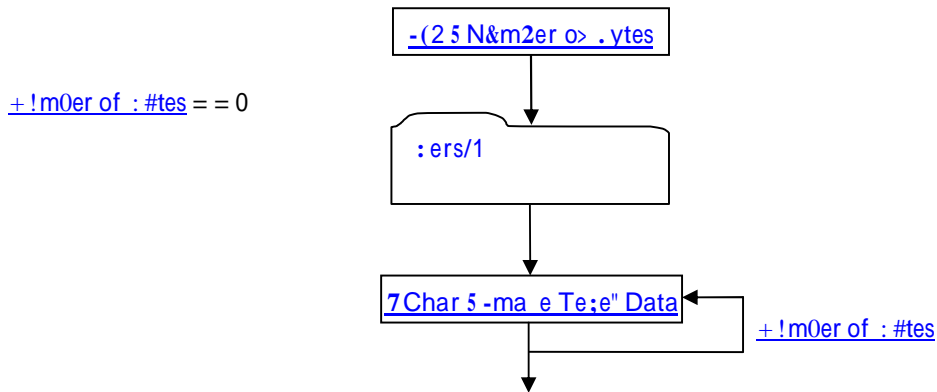
Version + !m0er is the version identifier for this element. The value of this Version + !m0er indicates the format of data fields to follow.

[1	- Version-1 Format
[2	- Version-2 Format

1.1.1.#.1 Te:ture Vers-1 Data

Texture Vers-1 Data format is stored in JT file if the Texture Image element is a vanilla OASIS texture image i.e. if texture does not use advanced features as described in [C.2.1.1.2.B.2 Texture Vers-2 Data](#).

!i &re 4(5 Te;t&re :ers/1 Data data #o"e#tion



-om/lete details for Vers-1 Image Format Description can be found in [C.2.1.1.2.B.1.1 Vers-1 Image Format Description](#).

-om/lete details for Vers-1 Te>t're Environment can be found in [C.2.1.1.2.B.1.2 Vers-1 Te>t're Environment](#).

1) # 2 'umber of 3 !tes

+!m0er of : #tes specifies the length() in 0#tes) of the on- is3 representation of the te>t're image. The te>t're image in a JT file is a single monolithic contiguous block of data beginning with the (ig(est-level mi/ image) and processing through the mi/ma/s on to a one-0#-one te>el image. If there are no mi/ma/s then the n!m0er of 0#tes is for a single te>t're image. If +!m0er of : #tes is zero then no other data is stored.

1 Character Image Te:el Data

Image Te>el Data is the single monolithic contiguous block of image data. The length() of this field in 0#tes is specified by the value of data field [+!m0er of : #tes](#).

C.2.1.1.2.B.1.1 Vers-1 Image Format Description

The Vers-1 Image Format Description is a collection of data defining the image format data type and other miscellaneous characteristics of the monolithic block of image data.

!i &re 445 :ers/1 -ma e !ormat Des#ri\$tion data #o"e#tion

1)#

1) # 2 Pixel Data Type

Pixel Data Type specifies the data type used to store the pixel data. If the [Pixel Format](#) represents a multi-component value (e.g. RGB) then each value represents the Pixel Data Type of the number of bytes of storage (e.g. a Pixel Format Type of 1KB means the Pixel Data Type of 1KB means 120 bytes of storage for each pixel).

[0	- +0 type specifies. Text rendering is not allowed.
[1	- Signed 8-bit integer
[2	- Single-precision 32-bit floating point
[B	- 4 signed 8-bit integers
[D	- Single Bytes in 4 signed 8-bit integers
[E	- 4 signed 16-bit integers
[F	- Signed 16-bit integer
[C	- 4 signed 32-bit integers
[8	- Signed 32-bit integer
[9	- 16-bit floating point according to IEEE-754 format (i.e. 1 sign bit, 5 exponent bits, 10 mantissa bits)

1. 2 Dimensionality

Dimensionality specifies the number of dimensions of the image (as valid values include 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292, 293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805, 806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 819, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832, 833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886, 887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 900, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913, 914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 927, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940, 941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 954, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967, 968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 981, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994, 995, 996, 997, 998, 999, 1000).

[1	- 1-dimensional text
[2	- 2-dimensional text

1) # 2 \$i*t%

\$i*t% specifies the image dimensionality of the image in terms of pixels.

1) # 2 0eig%

eig% specifies the eigenspace dimensionality of the image in terms of pixels. eig% is 1KB for one-dimensional images.

1) # 2 9i ma s Flag

\$i/ma/s Flag is a flag indicating whether the image is a multi-media/s.

[0	- +0 mi/ma/s
[1	- 9es (as mi/ma/s. Image Text Data is assumed to contain multiple text lines) each a mi/ma/ of the base text. If the text is in /o%er of t%o m%st 0e /rovi e 0et%een t(e 0ase te>t!re an a one-0#-one te>t!re.

1) # 2 S%are* Image Flag

S(are &image Flag is a flag indicating whether the image is a single element attribute.

[0	- &image is not a single element attribute.
[1	- &image is a single element attribute.

C.2.1.1.2.B.1.2 Vers-1 Te:ture 6nvironment

The Vers-1 Te>t!re .nvironment is a collection of data defining vario!s as/ects of (o% a te>t!re image is to 0e ma//e 5a//lie to a s!rface.

!i &re 455 :ers/1 Te;t&re)nvironment data #o"e#tion

e0

e0

l)# 2 9i ma 9agnification Filtera4a0J

[1	- nearest. The element with coordinates nearest to the center of the element is selected.
[2	- "linear" eigens. Linear average of the $2 > 2$ array of elements nearest to the center of the element is selected. For one-dimensional there is average of 2 elements. For three-dimensional there is $2 > 2 > 2$ array.
[B	- nearest in i/m . = it (in an individual i/m) the element with coordinates nearest to the center of the element is selected.
[D	- "linear in i/m . = it (in an individual i/m) a eigens. Linear average of the $2 > 2$ array of elements nearest to the center of the element is selected. For one-dimensional there is average of 2 elements. For three-dimensional there is $2 > 2 > 2$ array.
[E	- nearest between $i/m/s$. = it (in each of the adjacent $i/m/s$) selects the element with coordinates nearest to the center of the element and then interpolates linearly between these two selected i/m values.

	Information. [0 – +one. [1 – Decal. &nter/ret same as , /en6 " 1 ,DD)C* , environment mo e. [2 – \$o !late. &nter/ret same as , /en6 " 1 ,D0 4D7 , *T) environment mo e. [B – Re/lace. &nter/ret same as , /en6 " 1 ,DR)3 , *C) environment mo e. [D – : len . &nter/ret same as , /en6 " 1 ,D. ,)ND environment mo e. [E – ' . &nter/ret same as , /en6 " 1 ,D*DD environment mo e. [F – -om0ine. &nter/ret same as , /en6 " 1 ,DC4 0 .-N) environment mo e.
: it B	. nvironment \$ a//ing Flag. +ote t(at if t(is flag is , + li.e. [12) t(en a//lications /rocessing t(is JT ata for BD gra/(ical vis!aliation s(o!l a!tomaticall# t!rn , + te>t!re coor inate generation for s/(erical environment ma/s. [0 Q , FF [1 Q , +
: its D - B1	Reserve for f!t!re !se.

