

# **JEDEC PUBLICATION**

---

## **PartModel Package Guidelines for Electronic-Device Packages – XML Requirements**

---

### **JEP30-P100G**

(Revision of JEP30-P100F, October 2024)

**February 2025**

---

**JEDEC SOLID STATE TECHNOLOGY ASSOCIATION**



## NOTICE

JEDEC standards and publications contain material that has been prepared, reviewed, and approved through the JEDEC Board of Directors level and subsequently reviewed and approved by the JEDEC legal counsel.

JEDEC standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for use by those other than JEDEC members, whether the standard is to be used either domestically or internationally.

JEDEC standards and publications are adopted without regard to whether or not their adoption may involve patents or articles, materials, or processes. By such action JEDEC does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the JEDEC standards or publications.

The information included in JEDEC standards and publications represents a sound approach to product specification and application, principally from the solid state device manufacturer viewpoint. Within the JEDEC organization there are procedures whereby a JEDEC standard or publication may be further processed and ultimately become an ANSI standard.

No claims to be in conformance with this standard may be made unless all requirements stated in the standard are met.

Inquiries, comments, and suggestions relative to the content of this JEDEC standard or publication should be addressed to JEDEC at the address below, or refer to [www.jedec.org](http://www.jedec.org) under Standards and Documents for alternative contact information.

Published by  
©JEDEC Solid State Technology Association 2025  
3103 10<sup>th</sup> Street North  
Suite 240S  
Arlington, VA 22201

JEDEC retains the copyright on this material. By downloading this file the individual agrees not to charge for or resell the resulting material.

**PRICE: Contact JEDEC**

Printed in the U.S.A.  
All rights reserved

DO NOT VIOLATE  
THE  
LAW!

This document is copyrighted by JEDEC and may not be  
reproduced without permission.

For information, contact:

JEDEC Solid State Technology Association  
3103 10<sup>th</sup> Street North  
Suite 240S  
Arlington, VA 22201  
<https://www.jedec.org/contact>

This page intentionally left blank.



## PartModel Package Guidelines for Electronic-Device Packages – XML Requirements

### Contents

	Page
Contents .....	i
1 Scope.....	1
1.1 Purpose .....	1
2 Applicable Documents.....	1
2.1 JEDEC (www.jedec.org).....	1
2.2 IPC (www.ipc.org) .....	2
2.3 JEDEC/IPC (www.jedec.org).....	2
2.4 ASME (www.asme.org) .....	2
3 Requirements.....	2
3.1 Terms and Definitions .....	2
3.2 XML Schema Key Terms and Definitions .....	2
4 PartModel Schema Definition .....	3
4.1 PartModel - Package Section .....	3
4.2 Manufacturer Part Number-Array .....	5
4.3 Linking the MPN to a specific Package Family Data set.....	6
4.3.1 Linking the Manufacturing Part Number to Package Content .....	7
4.3.2 Linking the Manufacturing Part Number to Physical Model Content .....	8
4.3.3 Linking the Manufacturing Part Number to Die Content.....	9
5 Package Section - Package .....	10
5.1 Package Terminal Position.....	15
5.1.1 Bottom .....	16
5.1.2 Dual .....	17
5.1.2.1 Dual Left-to-Right and Back-to-Front.....	18
5.1.2.2 Dual Bottom-to-Upper .....	18
5.1.2.3 Dual Corners.....	19
5.1.3 Diagonal.....	20
5.1.4 Quad .....	21
5.1.5 Radial.....	22
5.1.6 Single.....	23
5.1.6.1 Single Side.....	24
5.1.6.2 Corner - Internal.....	24
5.1.6.3 Corner - External.....	25
5.1.7 Triple.....	26
5.1.7.1 Triple – Multi Side position .....	27
5.1.8 Upper .....	27
5.2 Package Outline Style.....	28
5.2.1 Microelectronics Assembly .....	29
5.2.2 Post Mount.....	29
5.3 Package Terminal Code.....	30
5.3.1 Ball.....	31
5.3.2 Lug.....	31
5.3.3 Flat.....	32
5.3.4 Gull-wing.....	32
5.3.5 Post Terminal.....	32
5.3.6 L-bend.....	33
5.3.7 Column .....	33
5.3.8 Surface Terminal.....	34

## Contents (cont'd)

	Page
5.3.9	Pressfit.....35
5.3.10	Pin.....35
5.3.11	Wraparound .....36
5.3.12	S-Bend.....37
5.3.13	Through-Hole.....37
5.3.14	Terminal Wire.....38
5.3.15	Screw.....38
5.4	Package Mounting Preparation .....38
5.5	Package Body Direction .....39
5.6	Mass .....39
5.7	Package Shape.....40
5.7.1	Contour .....42
5.7.2	Shape Center.....43
5.7.3	Package Vertical Dimension.....43
5.8	Part Height.....44
5.9	Clearance Region - Array .....45
5.10	Assembly Technology .....47
5.11	CTE - Array .....48
5.11.1	Temperature Condition.....49
5.11.2	Values.....50
5.11.3	RuleVsDirectional-xy-and-z-Rule .....51
5.11.4	CTE - Graph.....52
5.11.4.1	Formatting.....53
5.11.5	Data - Array.....56
5.11.5.1	Plot Test Condition.....57
5.11.6	Graph Formula.....58
5.12	Young Modulus - Array.....59
5.12.1	Youngs Modulus- Graph .....60
5.13	Heat Capacity - Array.....61
5.14	Terminal Groups .....62
5.14.1	Terminal Group - Array.....63
5.14.1.1	Terminal Position .....64
5.14.1.2	Terminal.....65
5.14.1.3	Terminal Material .....66
5.14.1.4	Terminal Function .....67
5.14.1.5	Terminal Location.....67
5.14.1.5.1	Standard Array.....68
5.14.1.5.2	Circular Array .....69
5.14.1.5.3	Random Array.....71
5.14.1.6	Pattern Group .....71
5.14.1.6.1	Pattern Relationship.....72
5.14.1.6.1.1	Relationship Transformations.....73
5.14.1.6.1.2	Duplicate.....74
5.14.1.7	Terminal Shape.....75
5.14.1.7.1	Ball Types .....85
5.14.1.7.2	C-bend .....86
5.14.1.7.3	Lug.....86
5.14.1.7.4	Flat.....87
5.14.1.7.5	Gull-wing.....88

## Contents (cont'd)

	Page
5.14.1.7.5.1 Modified Corner.....	89
5.14.1.7.5.1.1 Impacted Terminal .....	89
5.14.1.7.5.1.2 Impact – to – Terminal Group.....	90
5.14.1.7.5.1.2.1 Apply – to – all - Terminals .....	90
5.14.1.7.5.1.2.2 Rotation .....	91
5.14.1.7.5.1.2.3 Reflection .....	91
5.14.1.7.5.1.3 Shape Impacted Corner .....	92
5.14.1.7.5.1.4 Corner Type .....	93
5.14.1.7.5.2 Reference Terminal Contour Type .....	94
5.14.1.7.5.3 Neck.....	95
5.14.1.7.5.3.1 Neck Transition .....	95
5.14.1.7.5.3.1.1 Tapered .....	96
5.14.1.7.5.3.1.2 Right Angled .....	96
5.14.1.7.5.3.2 Configuration.....	97
5.14.1.7.5.4 Dambar .....	98
5.14.1.7.6 Post (Stud) Terminal .....	99
5.14.1.7.7 J-bend.....	100
5.14.1.7.8 L-bend.....	101
5.14.1.7.9 Column Types.....	102
5.14.1.7.10 Surface-terminal.....	103
5.14.1.7.11 Pin.....	104
5.14.1.7.11.1 Swage Fastening Pin .....	105
5.14.1.7.11.1.1 Tail Style .....	105
5.14.1.7.11.1.1.1 Slotted.....	106
5.14.1.7.11.1.1.2 Turret .....	106
5.14.1.7.12 Wraparound .....	107
5.14.1.7.13 S-bend Terminal.....	108
5.14.1.7.14 Through - Hole .....	109
5.14.1.7.15 J-inverted .....	110
5.14.1.7.16 Terminal Wire.....	111
5.14.1.7.17 Screw .....	112
5.14.1.7.18 Shoulder .....	113
5.14.1.7.18.1 Shoulder Transition .....	113
5.14.1.7.18.1.1 Tapered .....	114
5.14.1.7.18.1.2 Right Angled .....	114
5.14.1.7.18.2 Configuration.....	115
5.14.1.7.18.3 TH - Shoulder Cutout .....	115
5.14.1.7.19 Kinked.....	116
5.14.1.7.20 Terminal End Shape .....	116
5.14.1.7.21 Terminal Shape versus Dimensions .....	117
5.14.1.7.22 Castellations - Array.....	118
5.14.1.7.22.1 Standard - Array.....	119
5.14.1.7.22.2 Circular - Array.....	119
5.14.1.7.22.3 Status.....	120
5.14.1.7.22.4 Random - Array.....	120
5.14.1.7.22.5 Vertical Castellations Location On Terminal.....	121
5.14.1.7.22.5.1 Castellations to Terminal X - Offset.....	122
5.14.1.7.22.5.2 Castellations to Terminal Y - Offset.....	125
5.14.1.7.22.5.3 Corner .....	128

## Contents (cont'd)

	Page
5.14.1.7.23	Terminal Void - Array ..... 128
5.14.1.7.23.1	Standard Array ..... 130
5.14.1.7.23.2	Circular Array ..... 131
5.14.1.7.23.3	Void Status..... 133
5.14.1.7.23.4	Random Array ..... 134
5.14.1.8	Terminal Span..... 134
5.14.1.9	Terminal Spacing ..... 135
5.14.1.10	Terminal Group to Body Relationship ..... 135
5.14.1.10.1	Terminal Group to Body X-Offset ..... 136
5.14.1.10.2	Terminal Group to Body Y-Offset ..... 139
5.14.1.10.3	Terminal Group to Body Z-Offset ..... 142
5.14.2	Terminal Group to Terminal Group Relationship Array ..... 145
5.14.3	Terminal Detail Array ..... 146
5.14.3.1	Terminal Detail..... 147
5.14.3.1.1	Terminal Center ..... 148
5.14.3.1.2	Terminal Status ..... 150
5.14.3.2	First Terminal Location..... 160
5.14.3.3	Terminal Number Pattern ..... 162
5.14.3.3.1	Sequential ..... 163
5.14.3.3.2	Grid ..... 168
5.14.3.4	Terminal Detail Exceptions..... 172
5.14.3.4.1.1	Terminal Center - Array..... 173
5.14.3.4.1.2	Terminal Index ..... 174
5.14.3.4.1.3	Terminal Number ..... 175
5.14.4	Via Array ..... 177
5.14.4.1	Standard Array ..... 178
5.14.4.2	Circular Array ..... 179
5.14.4.3	Random Array ..... 179
5.14.4.4	Via Detail..... 180
5.15	Tiebar - Array ..... 181
5.15.1	Tiebar Pattern ..... 182
5.15.2	Pattern Group ..... 183
5.15.2.1	Relationship Transformations..... 183
5.15.3	Tiebar Shape - Array ..... 184
5.15.3.1	View Perspective..... 185
5.15.3.1.1	Rectangle Shape..... 186
5.15.3.1.2	Modified Rectangle Shape ..... 186
5.15.3.1.2.1	Impacted Corner ..... 187
5.15.3.1.2.2	Exterior Corner Type..... 187
5.15.3.2	View Perspective – to – View Perspective Alignment - Array..... 188
5.15.3.2.1	View Perspective – to – View Perspective Alignment ..... 188
5.15.3.2.1.1	From View Perspective Shape ID..... 189
5.15.3.3	Assembled Shape ..... 190
5.15.3.3.1	Tiebar Center ..... 190
5.15.3.3.2	Apply – to – all - Tiebars..... 191
5.15.3.3.3	Symmetry..... 191
5.15.3.3.3.1	Center ..... 192
5.15.3.3.3.2	Mirror Plane ..... 192
5.15.3.3.3.3	Inversion Center..... 192

## Contents (cont'd)

	Page
5.15.4	Tiebar Selection To Body Relationship - Array ..... 193
5.15.4.1	Tiebar Pattern or Tiebar Pattern Group to Body X-Offset ..... 194
5.15.4.2	Tiebar Pattern or Tiebar Pattern Group to Body Y-Offset ..... 197
5.15.4.3	Tiebar Pattern or Tiebar Pattern Group to Body Z-Offset ..... 200
5.16	Fiducial Marking ..... 201
5.16.1	Fiducial Shape ..... 201
5.16.1.1	Vertex - Array ..... 202
5.16.1.1.1	Edge ..... 203
5.16.1.2	Primitive Fiducial Shape ..... 204
5.16.2	Graphical Format - Array ..... 204
5.16.2.1	Graphical Format ..... 205
5.16.2.1.1	Color Type ..... 206
5.17	Geometric Dimensioning and Tolerancing ..... 207
5.17.1	Datum Array ..... 208
5.17.2	Feature Control Array ..... 210
5.17.2.1	Feature Control Type ..... 210
5.17.2.1.1	Form ..... 211
5.17.2.1.2	Orientation ..... 211
5.17.2.1.3	Profile ..... 212
5.17.2.1.4	Location ..... 212
5.17.2.1.5	Runout ..... 213
5.17.3	Modifier ..... 214
5.17.4	Datum Reference Array ..... 217
5.17.5	Feature Control Placement Array ..... 218
5.17.6	GD&T XML Example ..... 219
5.17.7	GD and T Datum-to-Element Map ..... 220
5.18	Recommended Footprint - Array ..... 221
5.18.1	Interconnect Technology - Array ..... 222
5.18.2	Recommended Pad Or Hole Shape - Array ..... 222
5.18.2.1	Recommended Pad or Hole Shape ..... 223
5.18.2.1.1	Surface Mount ..... 224
5.18.2.1.1.1	Pad Void - Array ..... 225
5.18.2.1.1.1.1	Pad Void Shape ..... 225
5.18.2.1.1.1.2	Pad Void Location ..... 226
5.18.2.1.1.1.2.1	Standard Array ..... 227
5.18.2.1.1.1.2.2	Circular Array ..... 228
5.18.2.1.2	Through Hole ..... 229
5.18.2.1.2.1	Hole Shape ..... 229
5.18.2.1.2.2	Mount Side Pad ..... 230
5.18.2.1.2.3	Internal Pad ..... 231
5.18.2.1.2.4	Opposite Side Pad ..... 232
5.18.2.1.3	Location ..... 233
5.18.2.1.3.1	Standard Array ..... 233
5.18.2.1.3.2	Circular Array ..... 234
5.18.2.1.3.3	Pad-or-Hole Status ..... 236
5.18.2.1.3.4	Random Array ..... 237
5.18.2.1.4	Pattern Groups ..... 237
5.18.2.1.4.1	Pattern Relationship ..... 238
5.18.2.1.4.1.1	Relationship Transformations ..... 238

## Contents (cont'd)

	Page
5.18.2.1.5	Land Pattern Span .....239
5.18.2.1.6	Land Pattern Spacing.....239
5.18.2.2	Pad Group To Pad Group Relationship .....240
5.18.3	Thermal Relief - Array .....240
5.18.3.1	Spoke Location .....241
5.18.4	Assembly Outline Layer .....241
5.18.5	Conductive Area - Array .....242
5.18.5.1	Conductive Area Void - Array .....242
5.18.5.1.1	Conductive Area Void Shape .....243
5.18.5.1.2	Conductive Area Void Location .....243
5.18.6	Placement Outline.....244
5.18.7	Keep-in Layer - Array .....245
5.18.7.1	Restrictive Layer .....246
5.18.7.2	From Outer Layer.....246
5.18.7.3	Restriction .....247
5.18.8	Keepout Region - Array.....247
5.18.9	Soldermask Layer - Array.....248
5.18.9.1	Soldermask Gang Relief- Array.....248
5.18.9.2	Soldermask Shape.....249
5.18.9.2.1	SM - Shape Location.....250
5.18.9.2.1.1	Standard Array.....250
5.18.9.2.1.2	Circular Array .....251
5.18.9.2.1.3	SM - Shape Status.....251
5.18.9.2.1.3.1	SM – Shape Center - Array .....252
5.18.9.2.1.3.2	SM - Shape Index .....253
5.18.9.2.1.4	Random Array .....254
5.18.9.2.2	Pattern Group .....254
5.18.9.2.2.1	RelationshipTransformations.....255
5.18.9.2.2.2	RelationshipTransformations.....255
5.18.9.2.3	SM – Shape Pattern Span.....256
5.18.9.2.4	SM – Shape Pattern Spacing .....256
5.18.10	Pastemask Layer- Array.....257
5.18.10.1	Aperture Shape.....258
5.18.10.1.1	Aperture Location.....259
5.18.10.1.1.1	Standard Array.....259
5.18.10.1.1.2	Circular Array .....260
5.18.10.1.1.3	Status.....261
5.18.10.1.1.4	RandomArray.....261
5.18.10.1.2	Pattern Groups.....262
5.18.10.1.2.1	Relationship Transformations.....262
5.18.10.1.2.2	Duplicate .....263
5.18.10.2	Pastemask Layer To Pastemask Layer Relationship.....263
5.19	Physical Model.....264
6	Package Section - Die-Array .....265
6.1	Die Zones.....267
6.2	Stepping Distance .....269
6.3	Mass .....270
6.4	Part Height.....270
6.5	Terminal Groups .....271

## Contents (cont'd)

	Page
6.5.1	Terminal Group - Array.....272
6.5.1.1	Terminal Group.....273
6.5.1.1.1	Terminal Pattern.....274
6.5.1.1.2	Pattern Group .....278
6.5.1.2	Relationship – to – Active Zone.....285
6.5.1.2.1	X-Offset.....286
6.5.1.2.2	Y-Offset.....289
6.5.1.3	Terminal Group to Terminal Group Relationships.....292
6.5.2	Terminal Regions - Array .....292
6.5.2.1	Area Selection.....293
6.5.2.2	Terminal Selection .....293
6.5.2.2.1	Select.....294
6.5.3	Terminal Materials - Array .....295
6.5.4	CTE - Array .....296
6.5.5	Terminal Shape - Array .....296
6.5.5.1	Terminal.....297
6.5.5.2	Soldermask Opening.....298
6.5.5.3	Pad .....299
6.5.6	Terminal Specification - Array .....300
6.5.7	Terminal Detail - Array .....300
6.5.7.1	Terminal Detail.....301
6.5.7.2	First Terminal Location.....303
6.5.7.3	Terminal Number Pattern .....305
6.5.7.4	Terminal Details Exception.....306
6.5.7.4.1	Associated .....307
Annex A (informative)	Shape Definitions & Dimensions .....313
A.1	Shape Definitions .....313
A.1.1	Rectangle.....313
A.1.2	Rounded Rectangle .....314
A.1.3	Rounded Concave .....314
A.1.4	Rectangle with Shoulder .....315
A.1.5	Rounded Rectangle with Tab .....316
A.1.6	Modified Rectangle .....317
A.1.7	Modified Rectangle with Shoulder .....319
A.1.8	Modified Rectangle with Tab .....319
A.1.9	Circle.....320
A.1.10	Circle with Shoulder .....320
A.1.11	Circle with Tab .....320
A.1.12	D-Shape.....321
A.1.13	Double-D.....321
A.1.14	Regular Polygon.....322
A.1.15	Segment .....322
A.1.16	Segmented Ring .....323
A.1.17	Para-truncated Circle .....325
A.1.18	Rounded Diamond .....325
A.1.19	Rounded Chamfered Diamond.....325
A.1.20	Isosceles Trapezoid .....326
A.1.21	Castellated Shape.....326
A.1.22	Castellated D-Shape Shape.....326

## Contents (cont'd)

	Page
A.1.23	Contour ..... 327
A.1.24	Reference Terminal Contour ..... 329
A.1.25	Modified Corner ..... 329
A.1.25.1	Impacted Terminal ..... 330
A.1.25.2	Impact – to – Terminal Group ..... 330
A.1.25.2.1	Apply – to – all - Terminals ..... 331
A.1.25.2.2	Rotation ..... 331
A.1.25.2.3	Reflection ..... 332
A.1.25.3	Shape Impacted Corner ..... 333
A.1.25.4	Corner Type ..... 334
A.2	Terminal Dimensions ..... 335
A.2.1	Ball ..... 335
A.2.1.1	Collapsing Ball ..... 335
A.2.1.2	Non-collapsing Ball ..... 335
A.2.2	C-Bend ..... 336
A.2.3	Lug ..... 337
A.2.4	Flat Terminal ..... 337
A.2.5	Gull-wing Terminal ..... 338
A.2.6	J-Bend ..... 340
A.2.7	L-bend ..... 341
A.2.8	Column ..... 342
A.2.9	Surface Terminal ..... 344
A.2.9.1	Vertical Castellation ..... 345
A.2.9.2	Horizontal Castellation ..... 346
A.2.10	Pressfit ..... 347
A.2.11	Pin ..... 348
A.2.12	Wraparound ..... 349
A.2.14	Through-Hole ..... 350
A.2.15	J-Inverted ..... 352
A.2.16	Wire ..... 353
A.2.17	Screw ..... 353
A.3	Land Pattern Shapes ..... 354
A.3.1	Rectangle ..... 354
A.3.2	Rounded Rectangle ..... 354
A.3.3	Half Rounded Rectangle ..... 355
A.3.4	Modified Rectangle ..... 355
A.3.5	Circle ..... 356
A.3.6	D-Shape ..... 356
A.3.7	Rounded Rectangle D-Shape ..... 357
A.3.8	Double-D ..... 357
A.3.9	Contour ..... 357
A.4	Value Set Types ..... 357
A.4.1	Value Type ..... 357
A.4.2	Value Set Type ..... 358
A.4.3	Unspecified Value Set Type ..... 359
A.4.4	Unspecified Value Set Group Type ..... 360
A.4.5	Dimensional Value Set Type ..... 361
A.4.6	Dimensional Value Set Group Type ..... 362
A.4.7	Unspecified and Uncontrolled Value Set Type ..... 363



---

**Contents (cont'd)**


---

	Page
A.4.8 Unspecified Dimensional Value Set Type.....	364
A.4.9 Unspecified and Uncontrolled Dimensional Value Set Type .....	365
Annex B (informative) Differences between JEP30-P100 and its predecessors .....	366

---

**Tables**


---

	Page
Table 1 - JESD30 Table and Section References for Package Elements.....	14
Table 2 - Package Shape versus Dimensions .....	41
Table 3 - Terminal Circular Array Elements Definition .....	70
Table 4 - Ball Dimensions .....	85
Table 5 - C-bend Dimensions.....	86
Table 6 - Lug Dimensions .....	86
Table 7 - Flat Terminal Type Dimensions.....	87
Table 8 - Gull-wing Dimensions .....	88
Table 9 - Post (Stud) Terminal Dimensions.....	99
Table 10 - J-bend Dimensions .....	100
Table 11 - L-bend Dimensions .....	101
Table 12 - Column Type Dimensions .....	102
Table 13 - Surface-terminal Dimensions .....	103
Table 14 - Pin Dimensions .....	104
Table 15 - Wraparound Dimensions.....	107
Table 16 - S-bend Dimensions.....	108
Table 17 - Through - Hole Dimensions.....	109
Table 18 - J-inverted Dimensions.....	110
Table 19 - Terminal Wire Dimensions .....	111
Table 20 - Screw Dimensions .....	112
Table 21 - Terminal Shape versus Dimensions.....	117
Table 22 - Castellations to Terminal X - Offset.....	124
Table 23 - Castellations to Terminal Y - Offset.....	127
Table 24 - Void Circular Array Elements Definition.....	132
Table 25 - Terminal Group to Package Body X-Offset.....	138
Table 26 - Terminal Group to Package Body Y-Offset.....	141
Table 27 - Terminal Group to Package Body Z-Offset.....	144
Table 28 - Location Relative to Package Center .....	161
Table 29 - Terminal Number Sequential Pattern .....	164
Table 30 - Terminal Number Patterns .....	165
Table 31 - Grid Terminal Numbering Pattern.....	170
Table 32 - Tiebar to Package Body X-Offset .....	196
Table 33 - Tiebar to Package Body Y-Offset .....	199
Table 34 - Terminal Group to Package Body Z-Offset.....	200
Table 35 - Pad-or-Hole Circular Array Elements Definition.....	235
Table 36 - Pattern Details for the UCle Instantiation .....	275
Table 37 - Pattern Group Construction.....	282
Table 38 - Terminal Group to Active Zone X-Offset.....	288
Table 39 - Terminal Group to Active Zone Y-Offset.....	291
Table 40 - Location Relative to Selection Center.....	304

## Figures

---

	Page
Figure 1 - Solder Mask Defined Pad .....	85
Figure 2 - Non Solder Mask Defined Pad .....	85
Figure 3 - Surface terminal With-Opening, plus Screw-Clearance Hole .....	129
Figure 4 - Flat Terminal with Hole .....	129
Figure 5 – Surface terminal With-Opening and Cavity .....	129
Figure 6 - Flat terminal with different shape Voids.....	130
Figure 7 - Terminal Group to Terminal Group Span and Spacing.....	145
Figure 8 - SOIC.....	149
Figure 9 - SOIC with Terminal 5 Missing.....	150
Figure 10 - SOIC with Terminal 5 Deleted.....	151
Figure 11 – Part with Special Terminal Pattern .....	153
Figure 12 – Terminal Numbering Pattern Samples.....	167
Figure 13 – A Multi grid Package .....	169
Figure 14 – Grid Array with Terminal Numbering .....	169
Figure 15 – Periphery Grid Array with Inner Array Matrix .....	175
Figure 16 – Example showing Datum and Feature Control Frames .....	208
Figure 17 – Constrained Degrees of Freedom for Primary Datum Features.....	209
Figure 18 – Modifying Symbols .....	216
Figure 19 – Sample for Representation in an XML Structure.....	219
Figure 20 - Zone shapes for Physical Dies on a Wafer.....	269
Figure 21 - Complex Array of 12 instantiations of a UCle Function .....	271
Figure 22 - Single UCle Instantiation in South Channel Inst 1 .....	275
Figure 23 - Power Nodes for South Channel Inst 1 & 2 .....	275
Figure 24 - Missing Terminal for Terminal Patterns.....	302
Figure 25 - Terminal Numbering Sequence for Bank 1.....	308
Figure 26 - An Array of Banks in a Zone .....	310
Figure 27 - An Array of Zones in One District.....	312

## PART MODEL PACKAGE GUIDELINE FOR ELECTRONIC-DEVICE PACKAGES - XML REQUIREMENTS

(From JEDEC Board Ballots JCB-17-48, JCB-23-10, JCB-23-27, JCB-23-33, JCB-24-08, JCB-24-29, JCB-24-50, and JCB-24-53 formulated under the cognizance of the JC-11 Committee on Mechanical Standardization.)

---

### 1 Scope

---

The JEP30 document establishes the requirements for exchanging part data between part manufacturers and their customers for electrical and electronic products. The JEP30 documents are part of a series to describe XML data exchange structure and hierarchy. The JEP30 document series will detail data exchange between companies for design at the next level, analysis, and interconnection. The parent JEP30 document specifically focuses on the parental structure, under which several sub-sections are listed, such as electrical, physical, thermal, supply chain, assembly process classification, design kit, generated ECAD models, and environment including material declaration. This document specifically focuses on the Package sub-section of the PartModel.

All releases of the [Package](#) sub-schema must be under the umbrella of the Part model Schema to ensure that the Part model schema is referencing the correct version of the [Package](#) sub-schema. In addition, this will enable the [Package](#) sub-schema to connect to the Manufacturer Part Number and the Manufacturer of the Part.

#### 1.1 Purpose

This standard is intended to benefit part manufacturers and their customers by providing consistency and efficiency to the transfer of part data from part manufacturer to customers. This standard specifically covers data applicable to the package definition of the device.

---

### 2 Applicable Documents

---

The following documents form a part of this standard to the extent specified herein. The revision of the document in effect at the time of solicitation shall take precedence.

#### 2.1 JEDEC ([www.jedec.org](http://www.jedec.org))

**JESD30L**, *Descriptive Designation System for Electronic-device Packages*

**JEP30**, *PartModel Guidelines for Electronic-Device Packages – XML Requirements*

**JEP95**, *JEDEC Registered and Standard Outlines for Solid State Products*

**JEP30-10**, *PartModel Schema*

**JEP30-P101**, *PartModel Package Schema*

**JEP30-D10**, *PartModel Schema Types Dictionary* (Required to support the PartModel Schema and each of its sectional sub-schemas.)

**SPP-010** Standard Procedures and Practices for Grid Array Terminal Position Numbering

## 2.2 IPC ([www.ipc.org](http://www.ipc.org))

**IPC-T-50**, *Terms and Definitions for Interconnecting and Packaging Electronic Circuits*

## 2.3 JEDEC/IPC ([www.jedec.org](http://www.jedec.org))

**J-Std-609**, *Marking, Symbols, and Labels of Leaded and Lead-Free Terminal Finished Materials Used in Electronic Assembly*

## 2.4 ASME ([www.asme.org](http://www.asme.org))

ASME Y14.5-2009 Dimensioning and Tolerancing

---

## 3 Requirements

---

The following terms and definitions are applicable to this XML Schema.

### 3.1 Terms and Definitions

All definitions and terms associated with the physical aspect of the part are in accordance with JESD30H or later versions. The physical details of the part are defined in the [PackageSection](#) of the XML Schema.

All common Terms and Definitions that are used by more than one sectional sub-schema, such as any of the Electrical, Package, Environmental, Assembly Process Classification, are defined in the “PartModel Schema Types Library”.

All other definitions and terms necessary to define the schema, are defined by this standard and included below.

**PartModel:** A PartModel is a data representation described in an XML file that conforms to the rules and structure of the PartModel XML Schema. Companies who use the PartModel XML Files and claim compliance to JEDEC, must ensure that their PartModel XML file conforms to the specific released version of the PartModel XML Schema released by JEDEC.

Section 4 will define the outline of the structure of the Package XML Schema. Specific components of the XML Schema and their hierarchy are specifically controlled by the JC-11 Standards Committee who retain the expertise for these structures.

The [PackageSection](#) of the schema forms part of the PartModel XML Schema and is not intended to act as a standalone schema. In addition, there is a “PartModel Schema Types Library” XML Schema, which is a common set of xml structures shared across the PartModel XML Schema and all its sub-section schemas.