R53+ 2 3len* Color

: len -olor s/ecifies t(e color to 0e !se for I: len K Te>t!re . nvironment \$ o e o/erations.

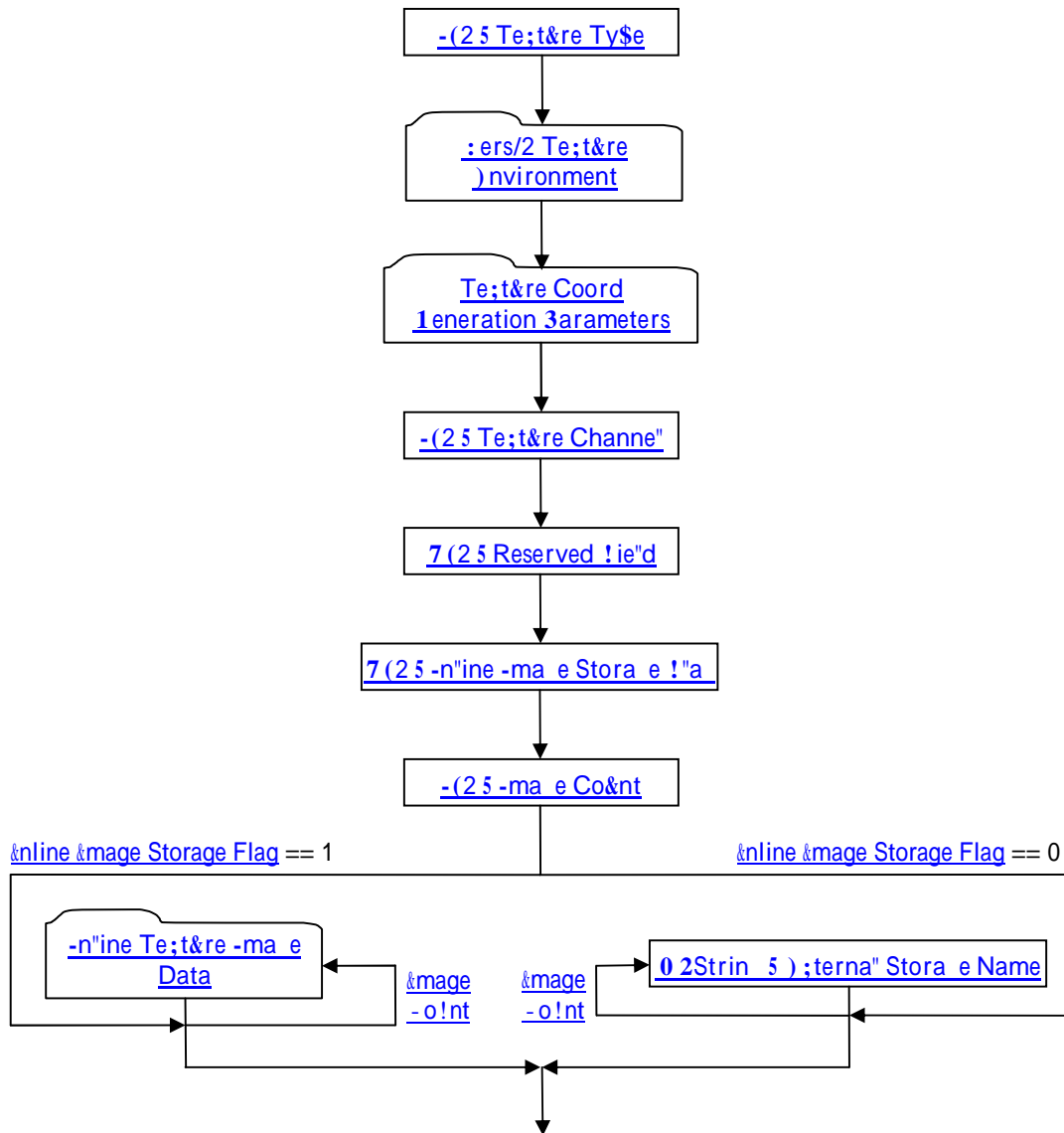
9 : ,F)# 2 Te:t!re Transform

Te>t!re Transform efines t(e te>t!re coor inate transformation matri>. ' ren erer of JT ata %o!! t#/icall# a//l# t(is transform to te>t!re coor inates /rior to a//l#ing t(e te>t!re.

/ #.1.1.#.)# Te:t!re Vers-# Data

Te>t!re Vers-2 Data collection s!//orts te>t!ring effects not re/resenta0le in t(e [Te>t!re Vers-1 Data](#) format le.g. m!lti/le te>t!res li.e. c(annels2) te>t!re image storage location e>ternal to t(e JT file) t(ree- imensional te>t!res) ot(er t(an !nsigne -0#te ata formats) mirror an e ge50or er coor inate clam/ mo es) etc.2. ' n# [Te>t!re &image 'ttri0!te .lement](#) !sing t(e Te>t!re Vers-2 Data format %ill contain a I egenerateK Te>t!re Vers-1 Data 0loc3) %(ere +!m0er of :#tes ata fiel (as a val!e of IOK.

!i &re 465 Te;t&re :ers/2 Data data #o"e#tion



-om/lete etails for Vers-2 Te>t!re .nvironment can 0e fo!n in [C.2.1.1.2.B.2.1 Vers-2 Te>t!re .nvironment](#).

-om/lete etails for Te>t!re -oor 6eneration Parameters can 0e fo!n in [C.2.1.1.2.B.2.2 Te>t!re -oor 6eneration Parameters](#).

-om/lete etails for &nline Te>t!re &image Data can 0e fo!n in [C.2.1.1.2.B.2.B &nline Te>t!re &image Data](#).

1) # 2 Te:ture T! e

Te>t!re T#/e s/ecifies t(e t#/e of te>t!re.

[0	- +one.
[1	- ,ne-Dimensional. ' one- imensional te>t!re (as a (eig(t 1T-Dimension2 an e/t(1R-Dimension2 eA!al to I1K an no to/ or 0ottom 0or er.
[2	- T%o-Dimensional. ' t%o- imensional te>t!re (as a e/t(1R-Dimension2 eA!al to I1.K
[B	- T(ree-Dimensional. ' t(ree- imensional te>t!re can Oe t(o!g(t of as la#ers of t%o- imensional s!0 image rectangles arrange in a seA!ence.
[D	- : !m/ \$a/. ' 0!m/ ma/ te>t!re is a te>t!re %(ere t(e image te>el ata le.g. R6 : color val!es2 re/resents s!rface normal 8 9L com/onents.
[E	- - !0e \$a/. ' c!0e ma/ te>t!re is a te>t!re c!0e centere at t(e origin an forme 0# a set of si>t%o- imensional te>t!re images.

1) # 2 Te:ture C%annel

Te>t!re - (annel s/ecifies t(e te>t!re c(annel n!m0er for t(e Te>t!re &image .lement. For /!r/oses of m!lti-te>t!ring) t(e JT conce/t of a te>t!re c(annel corres/on s irect# to t(e ,/en6" conce/t of a Ite>t!re !nit.K T(e Te>t!re - (annel val!e m!st Oe 0et%een 0 an B1 incl!sive. : est /ractices s!ggest t(at ren erer of JT ata ignore all 0!t c(annel-0 if t(e ren erer oes not s!//ort m!lti-te>t!re geometr#. ' Iso for /!r/oses of 0len ing) ren erer of JT ata s(o!l ass!me t(at ig(er n!m0ere te>t!re c(annels IOlen overK lo%er n!m0ere ones.

1) # 2 Reserve* Fiel*

Reserve Fiel is a ata fiel reserve for f!t!re JT format e>/ansion.

1) # 2 Inline Image Storage Flag

&inline &image Storage Flag is a flag t(at in icates %(et(er t(e te>t!re image is store %it(in t(e JT File li.e. inline2 or in some ot(er e>ternal file.

[0	- Te>t!re image store in an e>ternal file.
[1	- Te>t!re image store inline in t(is JT file.

1) # 2 Image Count

&image -o!nt s/ecifies t(e n!m0er of te>t!re images. ' I - !0e \$a/K [Te>t!re T#/e](#) m!st (ave si> images %(ile all ot(er Te>t!re T#/es s(o!l on!# (ave one image.

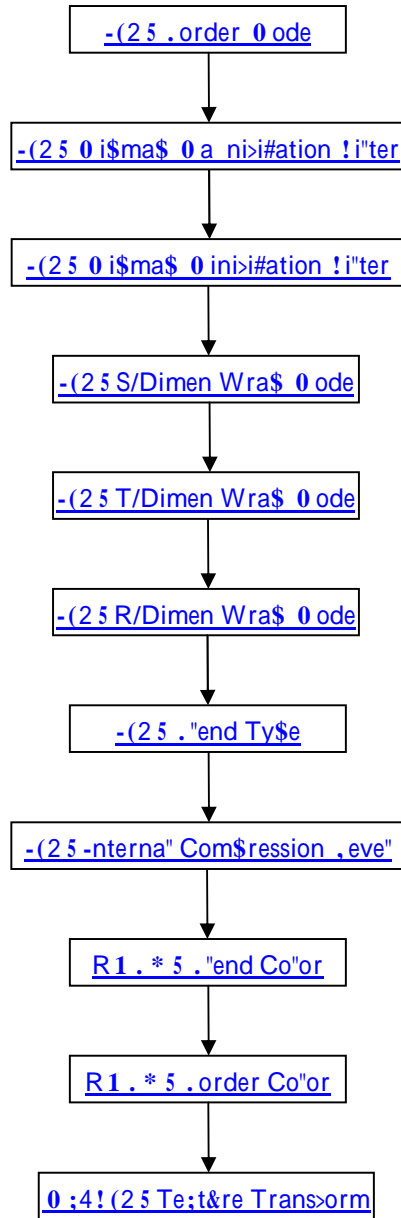
9bString 2 6 :ternal Storage +ame

.>ternal Storage +ame is a string i entif#ing t(e name of an e>ternal te>t!re image storage. .>ternal Storage +ame is on!# /resent if ata fiel [&inline &image Storage Flag](#) eA!als IO.K &f /resent t(ere %ill Oe ata fiel [&image -o!nt](#) n!m0er of .>ternal Storage +ame instances. T(is .>ternal Storage +ame string is a relative /at(0ase name for t(e te>t!re image file. = (ere Irelative /at(K s(o!! Oe inter/rete to mean t(e string contains t(e file name along %it(an# a itional /at(information t(at locates t(e te>t!re image file relative to t(e location of t(e referencing JT file.

C.2.1.1.2.B.2.1 Vers-# Te:ture 6nvironment

The Vers-2 Te>t!re .nvironment is a collection of data defining vario!s as/ects of (o% a te>t!re image is to Oe ma//e 5a//lie to a s!rface.

!i &re 4%5 :ers/2 Te;t&re)nvironment data #o"e#tion



l)# 2 3or*er 9o*e

: or er \$ o e s/ecifies t(e te>t!re 0or er mo e.

[0	- +o 0or er.
[1	- -onstant :or er -olor. &n icates t(at t(e te>t!re (as a constant 0or er color %(ose val!e is efine in ata fiel :or er -olor .
[2	- .>/licit. &n icates t(at a 0or er te>el ring is /resent in t(e te>t!re image efinition.

I)#2 9i ma 9agnification Filter

\$i/ma/ \$agnification Filter s/ecifies t(e te>t!re filtering met(o to a//l# %(en a single /i>el on screen ma/s to a tin# /ortion of a te>el.

[0	- +one.
[1	- +earest. Te>el %it(coor inates nearest t(e center of t(e /i>el is !se .
[2	- "inear. ' %eig(te linear average of t(e 2 > 2 arra# of te>els nearest to t(e center of t(e /i>el is !se . For one- imensional te>t!re is average of 2 te>els. For t(ree- imensional te>el is 2 > 2 > 2 arra#.

I)#2 9i ma 9inification Filter

\$i/ma/ \$inification Filter s/ecifies t(e te>t!re filtering met(o to a//l# %(en a single /i>el on screen ma/s to a large collection of te>els.

[0	- +one.
[1	- +earest. Te>el %it(coor inates nearest t(e center of t(e /i>el is !se .
[2	- "inear. ' %eig(te linear average of t(e 2 > 2 arra# of te>els nearest to t(e center of t(e /i>el is !se . For one- imensional te>t!re is average of 2 te>els. For t(ree- imensional te>t!re is 2 > 2 > 2 arra#.
[B	- +earest in \$i/ma/. = it(in an in ivi !al mi/ma/) t(e te>el %it(coor inates nearest t(e center of t(e /i>el is !se .
[D	- "inear in \$i/ma/. = it(in an in ivi !al mi/ma/) a %eig(te linear average of t(e 2 > 2 arra# of te>els nearest to t(e center of t(e /i>el is !se . For one- imensional te>t!re is average of 2 te>els. For t(ree- imensional te>t!re is 2 > 2 > 2 arra#
[E	- +earest 0et%een \$i/ma/s. = it(in eac(of t(e a 7acent t%o mi/ma/s) selects t(e te>el %it(coor inates nearest t(e center of t(e /i>el an t(en inter/olates linearl# 0et%een t(ese t%o selecte mi/ma/ val!es.
[F	- "inear 0et%een \$i/ma/s. = it(in eac(of t(e t%o a 7acent mi/ma/s) com/!tes val!e 0ase on a %eig(te linear average of t(e 2 > 2 arra# of te>els nearest to t(e center of t(e /i>el an t(en inter/olates linearl# 0et%een t(ese t%o com/!te mi/ma/ val!es.