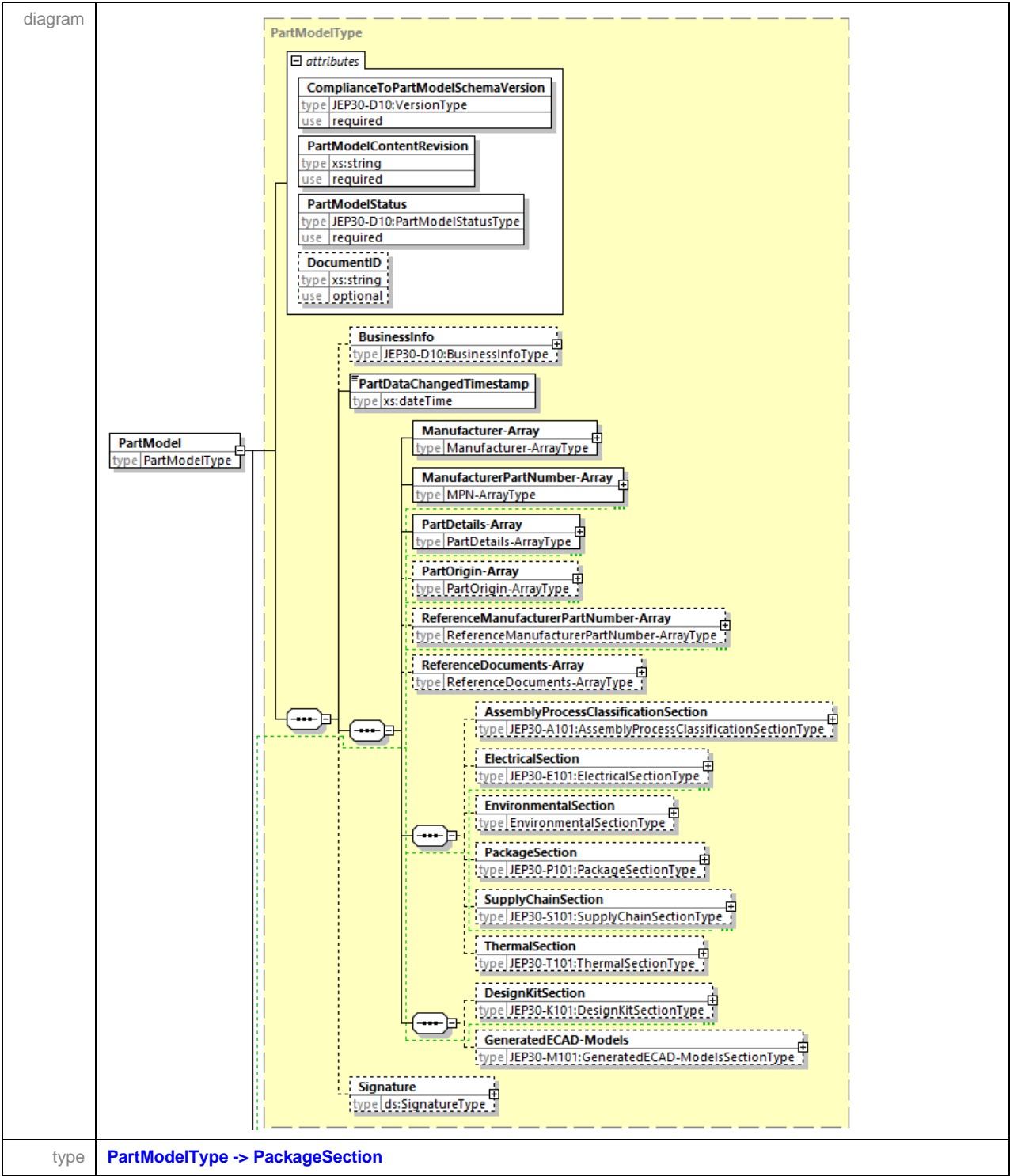
### 3.2 XML Schema Key Terms and Definitions

Reference JEP30 for details of the "XML Schema Key Terms and Definitions".

#### 4 PartModel Schema Definition

The following section describes the XML Schema structure.

##### 4.1 PartModel - Package Section



#### 4.1 PartModel - Package Section (cont'd)

The [PartModelType](#) belongs to the “PartModel XML Schema”. The [PackageSection](#) belongs to the “PartModel Package XML Schema”. The primary purpose of the PartModel Schema is to provide the structure for identifying unique parts (Manufacturer and MPN), and the structure to include the sub schemas which define the part details, as outlined in the JEP30 publication.

This document covers the [PackageSection](#), which is referenced from its parent’s structure, the [PartModel](#). The contents under the [PackageSection](#) is tied to the Manufacturer’s name and Manufacturer’s part number.

The [ComplianceToPartModelSchemaVersion](#) indicates the version of the Schema to which the XML file is to be validated against. All new releases to this document or XML Schema is governed by the rules outlined in the JEPXX, and must be release in sync with the PartModel.

*“Each time that a Sub-schema gets updated, then the part model version also gets updated in order to release that Sub-schema under the umbrella of the PartModel. This is because the PartModel must now reference the new version of sub-schema, since all subschemas have their own version number. The parent schema includes them by referring to a precise version, so a version bump in the subschema requires a version bump in the parent only at the time of release of the Parent.”*

The [PartModelContentRevision](#) indicates the revision of the data for the Part that is submitted in the XML file. This enables the Component Manufacturer to provide a new XML file for a Part each time they wish to upgrade a new set of data for a part, in any of the sub-sections such as this [PackageSection](#).

4.2 Manufacturer Part Number-Array

path	PartModel/ManufacturerPartNumber-Array.
diagram	
type	MPN-ArrayType, ManufacturerPartNumbersType, PartNumberType, OrderablePartNumber-ArrayType, JEP30-D10:SignatureDigestLinkType, ds:SignatureType.

The *ManufacturerPartNumber-Array/ManufacturerPartNumbers* provides the definition of the part number or a specific Standard, so that it can be connected to the technical specification details in the *PackageSection* via the *PartDetails-Array* section.

### 4.3 Linking the MPN to a specific Package Family Data set

The linking of the Parts to its technical data is done via the [PartDetails-Array](#) section as outline in the JEP30 - PartModel Guidelines for Electronic-Device Packages – XML Requirements. This consists of two sections called [PartsSelection-Array](#) and [Association-Array](#) which defines the relationship between identifying the specific set of parts and how they are associated with the package content. Reference the JEP30 parent document for more details on this association.

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Package-Array</a>
diagram at the Association level	<p>The diagram shows a <b>Package-Array</b> element (type <code>PackageAssociation-ArrayType</code>) connected to a <b>PackageAssociation-ArrayType</b> container. This container holds three elements: <b>Package</b> (type <code>PackageAssociationType</code>), <b>PhysicalModel</b> (type <code>PhysicalModelAssociationType</code>), and <b>Die</b> (type <code>DieAssociationType</code>). Each of these elements has a cardinality of <code>0..∞</code>.</p>
type	<a href="#">PackageAssociation-ArrayType</a> , <a href="#">PackageAssociationType</a> , <a href="#">PhysicalModelAssociationType</a> , <a href="#">DieAssociationType</a> .
diagram at the Package Section level	<p>The diagram shows a <b>PackageSection</b> element (type <code>PackageSectionType</code>) connected to a <b>PackageSectionType</b> container. This container holds two elements: <b>Package-Array</b> (type <code>Package-ArrayType</code>) and <b>Die-Array</b> (type <code>Die-ArrayType</code>). The <b>Package-Array</b> element is further linked to a <b>Package-ArrayType</b> container, which holds <b>Package</b> (type <code>PackageType</code>) and <b>PhysicalModel</b> (type <code>PhysicalModelType</code>). The <b>Die-Array</b> element is linked to a <b>Die-ArrayType</b> container, which holds <b>Die</b> (type <code>DieType</code>). The <b>Package</b> and <b>Die</b> elements have a cardinality of <code>1..∞</code>. There are also <b>constraints</b> boxes associated with the <b>Package-ArrayType</b> and <b>Die-ArrayType</b> containers.</p>
type	<a href="#">PackageSectionType</a> , <a href="#">Package-ArrayType</a> , <a href="#">PackageType</a> , <a href="#">PhysicalModelType</a> , <a href="#">Die-ArrayType</a> , <a href="#">DieType</a> .

The [Package-Array](#) content is now sub-grouped into 2 major sections as shown in the diagram. This enables each section to be digitally signed independently of each other. The linkage between the 2 sections plus the [Die](#) is shown below.



4.3.1 Linking the Manufacturing Part Number to Package Content

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Package-Array/Package
diagram at the Package Association level	<p>The diagram shows a dashed box labeled <b>PackageAssociationType</b>. Inside, there is a <b>Package</b> element with type <b>PackageAssociationType</b> and a cardinality of <b>0..∞</b>. To its right is a <b>PackageSignature</b> element with type <b>JEP30-D10:SignatureDigestLinkType</b> and a cardinality of <b>1</b>. A red arrow points from the <b>PackageSignature</b> element to the <b>PackageID</b> element in the diagram below.</p>
type	PackageAssociationType, JEP30-D10:SignatureDigestLinkType.
path	PartModel/PackageSection/Package-Array/Package
diagram at the Package level	<p>The diagram shows a dashed box labeled <b>PackageType</b>. Inside, there is an <b>ID</b> element with type <b>xs:string</b> and a cardinality of <b>1</b>. Below it is a <b>DescriptiveDesignation</b> element with type <b>xs:string</b> and a cardinality of <b>1</b>. Below that is a <b>ManufacturerPackageCode</b> element with type <b>xs:string</b> and a cardinality of <b>1</b>. Below that are two empty elements with cardinalities of <b>1</b>. At the bottom is a <b>PackageSignature</b> element with type <b>ds:SignatureType</b> and a cardinality of <b>1</b>. A red arrow points from the <b>PackageSignature</b> element in this diagram to the <b>PackageSignature</b> element in the diagram above.</p>
type	PackageType, ds:SignatureType, ...

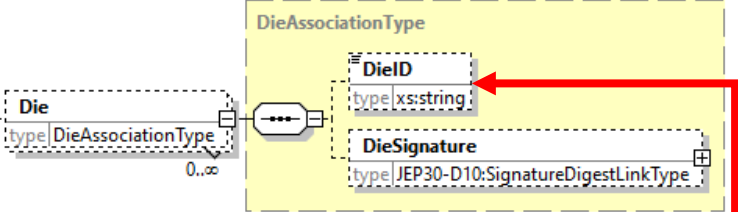
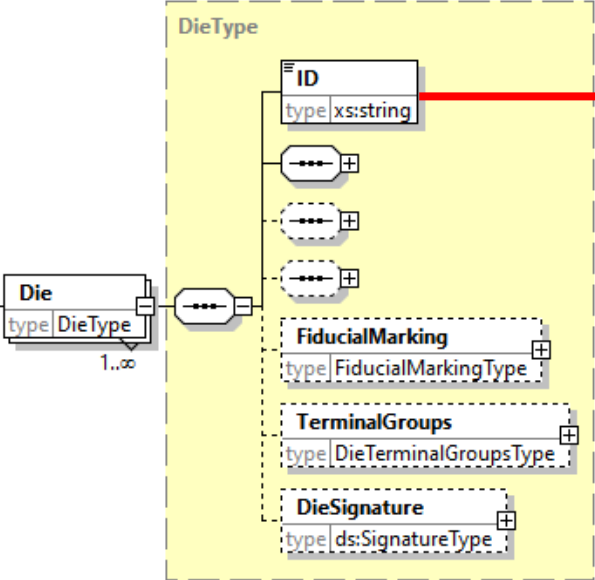
The *PackageID* references the *Package/ID* under the *PackageSection/Package-Array*. This is enforced by the key named as *PackageKey* that is assigned to the *Package/ID* element, which is referenced by the *PackageID* which has a KeyRef that refers to the *JEP30-P101:PackageKey*.

#### 4.3.2 Linking the Manufacturing Part Number to Physical Model Content

path	<a href="#">PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Package-Array/PhysicalModel</a>	
diagram at the Physical Model Association level		
type	<a href="#">PhysicalModelAssociationType</a> , <a href="#">JEP30-D10:SignatureDigestLinkType</a> .	
path	<a href="#">PartModel/PackageSection/Package-Array/PhysicalModel</a>	
diagram at the Physical Model level		
type	<a href="#">PhysicalModelType</a> , <a href="#">ModelType</a> , <a href="#">ds:SignatureType</a> .	

The [PhysicalModelID](#) references the [PhysicalModel/ID](#) under the [PackageSection/Package-Array](#). This is enforced by the key named as [PhysicalModelKey](#) that is assigned to the [PhysicalModel/ID](#) element, which is referenced by the [PhysicalModelID](#) which has a KeyRef that refers to the [JEP30-P101:PhysicalModelKey](#).

4.3.3 Linking the Manufacturing Part Number to Die Content

path	PartModel/PartDetails-Array/PartDetails/Association-Array/Association/Package-Array/Die	
diagram at the Physical Model Association level	 <p>The diagram shows a 'Die' element with a dashed border and a solid border, containing the text 'type DieAssociationType' and a cardinality of '0..∞'. It is connected to a 'DieAssociationType' element, which is a dashed box containing a 'DieID' element (type xs:string) and a 'DieSignature' element (type JEP30-D10:SignatureDigestLinkType). A red arrow points from the 'DieID' element to the 'Die' element.</p>	
type	DieAssociationType, JEP30-D10:SignatureDigestLinkType.	
path	PartModel/PackageSection/Die-Array/Die	
diagram at the Physical Model level	 <p>The diagram shows a 'Die' element with a dashed border and a solid border, containing the text 'type DieType' and a cardinality of '1..∞'. It is connected to a 'DieType' element, which is a dashed box containing an 'ID' element (type xs:string), three unnamed elements (each with a cardinality of '1..∞'), a 'FiducialMarking' element (type FiducialMarkingType), a 'TerminalGroups' element (type DieTerminalGroupsType), and a 'DieSignature' element (type ds:SignatureType). A red arrow points from the 'ID' element to the 'Die' element.</p>	
type	DieType, FiducialMarkingType, DieTerminalGroupsType, ds:SignatureType.	

The *DieID* references the *Die/ID* under the *PackageSection/Die-Array*. This is enforced by the key named as *DieKey* that is assigned to the *Die/ID* element, which is referenced by the *DieID* which has a KeyRef that refers to the *JEP30-P101:DieKey*.

## 5 Package Section - Package

path	<b>PartModel/PackageSection</b>
diagram part 1 of 4	<p>The diagram illustrates the structure of the <b>PackageSectionType</b> and its associated components. <b>PackageSectionType</b> is a complex type with the following attributes:</p> <ul style="list-style-type: none"> <li><b>ComplianceToJEP30-P101SchemaVersion</b>: type <code>JEP30-D10:VersionType</code>, use <code>required</code>.</li> <li><b>PackageContentRevision</b>: type <code>xs:string</code>, use <code>required</code>.</li> <li><b>PartModelPackageSectionStatus</b>: type <code>JEP30-D10:PartModelStatusType</code>, use <code>required</code>.</li> <li><b>DocumentID</b>: type <code>xs:string</code>, use <code>optional</code>.</li> </ul> <p><b>PackageSection</b> is a simple type that inherits from <b>PackageSectionType</b>.</p> <p><b>Package-Array</b> is a complex type that inherits from <b>Package-ArrayType</b>. It contains a <b>Package-Array</b> element (type <code>Package-ArrayType</code>) and a <b>Die-Array</b> element (type <code>Die-ArrayType</code>).</p> <p><b>Package-ArrayType</b> is a complex type that contains a <b>Package</b> element (type <code>PackageType</code>) and a <b>PhysicalModel</b> element (type <code>PhysicalModelType</code>). The <b>Package</b> element has a cardinality of <code>1..∞</code> and the <b>PhysicalModel</b> element has a cardinality of <code>0..∞</code>. A <b>constraints</b> box is also shown.</p>
type	<b>PackageSectionType, Package-ArrayType, PackageType, PhysicalModelType, Die-ArrayType.</b>
path	<b>PartModel/PackageSection/Package-Array/Package</b>

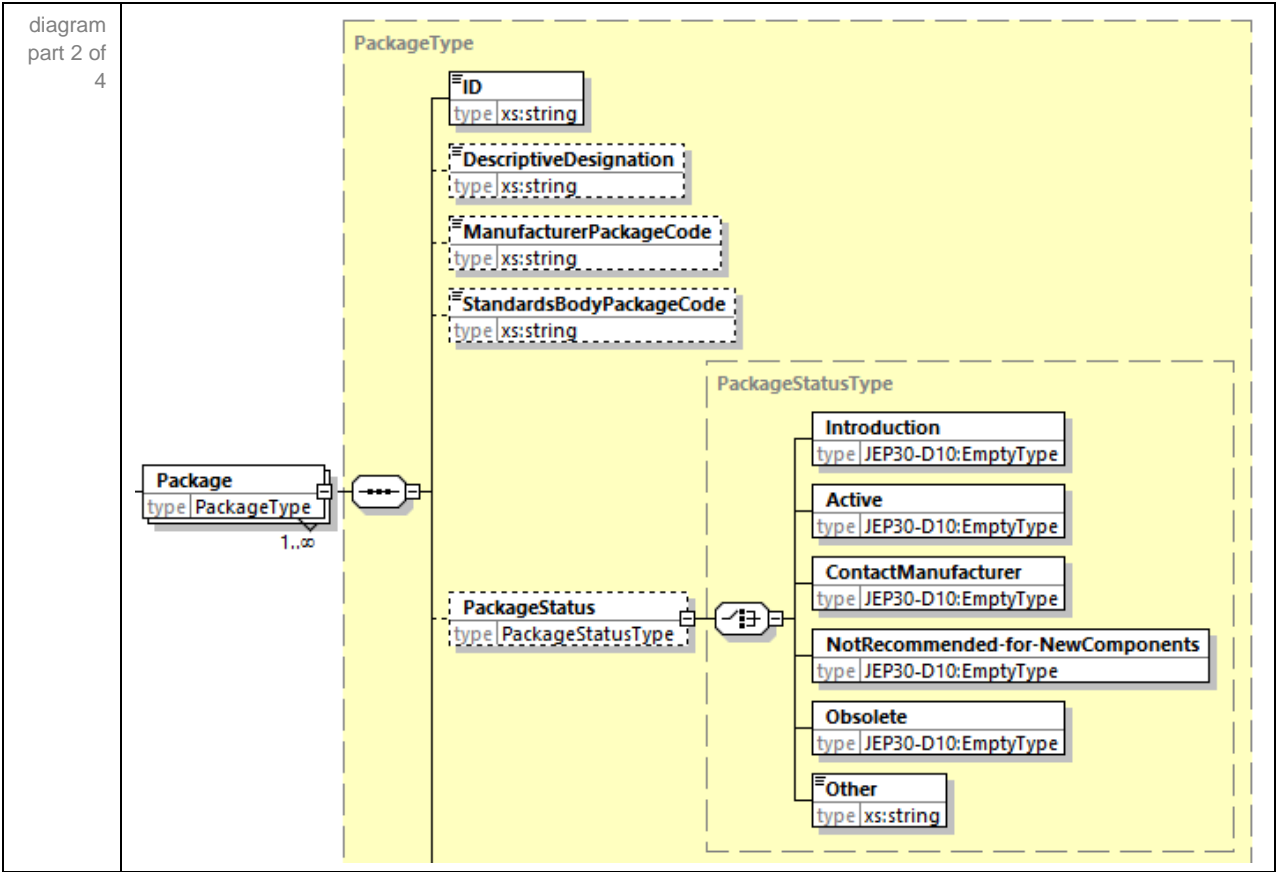
The [ComplianceToJEP30-P101SchemaVersion](#) attribute indicates the version of the Schema to which the JEP30-P101 XML file is to be validated against.

The [PartModelContentRevision](#) attribute indicates the revision of the data for the Part that is submitted in the XML file. This enables the Component Manufacturer to provide a new XML file for a Part each time they wish to upgrade a new set of data for a part, in any of the child sections.

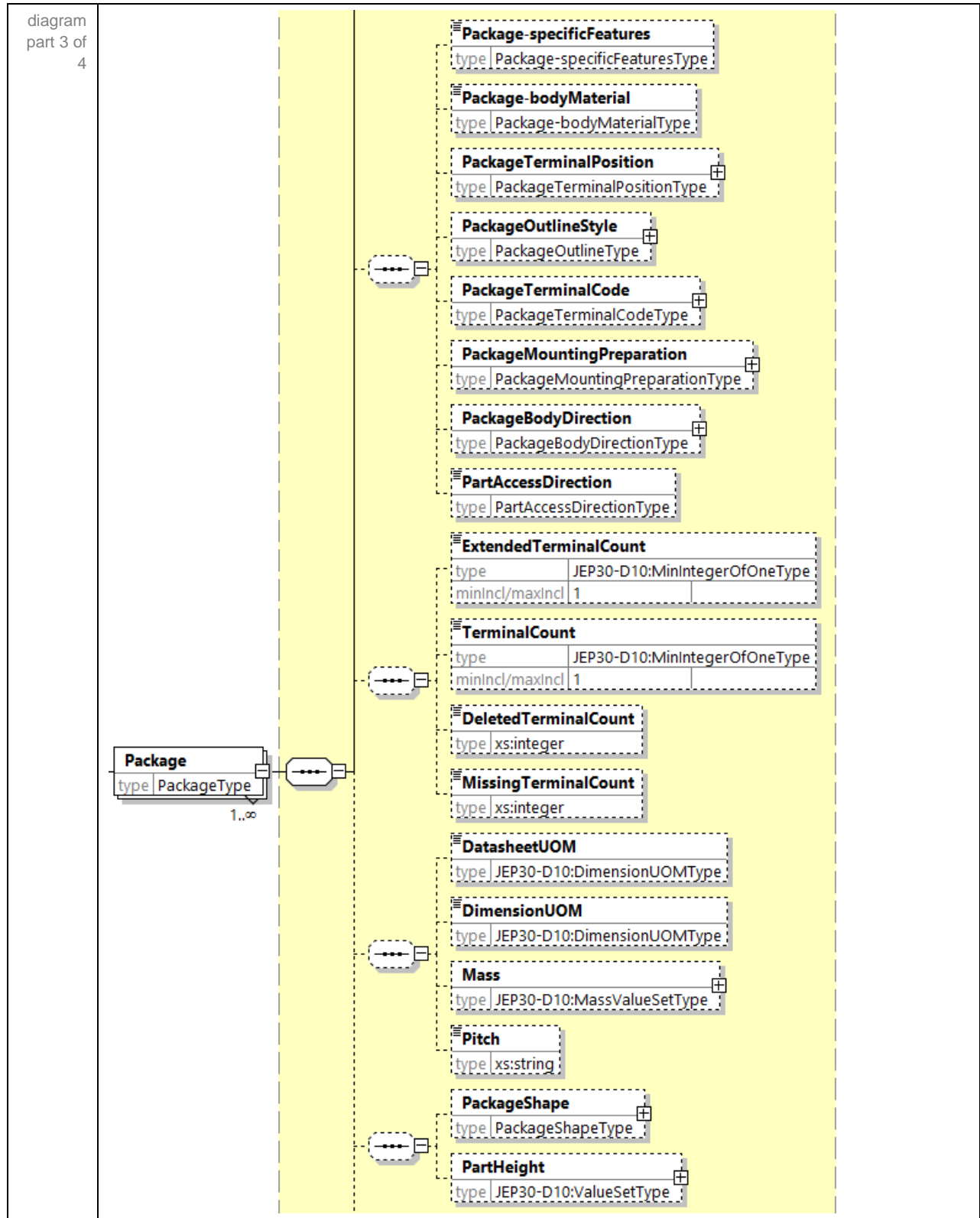
The [PartModelStatus](#) attribute is a mandatory attribute that determines the status of the PartModel XML file. It has enumerated values of [Pre-Release](#), [Released](#), [Superseded](#), and [Withdrawn](#).

The [DocumentID](#) attribute provides a unique ID for the JEP30 document that is being published.

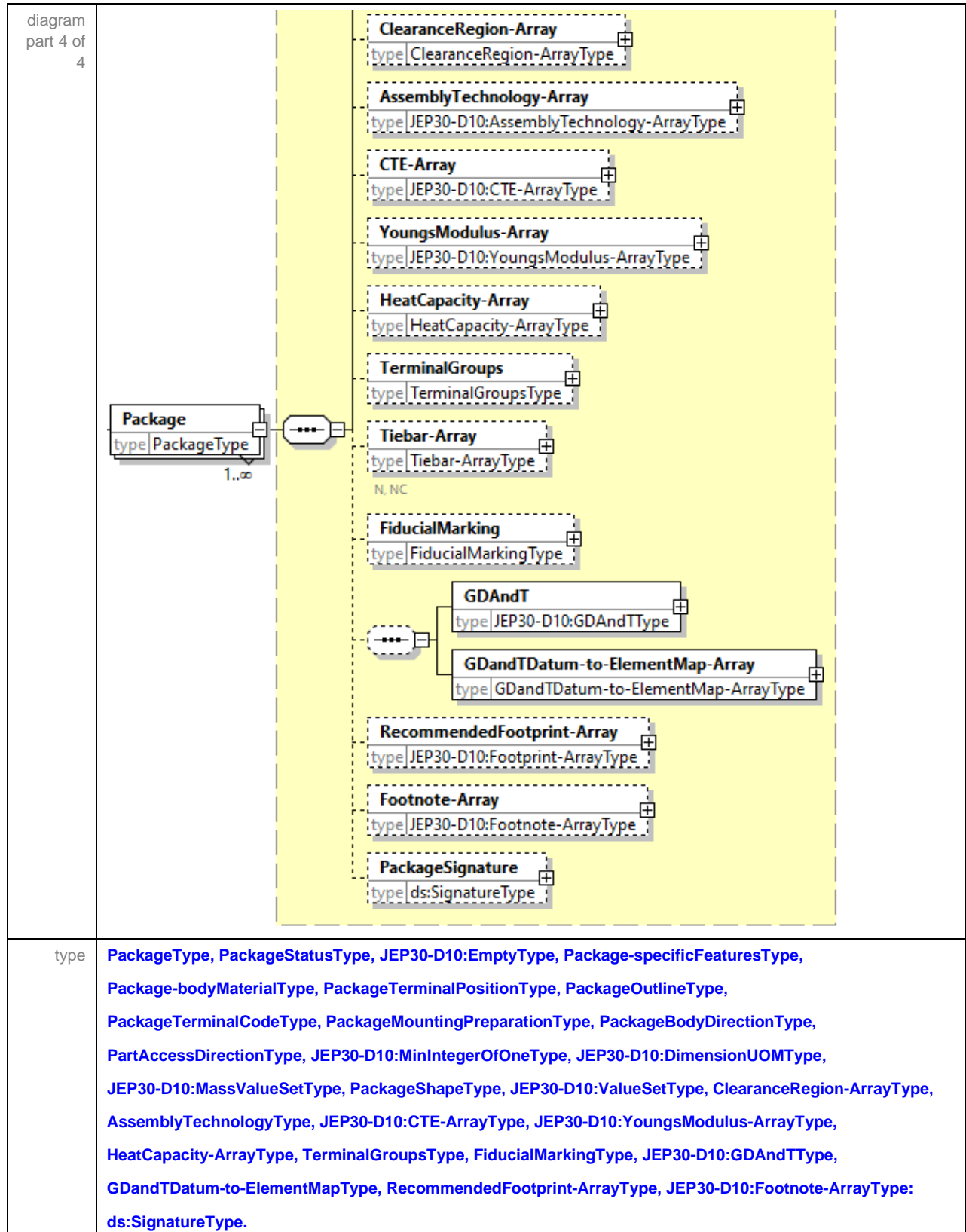
5 Package Section – Package (cont'd)



## 5 Package Section – Package (cont'd)



## 5 Package Section – Package (cont'd)



## 5 Package Section – Package (cont'd)

The [DescriptiveDesignation](#) is described in JESD30, “Descriptive Designation System for Electronic-device Packages and Footprints”. The [ManufacturerPackageCode](#) is a non-standardized package Code that is assigned to the Package by the component manufacturer. The definitions of the [ExtendedTerminalCount](#), [TerminalCount](#), [DeletedTerminalCount](#), [MissingTerminalCount](#) and [Package/Pitch](#) are also described in the JESD30 document.

The [PackageStatus](#) is an optional attribute that determines the status of the Package.

The enumerate values of the [JEP30-D10:DimensionUOMType](#) are [nm](#), [um](#), [mm](#), [m](#), [in](#) and [mil](#).

The enumerate values of the above lists of [Package-specificFeatures](#), [Package-bodyMaterial](#), and [PartAccessDirection](#), are contained in JESD30, as indicated in Table 1.

**Table 1 - JESD30 Table and Section References for Package Elements**

Enumerated Value List	JESD30 Tables / Section
Package-specific Features	Codes for package-specific features table
Package-body Material Type	Prefixes for predominant package-body material table
Part Access Direction	Part access Direction section



5.1 Package Terminal Position

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition.
diagram	<p>The diagram illustrates the <b>PackageTerminalPositionType</b> as a generalization of several specific package terminal position types. A dashed box on the left represents the generalization, labeled <b>PackageTerminalPosition</b> with the type <b>PackageTerminalPositionType</b>. This box is connected via a solid line with a hollow triangle arrow to a central box containing a package icon. This central box is then connected to a large dashed box on the right, which represents the <b>PackageTerminalPositionType</b> and contains a list of specific types: <b>Axial</b> (type: JEP30-D10:EmptyType), <b>Bottom</b> (type: BottomUpperPositionConfigurationType, marked with a +), <b>Dual</b> (type: DualPositionConfigurationType, marked with a +), <b>End</b> (type: JEP30-D10:EmptyType), <b>Diagonal</b> (type: DiagonalCornerConfigurationType, marked with a +), <b>Internal</b> (type: JEP30-D10:EmptyType), <b>MixedPosition</b> (type: JEP30-D10:EmptyType), <b>Quad</b> (type: QuadPositionConfigurationType, marked with a +), <b>Radial</b> (type: RadialPositionConfigurationType, marked with a +), <b>Single</b> (type: SinglePositionConfigurationType, marked with a +), <b>Triple</b> (type: TriplePositionConfigurationType, marked with a +), <b>Upper</b> (type: BottomUpperPositionConfigurationType, marked with a +), and <b>ZigZag</b> (type: JEP30-D10:EmptyType).</p>
type	PackageTerminalPositionType, JEP30-D10:EmptyType, BottomUpperPositionConfigurationType, DualPositionConfigurationType, DiagonalCornerConfigurationType, QuadPositionConfigurationType, RadialPositionConfigurationType, SinglePositionConfigurationType, TriplePositionConfigurationType.