I)#2 S-Dimen \$ra 9o*e

S-Dimen =ra/ \$ o e s/ecifies t(e mo e for (an ling te>t!re coor inates S-Dimension val!es o!tsi e t(e range R0) 1s.

[0	- +one.
[1	- -lam/. ' n# val!es greater t(an 1.0 are set to 1.0? an# val!es less t(an 0.0 are set to 0.0
[2	- Re/eat &ninteger /arts of t(e te>t!re coor inates are ignore i.e. retains onl# t(e fractional com/onent o te>t!re coor inates grater t(an 1.0 an onl# one-min!s t(e fractional com/onent of val!es less t(an H0ero2. Res!lting in co/ies of t(e te>t!re ma/ tiling t(e

	s!rface
[B	- \$irror Re/eat. "i3e Re/eat) e>ce/t t(e s!rface tiles Ifli/-flo/K res!lting in an alternating mirror /attern of s!rface tiles.
[D	- -lam/ to . ge. :or er is al%#s ignore an instea te>el at or near t(e e ge is c(osen for coor inates o!tsi e t(e range R0) 1\$. = (et(er t(e e>act nearest e ge te>el or some average of t(e nearest e ge te>els is !se is e/en ent !/on t(e mi/ma/ filtering val!e.
[E	- -lam/ to :or er. +earest 0or er te>el is c(osen for coor inates o!tsi e t(e range R0) 1\$. = (et(er t(e e>act nearest 0or er te>el or some average of t(e nearest 0or er te>els is !se is e/en ent !/on t(e mi/ma/ filtering val!e.

I)#2 T-Dimen \$ra 9o*e

T-Dimen =ra/ \$o e s/ecifies t(e mo e for (an ling te>t!re coor inates T-Dimension val!es o!tsi e t(e range R0) 1\$. Same mo e val!es as oc!mente for [S-Dimen =ra/ \\$o e](#).

I)#2 R-Dimen \$ra 9o*e

R-Dimen =ra/ \$o e s/ecifies t(e mo e for (an ling te>t!re coor inates R-Dimension val!es o!tsi e t(e range R0) 1\$. Same mo e val!es as oc!mente for [S-Dimen =ra/ \\$o e](#).

I)#2 3len* T! e

:len T#/e contains information in icating (o% t(e val!es in t(e te>t!re ma/ are to 0e mo !late 5com0ine 50len e %it(t(e original color of t(e s!rface or some ot(er alternative color to com/!te t(e final color to 0e /ainte on t(e s!rface. ' itional information on t(e inter/retation of t(e :len T#/e val!es an (o% one mig(t leverage t(em to ren er an image can 0e fo!n in reference [RDS](#) liste in section [B References an ' itional &nformation](#).

[0	- +one.
[1	- Decal. &nter/ret same as , /en6 " 1,DD)C*, environment mo e.
[2	- \$o !late. &nter/ret same as , /en6 " 1,D0 4D7, *T) environment mo e.
[B	- Re/lace. &nter/ret same as , /en6 " 1,DR)3, *C) environment mo e.
[D	- :len . &nter/ret same as , /en6 " 1,D. ,)ND environment mo e.
[E	- ' . &nter/ret same as , /en6 " 1,D*DD environment mo e.
[F	- -om0ine. &nter/ret same as , /en6 " 1,DC4 0 .-N) environment mo e.

I)#2 Internal Com ession "evel

&nternal -om/ression "evel s/ecifies a ata com/ression (intrecommen ation t(at a JT file loa er is free to follo% for internall# lin memor#2 storing te>el ata. T(is setting oes not affect (o% image te>el ata is act!all# store in JT files or ot(er e>ternall# reference files.

[0	- +one. +o com/ression of te>el ata.
[1	- -onservative. "ossless com/ression of te>el ata.
[2	- \$o erate. Te>el com/onents tr!ncate to 8-0its eac(.
[B	- ' ggressive. Te>el com/onents tr!ncates to D-0its eac(lor E 0its for R6 : images2.

R53+ 2 3len* Color

:len -olor s/ecifies t(e color to 0e !se for t(e I: len K mo e of [:len T#/e](#) o/erations.

R53+ 2 3or*er Color

: or er -olor s/ecifies t(e constant 0or er color to !se for I - lam/ to :or erK st#le %ra/ mo es %(en t(e te>t!re itself oes not (ave a 0or er.

9 : ,F)# 2 Te:ture Transform

Te>t!re Transform efines t(e te>t!re coor inate transformation matri>. ' ren erer of JT ata %o!! t#/icall# a//l# t(is transform to te>t!re coor inates /rior to a//l#ing t(e te>t!re.

C.2.1.1.2.B.2.2 Te:ture Coor* 5eneration Parameters

Te>t!re -oor 6eneration Parameters contains information in icating if an (o% te>t!re coor inate com/onents s(o!! 0e a!tomaticall# generate for eac(of t(e D com/onents IS) T) R) G2 of a te>t!re coor inate.

!i &re 485 Te;t&re Coord 1eneration 3arameters data #o""e#tion

l)# 2 Te: Coor* 5en 9o*e

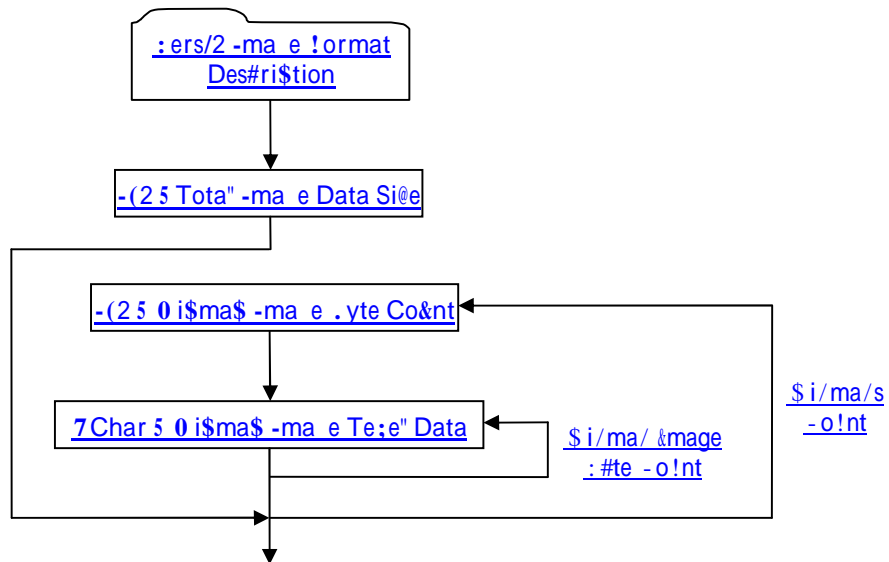
Te> -oor 6en \$o e s/ecifies t(e te>t!re coor inate generation mo e for eac(com/onent IS) T) R) G2 of te>t!re coor inate. T(ere are fo!r mo e val!es store) one for eac(com/onent of te>t!re coor inate. T(e mo e val!es are store in S) T) R) G or er.

- | | |
|-----|---|
| [0 | - +one. +o te>t!re coor inates a!tomaticall# generate . |
| [1 | - \$o el -oor inate S#stem "inear. Te>t!re coor inates com/!te as a istance from a reference /lane s/ecifie in mo el coor inates. |

C.2.1.1.2.B.2.B Inline Te:ture Image Data

Inline Te:ture Image Data is a collection of data defining the format/properties of an image te:el data for one te:ture image. Inline Te:ture Image Data is only present if the data field [Inline Image Storage Flag](#) equals 1. The field [Image -offset](#) is the number of Inline Te:ture Image Data instances.

!i &re 4'5 -n"ine Te;t&re -ma e Data data #o"#e#tion



The description for Vers-2 Image Format Description can be found in [C.2.1.1.2.B.2.B.1 Vers-2 Image Format Description](#).

1) # 2 Total Image Data Size

Total Image Data Size specifies the total length (in bytes) of the on-disk representation for all images. This is the total does not include the [B2 * \\$i/ma/ &image :#te -offset](#) data field storage ID bytes per image.

1) # 2 9i ma Image 3!te Count

[\\$i/ma/ &image :#te -offset](#) specifies the length (in bytes) of the on-disk representation of the next image.

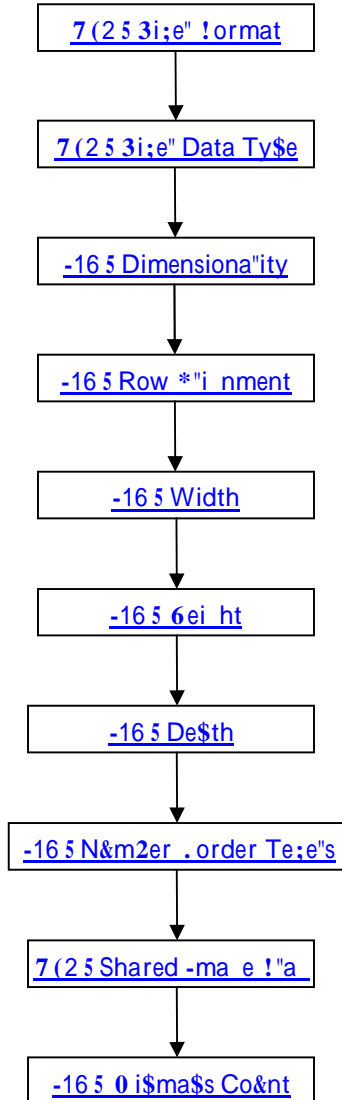
1) # 2 9i ma Image Te:el Data

[\\$i/ma/ &image Te:el Data](#) is the image data. The length of this field in bytes is specified by the value of the data field [\\$i/ma/ &image :#te -offset](#).

1) # 2 9i ma Image Te:el Data

The Vers-2 Image Format Description is a collection of data defining the image format and other miscellaneous characteristics of the image data.

!i &re 505 :ers/2 -ma e !ormat Des#ri\$tion data #o"e#tion



1)# 2 Pi:el Format

Pi>el format s/ecifies t(e format of t(e te>t!re image /i>el ata. De/en ing on t(e format) an#%(ere from one to fo!r elements of ata e>ists /er te>el.

[0	- +o format s/ecifie . Te>t!re ma//ing is not a//lie .
[1	- ' re color com/onent follo%e 0# green an 0!e color com/onents

[2 - ']

11.2 Number of Elements

The number of elements specifies the number of elements in the image definition. Valid values are 10K or 11K.

1) # 2 S are * Image Flag

The Image Flag is a flag indicating whether the image is scalable (0) or not (1).

[0	- Image is scalable (0) or not (1).
[1	- Image is not scalable (0) or not (1).

11.29 Images Count

The number of images specifies the number of images. The value of 11K indicates that no images are used. The value greater than 11K indicates that all the images are present to a 1-0#-1 level.

1.1.1.1, Dra (Style + attribute Element

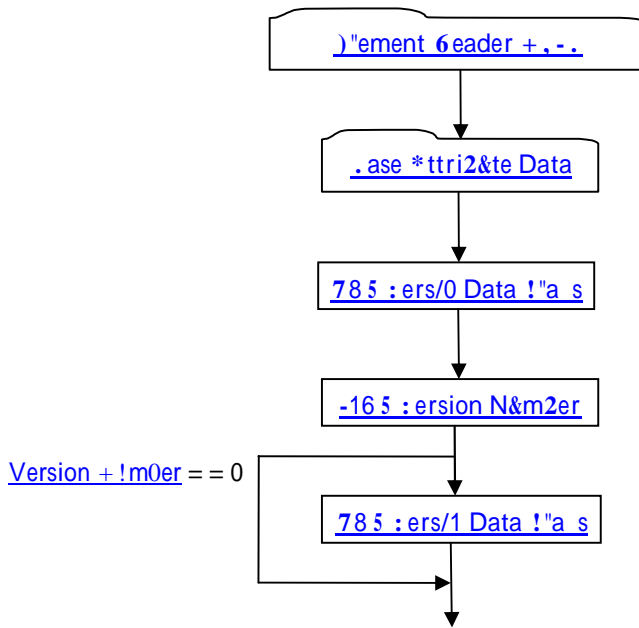
42?e#t Ty\$e -D5 0>10 101D) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

The Style attribute element contains information defining various aspects of the graphics state (at least one is required for rendering associated geometry). JT format "S6 traversal semantics dictate that the attribute values are as follows:

The field number flag (see [C.2.1.1.2.1.1: style attribute Data](#)) bit assignments for the Style attribute element data fields are as follows:

Field Number	Data Field Description
0	Toggle "Image Flag"
1	Toggle "Image Flag"
2	Toggle "Image Flag"
B	Toggle "Image Flag"
D	Toggle "Image Flag"
E	Toggle "Image Flag"

!i &re 515 Draw Style *ttri2&te)"ement data #o""e#tion



-om/lete escri/tion for .lement ;ea er L"&: can 0e fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

-om/lete escri/tion for :ase 'ttri0!te Data can 0e fo!n in [C.2.1.1.2.1.1 :ase 'ttri0!te Data.](#)

18 2 Vers-< Data Flags

Vers-0 Data Flags is a collection of flags. T(e flags are com0ine !sing t(e 0inar# ,R o/erator an store vario!s state settings for Dra% St#le 'ttri0!te .lements. 'll !n oc!mente 0its are reserve .