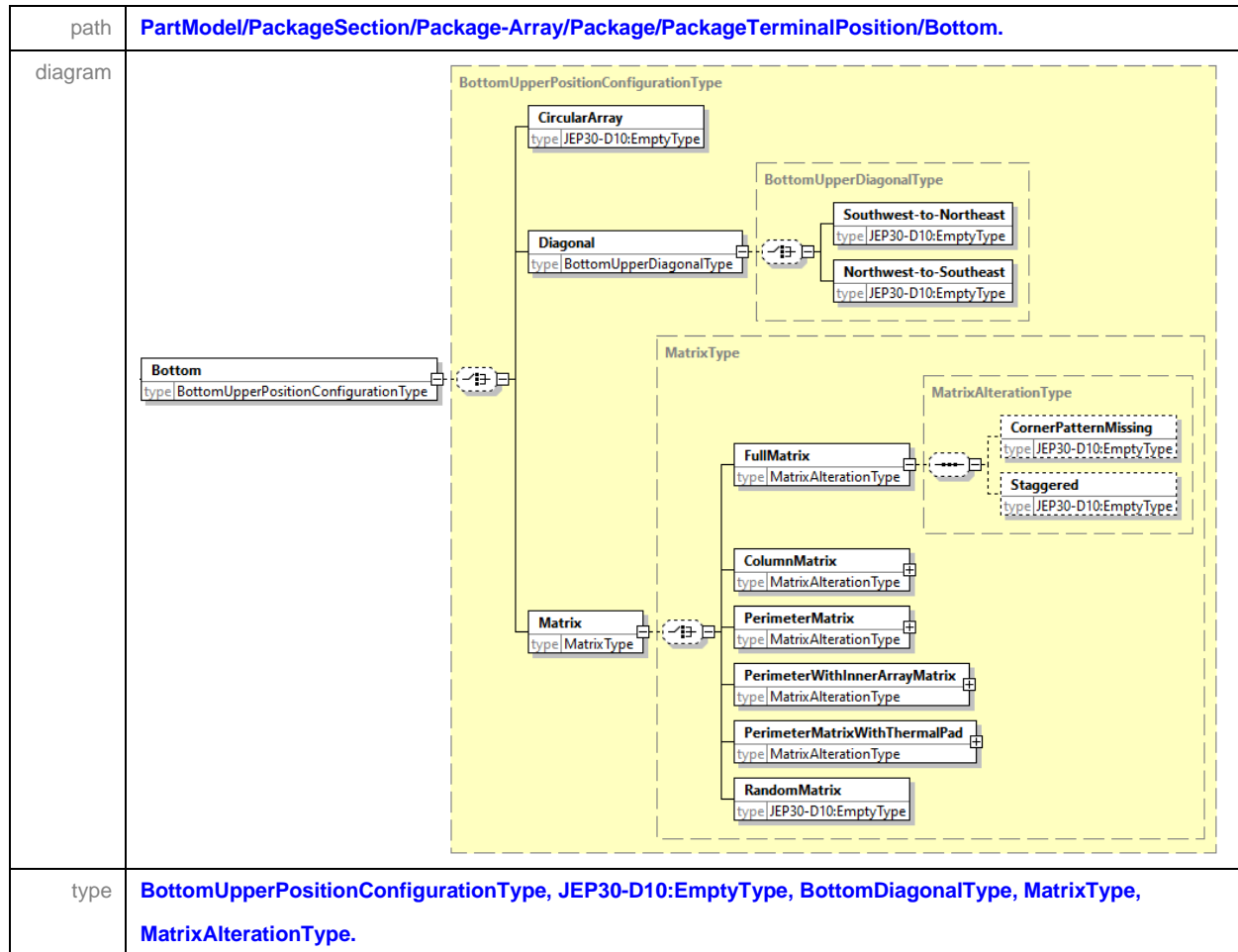
## 5.1 Package Terminal Position (cont'd)

The definition of the above [PackageTerminalPosition](#) data elements can be found in JESD30, under the “Terminal position Prefix” section which *includes* the “Prefixes for terminal position” table, plus three sections in Annex A, namely

- Terminal position with additional definition,
- Relationship concepts between the Terminal Contact Area and the Body outline, and
- Position Images.

NOTE As a result of the [TerminalGroup/TerminalPosition](#) (as defined in section 4.16 Terminal Position below), the rollup of the [TerminalPosition](#) values for all [TerminalGroup-Array/TerminalGroups](#) into the [PackageTerminalPosition](#), may be different than the value at the [PackageTerminalPosition](#) level. This is described in detail in JESD30, “Terminal-position prefix”.

### 5.1.1 Bottom



5.1.2 Dual

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Dual.	
diagram	<p>The diagram illustrates the structure of the <b>Dual</b> package terminal position. It features a <b>Dual</b> class (type <b>DualPositionConfigurationType</b>) which is associated with a <b>DualPositionConfigurationType</b> container. This container includes three relationship types: <b>Left-to-Right</b> (type <b>DualSidePositionRelationshipType</b>), <b>Back-to-Front</b> (type <b>DualSidePositionRelationshipType</b>), and <b>Bottom-to-Upper</b> (type <b>Bottom-to-UpperDualPositionRelationshipType</b>). Additionally, it contains a <b>Corners</b> class (type <b>DualCornerConfirurationType</b>), which is further associated with a <b>DualCornerConfirurationType</b> container. This container lists eight relationship types: <b>Left</b> (type <b>SideCornerPositionRelationshipType</b>), <b>Back</b> (type <b>SideCornerPositionRelationshipType</b>), <b>Right</b> (type <b>SideCornerPositionRelationshipType</b>), <b>Front</b> (type <b>SideCornerPositionRelationshipType</b>), <b>Left-to-RightTop</b> (type <b>DifferentSidePositionRelationshipType</b>), <b>Left-to-RightBottom</b> (type <b>DifferentSidePositionRelationshipType</b>), <b>Back-to-FrontLeft</b> (type <b>DifferentSidePositionRelationshipType</b>), and <b>Back-to-FrontRight</b> (type <b>DifferentSidePositionRelationshipType</b>). The diagram uses dashed boxes to group related elements and solid lines with open squares to denote associations.</p>	
type	<b>DualPositionConfigurationType</b> , <b>DualSidePositionRelationshipType</b> , <b>Bottom-to-UpperDualPositionRelationshipType</b> , <b>DualCornerConfirurationType</b> , <b>SideCornerPositionRelationshipType</b> , <b>DifferentSidePositionRelationshipType</b> .	

### 5.1.2.1 Dual Left-to-Right and Back-to-Front

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Dual/Left-to-Right</a></li> <li>2. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Dual/Back-to-Front</a></li> </ol>
diagram	
type	<a href="#">DualSidePositionRelationshipType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">SpanPositionRelationshipType</a> .

### 5.1.2.2 Dual Bottom-to-Upper

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Dual/Bottom-to-Upper</a>
diagram	
type	<a href="#">Bottom-to-UpperDualPositionRelationshipType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">Bottom-to-UpperSpanPositionRelationshipType</a>

5.1.2.3 Dual Corners

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Dual/Corners
diagram	<p>The diagram illustrates the structure of the Dual Corners configuration. It starts with a 'Corners' class (type DualCornerConfirurationType) which is connected to a 'DualCornerConfirurationType' class. This class contains several subclasses: 'Left', 'Back', 'Right', 'Front', 'Left-to-RightTop', 'Left-to-RightBottom', 'Back-to-FrontLeft', and 'Back-to-FrontRight'. The 'Left' and 'Left-to-RightTop' classes are further connected to a 'SideCornerPositionRelationshipType' class, which contains 'Outside', 'Edge-Out', 'Overlap', and 'Edge-In' subclasses. The 'Left-to-RightTop', 'Left-to-RightBottom', 'Back-to-FrontLeft', and 'Back-to-FrontRight' classes are connected to a 'DifferentSidePositionRelationshipType' class, which contains 'Outside', 'Edge-Out', and 'Overlap' subclasses. All relationship types are of type JEP30-D10:EmptyType.</p>
type	DualCornerConfigurationType, SideCornerPositionRelationshipType, DifferentSidePositionRelationshipType, JEP30-D10:EmptyType.

### 5.1.3 Diagonal

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Diagonal
diagram	<pre> classDiagram     class Diagonal {         type DiagonalCornerConfigurationType     }     class InternalCornerPositionRelationshipType {         type InternalCornerPositionRelationshipType     }     class DifferentSidePositionRelationshipType {         type DifferentSidePositionRelationshipType     }     class JEP30D10EmptyType {         type JEP30-D10:EmptyType     }     class Overlap {         type JEP30-D10:EmptyType     }     class EdgeIn {         type JEP30-D10:EmptyType     }     class Inside {         type JEP30-D10:EmptyType     }     class Outside {         type JEP30-D10:EmptyType     }     class EdgeOut {         type JEP30-D10:EmptyType     }      Diagonal --&gt; InternalCornerPositionRelationshipType : Southwest-to-Northeast     Diagonal --&gt; InternalCornerPositionRelationshipType : Northwest-to-Southeast     Diagonal --&gt; DifferentSidePositionRelationshipType : Back-left-to-Front-right     Diagonal --&gt; DifferentSidePositionRelationshipType : Front-left-to-Back-right     Diagonal --&gt; DifferentSidePositionRelationshipType : Left-bottom-to-Right-top     Diagonal --&gt; DifferentSidePositionRelationshipType : Left-top-to-Right-bottom     DifferentSidePositionRelationshipType --&gt; JEP30D10EmptyType : Outside     DifferentSidePositionRelationshipType --&gt; JEP30D10EmptyType : Edge-Out     DifferentSidePositionRelationshipType --&gt; JEP30D10EmptyType : Overlap     InternalCornerPositionRelationshipType --&gt; JEP30D10EmptyType : Overlap     InternalCornerPositionRelationshipType --&gt; JEP30D10EmptyType : Edge-In     InternalCornerPositionRelationshipType --&gt; JEP30D10EmptyType : Inside </pre>
type	DiagonalCornerConfigurationType, InternalCornerPositionRelationshipType, JEP30-D10:EmptyType, DifferentSidePositionRelationshipType.

### 5.1.4 Quad

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Quad.
diagram	<pre> classDiagram     class Quad {         type QuadPositionConfigurationType     }     class Span {         type SpanPositionRelationshipType     }     class Corners {         type QuadCornerConfigurationType     }     class Outside {         type JEP30-D10:EmptyType     }     class EdgeOut {         type JEP30-D10:EmptyType     }     class Overlap {         type JEP30-D10:EmptyType     }     class EdgeIn {         type JEP30-D10:EmptyType     }     class Inside {         type JEP30-D10:EmptyType     }     class OverlapSpan {         type JEP30-D10:EmptyType     }     class EdgeInSpan {         type JEP30-D10:EmptyType     }     class InsideSpan {         type JEP30-D10:EmptyType     }     class OverlapCorners {         type JEP30-D10:EmptyType     }     class EdgeInCorners {         type JEP30-D10:EmptyType     }     class InsideCorners {         type JEP30-D10:EmptyType     }     class LeftToRight {         type DifferentSidePositionRelationshipType     }     class BackToFront {         type DifferentSidePositionRelationshipType     }      Quad --&gt; Span : +     Quad --&gt; Corners : +     Span --&gt; Outside : +     Span --&gt; EdgeOut : +     Span --&gt; Overlap : +     Span --&gt; EdgeIn : +     Span --&gt; Inside : +     Span --&gt; OverlapSpan : +     Span --&gt; EdgeInSpan : +     Span --&gt; InsideSpan : +     Corners --&gt; OverlapCorners : +     Corners --&gt; EdgeInCorners : +     Corners --&gt; InsideCorners : +     Corners --&gt; LeftToRight : +     Corners --&gt; BackToFront : +     </pre> <p>The diagram illustrates the structure of the Quad class and its relationships. The Quad class has two main associations: one to Span and one to Corners. The Span association leads to a collection of relationship types (Outside, Edge-Out, Overlap, Edge-In, Inside) and another set of relationship types (Overlap, Edge-In, Inside). The Corners association leads to a collection of configuration types (Overlap, Edge-In, Inside) and two additional relationship types (Left-to-Right, Back-to-Front).</p>
type	QuadPositionConfigurationType, JEP30-D10:EmptyType, SpanPositionRelationshipType, QuadCornerConfigurationType, DifferentSidePositionRelationshipType.

5.1.5 Radial

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Radial
diagram	<p>The diagram illustrates the structure of the <b>RadialPositionConfigurationType</b> and its relationship to <b>RadialCornerPositionRelationshipType</b>. The <b>RadialPositionConfigurationType</b> is a container type that includes several sub-elements: <b>Outside</b>, <b>Edge-Out</b>, <b>Overlap</b>, <b>Edge-In</b>, <b>Inside</b>, and <b>Corners</b>. Each of these sub-elements is of type <b>JEP30-D10:EmptyType</b>. The <b>Corners</b> element is further detailed by a <b>RadialCornerPositionRelationshipType</b>, which is shown as a dashed box containing its own set of sub-elements: <b>Outside</b>, <b>Edge-Out</b>, <b>Overlap</b>, <b>Edge-In</b>, and <b>Inside</b>, all of which are also of type <b>JEP30-D10:EmptyType</b>. The <b>Radial</b> element is shown as a container type of type <b>RadialPositionConfigurationType</b>.</p>
type	RadialPositionConfigurationType, JEP30-D10:EmptyType, RadialCornerpositionRelationshipType.



5.1.6 Single

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single.
diagram	
type	SinglePositionConfigurationType, SingleSidePositionRelationshipType, SingleCornerConfigurationType, InternalCornerPositionRelationshipType, DifferentSidePositionRelationshipType.

### 5.1.6.1 Single Side

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Left</a></li> <li>2. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Back</a></li> <li>3. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Right</a></li> <li>4. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Front</a></li> </ol>
diagram	<p>The diagram illustrates the <b>SingleSidePositionRelationshipType</b> structure. On the left, a box labeled <b>Left</b> with the type <code>SingleSidePositionRelationshipType</code> is connected via a dashed line to a central circular node. This node is further connected to a larger dashed yellow box on the right, which is titled <b>SingleSidePositionRelationshipType</b>. Inside this yellow box, there are five stacked boxes, each with a type of <code>JEP30-D10:EmptyType</code>: <b>Outside</b>, <b>Edge-Out</b>, <b>Overlap</b>, <b>Edge-In</b>, and <b>Inside</b>.</p>
type	<b>SingleSidePositionRelationshipType</b> , JEP30-D10:EmptyType.

### 5.1.6.2 Corner - Internal

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Southwest</a></li> <li>2. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Southeast</a></li> <li>3. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Northeast</a></li> <li>4. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Northwest</a></li> </ol>
diagram	<p>The diagram illustrates the <b>InternalCornerPositionRelationshipType</b> structure. On the left, a box labeled <b>Southwest</b> with the type <code>InternalCornerPositionRelationshipType</code> is connected via a dashed line to a central circular node. This node is further connected to a larger dashed yellow box on the right, which is titled <b>InternalCornerPositionRelationshipType</b>. Inside this yellow box, there are three stacked boxes, each with a type of <code>JEP30-D10:EmptyType</code>: <b>Overlap</b>, <b>Edge-In</b>, and <b>Inside</b>.</p>
type	<b>InternalCornerPositionRelationshipType</b> , JEP30-D10:EmptyType.

5.1.6.3 Corner - External

path	<div><div>1. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Back-left</div><div>2. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Back-right</div><div>3. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Front-right</div><div>4. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Front-left</div><div>5. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Left-bottom</div><div>6. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Right-bottom</div><div>7. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Right-top</div><div>8. PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Single/Corner/Left-top</div></div>
diagram	<div><div><div>Back-left</div><div>type DifferentSidePositionRelationshipType</div></div><div><div>DifferentSidePositionRelationshipType</div><div><div>Outside</div><div>type JEP30-D10:EmptyType</div></div><div><div>Edge-Out</div><div>type JEP30-D10:EmptyType</div></div><div><div>Overlap</div><div>type JEP30-D10:EmptyType</div></div></div></div>
type	DifferentSidePositionRelationshipType, JEP30-D10:EmptyType.

### 5.1.7 Triple

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Triple
diagram	<pre> classDiagram     class Triple {         type TriplePositionConfigurationType     }     class TriplePositionConfigurationType {         NotLeft type MultiSidePositionRelationshipType         NotBack type MultiSidePositionRelationshipType         NotRight type MultiSidePositionRelationshipType         NotFront type MultiSidePositionRelationshipType     }     class TripleCornerConfigurationType {         Not-Southwest type InternalCornerPositionRelationshipType         Not-Southeast type InternalCornerPositionRelationshipType         Not-Northeast type InternalCornerPositionRelationshipType         Not-Northwest type InternalCornerPositionRelationshipType         Not-Back-left type DifferentSidePositionRelationshipType         Not-Back-right type DifferentSidePositionRelationshipType         Not-Front-right type DifferentSidePositionRelationshipType         Not-Front-left type DifferentSidePositionRelationshipType         Not-Left-bottom type DifferentSidePositionRelationshipType         Not-Right-bottom type DifferentSidePositionRelationshipType         Not-Right-top type DifferentSidePositionRelationshipType         Not-Left-top type DifferentSidePositionRelationshipType     }     Triple "1" -- "*" TripleCornerConfigurationType     TriplePositionConfigurationType "1" -- "*" TripleCornerConfigurationType     </pre>
type	TriplePositionConfigurationType, MultiSidePositionRelationshipType, TripleCornerConfigurationType, InternalCornerPositionRelationshipType, DifferentSidePositionRelationshipType.

### 5.1.7.1 Triple – Multi Side position

path	<ol style="list-style-type: none"> <li>1. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Triple/NotLeft</a></li> <li>2. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Triple/NotBack</a></li> <li>3. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Triple/NotRight</a></li> <li>4. <a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Triple/NotFront</a></li> </ol>
diagram	
type	<a href="#">MultiSidePositionRelationshipType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">SpanPositionRelationshipType</a> .

### 5.1.8 Upper

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalPosition/Upper.</a>
diagram	
type	<a href="#">BottomUpperPositionConfigurationType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">BottomUpperDiagonalType</a> , <a href="#">MatrixType</a> , <a href="#">MatrixAlterationType</a> .

5.2 Package Outline Style

path	PartModel/PackageSection/Package-Array/Package/PackageOutlineStyle.
diagram	
type	PackageOutlineType, JEP30-D10:EmptyType, MicroelectronicsAssemblyPackageOutlineType, PostMountPackageOutlineType

## 5.2 Package Outline Style (cont'd)

The definition of the above [PackageOutlineStyle](#) data elements can be found in the JESD30 publication, under the “Package-outline style codes” section which includes a table with the definitions of each of the above data elements. In addition, in JESD30, there is an “Illustrations of Package Outline Styles” in Annex A.

### 5.2.1 Microelectronics Assembly

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageOutlineStyle/MicroelectronicsAssembly</a>
diagram	<pre> graph LR     MA[MicroelectronicsAssembly type: MicroelectronicsAssemblyPackageOutlineType] --- JEP30((JEP30-D10))     JEP30 --- MC[MultiChipModule type: JEP30-D10:EmptyType]     JEP30 --- C[Chiplet type: JEP30-D10:EmptyType]     subgraph MicroelectronicsAssemblyPackageOutlineType         MC         C     end </pre> <p>The diagram illustrates the structure of the <b>MicroelectronicsAssemblyPackageOutlineType</b>. It shows a <b>MicroelectronicsAssembly</b> box (type: <b>MicroelectronicsAssemblyPackageOutlineType</b>) connected to a dashed circle containing a JEP30-D10 symbol. This symbol is further connected to two boxes: <b>MultiChipModule</b> (type: <b>JEP30-D10:EmptyType</b>) and <b>Chiplet</b> (type: <b>JEP30-D10:EmptyType</b>). These two boxes are enclosed within a dashed yellow box labeled <b>MicroelectronicsAssemblyPackageOutlineType</b>.</p>
type	<a href="#">MicroelectronicsAssemblyPackageOutlineType</a>

### 5.2.2 Post Mount

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageOutlineStyle/PostMount.</a>
diagram	<pre> graph LR     PM[PostMount type: PostMountPackageOutlineType] --- JEP30((JEP30-D10))     JEP30 --- BST[BifurcatedSolderTerminal type: JEP30-D10:EmptyType]     JEP30 --- TST[TurretSolderTerminal type: JEP30-D10:EmptyType]     subgraph PostMountPackageOutlineType         BST         TST     end </pre> <p>The diagram illustrates the structure of the <b>PostMountPackageOutlineType</b>. It shows a <b>PostMount</b> box (type: <b>PostMountPackageOutlineType</b>) connected to a dashed circle containing a JEP30-D10 symbol. This symbol is further connected to two boxes: <b>BifurcatedSolderTerminal</b> (type: <b>JEP30-D10:EmptyType</b>) and <b>TurretSolderTerminal</b> (type: <b>JEP30-D10:EmptyType</b>). These two boxes are enclosed within a dashed yellow box labeled <b>PostMountPackageOutlineType</b>.</p>
type	<a href="#">PostMountPackageOutlineType</a>

### 5.3 Package Terminal Code

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode	
diagram		
type	PackageTerminalCodeType, BallType, LugType, FlatType, Gull-wingType, PostTerminalType, L-BendType, ColumnType, SurfaceTerminalType, PressfitType, PinType, WraparoundType, S-BendType, Through-HoleType, WireType, ScrewType.	

The definition of the above [PackageTerminalCode](#) data elements and their following sub-structures can be found in JESD30, under the “Terminal Shape Suffix” section which *includes* the “Suffixes for terminal shape” table, and two sections in Annex A, namely “Suffixes for terminal shape with additional definition”, and “Illustrations of terminal shape”.

NOTE As a result of the [TerminalGroup/Terminal](#) (as defined in 4.18), the rollup of the [Terminal](#) values for all [TerminalGroup-Array/TerminalGroups](#) into the [PackageTerminalCode](#), may be different than the value at the [PackageTerminalCode](#) level. This is described in detail in JESD30, “Terminal Shape Suffix”.



5.3.1 Ball

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Ball
diagram	<p>The diagram illustrates the BallType hierarchy. A central 'Ball' box (type BallType) is connected to a dashed box labeled 'BallType'. Inside this dashed box are three subtypes: 'Bump' (type JEP30-D10:EmptyType), 'Collapsing' (type JEP30-D10:EmptyType), and 'Non-Collapsing' (type JEP30-D10:EmptyType). The connection is represented by a dashed line with a small square at the end of the line pointing to the Ball box.</p>
type	BallType, JEP30-D10:EmptyType

5.3.2 Lug

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Lug
diagram	<p>The diagram illustrates the LugType hierarchy. A central 'Lug' box (type LugType) is connected to a dashed box labeled 'LugType'. Inside this dashed box are six subtypes: 'PerforatedSolderTerminal' (type JEP30-D10:EmptyType), 'CupSolderTerminal' (type JEP30-D10:EmptyType), 'HookSolderTerminal' (type JEP30-D10:EmptyType), 'CrimpLug' (type JEP30-D10:EmptyType), 'RingTongueTerminal' (type JEP30-D10:EmptyType), and 'LugWithThreadedHole' (type JEP30-D10:EmptyType). The connection is represented by a dashed line with a small square at the end of the line pointing to the Lug box.</p>
type	LugType, JEP30-D10:EmptyType

### 5.3.3 Flat

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Flat</a>
diagram	
type	<a href="#">FlatType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.3.4 Gull-wing

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Gull-wing</a>
diagram	
type	<a href="#">Gull-wingType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.3.5 Post Terminal

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/PostTerminal</a>
diagram	
type	<a href="#">PostTerminalType</a> , <a href="#">JEP30-D10:EmptyType</a>

5.3.6 L-bend

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/L-Bend
diagram	
type	L-BendType, JEP30-D10:EmptyType

5.3.7 Column

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/ Column
diagram	
type	ColumnType, JEP30-D10:EmptyType

5.3.8 Surface Terminal

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/SurfaceTerminal
diagram	<p>The diagram illustrates the hierarchical structure of a Surface Terminal. It begins with a <b>Surface-terminal</b> class (type <code>Surface-terminalType</code>) which is associated with a <b>Surface-terminalType</b> class. This class is further associated with a <b>Castellated</b> class (type <code>CastellatedType</code>). The <b>Castellated</b> class is associated with a <b>CastellatedType</b> class, which is further associated with a <b>WettableFlank</b> class (type <code>JEP30-D10:EmptyType</code>). The <b>CastellatedType</b> class is also associated with a <b>D-Shape</b> class (type <code>JEP30-D10:EmptyType</code>) and a <b>Pullback</b> class (type <code>JEP30-D10:EmptyType</code>). The <b>CastellatedType</b> class is further associated with a <b>Horizontal</b> class (type <code>HorizontalCastellationType</code>) and a <b>Vertical</b> class (type <code>JEP30-D10:EmptyType</code>). The <b>Horizontal</b> class is associated with a <b>HorizontalCastellationType</b> class, which is further associated with a <b>StepCut</b> class (type <code>JEP30-D10:EmptyType</code>) and a <b>Dimple</b> class (type <code>JEP30-D10:EmptyType</code>). The <b>CastellatedType</b> class is also associated with a <b>Hole</b> class (type <code>JEP30-D10:EmptyType</code>) and a <b>With-opening</b> class (type <code>JEP30-D10:EmptyType</code>). The <b>CastellatedType</b> class is further associated with a <b>RingType</b> class, which is further associated with a <b>Open-Ring</b> class (type <code>RingType</code>). The <b>Open-Ring</b> class is associated with a <b>Castellated</b> class (type <code>RingCastellatedType</code>), which is further associated with a <b>WettableFlank</b> class (type <code>JEP30-D10:EmptyType</code>).</p>
type	SurfaceTerminalType, JEP30-D10:EmptyType, CastellatedType, HorizontalCastellationType, RingType, RingCastellatedType.

### 5.3.9 Pressfit

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Pressfit</a>
diagram	<p>The diagram illustrates the hierarchy for the <b>Pressfit</b> type. A <b>Pressfit</b> object (type <b>PressfitType</b>) is associated with a <b>PressfitType</b> container. This container branches into two main categories: <b>Compliant</b> (type <b>CompliantType</b>) and <b>Non-Compliant</b> (type <b>JEP30-D10:EmptyType</b>). The <b>Compliant</b> category further branches into four sub-types: <b>Eye-of-the-NeedlePin</b>, <b>ConicalPin</b>, <b>Multi-SpringPin</b>, and <b>ActionPin</b>, all of which are of type <b>JEP30-D10:EmptyType</b>. The <b>Non-Compliant</b> category is also of type <b>JEP30-D10:EmptyType</b>.</p>
type	<a href="#">PressfitType</a> , <a href="#">CompliantType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.3.10 Pin

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Pin</a>
diagram	<p>The diagram illustrates the hierarchy for the <b>Pin</b> type. A <b>Pin</b> object (type <b>PinType</b>) is associated with a <b>PinType</b> container. This container branches into six sub-types: <b>Kinked</b>, <b>Shoulder</b>, <b>Press-InSolderable</b>, <b>Surface-mount</b>, <b>SwageFasteningPin</b>, and <b>Press-InNon-Solderable</b>. All of these sub-types are of type <b>JEP30-D10:EmptyType</b>.</p>
type	<a href="#">PinType</a> , <a href="#">JEP30-D10:EmptyType</a>

5.3.11      Wraparound

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Wraparound
diagram	
type	WraparoundType, JEP30-D10:EmptyType, WraparoundCastellatedType, RingType, RingCastellatedType.

5.3.12 S-Bend

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/S-Bend
diagram	<p>The diagram illustrates the S-Bend structure. On the left, a box labeled 'S-bend' with 'type   S-bendType' is connected to a central connector box. This connector box is linked to a larger box on the right labeled 'S-bendType'. Inside the 'S-bendType' box, there are four sub-type boxes: 'InwardFlat' (type   JEP30-D10:EmptyType), 'InwardCurved' (type   JEP30-D10:EmptyType), 'OutwardFlat' (type   JEP30-D10:EmptyType), and 'OutwardCurved' (type   JEP30-D10:EmptyType).</p>
type	S-BendType, JEP30-D10:EmptyType

5.3.13 Through-Hole

path	PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Through-Hole
diagram	<p>The diagram illustrates the Through-Hole structure. On the left, a box labeled 'Through-Hole' with 'type   Through-HoleType' is connected to a central connector box. This connector box is linked to a larger box on the right labeled 'Through-HoleType'. Inside the 'Through-HoleType' box, there are two sub-type boxes: 'Kinked' (type   JEP30-D10:EmptyType) and 'Shoulder' (type   JEP30-D10:EmptyType).</p>
type	Through-HoleType, JEP30-D10:EmptyType

### 5.3.14 Terminal Wire

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/TerminalWire</a>
diagram	
type	<a href="#">TerminalWireType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.3.15 Screw

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageTerminalCode/Screw</a>
diagram	
type	<a href="#">ScrewType</a> , <a href="#">JEP30-D10:EmptyType</a>

## 5.4 Package Mounting Preparation

path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageMountingPreparation</a>
diagram	
type	<a href="#">PackageMountingPreparationType</a> , <a href="#">JEP30-D10:EmptyType</a> .



## 5.4 Package Mounting Preparation (cont'd)

*PackageMountingPreparation* definition can be found in JESD30, under the “Mounting Preparation” section which includes some graphical images to describe the values.

## 5.5 Package Body Direction

path	PartModel/PackageSection/Package-Array/Package/PackageBodyDirection
diagram	<p>The diagram shows a dashed box labeled <b>PackageBodyDirection</b> with the type <code>PackageBodyDirectionType</code>. This box is connected to a choice symbol (a circle with a vertical line and two horizontal lines) which branches into two options: <b>Horizontal</b> (type <code>JEP30-D10:EmptyType</code>) and <b>Vertical</b> (type <code>JEP30-D10:EmptyType</code>). The entire choice structure is enclosed in a yellow dashed box labeled <b>PackageBodyDirectionType</b>.</p>
type	<b>PackageBodyDirectionType</b> , <code>JEP30-D10:EmptyType</code> .

*PackageBodyDirection* definition can be found in the JESD30, under the “Body Direction” section which includes some graphical images to describe the values.

## 5.6 Mass

path	PartModel/PackageSection/Package-Array/Package/Mass
diagram	<p>The diagram shows a dashed box labeled <b>Mass</b> with the type <code>JEP30-D10:MassValueSetType</code>. This box is connected to a choice symbol which branches into <b>Value</b> (type <code>ValueSetType</code>) and <b>Units</b> (type <code>MassUOMType</code>). The <b>Value</b> box is further connected to a choice symbol which branches into <b>ValueSetGroup</b> and <b>FootnoteID</b> (type <code>xs:string</code>). The <b>ValueSetGroup</b> box is connected to a choice symbol which branches into <b>Standard</b> (type <code>StandardType</code>) and <b>Limits</b> (type <code>MinMaxLimitsType</code>). The <b>Limits</b> box is connected to a choice symbol which branches into <b>Nominal</b> (type <code>xs:decimal</code>) and <b>ToleranceUOM</b> (type <code>ToleranceUOMType</code>). The <b>ToleranceUOM</b> box is connected to a choice symbol which branches into <b>NegativeTolerance</b> (type <code>xs:decimal</code>), <b>PositiveTolerance</b> (type <code>xs:decimal</code>), and <b>TotalTolerance</b> (type <code>xs:decimal</code>). The <b>Nominal</b> box is connected to a choice symbol which branches into <b>Minimum</b> (type <code>xs:decimal</code>) and <b>Maximum</b> (type <code>xs:decimal</code>). The entire structure is enclosed in a yellow dashed box labeled <b>JEP30-D10:MassValueSetType</b>.</p>
type	<b>JEP30-D10:MassValueSetType</b> , <b>ValueSetType</b> , <b>StandardType</b> , <b>MinMaxLimitsType</b> , <b>ToleranceUOMType</b> , <b>MassUOMType</b> .
group	<b>ValueSetGroup</b>

The Mass *Units* of Measure are as follows; however, the standard preference is metric in grams.

- |                   |                  |
|-------------------|------------------|
| 1) Microgram (ug) | 4) Kilogram (kg) |
| 2) Miligram (mg)  | 5) Ounce (oz)    |
| 3) Gram (g)       | 6) Pound (lb)    |

## 5.7 Package Shape

path	<b>PartModel/PackageSection/Package-Array/Package/PackageShape.</b>
diagram	<pre> classDiagram     class PackageShapeType {         Rectangle         RectangleConcave         RoundedRectangle         ReferenceRoundedRectangleWithTab         ModifiedRectangle         ModifiedRectangleWithTab         Circle         CircleWithTab         D-Shape         Double-D         RegularPolygon         Segment         Para-truncatedCircle         RoundedDiamond         RoundedChamferedDiamond         IsoscelesTrapezoid         Contour         ShapeCenter         PackageVerticalDimension     }     class PackageShape {         PackageShapeType     }     PackageShapeType --&gt; PackageShape     PackageShapeType --&gt; Rectangle : type   ReferenceRectangleType     PackageShapeType --&gt; RectangleConcave : type   ReferenceRectangleConcaveType     PackageShapeType --&gt; RoundedRectangle : type   ReferenceRoundedRectangleType     PackageShapeType --&gt; ReferenceRoundedRectangleWithTab : type   ReferenceRoundedRectangleWithTabType     PackageShapeType --&gt; ModifiedRectangle : type   PackageReferenceModifiedRectangleType     PackageShapeType --&gt; ModifiedRectangleWithTab : type   ModifiedReferenceRectangleWithTabType     PackageShapeType --&gt; Circle : type   ReferenceCircleType     PackageShapeType --&gt; CircleWithTab : type   ReferenceCircleWithTabType     PackageShapeType --&gt; D-Shape : type   ReferenceD-ShapeType     PackageShapeType --&gt; Double-D : type   ReferenceDouble-DType     PackageShapeType --&gt; RegularPolygon : type   ReferenceRegularPolygonType     PackageShapeType --&gt; Segment : type   ReferenceSegmentType     PackageShapeType --&gt; Para-truncatedCircle : type   ReferencePara-truncatedCircleType     PackageShapeType --&gt; RoundedDiamond : type   ReferenceRoundedDiamondType     PackageShapeType --&gt; RoundedChamferedDiamond : type   ReferenceRoundedChamferedDiamondType     PackageShapeType --&gt; IsoscelesTrapezoid : type   ReferenceIsoscelesTrapezoidType     PackageShapeType --&gt; Contour : type   ContourShapeType     PackageShapeType --&gt; ShapeCenter : type   JEP30-D10:PointXYType     PackageShapeType --&gt; PackageVerticalDimension : type   PackageVerticalDimensionType     </pre>
type	<b>PackageShapeType, ReferenceRectangleType, ReferenceRectangleConcaveType, ReferenceRoundedRectangleType, ReferenceRoundedRectangleWithTabType, PackageReferenceModifiedRectangleType, ModifiedReferenceRectangleWithTabType, ReferenceCircleType, ReferenceCircleWithTabType, ReferenceD-ShapeType, ReferenceDouble-DType, ReferenceRegularPolygonType, ReferenceSegmentType, ReferencePara-truncatedCircleType, ReferenceRoundedDiamondType, ReferenceRoundedChamferedDiamondType, ReferenceIsoscelesTrapezoidType, ContourShapeType, JEP30-D10:PointXYType, PackageVerticalDimensionType.</b>

## 5.7 Package Shape (cont'd)

Table 2 shows an array of dimensions that are captured below each shape listed under [PackageShape](#).

**Table 2 - Package Shape versus Dimensions**

Package Shape	Dimension 1	Dimension 2	Dimension 3	Diameter	Radius	Angle	No. of Sides	Inner/Outer	Tab Length	Tab Width	Tab Orientation	Impacted Corner
Rectangle	Y	Y				Y						
Rectangle Concave	Y	Y			Y	Y						
Rounded Rectangle	Y	Y			Y	Y						
Rounded Rectangle with Tab	Y	Y			Y	Y			Y	Y	Y	
Modified Rectangle	Y	Y				Y						Y
Modified Rectangle with Tab	Y	Y				Y			Y	Y	Y	Y
Circle				Y								
Circle with Tab				Y					Y	Y	Y	
D-Shape	Y	Y				Y						
Double-D	Y	Y				Y						
Regular Polygon				Y		Y	Y	Y				
Segment	Y			Y		Y						
Para-truncated Circle	Y			Y		Y						
Rounded Diamond	Y			Y	Y	Y						
Rounded Chamfered Diamond	Y	Y		Y		Y						
Isosceles Trapezoid	Y	Y	Y			Y						

## 5.7 Package Shape (cont'd)

The Shape Center and the Package Vertical Dimensions are applicable to all shapes. The definition of each shape including Contour is outlined in Annex A (informative) Shape Dimensions.

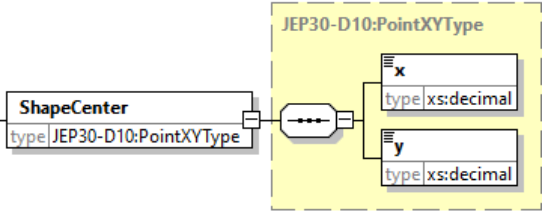
In the event that the modified corners of a modified rectangle shape is not dimensioned within a datasheet, and all corners are modified with the same shape, with one corner having a different dimension on the corner that defines the part orientation, then it is preferred to specify only the corner that defines the part orientation with the unspecified dimension.

### 5.7.1 Contour

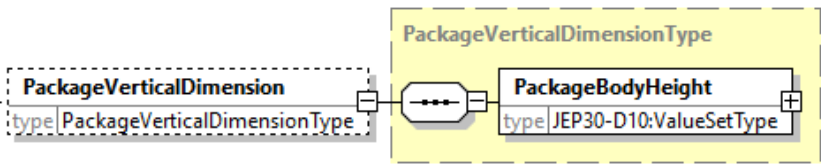
path	<a href="#">PartModel/PackageSection/Package-Array/Package/PackageShape/Contour</a>
diagram	
type	<a href="#">ContourShapeType</a> , <a href="#">ContourToleranceType</a> , <a href="#">JEP30-D10:ToleranceUOMType</a> , <a href="#">Outline-ArrayType</a> , <a href="#">OutlineSegmentType</a> , <a href="#">JEP30-D10:ArcSegmentRotationType</a> , <a href="#">JEP30-D10:FeatureControl-IDsType</a> .

The addition of the [Outline-Array](#) enables the GD&T to be assigned to any segment of the contour outline. The enumerated values for the [ArcSegmentRotation](#) are Clockwise and Anti-clockwise.

5.7.2      Shape Center

path	PartModel/PackageSection/Package-Array/Package/PackageShape/ShapeCenter
diagram	 <p>The diagram shows a class <b>ShapeCenter</b> with a type constraint <code>JEP30-D10:PointXYType</code>. It is connected via a composition relationship (indicated by a solid line with a filled diamond) to a container class. This container class is highlighted with a yellow dashed border and labeled <code>JEP30-D10:PointXYType</code>. Inside the container, there are two attributes: <b>x</b> and <b>y</b>, both with type constraints <code>xs:decimal</code>.</p>
type	JEP30-D10:PointXYType

5.7.3      Package Vertical Dimension

path	PartModel/PackageSection/Package-Array/Package/PackageShape/PackageVerticalDimension
diagram	 <p>The diagram shows a class <b>PackageVerticalDimension</b> with a type constraint <code>PackageVerticalDimensionType</code>. It is connected via a composition relationship (indicated by a solid line with a filled diamond) to a container class. This container class is highlighted with a yellow dashed border and labeled <code>PackageVerticalDimensionType</code>. Inside the container, there is an attribute <b>PackageBodyHeight</b> with a type constraint <code>JEP30-D10:ValueSetType</code> and a multiplicity of 1 (indicated by a '+' sign in a box).</p>
type	PackageVerticalDimensionType, JEP30-D10:ValueSetType,

## 5.8 Part Height

path	<b>PartModel/PackageSection/Package-Array/Package/PartHeight</b>
diagram	
type	<b>JEP30-D10:ValueSetType, StandardType, ToleranceUOMType, m.math.type.</b>
group	<b>ValueSetGroup.</b>

*PartHeight* represents the total height of the part from the seating plane – therefore the sum of the *PackageBodyHeight* plus the *Standoff* (as defined in section 4.13 below) is normally equal to the part height. The exception is when you also have terminals exiting from the “Topside” (as defined by *PartAccessDirection*) of the part, whereby *PartHeight* is then defined as the distance from the seating plane to the top of the terminal that exits from the topside of the part.

The Value Set Type is widely used throughout the Schema. It provides for all the following combinations in a structured way:

- 1) Nominal,
- 2) Positive and Negative Tolerances, or Total tolerance based on the same UOM or based on a percentage of the nominal value, and
- 3) Minimum and or Maximum.

For many applications, both the minimum and maximum Part Height details are required.

5.9 Clearance Region - Array

path	PartModel/PackageSection/Package-Array/Package/ClearanceRegion-Array.
diagram	
type	ClearanceRegionArrayType, JEP30-D10:UnspecifiedValueSetType, StandardType, MinMaxLimitsType, ClearanceRegionType, RectangleType, CircleType, ContourShapeType, ContourToleranceType, Outline-ArrayType, JEP30-D10:PointType.

*ClearanceRegion-Array* represents the space available underneath the part after mounting to the printed board. This space if large enough can allow other parts to be placed in the product design underneath this part. The space also determines the rules applied to the solder fillet for any terminal whose land pattern may encroach inside the package body outline. If there is insufficient space around the terminals in this area, changes to the land pattern may be necessary.

If there is a simple clearance as for example typically found under an SOIC, then the clearance value can be captured under the element *Standoff*. However, many parts have different *ClearanceRegions* under the part. These *ClearanceRegions* can have different shapes, the most common being *Rectangle* or *Circle* shapes, but additional shapes can be captured via the *Contour* element.

## 5.9 Clearance Region - Array (cont'd)

NOTE The element *Unspecified* under the *ClearanceRegion-Array/Standoff*. The purpose of this flag is to enable the user to specify that a Clearance does exist even though its value is not specified, and that the clearance is adequate to provide for a Heel, Toe, Side or Periphery fillet of solder around the Terminal Contact Area without the solder touching the package body, where this could lead to manufacturing defects such as solder balls.

The *Unspecified* element under *ClearanceRegion-Array/ClearanceRegion/Clearance* is used when there are some *ClearanceRegions* shapes that do not have a *Clearance* specified, but the *ClearanceRegion* shape is defined.

This document recommends that the clearance value is properly specified, and that the *Unspecified* field is not used.

The definition of each shape is outlined in Annex A (informative) Shape Dimensions.



5.10 Assembly Technology

path	PartModel/PackageSection/Package-Array/Package/AssemblyTechnology.
diagram	<p>The diagram illustrates the <b>AssemblyTechnologyType</b> hierarchy. It is a base class with several subclasses, each represented by a dashed box. The subclasses are:</p> <ul style="list-style-type: none"><li><b>SMT-MaskReflow</b>: type JEP30-D10:EmptyType</li><li><b>Paste-in-Hole</b>: type JEP30-D10:EmptyType</li><li><b>WaveSoldering</b>: type JEP30-D10:EmptyType</li><li><b>SelectiveMini-WaveSoldering</b>: type JEP30-D10:EmptyType</li><li><b>LaserSoldering</b>: type JEP30-D10:EmptyType</li><li><b>ManualSoldering</b>: type JEP30-D10:EmptyType</li><li><b>Pressfit</b>: type JEP30-D10:EmptyType</li><li><b>WireBonding</b>: type WireBondingType<ul style="list-style-type: none"><li><b>Chip-on-Board</b>: type JEP30-D10:EmptyType</li><li><b>Chip-on-Glass</b>: type JEP30-D10:EmptyType</li><li><b>Other</b>: type xs:string</li></ul></li><li><b>Copper-to-CopperHybridBonding</b>: type JEP30-D10:EmptyType</li><li><b>ThermoCompressionBonding</b>: type JEP30-D10:EmptyType</li><li><b>Other</b>: type xs:string</li></ul> <p>The <b>AssemblyTechnology</b> class is shown as a base class for the <b>AssemblyTechnologyType</b> class, with a type attribute of <b>AssemblyTechnologyType</b>.</p>
type	JEP30-D10:EmptyType, WireBondingType

5.11 CTE - Array

path	PartModel/PackageSection/Package-Array/Package/CTE-Array.
diagram	<p>The diagram illustrates the XSD structure for the CTE-Array. It features a main container 'CTE-Array' which includes a 'ParameterIdentityGroup' and a 'CTE' element. The 'ParameterIdentityGroup' contains several string-based fields: 'Name', 'Symbol', 'SymbolDescription', 'Definition', and 'FootnoteID'. The 'CTE' element is a complex type that includes 'ID', 'Name', 'TemperatureCondition', 'Values', and 'RuleVsDirectional-xy-and-z-Rule'. Additionally, a 'CTE-Graph' element is shown as an optional part of the structure.</p>
type	JEP30-D10:CTE-ArrayType, CTEType, TemperatureConditionType, CTE-ValuesType, CTE-UOMType, RuleVsDirectional-xy-and-z-RuleType, CTE-GraphType.
group	ParameterIdentityGroup,

5.11.1 Temperature Condition

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/TemperatureCondition
diagram	<p>The diagram illustrates the structure of the <b>TemperatureConditionType</b> and <b>ValueSetGroupType</b> classes. <b>TemperatureConditionType</b> is composed of a <b>ParameterIdentityGroup</b> (containing <b>Name</b>, <b>Symbol</b>, <b>SymbolDescription</b>, <b>Definition</b>, and <b>FootnoteID</b>), a <b>Value</b> of type <b>ValueSetGroupType</b>, and a <b>TemperatureUOM</b>. <b>ValueSetGroupType</b> includes <b>Nominal</b>, <b>NegativeTolerance</b>, <b>PositiveTolerance</b>, <b>TotalTolerance</b>, <b>ToleranceUOM</b>, <b>Minimum</b>, <b>Maximum</b>, and <b>FootnoteID</b>. <b>TemperatureUOM</b> is an enumeration with values <b>DegC</b> and <b>K</b>, with <b>DegC</b> as the default.</p>
type	TemperatureConditionType, ValueSetGroupType, Temperature-in-Celsius-or-KelvinUOMType.
group	ParameterIdentityGroup, ValueSetGroup.

The enumerated value for *TemperatureUOM* is *DegC* and *K* but it is set to *DegC* as default.

### 5.11.2 Values

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/Values
diagram	
type	CTE-ValuesType, ValueSetGroupType, CTE-UOMType.
group	ValueVsDirectional-xy-and-z-ValueGroup.

The enumerated value for *CTE-UOM* is *ppm/DegC* and *ppm/K* but it is set to *ppm/DegC* as default.

5.11.3 Rule Vs Directional – xy – and – z - Rule

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/RuleVsDirectional-xy-and-z-Rule
diagram	
type	RuleVsDirectional-xy-and-z-RuleType, RuleType, MinNomMaxRuleContextType, m:math.type, XY-DirectionRuleType, Z-DirectionRuleType.
group	RuleGroup.

#### 5.11.4 CTE - Graph

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph
diagram	<p>The diagram illustrates the XSD structure for CTE-GraphType. It features a main container with a choice between TestConditionDefinition and ParameterDefinition. TestConditionDefinition includes an AxisParameterIdentityGroup (with sub-elements like AxisTitle, Name, Symbol, etc.) and a TemperatureUOM. ParameterDefinition includes a Data-Array and a GraphFormula. Both paths include a Formatting element. A separate CTE-Graph element is also shown with a 0..∞ cardinality. A constraints section is located at the bottom of the diagram.</p>
type	CTE-GraphType, TemperatureParametricGraphChartX-AxisType, Temperature-in-Celsius-or-KelvinUOMType, GraphChartX-AxisFormattingType, CTE-ParametricGraphChartY-AxisType, CTE-UOMType, GraphChartY-AxisFormattingType, CTE-ParameterGraphData-ArrayType, GraphFormulaType, GraphFormattingType.
group	AxisParameterIdentityGroup,

5.11.4 CTE - Graph (cont'd)

A *ParameterGraph* has 2 axis that are defined by the *TestConditionType* (The X-axis definition), and the *ParameterType* (The Y-axis definition). Each axis is labelled by the *AxisTitle*. When possible, the *Symbol* which represents the *AxisTitle* should be added to the PartModel file and should represent a standards-based symbol as defined in the appropriate Terms and Definitions standards. If appropriate, a more detailed *Description* can be used to describe the definition of the *AxisTitle*. Each axis will also have a pre-defined set of *Units* but can be optionally excluded for those axis' which are unitless.

Note that the *ParameterDefinition* is unbounded whereas the *TestConditionDefinition* is bounded to a single instance. This is to cater for those graphs in which there are 2 or more y-axis, each with their own definition.

The graph can either be captured under the *Data-Array* or represented via a *GraphFormula* (A string representing the equation of the *ParameterDefinition* relationship to the *TestTestConditionDefinition*).

5.11.4.1 Formatting

The Formatting is an optional set of data that enables the user to re-create the graph for visualization purposes. Formatting applies to the following

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph/TestConditionDefinition/Formatting
diagram	<p>The diagram illustrates the structure of the <i>GraphChartXAxisFormattingType</i> and its associated subtypes. The main class, <i>GraphChartXAxisFormattingType</i>, is shown with a dashed border and contains several attributes: <i>Range</i> (type <i>GraphAxisRangeType</i>), <i>Inverted</i> (type <i>EmptyType</i>), <i>Scale</i> (type <i>GraphAxisScaleType</i>), and <i>Position</i> (type <i>GraphChartXAxisPositionType</i>). The <i>Range</i> attribute is linked to a <i>GraphAxisRangeType</i> class, which has <i>Minimum</i> (type <i>xs:int</i>) and <i>Maximum</i> (type <i>xs:int</i>) attributes. The <i>Scale</i> attribute is linked to a <i>GraphAxisScaleType</i> class, which has two subtypes: <i>Linear</i> (type <i>GraphAxisScaleLinearType</i>) and <i>Logarithmic</i> (type <i>GraphAxisScaleLogarithmicType</i>). The <i>Linear</i> subtype has a <i>Step</i> attribute (type <i>xs:float</i>). The <i>Logarithmic</i> subtype has two attributes: <i>Natural</i> (type <i>xs:string</i>) and <i>Base</i> (type <i>xs:float</i>, default 10.0). The <i>Position</i> attribute is linked to a <i>GraphChartXAxisPositionType</i> class.</p>
type	JEP30-D10:GraphChartXAxisFormattingType, GraphAxisRangeType, EmptyType, GraphAxisScaleType, GraphAxisScaleLinearType, GraphAxisScaleLogarithmicType, GraphChartXAxisPositionType.

### 5.11.1.1 Formatting (cont'd)

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph/ParameterDefinition/Formatting
diagram	<pre> classDiagram     class GraphChartYAxisFormattingType {         Range         Inverted         Scale         Position     }     class GraphAxisRangeType {         Minimum         Maximum     }     class GraphAxisScaleType {         Linear         Logarithmic     }     class GraphAxisScaleLinearType {         Step     }     class GraphAxisScaleLogarithmicType {         Natural         Base     }     class GraphChartYAxisPositionType {     }      GraphChartYAxisFormattingType "1" -- "*" GraphAxisRangeType : Range     GraphChartYAxisFormattingType "1" -- "*" EmptyType : Inverted     GraphChartYAxisFormattingType "1" -- "*" GraphAxisScaleType : Scale     GraphChartYAxisFormattingType "1" -- "*" GraphChartYAxisPositionType : Position     GraphAxisScaleType "1" -- "*" GraphAxisScaleLinearType : Linear     GraphAxisScaleType "1" -- "*" GraphAxisScaleLogarithmicType : Logarithmic     </pre>
type	JEP30-D10:GraphChartYAxisFormattingType, GraphAxisRangeType, EmptyType, GraphAxisScaleType, GraphAxisScaleLinearType, GraphAxisScaleLogarithmicType, GraphChartYAxisPositionType.

The axis range which is usually defined from minimum to maximum can be inverted to show a graph going from maximum to minimum. The scale can be defined in either a linear step amount, a natural logarithm, or a logarithm of the specified base. The *Base* log is set to a default of *Base 10* but can be defined to any base number.

The *Position* enumerated list for the *GraphChartXAxisPositionType* is

- Top
- Bottom

And for the *GraphChartYAxisPositionType*, the enumerated values are

- Left
- Right



5.11.1.1      Formatting (cont'd)

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph/Formatting
diagram	
type	JEP30-D10:GraphChartYAxisFormattingType, GraphAxisRangeType, EmptyType, GraphAxisScaleType, GraphAxisScaleLinearType, GraphAxisScaleLogarithmicType, GraphChartYAxisPositionType.

The body of the graph can be formatted under the *GraphFormattingType*. The *DisplayType* enumerated list is

- Line
- Bar

The graph Legend can also be positioned around the graph in any of the following locations:

- Location
  - Inside Graph,
  - Outside Graph,
- Vertical Position
  - Top,
  - Center,
  - Bottom,
- • Horizontal Position
  - Left,
  - Center,
  - Right.

5.11.5 Data - Array

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE-Graph/Data-Array
diagram	<p>The diagram illustrates the structure of the <b>CTE-ParameterGraphData-ArrayType</b>. It features several nested and associated classes:</p> <ul style="list-style-type: none"><li><b>CTE-ParameterGraphData-ArrayType</b> (Main Type):<ul style="list-style-type: none"><li>Attributes: <b>ParameterDefinitionID</b> (xs:string).</li><li>Associations:<ul style="list-style-type: none"><li><b>PlotTestCondition</b> (TemperatureGraphPlotConditionType) with multiplicity 1..∞.</li><li><b>Data</b> (GraphDataType) with multiplicity 1..∞.</li><li><b>Formatting</b> (GraphDataFormattingType) with multiplicity 1..∞.</li></ul></li></ul></li><li><b>TemperatureGraphPlotConditionType</b> (Nested Type):<ul style="list-style-type: none"><li>Attributes: <b>ParameterIdentityGroup</b>, <b>Value</b> (ValueSetGroupType), <b>TemperatureUOM</b> (Temperature-in-Celsius-or-KelvinUOMType, default: DegC), <b>ValueText</b> (xs:string), <b>RuleGroup</b>, <b>Legend</b> (GraphDataFormattingLegendType).</li></ul></li><li><b>GraphDataType</b> (Nested Type):<ul style="list-style-type: none"><li>Attributes: <b>TestConditionValue</b> (xs:decimal), <b>ParameterValue</b> (MinNomMaxValueSetType).</li></ul></li><li><b>GraphDataFormattingType</b> (Nested Type):<ul style="list-style-type: none"><li>Attributes: <b>Point</b> (GraphDataFormattingPointType), <b>Line</b> (GraphDataFormattingLineType).</li></ul></li><li><b>GraphDataFormattingPointType</b> (Nested Type):<ul style="list-style-type: none"><li>Attributes: <b>Color</b> (GraphDataFormattingColorType), <b>Style</b> (GraphDataFormattingPointStyleType).</li></ul></li><li><b>GraphDataFormattingLineType</b> (Nested Type):<ul style="list-style-type: none"><li>Attributes: <b>Color</b> (GraphDataFormattingColorType), <b>Style</b> (GraphDataFormattingLineStyleType).</li></ul></li></ul>
type	CTE-ParameterGraphData-ArrayType, TemperatureGraphPlotConditionType, ValueSetGroupType, Temperature-in-Celsius-or-KelvinUOMType, GraphDataFormattingLegendType, GraphDataType, MinNomMaxValueSetType, GraphDataFormattingType, GraphDataFormattingPointType, GraphDataFormattingColorType, GraphDataFormattingPointStyleType, GraphDataFormattingLineType, GraphDataFormattingLineStyleType
group	ParameterIdentityGroup,

5.11.5.1 Plot Test Condition

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph/Data-Array/PlotTestCondition
diagram	<p>The diagram illustrates the structure of the <code>TemperatureGraphPlotConditionType</code>. It features a central <code>TemperatureGraphPlotConditionType</code> container (yellow background) which includes the following elements:</p> <ul style="list-style-type: none"><li><code>PlotTestCondition</code> (type: <code>TemperatureGraphPlotConditionType</code>)</li><li><code>ParameterIdentityGroup</code></li><li><code>RuleGroup</code></li><li><code>Legend</code> (type: <code>GraphDataFormattingLegendType</code>)</li><li><code>ValueSetGroupType</code> (type: <code>ValueSetGroupType</code>)</li><li><code>TemperatureUOM</code> (type: <code>Temperature-in-Celsius-or-KelvinUOMType</code>, default: <code>DegC</code>)</li><li><code>ValueText</code> (type: <code>xs:string</code>)</li></ul> <p>The diagram also shows a <code>Value</code> element (type: <code>ValueSetGroupType</code>) and a <code>TemperatureUOM</code> element (type: <code>Temperature-in-Celsius-or-KelvinUOMType</code>, default: <code>DegC</code>) connected to the <code>ValueSetGroupType</code> and <code>TemperatureUOM</code> elements respectively.</p>
type	TemperatureGraphPlotConditionType, ValueSetGroupType, Temperature-in-Celsius-or-KelvinUOMType, GraphDataFormattingLegendType
group	ParameterIdentityGroup

### 5.11.6 Graph Formula

path	PartModel/PackageSection/Package-Array/Package/CTE-Array/CTE/CTE-Graph/GraphFormula
diagram	
type	GraphFormulaType, MinNomMaxRuleContextType, m:math.type
group	LaTeX-and-MathML-RuleGroup,

5.12 Young Modulus - Array

path	PartModel/PackageSection/Package-Array/Package/YoungsModulus-Array.
diagram	
type	JEP30-D10:YoungsModulus-ArrayType, YoungsModulusType, TemperatureConditionType, YoungsModulusValuesType, YoungsModulus-UOMType, RuleVsDirectional-xy-and-z-RuleType, YoungsModulus-GraphType.
group	ParameterIdentityGroup, ValueVsDirectional-xy-and-z-ValueGroup.

The enumerated values for *YoungsModulus-UOM* are

- N/m^2*, or
- Giga-Pascal (GPa)*.

### 5.12.1 Youngs Modulus- Graph

path	PartModel/PackageSection/Package-Array/Package/YoungsModulus-Array/YoungsModulus-Graph	
diagram		
type	YoungsModulus-GraphType, TemperatureParametricGraphChartX-AxisType, YoungsModulusParameterGraphChartY-AxisType, YoungsModulus-UOMType, GraphChartY-AxisFormattingType, YoungsModulusParameter-GraphData-ArrayType, TemperatureGraphPlotConditionType, GraphDataType, GraphDataFormattingType, GraphFormulaType, GraphFormattingType.	
group	AxisParameterIdentityGroup,	

5.13 Heat Capacity - Array

path	PartModel/PackageSection/Package-Array/Package/YoungsModulus-Array.
diagram	
type	HeatCapacity-ArrayType, HeatCapacityType, HeatCapacityUOMType, JEP30-D10:RuleType
group	JEP30-D10:ParameterIdentityGroup

The enumerated values for *HeatCapacityUOM* are

- *J/DegC,*
- *J/K.*

## 5.14 Terminal Groups

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups</a>
diagram	
type	<a href="#">TerminalsGroupsType</a> , <a href="#">TerminalGroup-ArrayType</a> , <a href="#">TerminalGroupToTerminalGroupRelationships-ArrayType</a> , <a href="#">TerminalDetail-ArrayType</a> , <a href="#">ViaRegion-ArrayType</a> .

Since a package may have several terminal groups under the [TerminalGroup-Array](#), and since there may be dimensional relationships between some or all these groups, an [ID](#) is assigned to each [TerminalGroup](#) that can then be referenced in the [TerminalGroupToTerminalGroup Relationships-Array](#) branch.



5.14.1 Terminal Group - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array
diagram	<p>The diagram illustrates the structure of the TerminalGroup-Array. It is a class hierarchy where TerminalGroup-Array (type TerminalGroup-ArrayType) contains an array of TerminalGroup (type TerminalGroupType). TerminalGroupType contains various attributes: ID (type xs:string), TerminalPosition (type TerminalPositionType), Terminal (type TerminalType), TerminalMaterial (type TerminalMaterialType), CTE-Array (type JEP30-D10:CTE-ArrayType), Coplanarity (type xs:decimal), FlexibleTerminal (type xs:boolean), TerminalFunction (type TerminalBasicFunctionType), TerminalLocation (type TerminalLocationType), PatternGroup (type PackageTerminalPatternGroupType), TerminalShape (type TerminalShapeType), TerminalSpan (type TerminalSpanType), TerminalSpacing (type TerminalSpacingType), and TerminalGroupToBodyRelationship (type TerminalGroupToBodyRelationshipType). There are also constraints on the diagram.</p>
type	TerminalsGroup-ArrayType, TerminalGroupType, TerminalPositionType, TerminalType, TerminalMaterialType, CTE-ArrayType, TerminalBasicFunctionType, TerminalLocationType, TerminalShapeType, TerminalSpanType, TerminalSpacingType, TerminalGroupToBodyRelationship.

*Coplanarity* is defined in JESD88, JEDEC Dictionary of Terms for Solid-State Technology.

*FlexibleTerminal* is a new type of material applied to mostly ceramic chip component terminals for the purpose of reducing the mechanical stress between the component terminal and the printed board. This element is optional and can be set by the component manufacturer when they have incorporated this technology to the part construction.

### 5.14.1.1 Terminal Position

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalPosition.</a>
diagram	
type	<a href="#">TerminalPositionType</a> , <a href="#">BottomUpperPositionConfigurationType</a> , <a href="#">DualPositionConfigurationType</a> , <a href="#">DiagonalCornerConfigurationType</a> , <a href="#">QuadConfigurationType</a> , <a href="#">RadialPositionConfigurationType</a> , <a href="#">SinglePositionConfigurationType</a> , <a href="#">TriplePositionConfigurationType</a> .

The definition of the above [TerminalPosition](#) data elements and their following sub-structures can be found in JESD30, under the “Terminal position Prefix” section which *includes* the “Prefixes for terminal position” table, plus three sections in Annex A, namely:

- Terminal position with additional definition,
- Relationship concepts between the Terminal Contact Area and the Body outline, and
- Position Images.

The only difference between this [TerminalPosition](#) type and the [PackageTerminalPosition](#) type is that the [MixedPosition](#) option is not available to a single [TerminalGroup](#). Refer to sub-sections of 4.5 Package Terminal Position for details of each of the sub-sections under [TerminalPosition](#).