0>01	<p>- :ac3-face - !lling Flag. &n icates if 0ac3-facing /ol#gons s(o!! 0e iscar e !c!lle 2. [0 Q : ac3-facing /ol#gons not c!lle . [1 Q : ac3-facing /ol#gons c!lle .</p>
0>02	<p>- T%o Si e "lig(ting Flag. &n icates if t%o si e lig(ting s(o!! 0e ena0le to ins!re t(at 0ac3-facing /ol#gons are ill!minate . [0 Q Disa0le t%o si e lig(ting. [1 Q .na0le t%o si e lig(ting.</p>
0>0D	<p>- , !tline Pol#gons Flag &n icates if /ol#gons s(o!! 0e ra% in I%ire frame mo eK i.e. not fille ? onl# o!tlines ra%n. [0 Q Pol#gons ra%n as fille . [1 Q , nl# /ol#gons o!tline ra%n.</p>

11.2 Version Number

Version +!m0er is the version identifier for this element. The value of this Version +!m0er indicates if additional data follows.

[0	– Version-0 Format
[1	– Version-1 Format

18.2 Vers-1 Data Flags

Vers-1 Data Flags is a collection of flags. The flags are combined using the Or operator and store various state settings for Draft/State elements. Vers-1 Data Flags field is only present if [Version +!m0er](#) equals 1.1. The Vers-1 Data Flags includes the [Vers-0 Data Flags](#) data items some data flags are repeated 5 times along with some additional flags. All non-occupied bits are reserved.

0>01	– ac3-face -!ling Flag. Indicates if ac3-facing /ol#gons so! Oe iscar e ic!le 2. [0 Q : ac3-facing /ol#gons not c!lle . [1 Q : ac3-facing /ol#gons c!lle .
0>02	– T%o Si e "ig(ting Flag. Indicates if t%o si e lig(ting s(o!l Oe ena0le to ins!re t(at ac3-facing /ol#gons are ill!minate . [0 Q Disa0le t%o si e lig(ting. [1 Q .na0le t%o si e lig(ting.
0>0D	– , !tline Pol#gons Flag Indicates if /ol#gons s(o!l Oe ra% in I%ire frame mode i.e. not filled? on!# o!tlines ra%n. [0 Q Pol#gons ra%n as fille . [1 Q ,nl# /ol#gons o!tline ra%n.
0>08	– "ig(ting .na0le Flag Indicates if lig(ting s(o!l Oe ena0le . &f lig(ting isa0le)t(en ren erer s(o!l /erform no calculations concerning normals) lig(t so!rces) material /ro/erties) etc. [0 Q Disa0le lig(ting. [1 Q .na0le lig(ting.
0>10	– Flat S(a ing Flag Indicates if the geometr# s(o!l Oe ren ere %it(single color iflat s(a ing2 or %it(man# different color ifsmoot(56o!ra! 2 s(a ing. [0 Q Disa0le flat s(a ing li.e. !se smoot(56o!ra! s(a ing2. [1 Q .na0le flat s(a ing.
0>20	– Se/arate S/ec!lar Flag. Indicates if the application of the s/ec!lar color s(o!l Oe ela#e !ntil after te>t!ring. &f no te>t!re ma//ing t(en t(is flag setting is irrelevant. [0 Q ' //l# s/ec!lar color contri0!tion Oefore te>t!re ma//ing. [1 Q ' //l# s/ec!lar color contri0!tion after te>t!re ma//ing.

/.#.1.1.#.- "ig%t Set +ttribute 6lement

42?e#t Ty\$e -D5 0>10 109F) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

"ig(t Set 'ttri0!te .lement (ol s an !nor ere list of "ig(ts. JT format "S6 traversal semantics ictate t(at lig(t set attri0!tes acc!m!late o%n t(e "S6 t(ro!g(a ition of lig(ts to an attri0!te list.

"ig(t Set 'ttri0!te .lement oes not (ave an# Fiel &n(i0it flag lsee [C.2.1.1.2.1.1 : ase 'ttri0!te Data](#) 0it assignments.

!i &re 525 ,i ht Set *ttri2&te)"ement data #o"e#tion

-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement :ea er L"&:.](#)

-om/lete escri/tion for : ase 'ttri0!te Data can Oe fo!n in [C.2.1.1.2.1.1 : ase 'ttri0!te Data.](#)

l)# 2 "ig%t Count

"ig(t -o!nt s/ecifies t(e n!m0er of lig(ts in t(e "ig(t Set

l)# 2 "ig%t 4 b7ect ID

"ig(t , 07ect &D is t(e i entifier for a reference "ig(t , 07ect.

/.#.1.1.#.. Infinite "ig%t +ttribute 6lement

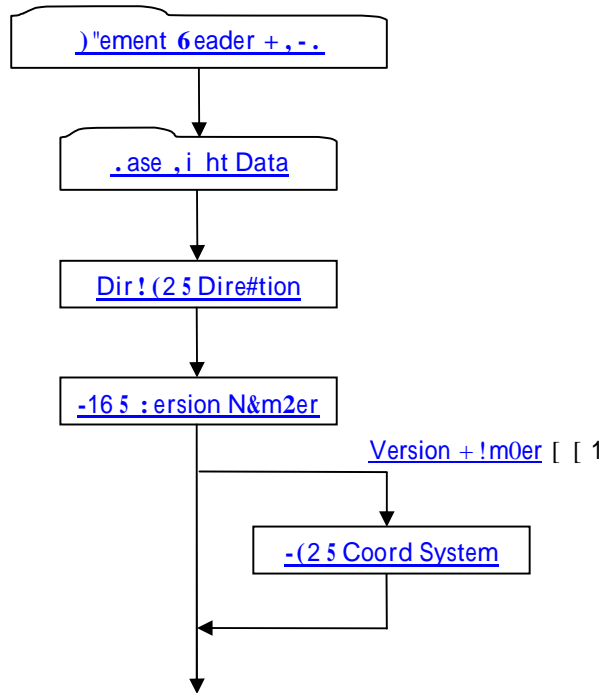
42?e#t Ty\$e -D5 0>10 1028) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

&infinite "ig(t 'ttri0!te .lement s/ecifies a lig(t so!rce emitting !natten!ate lig(t in a single irection from ever# /oint on an infinite /lane. T(e infinite location in icates t(at t(e ra#s of lig(t can Oe consi ere /arallel Oatcaif g(t#ig(

JT format "S6 traversal semantics ictate t(at infinite lig(t attri0!tes acc!m!late o%n t(e "S6 t(ro!g(a ition of lig(ts to an attri0!te list.

&finite "ig(t 'ttri0!te .lement oes not (ave an# Fiel &n(i0it flag lsee [C.2.1.1.2.1.1 : ase 'ttri0!te Data2](#) 0it assignments.