5.14.1.2 Terminal

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/Terminal
diagram	<p>The diagram illustrates the structure of the <b>Terminal</b> data element. It is defined as a class with a <b>type</b> attribute of type <b>TerminalType</b>. The <b>TerminalType</b> class is a base class for various terminal shapes, each with its own <b>type</b> attribute:</p> <ul style="list-style-type: none"><li><b>Ball</b> (type: BallType)</li><li><b>C-bend</b> (type: JEP30-D10:EmptyType)</li><li><b>Lug</b> (type: LugType)</li><li><b>Flat</b> (type: FlatType)</li><li><b>Gull-wing</b> (type: Gull-wingType)</li><li><b>CompressedMountTechnology</b> (type: JEP30-D10:EmptyType)</li><li><b>Post-Terminal</b> (type: PostTerminalType)</li><li><b>J-bend</b> (type: JEP30-D10:EmptyType)</li><li><b>L-bend</b> (type: L-bendType)</li><li><b>Column</b> (type: ColumnType)</li><li><b>Surface-terminal</b> (type: Surface-terminalType)</li></ul> <p>Additionally, there are two separate groups of terminal shapes, each with its own <b>type</b> attribute:</p> <ul style="list-style-type: none"><li><b>Surface-terminal</b> (type: Surface-terminalType)</li><li><b>Pressfit</b> (type: PressfitType)</li><li><b>Pin</b> (type: PinType)</li><li><b>Quick-connect</b> (type: JEP30-D10:EmptyType)</li><li><b>Wraparound</b> (type: WraparoundType)</li><li><b>S-bend</b> (type: S-bendType)</li><li><b>Through-Hole</b> (type: Through-HoleType)</li><li><b>J-inverted</b> (type: JEP30-D10:EmptyType)</li><li><b>TerminalWire</b> (type: WireType)</li><li><b>Screw</b> (type: ScrewType)</li></ul>
type	TerminalType, BallType, JEP30-D10:EmptyType, LugType, FlatType, Gull-wingType, PostTerminalType, L-BendType, ColumnType, SurfaceTerminalType, PressfitType, PinType, WraparoundType, S-BendType, Through-HoleType, WireType, ScrewType.

The definition of the [TerminalCode](#) data elements and their following sub-structures can be found in the JESD30, under the “Terminal Shape Suffix” section which *includes* the “Suffixes for terminal shape” table, plus two sections in Annex A, namely

- Suffixes for terminal shape with additional definition, and
- Illustrations of terminal shape.

NOTE The structure for the [TerminalGroup/Terminal](#) that is under [Package/TerminalGroups/TerminalGroup-Array](#) is exactly the same as for [Package/PackageTerminalCode](#) as outlined in section 4.7 Package Terminal Code above, with the exception that the following three elements that are contained in the [Package/PackageTerminalCode](#) structure are not contained in the [TerminalGroup/Terminal](#) structure. This is because each single [TerminalGroup](#) can only represent a single [Terminal](#) Type.

### 5.13.1.2 Terminal (cont'd)

- Mixed-SMT,
- Mixed-TH, or
- Mixed-Technology.

Refer to 4.7 Package Terminal Code for details of each of the sub-sections under *Terminal*.

### 5.14.1.3 Terminal Material

path	<b>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalMaterial.</b>
diagram	<pre> classDiagram     class TerminalMaterialType {         type TerminalMaterialType     }     class JStd609AlloyCompositionType {         type J-Std-609AlloyCompositionType     }     class JStd609eCodeType {         type J-Std-609e-codeType     }     class BaseMaterialType {         type BaseMaterialType     }     class OtherBaseMaterialType {         type xs:string     }     class PlatingArrayType {         type Plating-ArrayType     }     class PlatingType {         type PlatingType     }     class PlatingIndex {         type xs:integer     }     class PlatingMaterial {         type xs:string     }     class PlatingThickness {         type xs:decimal     }     class PlatingThicknessUOMType {         type PlatingThicknessUOMType     }      TerminalMaterialType --&gt; JStd609AlloyCompositionType     TerminalMaterialType --&gt; JStd609eCodeType     TerminalMaterialType --&gt; BaseMaterialType     TerminalMaterialType --&gt; OtherBaseMaterialType     TerminalMaterialType --&gt; PlatingArrayType     PlatingArrayType --&gt; PlatingType "1..∞"     PlatingType --&gt; PlatingIndex     PlatingType --&gt; PlatingMaterial     PlatingType --&gt; PlatingThickness     PlatingThickness --&gt; PlatingThicknessUOMType </pre>
type	<b>TerminalMaterialType, J-Std-609AlloyCompositionType, J-Std-609e-codeType, BaseMaterialType, Plating-ArrayType, PlatingType, PlatingThicknessUOMType.</b>

The *J-Std-609AlloyComposition* and *J-Std-609e-code* enumerated values are defined in “Annex A (informative) Example Alloys and Associated Material Codes” in the J-Std-609, “Marking, Symbols, and Labels of Leaded and Lead-Free Terminal Finished Materials Used in Electronic Assembly” standard.

The is *PlatingThicknessUOM* specified in *uM*.

5.14.1.4 Terminal Function

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalFunction.
diagram	
type	TerminalBasicFunctionType, JEP30-D10:EmptyType.

5.14.1.5 Terminal Location

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalLocation.
diagram	
type	TerminalLocationType, StandardArrayType, CircularArrayType, TerminalRandomArrayType.

5.14.1.5.1      **Standard Array**

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalLocation
diagram	<p>The diagram illustrates the structure of the <b>StandardArray</b> type and its associated components. The <b>StandardArray</b> type is defined as a <b>StandardArrayType</b> and contains several attributes: <b>nx</b> (xs:integer), <b>ny</b> (xs:integer), <b>dx</b> (JEP30-D10:PitchValueSetType), <b>dy</b> (JEP30-D10:PitchValueSetType), <b>Angle</b> (xs:integer), and <b>TerminalGroupLowerLeftTerminalCenter</b> (JEP30-D10:PointXYType). The <b>dx</b> and <b>dy</b> attributes are associated with the <b>JEP30-D10:PitchValueSetType</b>, which is further detailed in a dashed box. This type includes a <b>Standard</b> attribute (StandardType), a <b>Limits</b> attribute (MinMaxLimitsType), and a <b>PitchToleranceCumulative</b> attribute (xs:boolean). The <b>Standard</b> and <b>Limits</b> attributes are grouped together in a dashed box, and the <b>PitchToleranceCumulative</b> attribute is also grouped with them. The <b>Standard</b> attribute is further associated with a <b>ValueSetGroup</b> type.</p>
type	StandardArrayType, JEP30-D10:PitchValueSetType, StandardType, MinMaxLimitsType, JEP30-D10:PointType.


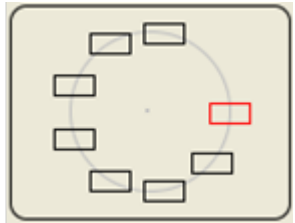
5.14.1.5.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalLocation/CircularArray.
diagram	<p>The diagram illustrates the structure of the <b>CircularArray</b> class and its associated types. The <b>CircularArray</b> class (type <b>CircularArrayType</b>) is shown on the left, connected to a dashed box representing the <b>CircularArrayType</b> structure. This structure contains several attributes and a nested <b>RotationType</b> structure.</p> <ul style="list-style-type: none"><li><b>PitchRadius</b>: type <b>xs:decimal</b></li><li><b>Center</b>: type <b>JEP30-D10:PointXYType</b></li><li><b>StartAngle</b>: type <b>xs:integer</b></li><li><b>AngleToFill</b>: type <b>xs:integer</b></li><li><b>AngleBetweenTerminals</b>: type <b>xs:integer</b></li><li><b>NumberOfTerminals</b>: type <b>xs:integer</b></li><li><b>Rotation</b>: type <b>RotationType</b></li></ul> <p>The <b>RotationType</b> structure (dashed box) contains two attributes:</p> <ul style="list-style-type: none"><li><b>RotateWithCircle</b>: type <b>JEP30-D10:EmptyType</b></li><li><b>RotationAngle</b>: type <b>xs:integer</b></li></ul>
type	CircularArrayType, JEP30-D10:PointXYType, RotationType, JEP30-D10:EmptyType.

### 5.13.1.5.2 Circular Array (cont'd)

Under the [CircularArray](#), Table 3 provides the definition of each elements.

**Table 3 - Terminal Circular Array Elements Definition**

Element	Explanation
Pitch Radius	The radius of the circle of terminals.
Center	The center of the circle of terminals.
Start Angle	The angle at which to place the first terminal in the array. By default, the terminal is at the right-most point on the circle (i.e. the number 3 position on a clock face). This angle specifies a rotation clockwise around the circle from this point.
Angle to Fill	The angle through which the terminals are distributed, starting from the first terminal and continuing clockwise, until the last terminal is reached.
Angle between Terminals	Instead of specifying <b>Angle to fill</b> , you can specify the angle between the centers of each terminal.
Number of Terminals	The number of terminals to be arrayed around the circle. The first terminal is displayed in red. This number includes any missing or deleted terminals in the array.
Rotate with Circle	 <p>If the terminals are rotated so that they are oriented perpendicular to the circle. This does not apply to circular or contour terminal shapes.</p>
Rotation Angle	 <p>If the terminals are not oriented perpendicular to the circle, then the terminals could be rotated around the center of the terminal itself. In this image, each terminal has a “0” degree rotation with respect to the “3 O’clock position”. This does not apply to circular or contour terminal shapes.</p>



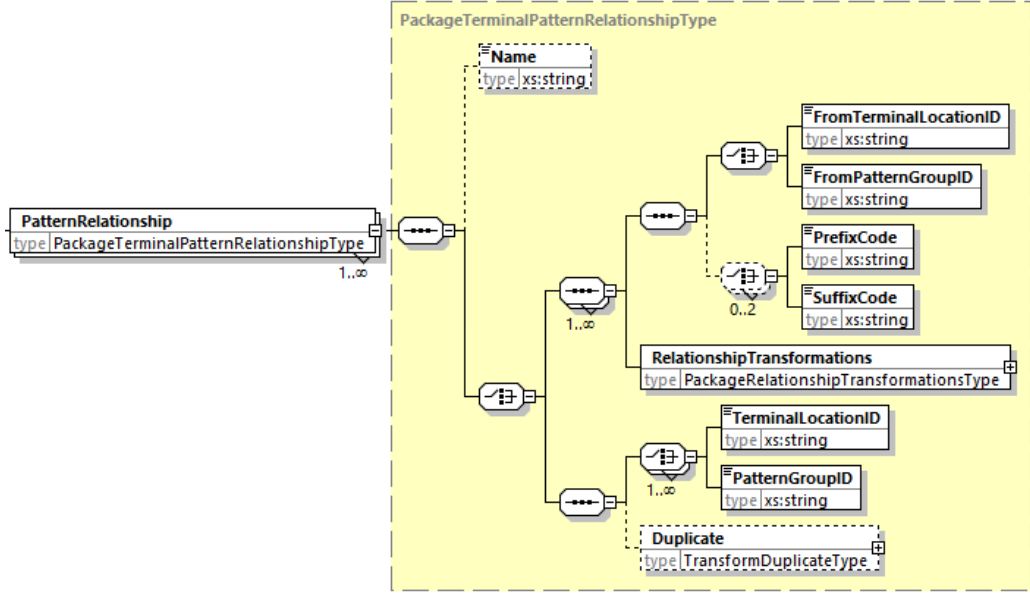
### 5.14.1.5.3 Random Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalLocation/RandomArray</a>
diagram	<pre> classDiagram     class RandomArray {         type TerminalRandomArrayType     }     class TerminalLocation {         type JEP30-D10:PointXYType     }     class JEP30D10PointXYType {         x xs:decimal         y xs:decimal     }     RandomArray "1..∞" -- "1..∞" TerminalLocation     TerminalLocation "1..∞" -- "1..∞" JEP30D10PointXYType         </pre>
type	<a href="#">TerminalRandomArrayType</a> , <a href="#">JEP30-D10:PointXYType</a> .

### 5.14.1.6 Pattern Group

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/PackageGroup</a>
diagram	<pre> classDiagram     class PatternGroup {         type PackageTerminalPatternGroupType     }     class PackageTerminalPatternGroupType {         ID xs:string         Name xs:string         choice             PrefixCode xs:string             SuffixCode xs:string         endchoice         PatternRelationship PackageTerminalPatternRelationshipType     }     class PackageTerminalPatternRelationshipType {         type PackageTerminalPatternRelationshipType     }     PatternGroup "0..∞" -- "1..∞" PackageTerminalPatternGroupType         </pre>
type	<a href="#">PackageTerminalPatternGroupType</a> , <a href="#">PackageTerminalPatternRelationshipType</a> .

5.14.1.6.1 Pattern Relationship

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/PackageGroup/PatternRelationship.
diagram	 <p>The diagram illustrates the XSD structure for the <code>PatternRelationship</code> element. The root element is <code>PatternRelationship</code>, which is of type <code>PackageTerminalPatternRelationshipType</code> and has a cardinality of <code>1..∞</code>. It is composed of the following elements:</p> <ul style="list-style-type: none"><li><code>Name</code> (type <code>xs:string</code>): A required element.</li><li><code>FromTerminalLocationID</code> (type <code>xs:string</code>): A required element.</li><li><code>FromPatternGroupID</code> (type <code>xs:string</code>): A required element.</li><li><code>PrefixCode</code> (type <code>xs:string</code>): An optional element with a cardinality of <code>0..2</code>.</li><li><code>SuffixCode</code> (type <code>xs:string</code>): An optional element with a cardinality of <code>0..2</code>.</li><li><code>RelationshipTransformations</code> (type <code>PackageRelationshipTransformationsType</code>): A required element.</li><li><code>TerminalLocationID</code> (type <code>xs:string</code>): A required element.</li><li><code>PatternGroupID</code> (type <code>xs:string</code>): A required element.</li><li><code>Duplicate</code> (type <code>TransformDuplicateType</code>): An optional element.</li></ul>
type	<code>PackageTerminalPatternRelationshipType</code> , <code>PackageRelationshipTransformationsType</code> , <code>TransformDuplicateType</code> .

5.14.1.6.1.1 Relationship Transformations

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/PackageGroup/PatternRelationship/RelationshipTransformations.
diagram	<p>The diagram illustrates the XSD structure for RelationshipTransformations. The root element is <b>RelationshipTransformations</b> (type <b>PackageRelationshipTransformationsType</b>). It is composed of two main parts:</p> <ul style="list-style-type: none"><li><b>Sequence of Transformations (0..2):</b> A sequence of up to two transformations, each containing:<ul style="list-style-type: none"><li><b>ToTerminalLocationID</b> (type <b>xs:string</b>)</li><li><b>ToPatternGroupID</b> (type <b>xs:string</b>)</li><li><b>PrefixCode</b> (type <b>xs:string</b>)</li><li><b>SuffixCode</b> (type <b>xs:string</b>)</li></ul></li><li><b>Single Transformation (1..∞):</b> A single transformation, which can be either a <b>Mirror</b> or a <b>Rotate</b>.<ul style="list-style-type: none"><li><b>Mirror</b> (type <b>TransformMirrorType</b>):<ul style="list-style-type: none"><li><b>Origin</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>SelectionCenter</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>Horizontal</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>Vertical</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>Coordinate</b> (type <b>JEP30-D10:PointXYType</b>)</li></ul></li><li><b>Rotate</b> (type <b>TransformRotateType</b>):<ul style="list-style-type: none"><li><b>Origin</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>SelectionCenter</b> (type <b>JEP30-D10:EmptyType</b>)</li><li><b>Coordinate</b> (type <b>JEP30-D10:PointXYType</b>)</li><li><b>Angle</b> (type <b>xs:decimal</b>)</li></ul></li></ul></li></ul> <p>The <b>JEP30-D10:PointXYType</b> base type is defined as:</p> <ul style="list-style-type: none"><li><b>x</b> (type <b>xs:decimal</b>)</li><li><b>y</b> (type <b>xs:decimal</b>)</li></ul>
type	PackageRelationshipTransformationsType, TransformMirrorType, TransformRotateType, JEP30-D10:EmptyType, JEP30-D10:PointXYType,

### 5.14.1.6.1.2 Duplicate

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/PatternGroup/PatternRelationship/Duplicate.
diagram	<p>JEP30-D10:TransformDuplicateType</p> <p><b>nx</b> type xs:integer</p> <p><b>ny</b> type xs:integer</p> <p><b>Duplicate</b> type JEP30-D10:TransformDuplicateType</p> <p><b>X-PrefixCode</b> type xs:string 1..∞ The max occurrence = nx</p> <p><b>X-PrefixRowCode</b> type RowTerminalNumberOrderingType</p> <p><b>X-SuffixCode</b> type xs:string 1..∞ The max occurrence = nx</p> <p><b>X-SuffixRowCode</b> type RowTerminalNumberOrderingType</p> <p><b>Y-PrefixCode</b> type xs:string 1..∞ The max occurrence = ny</p> <p><b>Y-PrefixColumnCode</b> type ColumnTerminalNumberOrderingType</p> <p><b>Y-SuffixCode</b> type xs:string 1..∞ The max occurrence = ny</p> <p><b>Y-SuffixColumnCode</b> type ColumnTerminalNumberOrderingType</p> <p><b>PrefixCode</b> type xs:string 1..∞</p> <p><b>PrefixNumericalSequence</b> type NumericalSequenceType</p> <p><b>PrefixAlphabeticalSequence</b> type AlphabeticalSequenceType</p> <p><b>PrefixSeperator</b> type xs:string</p> <p><b>SuffixCode</b> type xs:string 1..∞</p> <p><b>SuffixNumericalSequence</b> type NumericalSequenceType</p> <p><b>SuffixAlphabeticalSequence</b> type AlphabeticalSequenceType</p> <p><b>SuffixSeperator</b> type xs:string</p> <p><b>dx</b> type xs:decimal</p> <p><b>dy</b> type xs:decimal</p> <p><b>FootnoteID</b> type xs:string 0..∞</p>
type	TransformDuplicateType, RowTerminalNumberOrderingType, ColumnTerminalNumberOrderingType, NumericalSequenceType, AlphabeticalSequenceType.

### 5.14.1.7 Terminal Shape

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape
diagram part 1 of 9	
type	TerminalShapeType, JEP30-D10:EmptyType, Pad-on-PackageType, SoldermaskOpening-on-PackageType, BallTerminalShapeType, ColumnTerminalShapeType.
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Pad-on-Package
diagram part 1.1 of 9	
type	Pad-on-PackageType, ReferenceRectangleGroupType, ReferenceRoundedRectangleGroupType, ReferenceModifiedRectangleGroupType, ReferenceCircleGroupType, ReferenceDouble-DGroupType, ReferenceRegularPolygonGroupType, ReferenceTerminalContourGroupType, JEP30-D10:UnspecifiedDimensionalValueSetGroupType

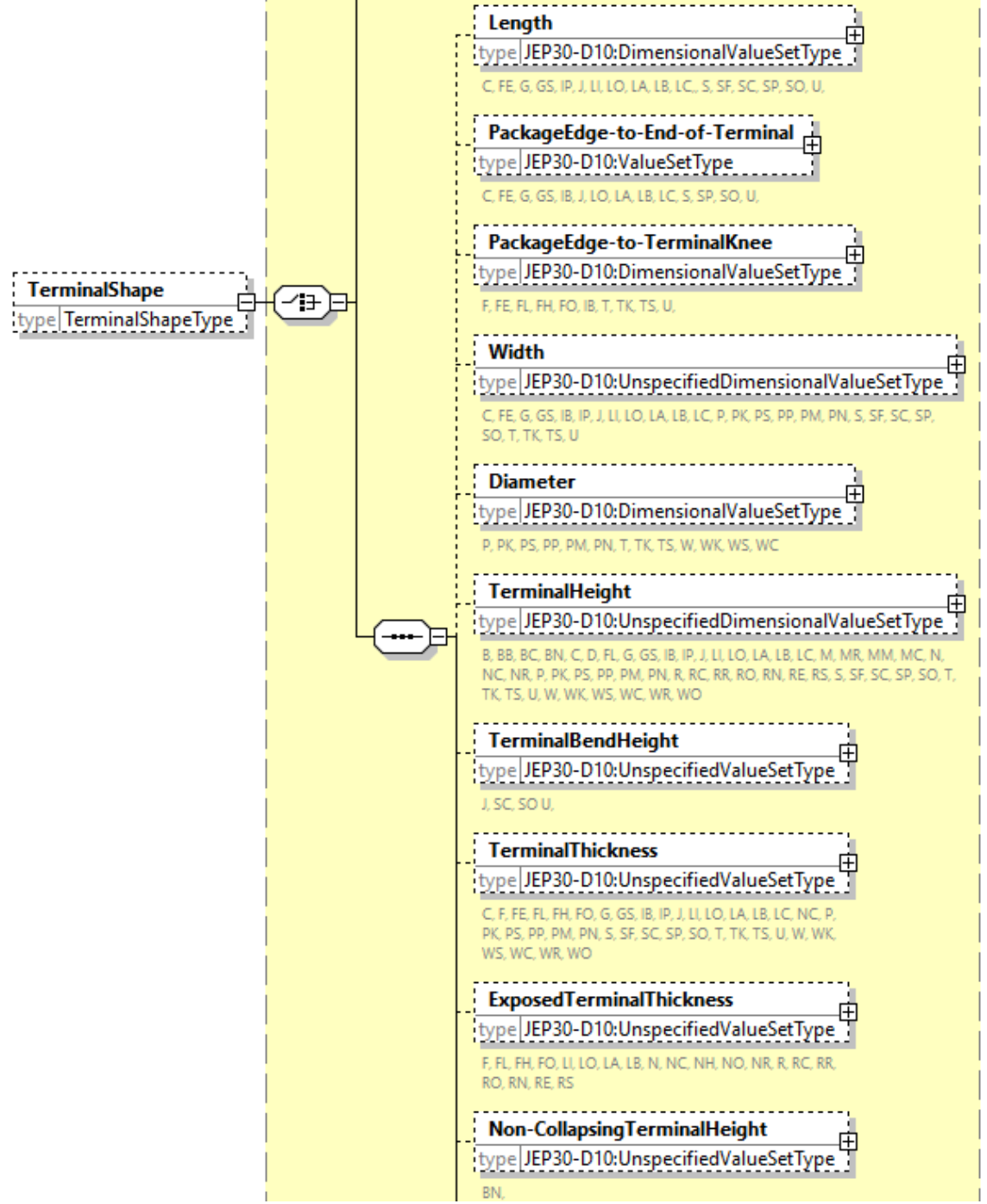
### 5.14.1.7 Terminal Shape (cont'd)

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/SoldermaskOpening-on-Package</a>
diagram part 1.2 of 9	
type	<a href="#">SoldermaskOpening-on-PackageType</a> , <a href="#">ReferenceRectangleGroupType</a> , <a href="#">ReferenceRoundedRectangleGroupType</a> , <a href="#">ReferenceModifiedRectangleGroupType</a> , <a href="#">ReferenceCircleGroupType</a> , <a href="#">ReferenceDouble-DGroupType</a> , <a href="#">ReferenceRegularPolygonGroupType</a> , <a href="#">ReferenceTerminalContourGroupType</a> , <a href="#">JEP30-D10:UnspecifiedDimensionalValueSetGroupType</a>
path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Ball</a>
diagram part 1.2 of 9	
type	<a href="#">BallTerminalShapeType</a> , <a href="#">ReferenceRoundedRectangleGroupType</a> , <a href="#">ReferenceCircleGroupType</a> , <a href="#">ReferenceRegularPolygonGroupType</a> , <a href="#">JEP30-D10:UnspecifiedDimensionalValueSetGroupType</a> , <a href="#">JEP30-D10:UnspecifiedValueSetGroupType</a>

5.14.1.7 Terminal Shape (cont'd)

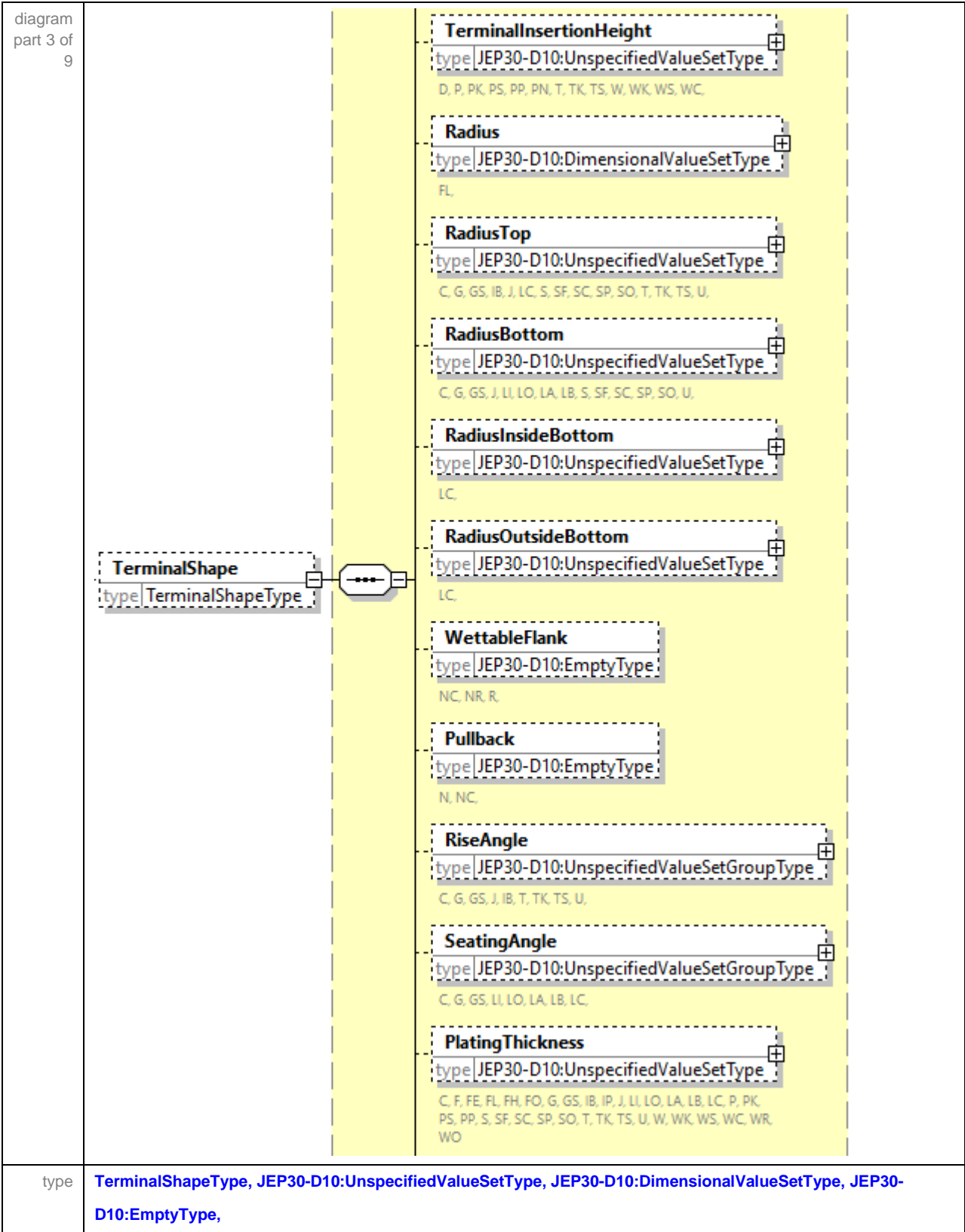
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Column
diagram part 1.4 of 9	
type	ColumnTerminalShapeType, ReferenceCircleGroupType, ReferenceRegularPolygonType, UpperSide-of-ColumnShapeType, ReferenceCircleGroupType, ReferenceRegularPolygonGroupType, LowerSide-of-ColumnShapeType, JEP30-D10:UnspecifiedDimensionalValueSetType, ColumnCoreType, JEP30-D10:DimensionalValueSetType, JEP30-D10:ValueSetType, JEP30-D10:UnspecifiedDimensionalValueSetType, JEP30-D10:UnspecifiedValueSetType.

### 5.14.1.7 Terminal Shape (cont'd)

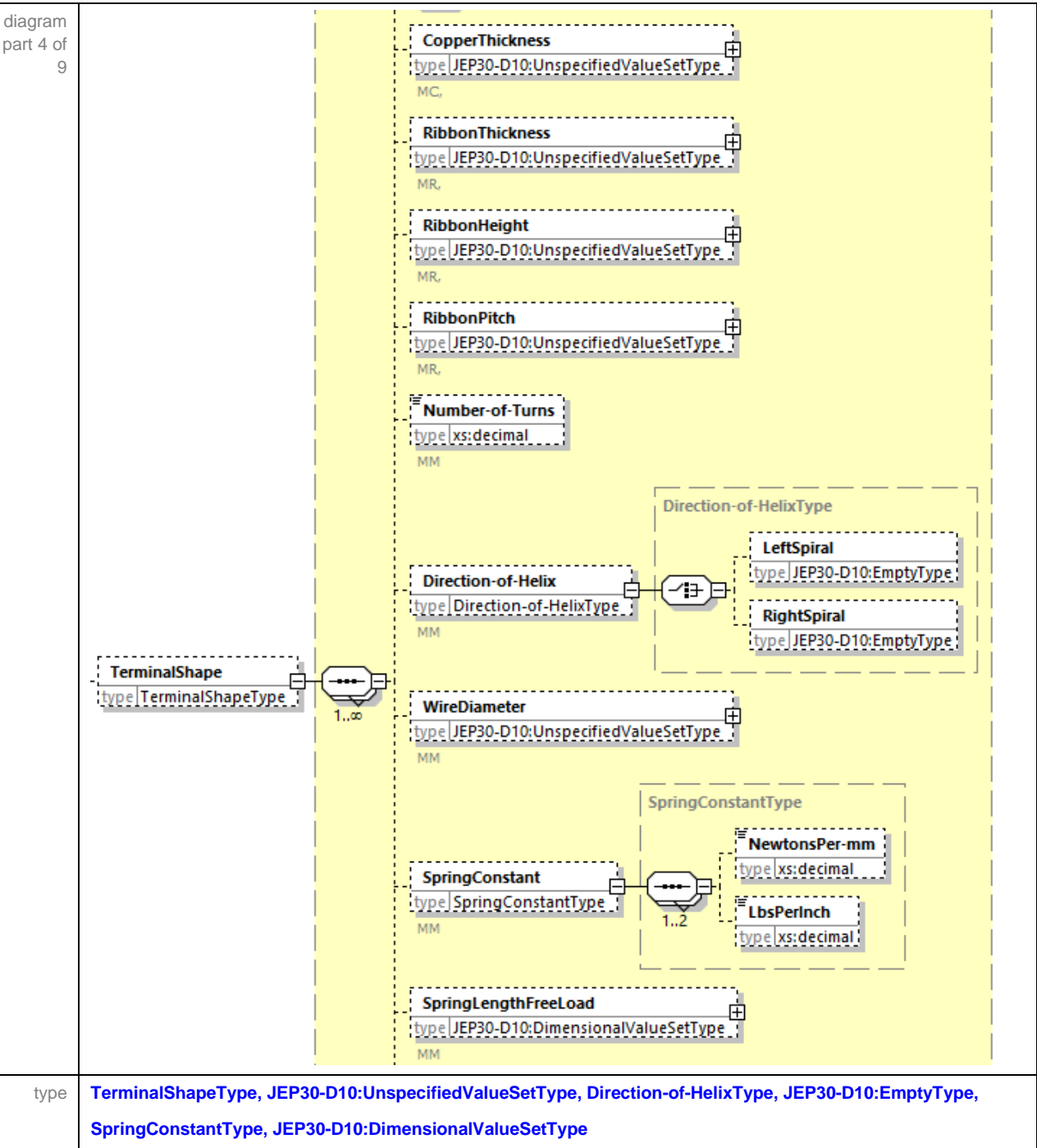
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Column		
diagram part 2 of 9	 <p><b>TerminalShape</b> type TerminalShapeType</p> <p><b>Length</b> type JEP30-D10:DimensionalValueSetType C, FE, G, GS, IP, J, LI, LO, LA, LB, LC, S, SF, SC, SP, SO, U,</p> <p><b>PackageEdge-to-End-of-Terminal</b> type JEP30-D10:ValueSetType C, FE, G, GS, IB, J, LO, LA, LB, LC, S, SP, SO, U,</p> <p><b>PackageEdge-to-TerminalKnee</b> type JEP30-D10:DimensionalValueSetType F, FE, FL, FH, FO, IB, T, TK, TS, U,</p> <p><b>Width</b> type JEP30-D10:UnspecifiedDimensionalValueSetType C, FE, G, GS, IB, IP, J, LI, LO, LA, LB, LC, P, PK, PS, PP, PM, PN, S, SF, SC, SP, SO, T, TK, TS, U</p> <p><b>Diameter</b> type JEP30-D10:DimensionalValueSetType P, PK, PS, PP, PM, PN, T, TK, TS, W, WK, WS, WC</p> <p><b>TerminalHeight</b> type JEP30-D10:UnspecifiedDimensionalValueSetType B, BB, BC, BN, C, D, FL, G, GS, IB, IP, J, LI, LO, LA, LB, LC, M, MR, MM, MC, N, NC, NR, P, PK, PS, PP, PM, PN, R, RC, RR, RO, RN, RE, RS, S, SF, SC, SP, SO, T, TK, TS, U, W, WK, WS, WC, WR, WO</p> <p><b>TerminalBendHeight</b> type JEP30-D10:UnspecifiedValueSetType J, SC, SO, U,</p> <p><b>TerminalThickness</b> type JEP30-D10:UnspecifiedValueSetType C, F, FE, FL, FH, FO, G, GS, IB, IP, J, LI, LO, LA, LB, LC, NC, P, PK, PS, PP, PM, PN, S, SF, SC, SP, SO, T, TK, TS, U, W, WK, WS, WC, WR, WO</p> <p><b>ExposedTerminalThickness</b> type JEP30-D10:UnspecifiedValueSetType F, FL, FH, FO, LI, LO, LA, LB, N, NC, NH, NO, NR, R, RC, RR, RO, RN, RE, RS</p> <p><b>Non-CollapsingTerminalHeight</b> type JEP30-D10:UnspecifiedValueSetType BN,</p>		
type	TerminalShapeType, JEP30-D10:DimensionalValueSetType, JEP30-D10:ValueSetType, JEP30-D10:UnspecifiedDimensionalValueSetType, JEP30-D10:UnspecifiedValueSetType.		