!i &re 5(5 -n>inite ,i ht *ttri2&te)"ement data #o"#e#tion



-om/lete escri/tion for .lement ;ea er L"&: can Oe fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#).

DirF)# 2 Direction

Direction s/ecifies t(e irection t(e lig(t is /ointing in.

11. 2 Version ' umber

Version +!m0er is t(e version i entifier for t(is element. T(e val!e of t(is Version +!m0er in icates t(e format of ata fiel s to follo%.

[0	- Version-0 Format
[1	- Version-1 Format

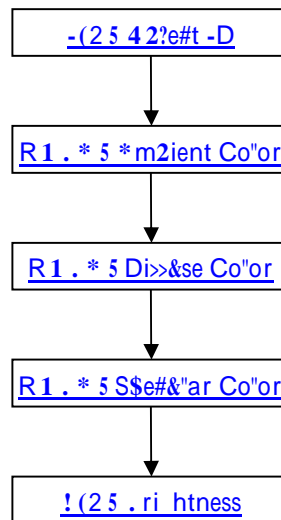
l)# 2 Coord* S!stem

-oor S#stem s/ecifies t(e coor inate s/ace in %(ic("ig(t so!rce is efine . Vali val!es incl! e t(e follo%ing*

[1	- Vie%/oint -oor inate S#stem. "ig(t so!rce is to move toget(er %it(t(e vie%/oint
[2	- \$o el -oor inate S#stem. "ig(t so!rce is affecte 0# %(atever mo el transforms t(at are c!rrent %(en t(e lig(t so!rce is enco!ntere in "S6.
[B	- =orl -oor inate s#stem. "ig(t so!rce is not affecte 0# mo el transforms in t(e "S6.

/.#.1.1.#...1 3ase "ig%t Data

!i &re 545 .ase ,i ht Data data #o"e#tion



I)# 2 4b7ect ID

,07ect &D is t(e i entifier for t(is ,07ect. , t(er o07ects referencing t(is /artic!lar o07ect o so !sing t(e ,07ect &D.

R53+ 2 +mbient Color

' m0ient -olor s/ecifies t(e am0ient re) green) 0l!e) al/ (a color val!es of t(e lig(t.

R53+ 2 Diffuse Color

Diff!se -olor s/ecifies t(e iff!se re) green) 0l!e) al/ (a color val!es of t(e lig(t.

R53+ 2 S ecular Color

S/ec!lar -olor s/ecifies t(e s/ec!lar re) green) 0l!e) al/ (a color val!es of t(e lig(t.

F) # 2 3 rig%tness

: rig(tness s/ecifies t(e "ig(t Orig(tness. T(e : rig(tness val!e m!st 0e greater t(an or eA!al to I-1K.

/.#.1.1.#./Point "ig%t +ttribute 6lement

42?e#t Ty\$e -D5 0>10 10DE) 0>2ac8) 0>11 1) 0>90) 0>F0) 0>00) 0>80) 0>cC) 0>00) 0>E9) 0>9C

Point "ig(t 'ttri0!te . lement s/ecifies a lig(t so!rce emitting lig(t from a s/ecifie /osition) along a s/ecifie irection) an %it(a s/ecifie s/rea angle

JT format "S6 traversal semantics ictate t(at /oint lig(t attri0!tes acc!m!late o%n t(e "S6 t(ro!g(a ition of lig(ts to an attri0!te list.

Point "ig(t 'ttri0!te . lement oes not (ave an# Fiel &n(i0it flag lsee [C.2.1.1.2.1.1 : ase 'ttri0!te Data? 0it assignments.](#)

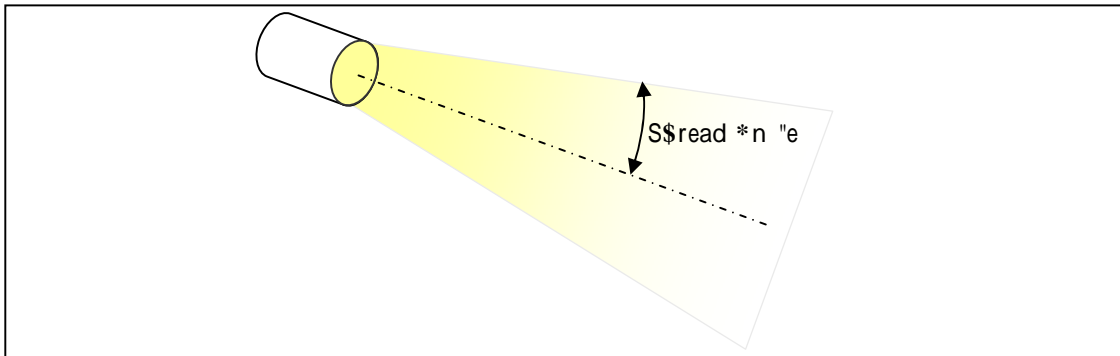
!i &re 555 3oint ,i ht *ttri2&te)"ement data #0"e#tion

e -om
rsmft nnf* e erm el.t00 e01 x00 g n - 00007149 6 d m(6-9)3 0 T dm(e)T jm-0.00002
-om

F) #2 S read * +ngle

S/read * +ngle specifies the angle in degrees (half angle of the light cone). Valid S/read * +ngle values are clamp/ed and inter/related as follows*

angle [[180.0	- Sim/le /oint lig(t
0.0 Z[angle Y[90.0	- S/ot "ig(t



!i &re 565 S/read * n 'e va"&e with res\$e#t to the "i ht #one

DirF) #2 S ot Direction

S/ot Direction specifies the direction (the s/ot lig(t is /ointing in.

I) #2 S ot Intensit!

S/ot &intensit# specifies the intensit# istri0!tion of the lig(t %it(in the s/ot lig(t cone. S/ot &intensit# is real# a Is/ot e>/onentK in a lig(ting eA!ation an in icates (o% foc!se t(e lig(t is at t(e center. T(e larger t(e val!e) t(e more foc!se t(e lig(t so!rce. , n!# non-negative S/ot intensit# val!es are vali .

11. 2 Version ' umber

Version +!m0er is the version identifier for this element. The value of this Version +!m0er indicates the format of data fields to follow.

[0	- Version-0 Format
[1	- Version-1 Format

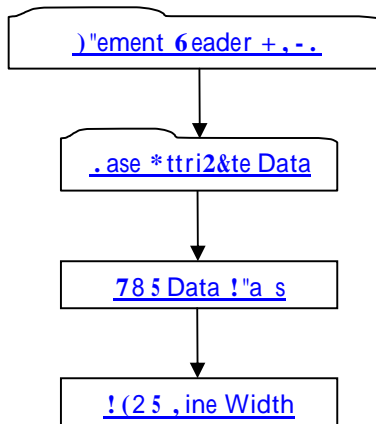
I) #2 Coor* S!stem

-oor S#stem specifies the coordinate space in which the lig(t so!rce is defined. Valid values include the following*

[1	- Vie%/oint -oor inate S#stem. "ig(t so!rce is to move together %it(t(e vie%/oint
[2	- \$o el -oor inate S#stem. "ig(t so!rce is affected 0# %(atever mo el transforms t(at are c!rrent %(en t(e lig(t so!rce is enco!ntere in "S6.
[B	- =orl -oor inate s#stem. "ig(t so!rce is not affected 0# mo el transforms in t(e "S6.

JT 111

!i &re 585 , inesty"e *ttri2&te)"ement data #o"e#tion



-om/lete escri/tion for .lement ;ea er L"&: can 0e fo!n in [C.1.B.2.2 .lement ;ea er L"&:](#)

-om/lete escri/tion for :ase 'ttri0!te Data can 0e fo!n in [C.2.1.1.2.1.1 :ase 'ttri0!te Data.](#)

18 2 Data Flags

Data Flags is a collection of flags an line t#/e ata. T(e flags an line t#/e ata are com0ine !sing t(e 0inar# , R o/erator an store vario!s /ol#line ren ering attri0!tes. 'll !n oc!mente 0its are reserve .

0>0F	<p>– "ine T#/e lstore in 0its 0 Q B or in 0inar# notation 000011112 "ine t#/e s/ecifies t(e /ol#line ren ering sti//le-/attem. [0 - Soli _____ [1 Q Das(----- [2 Q Dot [B Q Das(VDot ----- [D Q Das(VDotVDot ----- [E Q "ongVDas(----- [F Q -enterVDas(----- [C Q -enterVDas(VDas(-----</p>
0>10	<p>– 'ntialiasing Flag lstore in 0it D or in 0inar# notation 000100002 &n icates if antialiasing s(o!! 0e a//lie as /art of ren ering /ol#lines. [0 Q 'ntialiasing isa0le . [1 Q 'ntialiasing ena0le .</p>

F)# 2 "ine \$i*t%

"ine =i t(s/ecifies t(e %i t(in /i>els t(at s(o!! 0e !se for ren ering /ol#lines

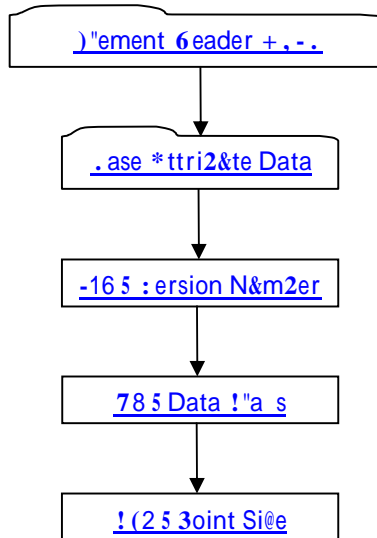
/.#.1.1.#.;Pointst!le +ttribute 6lement

42?e#t Ty\$e -D5 0>8 ECc010) 0>eEc0) 0>11 D) 0>8D) 0>e) 0>00) 0>a0) 0> 2) 0>18) 0>2f) 0>9

Pointst#le ' ttri0!te . lement contains information defining t(e gra/(ical /ro/erties t(at s(o!! Oe !se for ren ering /oints. JT format "S6 traversal semantics ictate t(at /ointst#le attri0!tes acc!m!late o%n t(e "S6 0# re/lacement.

Pointst#le ' ttri0!te . lement oes not (ave an# Fiel &n(ioit flag lsee [C.2.1.1.2.1.1 : ase ' ttri0!te Data](#) 0it assignments.

!i &re 5'5 3ointsty"e *ttri2&te)"ement data #o"e#tion



-om/lete escri/tion for . lement ; ea er L"&: can Oe fo!n in [C.1.B.2.2 . lement ; ea er L"&:](#).

-om/lete escri/tion for : ase ' ttri0!te Data can Oe fo!n in [C.2.1.1.2.1.1 : ase ' ttri0!te Data](#).

11. 2 Version ' umber

Version +!m0er is t(e version i entifier for t(is element. Version n!m0er IO>0001K is c!rrent!# t(e on!# vali val!e.

18 2 Data Flags

Data Flags is a collection of flags an /oint t#/e ata. T(e flags an /oint t#/e ata are com0ine !sing t(e 0inar# , R o/erator an store vario!s /oint ren ering attri0!tes. 'll !n oc!mente 0its are reserve .

0>0F	- Point T#/e lstore in 0its 0 Q B or in 0inar# notation 000011112 T(ese 0its are reserve for f!t!re e>/ansion of t(e format to s!//ort Point T#/es.
0>10	- 'ntialiasing Flag lstore in 0it D or in 0inar# notation 000100002 &n icates if antialiasing s(o!! Oe a//lie as /art of ren ering /oints. [0 Q 'ntialiasing isa0le .

-om/lete escri/tion for .lement ;ea er L": can Oe fo!n in [C.1.B.2.2 .lement ;ea er L":](#).

-om/lete escri/tion for :ase 'ttri0!te Data can Oe fo!n in [C.2.1.1.2.1.1 :ase 'ttri0!te Data](#).

11.2 Store* Values 9as?

Store Val!es mas3 is a 1F-0it mas3 % (ere eac(0it is a flag in icating % (et(er t(e corres/on ing element in t(e matri> is iffereent from t(e i entit# matri>. ,nl# elements % (ic(are iffereent from t(e i entit# matri> are act!all# store . T(e 0its are assigne to matri> elements as follo%*

```
: it1E : it1D : it1B : it12
: it11 : it10 : it9 : it8
: itC : itF : itE : itD
: itB : it2 : it1 : it0
```

T(e in ivi !al 0it-flag val!es are inter/rete as follo%*

[0	- Val!e not store !matri> val!e same as corres/on ing element in i entit# matri>2
[1	- Val!e store

F)# 2 6lement Value

.lement Val!e s/ecifies a /artic!lar matri> element val!e.

/.#.1.1.#.11 S%a*er 6ffects +ttribute 6lement

42?e#t Ty\$e -D5 0>aa108B1) 0>FeDC) 0>Dfee) 0>a8) 0>FE) 0>c) 0>Ce) 0>1f) 0>2f) 0>B9) 0> 0

S(a er .ffects 'ttri0!te .lement contains information s/ecif#ing I(ig(-levelK s(a er f!nctionalit# le.g. P(ong s(a ing) 0!m/ ma//ing) etc.2 t(at s(o!l Oe !se for ren ering t(e geometr# t(is attri0!te element is associate %it(.

JT format "S6 traversal semantics ictate t(at s(a er effects attri0!tes acc!m!late o%n t(e "S6 0# re/lacement.

S(a er .ffects 'ttri0!te .lement oes not (ave an# Fiel &n(i0it flag lsee [C.2.1.1.2.1.1 :ase 'ttri0!te Data2](#) 0it assignments.