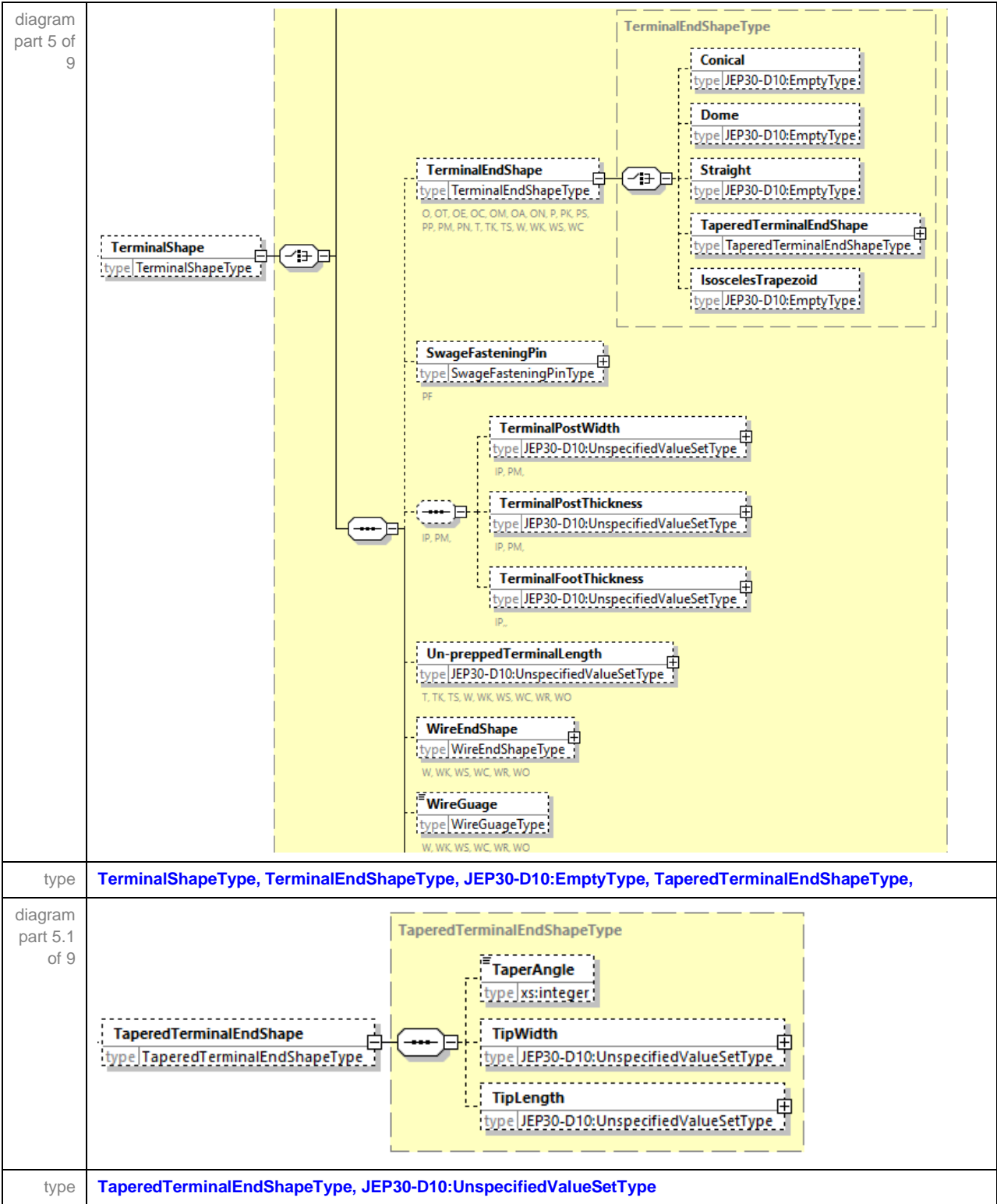
5.14.1.7 Terminal Shape (cont'd)



5.14.1.7 Terminal Shape (cont'd)



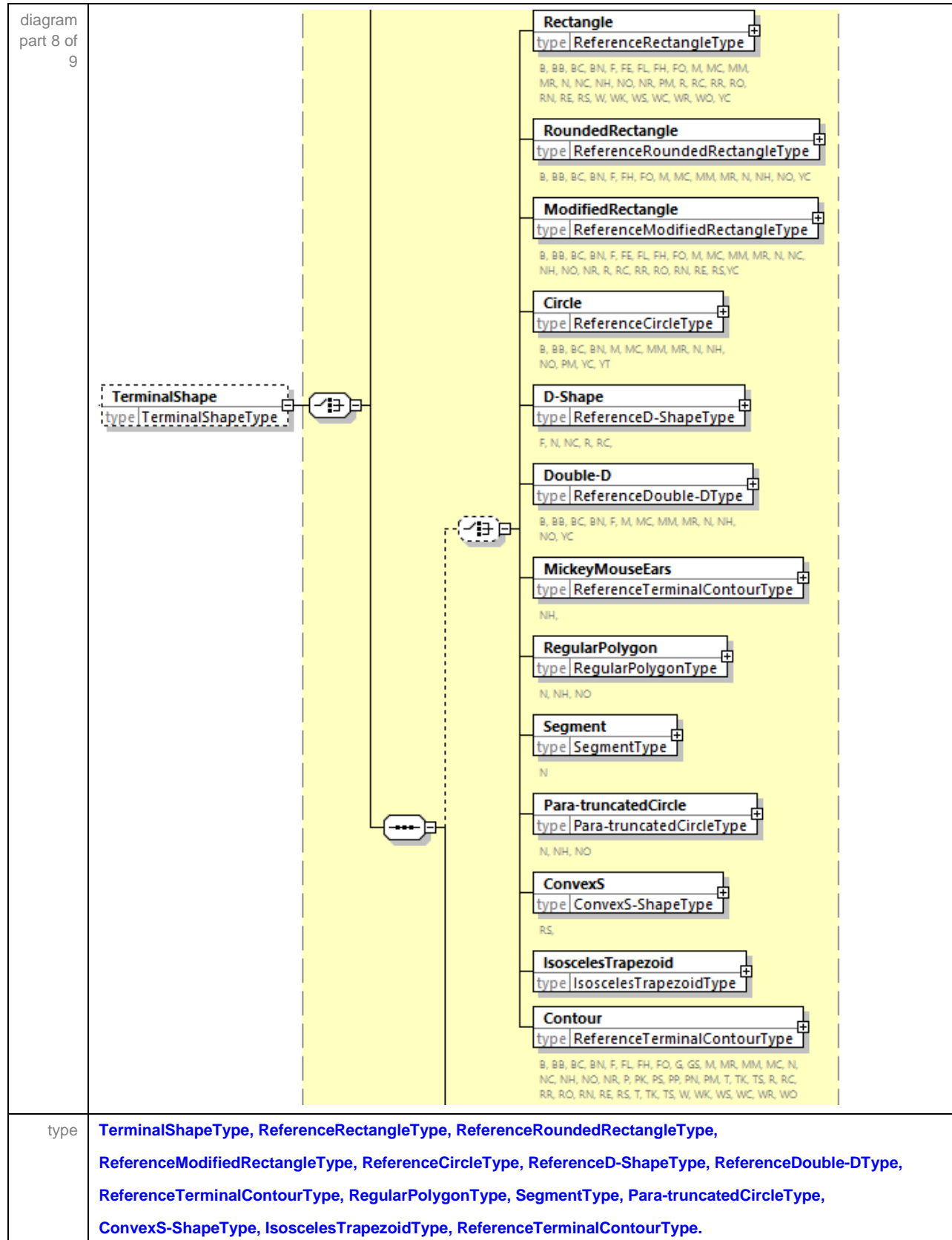
5.14.1.7 Terminal Shape (cont'd)



### 5.14.1.7 Terminal Shape (cont'd)

<p>diagram part 6 of 9</p>	
<p>type</p>	<p><b>TerminalShapeType, JEP30-D10:UnspecifiedValueType</b></p>
<p>diagram part 7 of 9</p>	
<p>type</p>	<p><b>TerminalShapeType, ModifiedCornerType, JEP30-D10:EmptyType, ConfigurableShoulderType, NeckType, DambarType, KinkedType.</b></p>

### 5.14.1.7 Terminal Shape (cont'd)



5.14.1.7      Terminal Shape (cont'd)

diagram part 9 of 9	
type	TerminalShapeType, Castellation-ArrayType, TerminalVoid-ArrayType, JEP30-D10:EmptyType, JEP30-D10:UnspecifiedValueSetType.

### 5.14.1.7.1 Ball Types

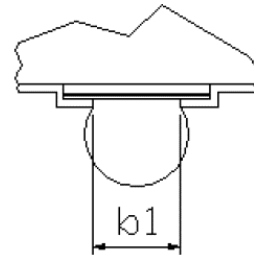
Table 4 shows an array of dimensions that are applicable to this terminal:

**Table 4 - Ball Dimensions**

Schema Labels	Ball Types	Applicable Shapes to capture			Schema Types
		Terminal Shape	Soldermask Opening	Pad Shape	
Rectangle	All		NSMD	SMD	ReferenceRectangleType
Rounded Rectangle	Bump	SMD, NSMD	SMD, NSMD	SMD, NSMD	ReferenceRoundedRectangleType
	Collapsing, Non-Collapsing		NSMD	SMD	
Modified Rectangle	All		NSMD	SMD	ReferenceModifiedRectangleType
Circle	All	SMD, NSMD	SMD, NSMD	SMD, NSMD	ReferenceCircleType
Double-D	All		NSMD	SMD	ReferenceDouble-DType
Regular Polygon	All	SMD, NSMD	SMD, NSMD	SMD, NSMD	ReferenceRegularPolygonType
Contour	Bump		NSMD	SMD	ReferenceTerminalContourType
Terminal Height	All	SMD, NSMD	n/a	n/a	JEP30-D10: UnspecifiedDimensionalValueSetType
Non-Collapsing Terminal Height	All	SMD, NSMD	n/a	n/a	JEP30-D10:UnspecifiedValueSetType

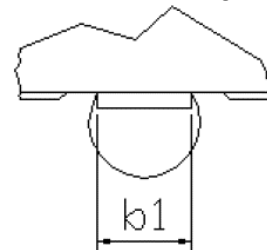
The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

The Terminal shapes is mandatory for all combinations of Ball types. The solderable surface of the terminal to the package may be defined by an opening in the soldermask resist layer as defined in Figure 1 – Solder Mask Defined Pad. For this type of SMD Pad, the Soldermask Opening shape is mandatory while the Pad shape is optional.



**Figure 1 - Solder Mask Defined Pad**

Alternatively, the solderable surface of the terminal to the package may be defined by the size of the metalized pad as defined in Figure 2 –Non Solder Mask Defined Pad. For this type of NSMD Pad, the Soldermask Opening shape is optional while the Pad shape is mandatory.



**Figure 2 - Non Solder Mask Defined Pad**

The preferred method of capturing ball dimensions is via the [BallTerminalShapeType](#) as shown in diagram 1.3 of 9. Information as specified in diagrams 1, 1.1, 1.2 and 1.3 are applicable for Ball terminals.

### 5.14.1.7.2 C-bend

Table 5 shows an array of dimensions that are applicable to this terminal.

**Table 5 - C-bend Dimensions**

Schema Labels	Symbol	Schema Types	Data Requirements
Length	L	JEP30-D10:DimensionalValueSetType	Mandatory – but can be derived by other dimensions provided
PackageEdge-to-end-of-Terminal	L1	JEP30-D10:ValueSetType	Mandatory – but can be derived by other dimensions provided
Width	b	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalHeight	A3	JEP30-D10:UnspecifiedDimensionalValueSetType	Optional
TerminalThickness	c	JEP30-D10:UnspecifiedValueSetType	Mandatory
RadiusTop	R1	JEP30-D10:UnspecifiedValueSetType	Optional
RadiusBottom	R2	JEP30-D10:UnspecifiedValueSetType	Optional
SeatingAngle	$\Phi 1$	JEP30-D10:UnspecifiedValueSetType	Optional
RiseAngle	$\Phi 2$	JEP30-D10:UnspecifiedValueSetType	Optional
PlatingThickness	p	JEP30-D10:UnspecifiedValueSetType	Mandatory

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions

### 5.14.1.7.3 Lug

Table 6 shows an array of dimensions that are applicable to this terminal.

**Table 6 - Lug Dimensions**

Schema Labels	Symbol	Schema Types	Data Requirements
TerminalHeight	A3	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalInsertionHeight		JEP30-D10:UnspecifiedValueSetType	Mandatory
Shoulder		ConfigurableShoulderType	Optional

In addition to this table of dimensions, any one of the 2D shapes as defined in diagram 2 of X in this section. The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.



#### 5.14.1.7.4 Flat

Table 7 shows an array of dimensions that are applicable to this terminal.

**Table 7 - Flat Terminal Type Dimensions**

Schema Labels	Flat	Elevated	Flat – L-bend	Hole	With- opening	Schema Types
One of the following 2D shapes is applicable to this Terminal type						
Rectangle	Y	Y	Y	Y	Y	ReferenceRectangleType
RoundedRectangle	Y			Y	Y	ReferenceRoundedRectangleType
ModifiedRectangle	Y	Y	Y	Y	Y	ReferenceModifiedRectangleType
D-Shape	Y					ReferenceD-ShapeType
Double-D	Y			Y	Y	ReferenceDouble-DType
The dimensions below are mandatory if applicable						
Terminal Height			Y			JEP30-D10: UnspecifiedDimensionalValueSetType
TerminalThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
ExposedTerminalThickness	Opt		Opt	Opt	Opt	JEP30-D10:UnspecifiedValueSetType
PackageEdge-to-End-of- Terminal	Y	Y	Y	Y	Y	JEP30-D10:ValueSetType
PlatingThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
TerminalVoid-Array				Y	Y	TerminalVoid-ArrayType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

### 5.14.1.7.5 Gull-wing

Table 8 shows an array of dimensions that are applicable to this terminal.

**Table 8 - Gull-wing Dimensions**

Schema Labels	Symbol	Schema Types	Data Requirements
Length	L	JEP30-D10:DimensionalValueSetType	Mandatory – but can be derived by other dimensions provided
PackageEdge-to-end-of-Terminal	L1	JEP30-D10:ValueSetType	Mandatory – but can be derived by other dimensions provided
Width	b	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
ModifiedCorner		ImpactedCornerType	Optional
Contour		ReferenceTerminalContourType	Optional
TerminalHeight	A3	JEP30-D10:UnspecifiedDimensionalValueSetType	Optional
TerminalThickness	c	JEP30-D10:UnspecifiedValueSetType	Mandatory
RadiusTop	R1	JEP30-D10:UnspecifiedValueSetType	Optional
RadiusBottom	R2	JEP30-D10:UnspecifiedValueSetType	Optional
SeatingAngle	Φ1	JEP30-D10:UnspecifiedValueSetType	Optional
RiseAngle	Φ2	JEP30-D10:UnspecifiedValueSetType	Optional
PlatingThickness	p	JEP30-D10:UnspecifiedValueSetType	Mandatory
One of the following two elements is applicable to this Terminal type			
ToeUp		JEP30-D10:EmptyType	If omitted, and seating angle is specified, then the default is <b>ToeDown</b> .
ToeDown		JEP30-D10:EmptyType	
One of the following elements is applicable if there is a Shoulder or Neck on this Gull-wing			
Shoulder		ConfigurableShoulderType	Optional
Neck		NeckType	Optional
If the Gull-wing has a Dambar, then the following element is Optional			
Dambar		DambarType	Optional

Shoulder and Kinked are described in sections 4.15.1.5.15 Shoulder and 4.15.1.5.16 Kinked below, since they are applicable to other terminals.

#### 5.14.1.7.5.1 Modified Corner

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner</a>
diagram	
type	<a href="#">ModifiedCornerType</a> , <a href="#">ImpactedTerminalType</a> , <a href="#">TerminalCenterType</a> , <a href="#">Impact-to-TerminalGroupType</a> , <a href="#">Apply-to-all-TerminalsType</a> , <a href="#">SymmetryType</a> , <a href="#">ImpactedCornerType</a> , <a href="#">CornerType</a> .

#### 5.14.1.7.5.1.1 Impacted Terminal

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ImpactedTerminal</a>
diagram	
type	<a href="#">ImpactedTerminalType</a> , <a href="#">TerminalCenterType</a> , <a href="#">JEP30-D10:PointXYType</a> , <a href="#">JEP30-D10:MinIntegerOfOneType</a> .

#### 5.14.1.7.5.1.2 Impact – to – Terminal Group

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup</a>
diagram	
type	<a href="#">Impact-to-TerminalGroupType</a> , <a href="#">Apply-to-all-TerminalsType</a> , <a href="#">SymmetryType</a> , <a href="#">SymmetryRotationType</a> , <a href="#">ReflectionType</a> ,

#### 5.14.1.7.5.1.2.1 Apply – to – all - Terminals

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner</a>
diagram	
type	<a href="#">Apply-to-all-TerminalsType</a> , <a href="#">CornerImpact-to-StandardArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> .

#### 5.14.1.7.5.1.2.2 Rotation

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup/Symmetry/Rotation
diagram	
type	SymmetryRotationType, SymmetryRotationAxisType, SymmetryRotationCenterType, JEP30-D10:EmptyType.

#### 5.14.1.7.5.1.2.3 Reflection

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup/Symmetry/Reflection
diagram	
type	ReflectionType, ReflectionAxisType, JEP30-D10:EmptyType, ReflectionInversionCenterType.

5.14.1.7.5.1.3      Shape Impacted Corner

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner
diagram	
type	ImpactedCornerType, CornerType, ChamferedCornerType, RectangularInCornerType, ConvexCornerType, ConcaveCornerType, CornerArcType.

#### 5.14.1.7.5.1.4 Corner Type

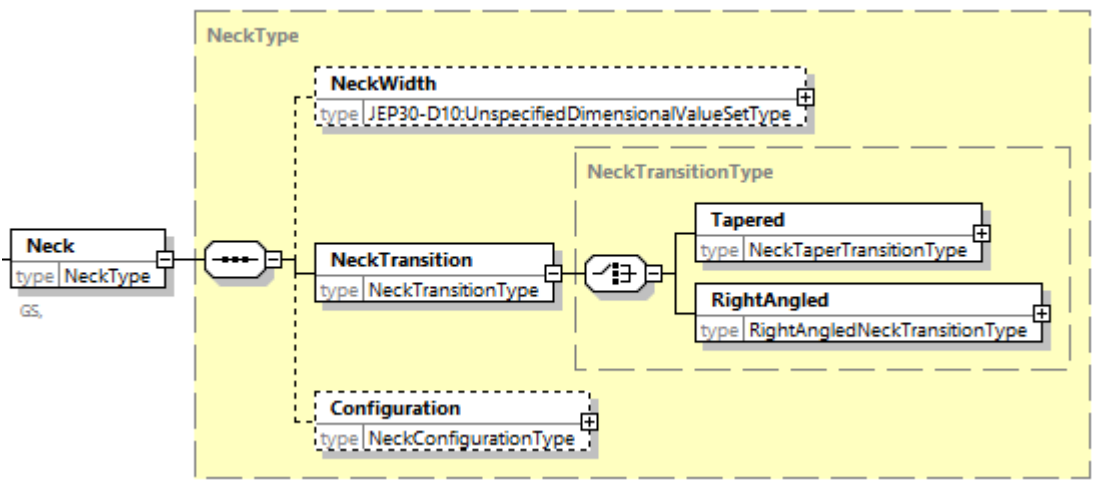
path	<p>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/NE</p> <p>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/SE</p> <p>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/SW</p> <p>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/NW</p>
diagram	<pre> classDiagram     class CornerType {         &lt;&lt;abstract&gt;&gt;     }     class ChamferedCorner {         DX         DY     }     class RectangularInCorner {         DX         DY     }     class ConvexCorner {         Radius         ConvexCornerCenter     }     class ConcaveCorner {         Radius         ConcaveCornerCenter     }     class CornerArc {         Radius1         Radius2         Neck         CornerArcCenter     }     CornerType &lt; -- ChamferedCorner     CornerType &lt; -- RectangularInCorner     CornerType &lt; -- ConvexCorner     CornerType &lt; -- ConcaveCorner     CornerType &lt; -- CornerArc     </pre>
type	<p>ImpactedCornerType, CornerType, ChamferedCornerType, RectangularInCornerType, ConvexCornerType, ConcaveCornerType, CornerArcType.</p>

5.14.1.7.5.2 Reference Terminal Contour Type

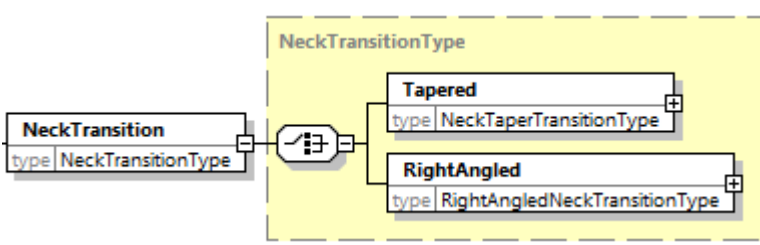
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Contour
diagram	<div><div><div><div><div>Contour</div><div>type ReferenceTerminalContourType</div><div>B, BB, BC, BN, F, FL, FH, FO, G, GS, M, MR, MM, MC, N, NC, NH, NO, NR, P, PK, PS, PP, PN, T, TK, TS, R, RC, RR, RO, RN, RE, RS, T, TK, TS, W, WK, WS, WC, WR, WO</div></div></div><div><div>ReferenceTerminalContourType</div><div><div><div>ID</div><div>type xs:string</div></div><div><div>X-Tolerance</div><div>type ContourToleranceType</div></div><div><div>Y-Tolerance</div><div>type ContourToleranceType</div></div><div><div>OutlineReference</div><div>type OutlineReferenceType</div></div><div><div>ImpactedTerminal</div><div>type ImpactedTerminalType</div></div><div><div>Impact-to-TerminalGroup</div><div>type Impact-to-TerminalGroupType</div></div><div><div>FootnoteID</div><div>type xs:string</div></div></div><div>0..∞</div></div><div><div>OutlineReferenceType</div><div><div><div>TerminalCenter</div><div>type JEP30-D10:EmptyType</div></div><div><div>PackageBodyCenter</div><div>type JEP30-D10:EmptyType</div></div><div><div>Outline-Array</div><div>type Outline-ArrayType</div></div></div></div></div></div>
type	ReferenceTerminalContourType, ContourToleranceType, OutlineReferenceType, JEP30-D10:EmptyType, Outline-ArrayType, ImpactedTerminalType, Impact-to-TerminalGroupType.



5.14.1.7.5.3 Neck

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Neck
diagram	 <p>The diagram illustrates the structure of the Neck class. It is a class with a type NeckType and a generalization relationship (GS) to the NeckTransition class. The Neck class has a dashed box labeled NeckType containing three elements: NeckWidth (type JEP30-D10:UnspecifiedDimensionalValueSetType), NeckTransitionType (type NeckTransitionType), and Configuration (type NeckConfigurationType). The NeckTransition class has a dashed box labeled NeckTransitionType containing two elements: Tapered (type NeckTaperTransitionType) and RightAngled (type RightAngledNeckTransitionType).</p>
type	NeckType, JEP30-D10:UnspecifiedDimensionalValueSetType, NeckTransitionType, NeckTaperTransitionType, RightAngledNeckTransitionType, ShoulderConfigurationType

5.14.1.7.5.3.1 Neck Transition

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Neck/NeckTransition
diagram	 <p>The diagram illustrates the structure of the NeckTransition class. It is a class with a type NeckTransitionType and a generalization relationship (GS) to the NeckTransitionType class. The NeckTransitionType class has a dashed box labeled NeckTransitionType containing two elements: Tapered (type NeckTaperTransitionType) and RightAngled (type RightAngledNeckTransitionType).</p>
type	ShoulderTransitionType, ShoulderTaperTransitionType, RightAngledShoulderTransitionType.

### 5.14.1.7.5.3.1.1 Tapered

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Neck/NeckTransition/Tapered
diagram	<p>The diagram illustrates the <b>NeckTaperTransitionType</b> structure. It features a central box labeled <b>Tapered</b> with the type <b>NeckTaperTransitionType</b>. This box is connected to a larger dashed box representing the transition type. Inside this dashed box, there is a sub-section labeled <b>NeckTaperAngle</b> with the type <b>xs:integer</b>. Below this, there are eight connection points, each represented by a small box with a plus sign and a minus sign. These connection points are labeled as follows:</p> <ul style="list-style-type: none"> <li><b>End-of-Terminal-to-start-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>End-of-Terminal-to-end-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>PackageEdge-to-start-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>PackageEdge-to-end-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>SeatingPlane-to-start-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>SeatingPlane-to-end-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>Top-of-TerminalExit-from-Package-to-start-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>Top-of-TerminalExit-from-Package-to-end-of-NeckTaperTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> </ul>
type	NeckTaperTransitionType, JEP30-D10:UnspecifiedValueSetType

### 5.14.1.7.5.3.1.2 Right Angled

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Neck/NeckTransition/RightAngled
diagram	<p>The diagram illustrates the <b>RightAngledNeckTransitionType</b> structure. It features a central box labeled <b>RightAngled</b> with the type <b>RightAngledNeckTransitionType</b>. This box is connected to a larger dashed box representing the transition type. Inside this dashed box, there are four connection points, each represented by a small box with a plus sign and a minus sign. These connection points are labeled as follows:</p> <ul style="list-style-type: none"> <li><b>End-of-Terminal-to-NeckTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>PackageEdge-to-NeckTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>SeatingPlane-to-NeckTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> <li><b>Top-of-TerminalExit-from-Package-to-NeckTransition</b> (type: JEP30-D10:UnspecifiedValueSetType)</li> </ul>
type	RightAngledNeckTransitionType, JEP30-D10:UnspecifiedValueSetType

5.14.1.7.5.3.2 Configuration

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Neck/Configuration
diagram	
type	NeckConfigurationType, JEP30-D10:EmptyType, NeckOffsetType, JEP30-D10:UnspecifiedValueSetType

5.14.1.7.5.4      Dambar

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Dambar
diagram	<p>The diagram illustrates the structure of the <b>DambarType</b> and its associated shapes. <b>DambarType</b> is the base class, with <b>DambarProtrusionType</b> and <b>DambarIntrusionType</b> as subclasses. <b>DambarType</b> includes attributes: <b>End-of-Terminal-to-Dambar</b>, <b>PackageEdge-to-Dambar</b>, <b>SeatingPlane-to-Dambar</b>, <b>Top-of-TerminalExit-from-Package-to-Dambar</b>, and <b>DambarLength</b>. <b>DambarProtrusionType</b> includes <b>Protrusion</b> and <b>DambarWidth</b>. <b>DambarIntrusionType</b> includes <b>Intrusion</b> and <b>DambarWidth</b>. All attributes are of type <b>JEP30-D10:UnspecifiedValueSetType</b> or <b>JEP30-D10:UnspecifiedUncontrolledValueSetType</b>.</p>
type	DambarType, DambarProtrusionType,

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions

#### 5.14.1.7.6 Post (Stud) Terminal

Table 9 shows an array of dimensions that are applicable to this terminal.

**Table 9 - Post (Stud) Terminal Dimensions**

Schema Labels	Butt	Flatten Post Connection	Schema Types	Data Requirements
Length		Y	JEP30-D10:DimensionalValueSetType	Mandatory
PackageEdge-to-end-of-Terminal	Y		JEP30-D10:ValueSetType	Mandatory
PackageEdge-to-TerminalKnee	Y		JEP30-D10:DimensionalValueSetType	Optional
Width	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalHeight	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalThickness	Y	Opt	JEP30-D10:UnspecifiedValueSetType	Mandatory
RadiusTop	Y		JEP30-D10:UnspecifiedValueSetType	Optional
RiseAngle	Y		JEP30-D10:UnspecifiedValueSetType	Optional
TerminalPostWidth		Y	JEP30-D10:UnspecifiedValueSetType	Optional
TerminalPostThickness		Y	JEP30-D10:UnspecifiedValueSetType	Optional
TerminalFootThickness		Y	JEP30-D10:UnspecifiedValueSetType	Optional
PlatingThickness	Y	Y	JEP30-D10:UnspecifiedValueSetType	Mandatory

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.7 J-bend

Table 10 shows an array of dimensions that are applicable to this terminal.

**Table 10 - J-bend Dimensions**

Schema Labels	Symbol	Schema Types	Data Requirements
Length	L	JEP30-D10:DimensionalValueSetType	Mandatory – but can be derived by other dimensions provided
PackageEdge-to-end-of-Terminal	L1	JEP30-D10:ValueSetType	Mandatory – but can be derived by other dimensions provided
Width	b	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalHeight	A3	JEP30-D10:UnspecifiedDimensionalValueSetType	Optional
TerminalBendHeight		JEP30-D10:UnspecifiedValueSetType	Optional
TerminalThickness	c	JEP30-D10:UnspecifiedValueSetType	Mandatory
RadiusTop	R1	JEP30-D10:UnspecifiedValueSetType	Optional
RadiusBottom	R2	JEP30-D10:UnspecifiedValueSetType	Optional
RiseAngle	Φ2	JEP30-D10:UnspecifiedValueSetType	Optional
PlatingThickness	p	JEP30-D10:UnspecifiedValueSetType	Mandatory
If the Terminal has a Shoulder, then the following elements is applicable.			
Shoulder		ConfigurableShoulderType	Optional

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.8 L-bend

Table 11 shows an array of dimensions that are applicable to this terminal.

**Table 11 - L-bend Dimensions**

Schema Labels	Inward	Outward	Side Inward	Side Outward	LC-bend	Schema Types
Length	Y	Y	Y	Y	Y	JEP30-D10:DimensionalValueSetType
PackageEdge-to-End-of-Terminal		Y	Opt	Y	Y	JEP30-D10:ValueSetType
Width	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType
Terminal Height	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType
TerminalThickness	Y	Y	Opt	Opt	Y	JEP30-D10:UnspecifiedValueSetType
ExposedTerminalThickness	Y	Y	Y	Y		JEP30-D10:UnspecifiedValueSetType
RadiusBottom	Y	Y	Y	Y		JEP30-D10:UnspecifiedValueSetType
RadiusInsideBottom					Y	JEP30-D10:UnspecifiedValueSetType
RadiusOutsideBottom					Y	JEP30-D10:UnspecifiedValueSetType
SeatingAngle	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
PlatingThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.9 Column Types

Table 12 shows an array of dimensions that are applicable to this terminal:

**Table 12 - Column Type Dimensions**

Schema Labels	Column Types	Applicable Shapes to capture			Schema Types
		Terminal Shape	Soldermask Opening	Pad Shape	
Rectangle	All		NSMD	SMD	ReferenceRectangleType
Rounded Rectangle	All		NSMD	SMD, NSMD	ReferenceRoundedRectangleType
Modified Rectangle	All		NSMD	SMD	ReferenceModifiedRectangleType
Circle	All	SMD, NSMD	SMD, NSMD	SMD, NSMD	ReferenceCircleType
Double-D	All		NSMD	SMD	ReferenceDouble-DType
Contour	All		NSMD	SMD	ReferenceTerminalContourType
Terminal Height	All	SMD, NSMD	n/a	n/a	JEP30-D10: UnspecifiedDimensionalValueSetType
UpperSide-of-Column	Column	Y			UpperSide-of-ColumnShapeType
LowerSide-of-Column	Column	Y			LowerSide-of-ColumnShapeType

All Column variations have only a circular terminal shape, even if mounted on the package via a SMD or a NSMD pad. The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

When *Column* has an no extended terminal code, then the Terminal shape dimensions should be classified for either the *LowerSide-of-Column* and/or the *UpperSide-of-Column*. If both ends of the terminal are different (i.e., Column is tapered), then two entries are required to capture both shape dimensions. The *LowerSide-of-Column* terminal shape is mandatory, whereas the *UpperSide-of-Column* terminal shape dimensions are optional. If the column is not tapered, then both of these flags are set for the same dimension captured.

When the *Column* has an extended terminal code to denote *Ribbon Wrap*, then the element *RibbonThickness*, *RibbonHeight*, *RibbonPitch* are applicable. These elements have an *UnspecifiedValueSetType* and are optional. The *Number-of-Turns* in the helix can also be captured in addition to the *Direction-of-Helix* as in *LeftSpiral* or *RightSpiral*.

When the *Column* has an extended terminal code to denote *Microspring*, then the element *SpringConstant*, *WireDiameter* and *SpringLengthFreeLoad* are applicable. The *WireDiameter* element has an *UnspecifiedValueSetType* and is optional. The *SpringLengthFreeLoad* element has an *DimensionalValueSetType* and is mandatory. When the *Column* has an extended terminal code to denote *Copper Coated Solder Column*, then the element *CopperThickness* is applicable. This has an *UnspecifiedValueSetType* and is optional. An additional entry is required to capture the *Core* shape dimensions.

The preferred method of capturing Column dimensions is via the *ColumnTerminalShapeType* as shown in diagram 1.4 of 9, coupled with diagrams 1, 1.1, and 1.2 similar to Ball terminals.



#### 5.14.1.7.10 Surface-terminal

Table 13 shows an array of dimensions that are applicable to this terminal.

**Table 13 - Surface-terminal Dimensions**

Schema Labels	Surface-terminal	Castellated	Hole	With-opening	Open-Ring	Schema Types
One of the following 2D shapes is applicable to this Terminal type						
Rectangle	Y	Y	Y	Y	Y	ReferenceRectangleType
RoundedRectangle	Y		Y	Y		ReferenceRoundedRectangleType
ModifiedRectangle	Y	Y	Y	Y	Y	ReferenceModifiedRectangleType
Circle	Y		Y	Y		ReferenceCircleType
D-Shape	Y	Y				ReferenceD-ShapeType
Double-D	Y		Y	Y		ReferenceDouble-DType
MickeyMouseEars			Y			ReferenceTerminalContourType
RegularPolygon	Y		Y	Y		RegularPolygonType
Segment	Y					SegmentType
Para-truncatedCircle	Y		Y	Y		
Contour	Y	Y	Y	Y	Y	ReferenceTerminalContourType
The elements below are mandatory if applicable						
ExposedTerminalThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
Terminal Height	Y	Y			Y	JEP30-D10:UnspecifiedDimensionalValueSetType
TerminalVoid-Array			Y	Y		TerminalVoid-ArrayType
The elements below are optional if applicable						
CircularCastellation		Y			Y	CircularCastellationType
TriangularCastellation		Y			Y	TriangularCastellationType
ChamferredRectangleCastellation		Y			Y	ChamferredRectangleCastellationType
RoundedRectangleCastellation		Y			Y	RoundedRectangleCastellationType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.11 Pin

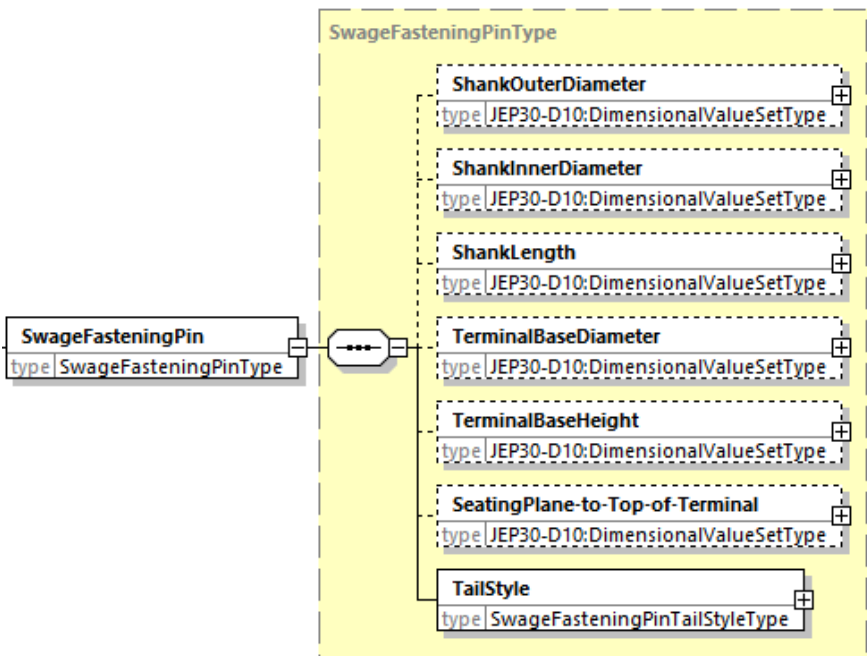
Table 14. shows an array of dimensions that are applicable to this terminal.

**Table 14 - Pin Dimensions**

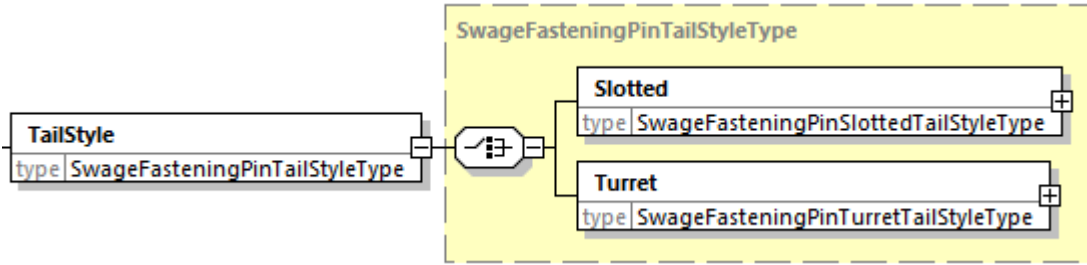
Schema Labels	Pin	Kinked	Shoulder	Press-in Solderable	Surface-mount	Swage Fastening Pin	Press-in Non-Solderable	Schema Types
TerminalInsertionHeight	Y	Y	Y	Y			Y	JEP30-D10:UnspecifiedValueSetType
Diameter	Y	Y	Y	Y	Y		Y	JEP30-D10: DimensionalValueSetType
Width	Y	Y	Y	Y	Y		Y	JEP30-D10:UnspecifiedDimensionalValueSetType
TerminalThickness	Y	Y	Y	Y	Y		Y	JEP30-D10:UnspecifiedValueSetType
TerminalHeight	Y	Y	Y	Y	Y		Y	JEP30-D10:UnspecifiedDimensionalValueSetType
PlatingThickness	Y	Y	Y	Y	Y			JEP30-D10:UnspecifiedValueSetType
Contour	Y	Y	Y	Y	Y		Y	ReferenceTerminalContourType
Kinked		Y						KinkedType
Shoulder			Y	Y			Y	ConfigurableShoulderType
SwageFasteningPin						Y		SwageFasteningPinType
TerminalEndShape	Y	Y	Y	Y	Y		Y	TerminalEndShapeType
TerminalPostWidth					Y			JEP30-D10:UnspecifiedValueSetType
TerminalPostThickness					Y			JEP30-D10:UnspecifiedValueSetType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.11.1 Swage Fastening Pin

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/SwageFasteningPin
diagram	 <p>The diagram shows a class hierarchy for <b>SwageFasteningPinType</b>. A box labeled <b>SwageFasteningPin</b> with <code>type SwageFasteningPinType</code> is connected to a larger box labeled <b>SwageFasteningPinType</b>. Inside <b>SwageFasteningPinType</b>, there are several attributes, each in a dashed box with a '+' icon in the top right corner: <b>ShankOuterDiameter</b> (type JEP30-D10:DimensionalValueSetType), <b>ShankInnerDiameter</b> (type JEP30-D10:DimensionalValueSetType), <b>ShankLength</b> (type JEP30-D10:DimensionalValueSetType), <b>TerminalBaseDiameter</b> (type JEP30-D10:DimensionalValueSetType), <b>TerminalBaseHeight</b> (type JEP30-D10:DimensionalValueSetType), <b>SeatingPlane-to-Top-of-Terminal</b> (type JEP30-D10:DimensionalValueSetType), and <b>TailStyle</b> (type SwageFasteningPinTailStyleType).</p>
type	SwageFasteningPinType, JEP30-D10:DimensionalValueSetType, SwageFasteningPinTailStyleType,

#### 5.14.1.7.11.1.1 Tail Style

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/SwageFasteningPin/TailStyle
diagram	 <p>The diagram shows a class hierarchy for <b>SwageFasteningPinTailStyleType</b>. A box labeled <b>TailStyle</b> with <code>type SwageFasteningPinTailStyleType</code> is connected to a larger box labeled <b>SwageFasteningPinTailStyleType</b>. Inside <b>SwageFasteningPinTailStyleType</b>, there are two attributes, each in a dashed box with a '+' icon in the top right corner: <b>Slotted</b> (type SwageFasteningPinSlottedTailStyleType) and <b>Turret</b> (type SwageFasteningPinTurretTailStyleType).</p>
type	SwageFasteningPinTailStyleType, SwageFasteningPinSlottedTailStyleType, SwageFasteningPinTurretTailStyleType

#### 5.14.1.7.11.1.1 Slotted

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/SwageFasteningPin/TailStyle/Slotted</a>
diagram	
type	<a href="#">SwageFasteningPinSlottedTailStyleType</a> , <a href="#">JEP30-D10:DimensionalValueSetType</a> .

#### 5.14.1.7.11.1.2 Turret

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/SwageFasteningPin/TailStyle/Turret</a>
diagram	
type	<a href="#">SwageFasteningPinTurretTailStyleType</a> , <a href="#">JEP30-D10:DimensionalValueSetType</a> .

#### 5.14.1.7.12 Wraparound

Table 15 shows an array of dimensions that are applicable to this terminal.

**Table 15 - Wraparound Dimensions**

Schema Labels	Wraparound	Castellated	Ring	Open-Ring	Nibble	Convex-E	Convex-S	Schema Types
One of the following 2D shapes is applicable to this Terminal type								
Rectangle	Y	Y	Y	Y	Y	Y	Y	ReferenceRectangleType
ModifiedRectangle	Y	Y	Y	Y	Y		Y	ReferenceModifiedRectangleType
Contour	Y	Y	Y	Y	Y		Y	ReferenceTerminalContourType
The elements below are mandatory if applicable								
ExposedTerminalThickness	Y	Y	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
Terminal Height	Y	Y	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType
The elements below are optional if applicable								
CircularCastellation		Y	Y	Y				CircularCastellationType
TriangularCastellation		Y	Y	Y				TriangularCastellationType
ChamferedRectangleCastellation		Y	Y	Y				ChamferedRectangleCastellationType
RoundedRectangleCastellation		Y	Y	Y				RoundedRectangleCastellationType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.13 S-bend Terminal

Table 16 shows an array of dimensions that are applicable to this terminal.

**Table 16 - S-bend Dimensions**

Schema Labels	S-bend	Inward Flat	Inward Curved	Outward Flat	Outward Curved	Schema Types
Length	Y	Y	Y	Y	Y	JEP30-D10:DimensionalValueSetType
PackageEdge-to-End-of-Terminal	Y			Y	Y	JEP30-D10:ValueSetType
Width	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType
Terminal Height	Opt	Opt	Opt	Opt	Opt	JEP30-D10:UnspecifiedDimensionalValueSetType
TerminalBendHeight			Opt		Opt	JEP30-D10:UnspecifiedValueSetType
TerminalThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
RadiusTop	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
RadiusBottom	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
PlatingThickness	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.14 Through - Hole

Table 17 — Through - Hole Dimensions shows an array of dimensions that are applicable to this terminal.

**Table 17 - Through - Hole Dimensions**

Schema Labels	Through - Hole	Kinked	Shoulder	Schema Types
PackageEdge-to-TerminalKnee	Y	Y	Y	JEP30-D10: DimensionalValueSetType
Diameter	Y	Y	Y	JEP30-D10: DimensionalValueSetType
Width	Y	Y	Y	JEP30-D10: UnspecifiedDimensionalValueSetType
TerminalThickness	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
TerminalHeight	Y	Y	Y	JEP30-D10: UnspecifiedDimensionalValueSetType
TerminalInsertionHeight	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
Un-preppedTerminalLength	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
RadiusTop	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
RiseAngle	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
PlatingThickness	Y	Y	Y	JEP30-D10: UnspecifiedValueSetType
TerminalEndShape	Y	Y	Y	TerminalEndShapeType
Contour	Y	Y	Y	ReferenceTerminalContourType
Kinked		Y		KinkedType
Shoulder			Y	ConfigurableShoulderType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions

#### 5.14.1.7.15 J-inverted

Table 18 shows an array of dimensions that are applicable to this terminal.

**Table 18 - J-inverted Dimensions**

Schema Labels	Symbol	Schema Types	Data Requirements
Length	L	JEP30-D10:DimensionalValueSetType	Mandatory – but can be derived by other dimensions provided
PackageEdge-to-end-of-Terminal	L1	JEP30-D10:ValueSetType	Mandatory – but can be derived by other dimensions provided
PackageEdge-to-TerminalKnee		JEP30-D10:DimensionalValueSetType	Optional
Width	b	JEP30-D10:UnspecifiedDimensionalValueSetType	Mandatory
TerminalHeight	A3	JEP30-D10:UnspecifiedDimensionalValueSetType	Optional
TerminalBendHeight		JEP30-D10:UnspecifiedValueSetType	Optional
TerminalThickness	c	JEP30-D10:UnspecifiedValueSetType	Mandatory
RadiusTop	R1	JEP30-D10:UnspecifiedValueSetType	Optional
RadiusBottom	R2	JEP30-D10:UnspecifiedValueSetType	Optional
RiseAngle	Φ2	JEP30-D10:UnspecifiedValueSetType	Optional
PlatingThickness	p	JEP30-D10:UnspecifiedValueSetType	Mandatory
If the Terminal has a Shoulder, then the following elements is applicable.			
Shoulder		ConfigurableShoulderType	Optional

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.



#### 5.14.1.7.16 Terminal Wire

Table 19 shows an array of dimensions that are applicable to this terminal.

**Table 19 - Terminal Wire Dimensions**

Schema Labels	Terminal Wire	Kinked (Bent)	SM-Wire	SM-Coined Wire	Wraparound SMT Mount	Coined Wraparound SMT Mount	Schema Types
TerminalInsertionHeight	Y	Y	Y	Y			JEP30-D10:UnspecifiedValueSetType
Diameter	Y	Y	Y	Y			JEP30-D10: DimensionalValueSetType
WireGuage	Y	Y	Y	Y	Y	Y	WireGuageType
TerminalHeight	Y	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedDimensionalValueSetType
TerminalThickness	Y	Y	Y	Y	Y	Y	JEP30-D10:UnspecifiedValueSetType
PlatingThickness	Y	Y	Y	Y			JEP30-D10:UnspecifiedValueSetType
Kinked		Y					KinkedType
Un-preppedTerminalLength	Y						JEP30-D10:UnspecifiedValueSetType
Contour	Y	Y	Y	Y	Y	Y	ReferenceTerminalContourType
Rectangle	Y	Y	Y	Y	Y	Y	ReferenceRectangleType
WireEndShape	Y	Y	Y	Y			WireEndShapeType
TerminalEndShape	Y	Y	Y	Y			TerminalEndShapeType

*WireGuage* is a numerical integer value between 10 and 40. The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

#### 5.14.1.7.17 Screw

Table 20 shows an array of dimensions that are applicable to this terminal.

**Table 20 - Screw Dimensions**

Schema Labels	Clearance Hole	Threaded Hole	Screw	Schema Types
Rectangle	Y			ReferenceRectangleType
RoundedRectangle	Y			ReferenceRoundedRectangleType
ModifiedRectangle	Y			ReferenceModifiedRectangleType
Circle	Y	Y		ReferenceCircleType
Double-D	Y			ReferenceDouble-DType
ScrewHeadType			Y	Xs: string
ScrewHeadHeight			Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadLength			Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadPitch		Y	Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadDepth		Y	Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadMajorDiameter		Y	Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadMinorDiameter		Y	Y	JEP30-D10:UnspecifiedValueSetType
ScrewThreadPitchDiameter		Y	Y	JEP30-D10:UnspecifiedValueSetType

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

5.14.1.7.18 Shoulder

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder
diagram	
type	ConfigurableShoulderType, JEP30-D10:UnspecifiedDimensionalValueSetType, ShoulderTransitionType, ShoulderConfigurationType, JEP30-D10:EmptyType.

5.14.1.7.18.1 Shoulder Transition

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder/ShoulderTransition
diagram	
type	ShoulderTransitionType, ShoulderTaperTransitionType, RightAngledShoulderTransitionType.

### 5.14.1.7.18.1.1 Tapered

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder/ShoulderTransition/Tapered
diagram	
type	ShoulderTaperTransitionType, JEP30-D10:UnspecifiedValueSetType

### 5.14.1.7.18.1.2 Right Angled

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder/ShoulderTransition/RightAngled
diagram	
type	RightAngledShoulderTransitionType, JEP30-D10:UnspecifiedValueSetType

### 5.14.1.7.18.2 Configuration

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder/Configuration</a>
diagram	
type	<a href="#">ShoulderConfigurationType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">ShoulderOffsetType</a> , <a href="#">JEP30-D10:UnspecifiedValueSetType</a>

### 5.14.1.7.18.3 TH - Shoulder Cutout

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Shoulder/TH-ShoulderCutout</a>
diagram	
type	<a href="#">TH-ShoulderCutoutType</a> , <a href="#">CutoutConfigurationType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">JEP30-D10:UnspecifiedValueSetType</a> .

#### 5.14.1.7.19 Kinked

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Kinked</a>
diagram	
type	<a href="#">KinkedType</a> , <a href="#">JEP30-D10:DimensionalValueSetType</a> ,

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions

#### 5.14.1.7.20 Terminal End Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalEndShape</a>
diagram	
type	<a href="#">TerminalEndShapeType</a> , <a href="#">TaperedTerminalEndShapeType</a> , <a href="#">JEP30-D10:UnspecifiedValueSetType</a> , <a href="#">JEP30-D10:EmptyType</a> .

#### 5.14.1.7.21 Terminal Shape versus Dimensions

Table 21 — Terminal Shape versus Dimensions shows an array of dimensions that are captured below each shape listed above:

**Table 21 - Terminal Shape versus Dimensions**

Terminal Shape	Dimension 1	Dimension 2	Dimension 3	Diameter	Radius	Angle	No. of Sides	Inner/Outer	Shoulder	Impacted Corner
Rectangle	Y	Y				Y				
Rectangle with Shoulder	Y	Y				Y			Y	
Rounded Rectangle	Y	Y			Y	Y				
Modified Rectangle	Y	Y				Y				Y
Modified Rectangle with Shoulder	Y	Y				Y			Y	Y
Circle				Y						
Circle with Shoulder				Y					Y	
D-Shape	Y	Y				Y				
Double-D	Y	Y				Y				
Regular Polygon				Y		Y	Y	Y		
Segment	Y			Y		Y				
Para-truncated Circle	Y			Y		Y				
Concave	Y	Y			Y					
Concave D-Shape	Y	Y			Y					
Isosceles Trapezoid	Y	Y	Y			Y				
Contour						Y				

The definition of each shape is outlined in Annex A (informative) Shape Dimensions.

### 5.14.1.7.22 Castellation - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array
diagram	<pre> classDiagram     class CastellationArrayType {         +CastellationType 1..∞     }     class CastellationType {         +CastellationLocationType         +CastellationShapeType     }     class CastellationLocationType {         +StandardArray         +CircularArray         +Status         +RandomArray         +VerticalCastellationLocationOnTerminal     }     class CastellationShapeType {         +VerticalCastellation         +HorizontalCastellation     }     class CastellationLocation {         type CastellationLocationType     }     class CastellationShape {         type CastellationShapeType     }     class Castellation {         type CastellationType     }     class StandardArray {         type CastellationStandardArrayType     }     class CircularArray {         type CastellationCircularArrayType     }     class Status {         type DeletedStatusType     }     class RandomArray {         type CastellationRandomArrayType     }     class VerticalCastellationLocationOnTerminal {         type VerticalCastellationLocationOnTerminalType     }     class VerticalCastellation {         type CircularCastellation         type ChamferedRectangleCastellation         type RoundedRectangleCastellation         type TriangularCastellation     }     class HorizontalCastellation {         type StepCut         type Dimple     }     class CircularCastellation {         type CircularCastellationType     }     class ChamferedRectangleCastellation {         type ChamferedRectangleCastellationType     }     class RoundedRectangleCastellation {         type RoundedRectangleCastellationType     }     class TriangularCastellation {         type TriangularCastellationType     }     class StepCut {         type HorizontalStepCutType     }     class Dimple {         type HorizontalCastellationDimpleType     }      CastellationArrayType --&gt; CastellationType     CastellationType --&gt; CastellationLocationType     CastellationType --&gt; CastellationShapeType     CastellationLocationType --&gt; StandardArray     CastellationLocationType --&gt; CircularArray     CastellationLocationType --&gt; Status     CastellationLocationType --&gt; RandomArray     CastellationLocationType --&gt; VerticalCastellationLocationOnTerminal     CastellationShapeType --&gt; VerticalCastellation     CastellationShapeType --&gt; HorizontalCastellation     VerticalCastellation --&gt; CircularCastellation     VerticalCastellation --&gt; ChamferedRectangleCastellation     VerticalCastellation --&gt; RoundedRectangleCastellation     VerticalCastellation --&gt; TriangularCastellation     HorizontalCastellation --&gt; StepCut     HorizontalCastellation --&gt; Dimple     </pre>
type	Castellation-ArrayType, CastellationType, CastellationLocationType, CastellationStandardArrayType, CastellationCircularArrayType, DeletedStatusType, CastellationRandomArrayType, VerticalCastellationLocationOnTerminalType, CastellationShapeType, TriangularCastellationType, ChamferedRectangleCastellationType, RoundedRectangleCastellationType, CircularCastellationType, HorizontalStepCutType, HorizontalCastellationDimpleType.
group	VerticalCastellation, HorizontalCastellation

All *CastellationShapes* are defined in Annex A.1



5.14.1.7.22.1     Standard - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/StandardArray
diagram	
type	CastellationStandardArrayType, JEP30-D10:PitchValueSetType, JEP30-D10:PointXYType.

5.14.1.7.22.2     Circular - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/CircularArray
diagram	
type	CastellationCircularArrayType, JEP30-D10:PitchValueSetType, JEP30-D10:PointXYType.

5.14.1.7.22.3 Status

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/Status
diagram	
type	DeletedStatusType, JEP30-D10:PointXYType, JEP30-D10:MinIntegerOfOneType, JEP30-D10:EmptyType.

5.14.1.7.22.4 Random - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/RandomArray
diagram	
type	CastellationRandomArrayType, JEP30-D10:PointXYType.

5.14.1.7.22.5    Vertical Castellation Location On Terminal

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/VerticalCastellationLocationOnTerminal
diagram	
type	VerticalCastellationLocationOnTerminalType, JEP30-D10:EmptyType, CastellationToTerminal-X-OffsetType, CastellationToTerminal-X-OffsetType, VerticalCastellationCornerLocationsOnTerminalType.

### 5.14.1.7.22.5.1 Castellation to Terminal X - Offset



### 5.14.1.7.22.5.1 Castellations to Terminal X - Offset (cont'd)


diagram part 2 of 2	 <p><b>X-Offset</b></p> <p>type   CastellationsToTerminal-X-OffsetType</p> <p>X14 - CenterOfTerminal-to-CenterlineOfCastellationGroup type   JEP30-D10:ValueSetGroupType</p> <p>X15 - CenterOfTerminal-to-RightSideOfCenterCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X16 - CenterOfTerminal-to-LeftSideOfRightCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X17 - CenterOfTerminal-to-CenterOfRightCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X18 - CenterOfTerminal-to-RightSideOfRightCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X19 - RightSideOfTerminal-to-LeftSideOfLeftCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X20 - RightSideOfTerminal-to-CenterOfLeftCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X21 - RightSideOfTerminal-to-RightSideOfLeftCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X22 - RightSideOfTerminal-to-LeftSideOfCenterCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X23 - RightSideOfTerminal-to-CenterlineOfCastellationGroup type   JEP30-D10:ValueSetGroupType</p> <p>X24 - RightSideOfTerminal-to-RightSideOfCenterCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X25 - RightSideOfTerminal-to-LeftSideOfRightCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X26 - RightSideOfTerminal-to-CenterOfRightCastellation type   JEP30-D10:ValueSetGroupType</p> <p>X27 - RightSideOfTerminal-to-RightSideOfRightCastellation type   JEP30-D10:ValueSetGroupType</p>
type	CastellationsToTerminal-X-OffsetType, JEP30-D10:ValueSetGroupType

Table 22 - Castellations to Terminal X - Offset lists out all the various dimensions that can be selected from any major point of the terminal to any major point of the Castellations in X direction.

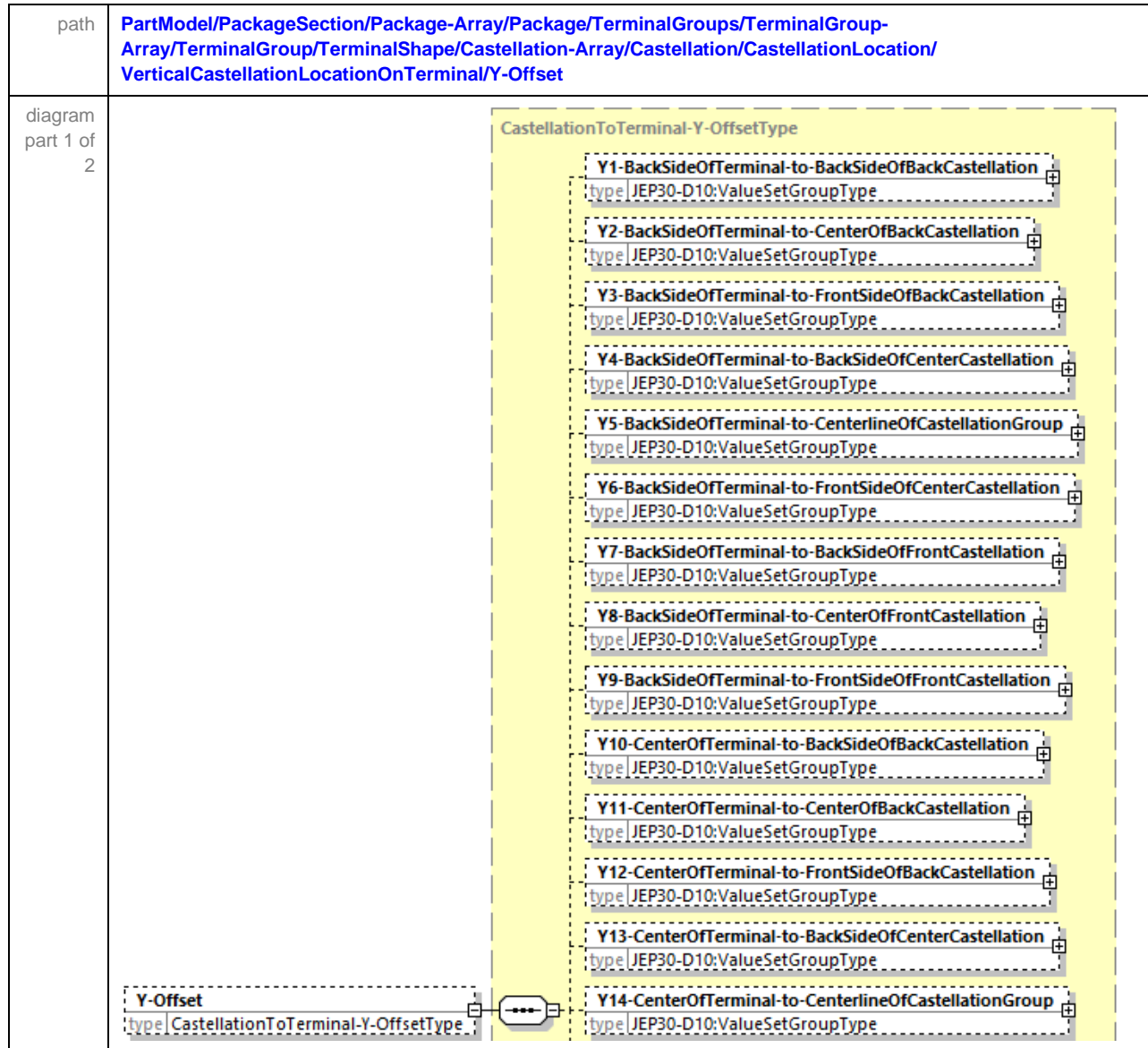
Center of Center Castellations is the same as Centerline of Castellations Group, however the use of the "...Centerline of Castellations Group" is more encompassing since it also includes the centerline of a row or column of an even number of Castellations, i.e., midway between castellations 2 and 3 of 4 castellations in a row or column.

#### 5.14.1.7.22.5.1 Castellations to Terminal X - Offset (cont'd)

**Table 22 - Castellations to Terminal X - Offset**

Symbol	Description
X1	Left side of Terminal to Left Side of Left Castellations
X2	Left side of Terminal to Center of Left Castellations
X3	Left side of Terminal to Right side of Left Castellations
X4	Left side of Terminal to Left Side of Center Castellations
X5	Left side of Terminal to Centerline of Castellations Group
X6	Left side of Terminal to Right side of Center Castellations
X7	Left side of Terminal to Left side of Right Castellations
X8	Left side of Terminal to Center of Right Castellations
X9	Left side of Terminal to Right side of Right Castellations
X10	Center of Terminal to Left side of Left Castellations
X11	Center of Terminal to Center of Left Castellations
X12	Center of Terminal to Right side of Left Castellations
X13	Center of Terminal to Left side of Center Castellations
X14	Center of Terminal to Centerline of Castellations Group
X15	Center of Terminal to Right side of Center Castellations
X16	Center of Terminal to Left side of Right Castellations
X17	Center of Terminal to Center of Right Castellations
X18	Center of Terminal to Right side of Right Castellations
X19	Right side of Terminal to Left side of Left Castellations
X20	Right side of Terminal to Center of Left Castellations
X21	Right side of Terminal to Right side of Left Castellations
X22	Right side of Terminal to Left side of Center Castellations
X23	Right side of Terminal to Centerline of Castellations Group
X24	Right side of Terminal to Right side of Center Castellations
X25	Right side of Terminal to Left side of Right Castellations
X26	Right side of Terminal to Center of Right Castellations
X27	Right side of Terminal to Right side of Right Castellations

### 5.14.1.7.22.5.2 Castellation to Terminal Y - Offset



### 5.14.1.7.22.5.2 Castellations to Terminal Y - Offset (cont'd)

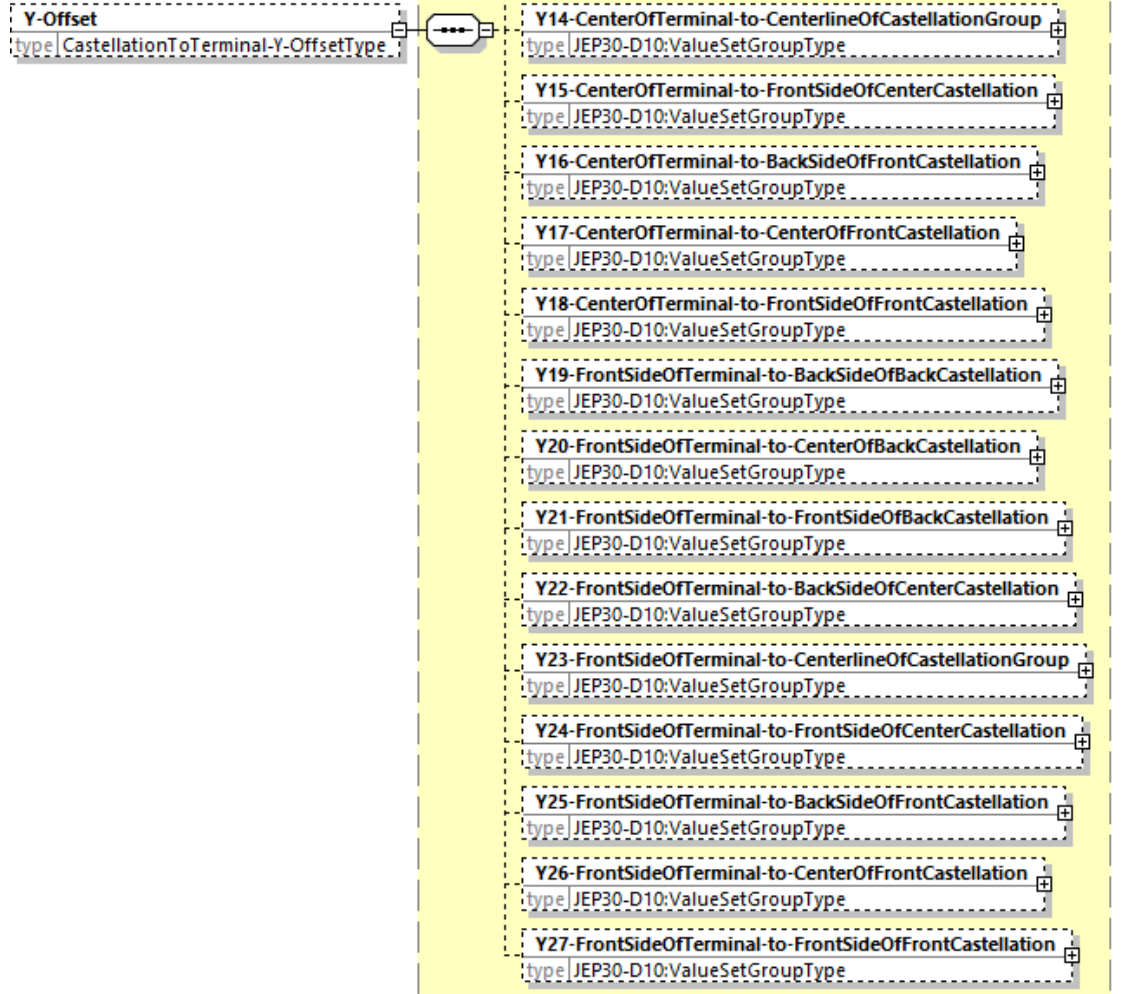
<p>diagram part 2 of 2</p>	
<p>type</p>	<p>CastellationToTerminal-Y-OffsetType, JEP30-D10:ValueSetGroupType</p>

Table 23 - Castellations to Terminal Y - Offset lists out all the various dimensions that can be selected from any major point of the terminal to any major point of the Castellations in Y direction.



5.14.1.7.22.5.2

Castellation to Terminal Y - Offset (cont'd)

Table 23 - Castellation to Terminal Y - Offset

Symbol	Description
Y1	Back side of Terminal to Back side of Back Castellation
Y2	Back side of Terminal to Center of Back Castellation
Y3	Back side of Terminal to Front side of Back Castellation
Y4	Back side of Terminal to Back side of Center Castellation
Y5	Back side of Terminal to Centerline of Castellation Group
Y6	Back side of Terminal to Front side of Center Castellation
Y7	Back side of Terminal to Back side of Front Castellation
Y8	Back side of Terminal to Center of Front Castellation
Y9	Back side of Terminal to Front side of Front Castellation
Y10	Center of Terminal to Back side of Back Castellation
Y11	Center of Terminal to Center of Back Castellation
Y12	Center of Terminal to Front side of Back Castellation
Y13	Center of Terminal to Back side of Center Castellation
Y14	Center of Terminal to Centerline of Castellation Group
Y15	Center of Terminal to Front side of Center Castellation
Y16	Center of Terminal to Back side of Front Castellation
Y17	Center of Terminal to Center of Front Castellation
Y18	Center of Terminal to Front side of Front Castellation
Y19	Front side of Terminal to Back side of Back Castellation
Y20	Front side of Terminal to Center of Back Castellation
Y21	Front side of Terminal to Front side of Back Castellation
Y22	Front side of Terminal to Back side of Center Castellation
Y23	Front side of Terminal to Centerline of Castellation Group
Y24	Front side of Terminal to Front side of Center Castellation
Y25	Front side of Terminal to Back side of Front Castellation
Y26	Front side of Terminal to Center of Front Castellation
Y27	Front side of Terminal to Front side of Front Castellation

### 5.14.1.7.22.5.3 Corner

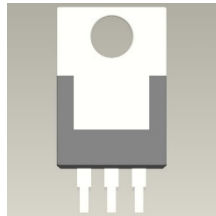
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/Castellation-Array/Castellation/CastellationLocation/VerticalCastellationLocationOnTerminal/Corner
diagram	
type	VerticalCastellationCornerLocationsOnTerminalType, JEP30-D10:EmptyType

### 5.14.1.7.23 Terminal Void - Array

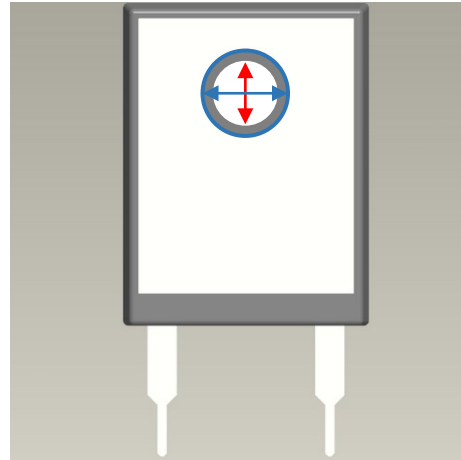
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalVoid-Array.
diagram	
type	TerminalVoid-ArrayType, TerminalVoidType, VoidlocationType, VoidStandardArrauType, VoidCircularArrayType, VoidStatusType, VoidRandomArrayType, TerminalVoidShapeType, RoundedRectangleType, CircleType, Double-DType, ContourShapeType.

### 5.13.1.7.23 Terminal Void – Array (cont'd)

Some terminals such as Flat terminals and Surface terminals may have voids within the Terminal Contact Area, as shown in JESD30, “Illustrations of terminal shape” table. The example in Figure 4 is a Flat terminal with Hole, whereas the example in Figure 3 is a Surface terminal With-opening, plus a 2<sup>nd</sup> terminal called a Screw - Clearance Hole.



**Figure 4 - Flat Terminal with Hole**



**Figure 3 - Surface terminal With-Opening, plus Screw-Clearance Hole**

When capturing the details of the Surface terminal With-opening, the dimensions associated with the larger diameter (as denoted by the horizontal arrow) is used to define the diameter of the circular opening within the Surface terminal. When the 2<sup>nd</sup> terminal group is defined to capture the definition of the Screw - Clearance Hole, the smaller diameter (as denoted by the vertical arrow) is used to define the diameter of the hole. The dimension then as calculated by the (larger diameter minus the smaller diameter) / 2 = the clearance between the edge of the hole and the metallization of the Surface terminal.

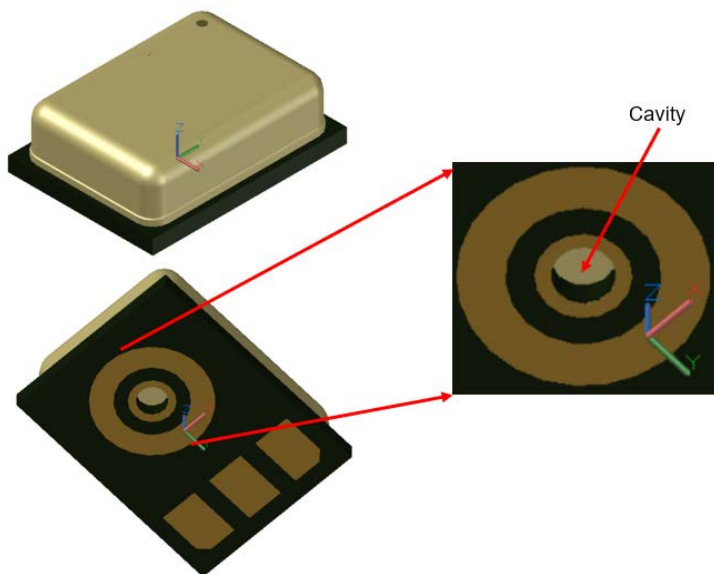


Figure 5 shows a package where a cavity exists within the inner terminal void. This is not a hole that goes through the part like a flange mount device as can be seen from the lack of a hole exiting from the top of the package. In this case, the depth of the cavity can be captured via the element called *CavityDepth*.

**Figure 5 – Surface terminal With-Opening and Cavity**

Since there may be more than one void within the Terminal shape shown in Figure 6, and since these voids may be different in shape (e.g., A Circle and a Double-D shape holes), the following sections outlines the array location of the voids and the different shapes that can be defined.

5.13.1.7.23    Terminal Void – Array (cont’d)



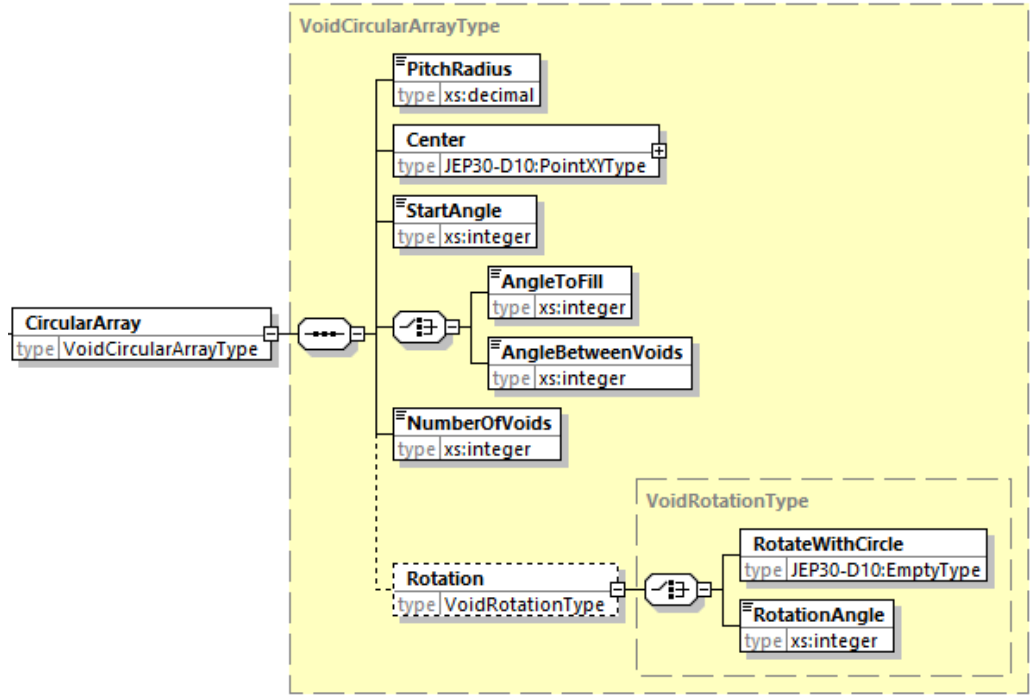
Figure 6 - Flat terminal with different shape Voids

The definition of each shape is outlined in Annex A (informative) Shape Dimensions.

5.14.1.7.23.1    Standard Array


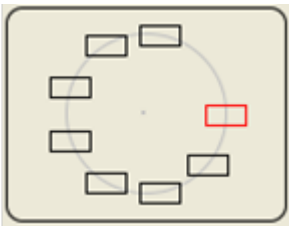
path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalVoid-Array/TerminalVoid/TerminalVoidLocation/StandardArray.
diagram	<p>The diagram illustrates the relationship between the <b>StandardArray</b> class and the <b>VoidStandardArrayType</b> class. The <b>StandardArray</b> class is associated with the <b>VoidStandardArrayType</b> class via a dashed line with a solid arrow. The <b>VoidStandardArrayType</b> class is a complex type with several attributes: <b>ny</b> (type: xs:integer), <b>nx</b> (type: xs:integer), <b>dx</b> (type: JEP30-D10:PitchValueSetType), <b>dy</b> (type: JEP30-D10:PitchValueSetType), <b>Angle</b> (type: xs:integer), and <b>VoidGroupLowerLeftVoidCenter</b> (type: JEP30-D10:PointXYType).</p>
type	VoidStandardArrayType, JEP30-D10:PitchValueSetType, JEP30-D10:PointType.

5.14.1.7.23.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalVoid-Array/TerminalVoid/TerminalVoidLocation/CircularArray.
diagram	 <p>The diagram illustrates the structure of the <b>CircularArray</b> type. It is defined as a <b>VoidCircularArrayType</b>. The structure includes the following elements:</p> <ul style="list-style-type: none"><li><b>PitchRadius</b>: type <code>xs:decimal</code></li><li><b>Center</b>: type <code>JEP30-D10:PointXYType</code></li><li><b>StartAngle</b>: type <code>xs:integer</code></li><li><b>AngleToFill</b>: type <code>xs:integer</code></li><li><b>AngleBetweenVoids</b>: type <code>xs:integer</code></li><li><b>NumberOfVoids</b>: type <code>xs:integer</code></li><li><b>Rotation</b>: type <code>VoidRotationType</code></li></ul> <p>The <b>Rotation</b> element is further detailed within a <b>VoidRotationType</b> container, which includes:</p> <ul style="list-style-type: none"><li><b>RotateWithCircle</b>: type <code>JEP30-D10:EmptyType</code></li><li><b>RotationAngle</b>: type <code>xs:integer</code></li></ul>
type	VoidCircularArrayType, JEP30-D10:PointType, VoidRotationType.

### 5.13.1.6.22.2 Circular Array (cont'd)

**Table 24 - Void Circular Array Elements Definition**

Element	Explanation
Pitch Radius	The radius of the circle of voids.
Center	The center of the circle of voids.
Start Angle	The angle at which to place the first void in the array. By default, the void is at the right-most point on the circle (i.e. the number 3 position on a clock face). This angle specifies a rotation clockwise around the circle from this point.
Angle to Fill	The angle through which the voids are distributed, starting from the first void and continuing clockwise, until the last void is reached.
Angle between voids	Instead of specifying Angle to fill, you can specify the angle between the centers of each void.
Number of voids	The number of voids to be arrayed around the circle. The first void is displayed in red. This number includes any deleted voids in the array.
Rotate with Circle	 <p>If the voids are rotated so that they are oriented perpendicular to the circle. This does not apply to circular or contour void shapes.</p>
Rotation Angle	 <p>If the voids are not oriented perpendicular to the circle, then the voids could be rotated around the center of the voids itself. In this image, each void has a "0" degree rotation with respect to the "3 O'clock position". This does not apply to circular or contour void shapes.</p>

5.14.1.7.23.3 Void Status

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalVoid-Array/TerminalVoid/TerminalVoidLocation/VoidStatus.
diagram	<p>The diagram illustrates the structure of the <b>VoidStatusType</b> within a dashed yellow boundary. It features a central <b>VoidStatus</b> entity (type: VoidStatusType) connected to several branches. The top branch leads to <b>VoidCenter</b> (type: JEP30-D10:PointXYType) with a multiplicity of 1..∞. The middle branch leads to a choice of three index types: <b>RowTerminalIndex</b>, <b>ColumnTerminalIndex</b>, and <b>PolarTerminalIndex</b>, each with a multiplicity of 1..∞. Each index type is further expanded into a choice of <b>From</b> and <b>To</b> sub-indices. The bottom branch leads to a <b>Deleted</b> entity (type: JEP30-D10:EmptyType).</p>
type	VoidStatusType, JEP30-D10:PointType, JEP30-D10:MinIntegerOfOneType.

For *TerminalVoidLocation* defined via *StandardArray* or *CircularArray*, some voids in the array may be deleted. This is covered in the *VoidStatus* branch.

#### 5.14.1.7.23.4 Random Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/TerminalVoid-Array/TerminalVoid/TerminalVoidLocation/RandomArray.</a>
diagram	
type	<a href="#">VoidRandomArrayType</a> , <a href="#">JEP30-D10:PointXYType</a> .

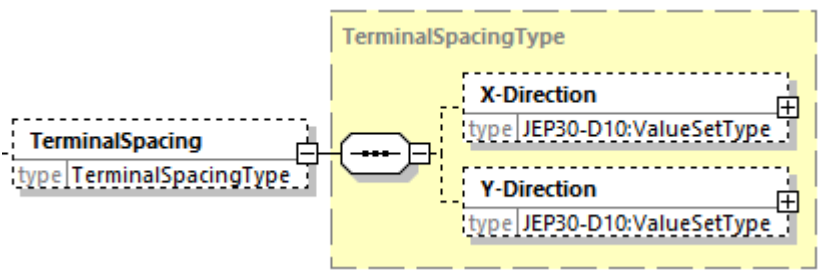
[TerminalVoidLocation](#) can also be defined via [RandomArray](#) when there is just 1 void, or when there is no logical structure to the location of the voids.

#### 5.14.1.8 Terminal Span

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalSpan.</a>
diagram	
type	<a href="#">TerminalSpanType</a> , <a href="#">JEP30-D10:ValueSetType</a> .

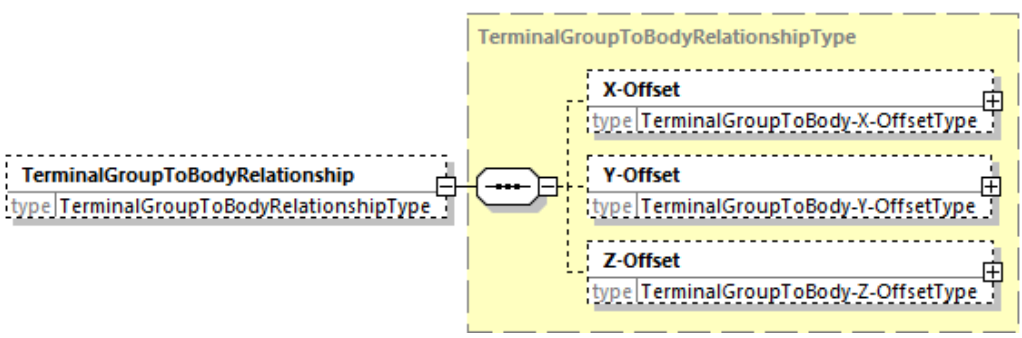


### 5.14.1.9 Terminal Spacing

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalSpacing.</a>
diagram	 <p>The diagram illustrates the structure of the <b>TerminalSpacingType</b>. It shows a dashed box labeled <b>TerminalSpacing</b> with the attribute <code>type TerminalSpacingType</code>. This box is connected to a central octagonal symbol, which is then connected to a larger dashed box labeled <b>TerminalSpacingType</b>. Inside this larger box, there are two sub-sections: <b>X-Direction</b> and <b>Y-Direction</b>. Each sub-section contains the attribute <code>type JEP30-D10:ValueSetType</code> and a plus sign (+) in a small box, indicating a required or optional attribute.</p>
type	<a href="#">TerminalSpacingType</a> , <a href="#">JEP30-D10:ValueSetType</a> .

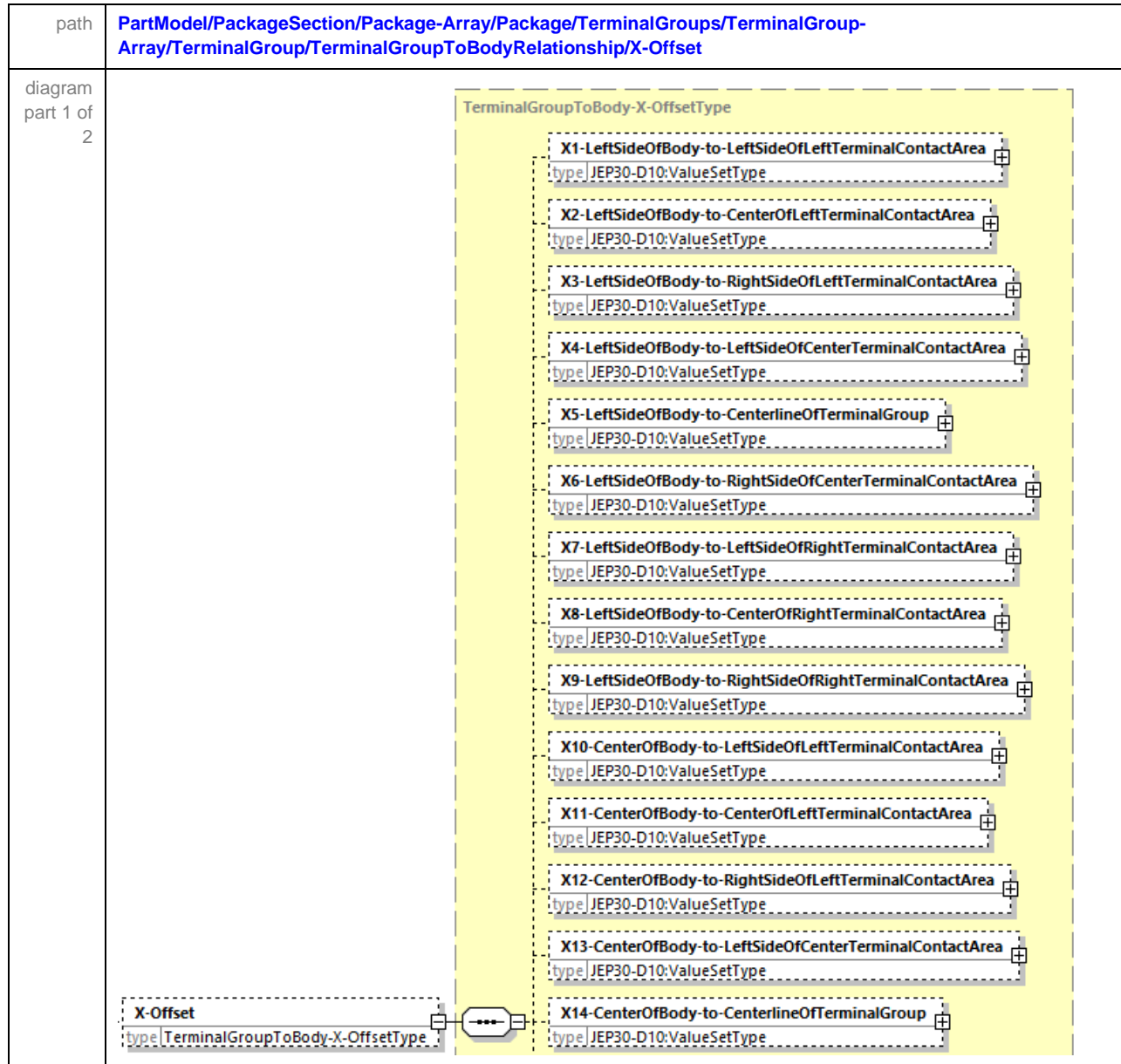
[TerminalSpan](#) and [TerminalSpacing](#) values are critical in the calculation of the most efficient land patterns for several [Terminal](#) types. The requirements for the provision of the above data elements can be found in JESD30, under the “Terminal Span and Terminal Spacing” section which includes a table titled “Terminal Shapes requiring Terminal Span and Terminal Spacing dimensions”.

### 5.14.1.10 Terminal Group to Body Relationship

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalGroupToBodyRelationship</a>
diagram	 <p>The diagram illustrates the structure of the <b>TerminalGroupToBodyRelationshipType</b>. It shows a dashed box labeled <b>TerminalGroupToBodyRelationship</b> with the attribute <code>type TerminalGroupToBodyRelationshipType</code>. This box is connected to a central octagonal symbol, which is then connected to a larger dashed box labeled <b>TerminalGroupToBodyRelationshipType</b>. Inside this larger box, there are three sub-sections: <b>X-Offset</b>, <b>Y-Offset</b>, and <b>Z-Offset</b>. Each sub-section contains the attribute <code>type TerminalGroupToBody-X-OffsetType</code>, <code>type TerminalGroupToBody-Y-OffsetType</code>, and <code>type TerminalGroupToBody-Z-OffsetType</code> respectively, and a plus sign (+) in a small box, indicating a required or optional attribute.</p>
type	<a href="#">TerminalGroupToBodyRelationshipType</a> , <a href="#">TerminalGroupToBody-X-OffsetType</a> , <a href="#">TerminalGroupToBody-Y-OffsetType</a> , <a href="#">TerminalGroupToBody-Z-OffsetType</a> .

For non-symmetrical Terminals around the package center, dimensions may be referenced from the Terminal Contact Area to the Package body area. This data can be captured in the [TerminalGroupToBodyRelationship](#) section.

### 5.14.1.10.1 Terminal Group to Body X-Offset



5.13.1.9.1 Terminal Group to Body X-Offset (cont'd)

diagram part 2 of 2	<p><b>X-Offset</b> type TerminalGroupToBody-X-OffsetType</p> <p>X14-CenterOfBody-to-CenterlineOfTerminalGroup type JEP30-D10:ValueSetType</p> <p>X15-CenterOfBody-to-RightSideOfCenterTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X16-CenterOfBody-to-LeftSideOfRightTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X17-CenterOfBody-to-CenterOfRightTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X18-CenterOfBody-to-RightSideOfRightTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X19-RightSideOfBody-to-LeftSideOfLeftTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X20-RightSideOfBody-to-CenterOfLeftTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X21-RightSideOfBody-to-RightSideOfLeftTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X22-RightSideOfBody-to-LeftSideOfCenterTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X23-RightSideOfBody-to-CenterlineOfTerminalGroup type JEP30-D10:ValueSetType</p> <p>X24-RightSideOfBody-to-RightSideOfCenterTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X25-RightSideOfBody-to-LeftSideOfRightTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X26-RightSideOfBody-to-CenterOfRightTerminalContactArea type JEP30-D10:ValueSetType</p> <p>X27-RightSideOfBody-to-RightSideOfRightTerminalContactArea type JEP30-D10:ValueSetType</p>
type	TerminalGroupToBody-X-OffsetType, JEP30-D10:ValueSetType.

Table 25 - Terminal Group to Package Body X-Offset lists out all the various dimensions that can be selected from any major point of the package body to any point of the terminal contact area in X direction.

Center of Center Terminal Contact Area is the same as Centerline of Terminal Group, however the use of the "...Centerline of Terminal Group" is more encompassing since it also includes the centerline of a row or column of an even number of terminals, i.e., midway between terminal 2 and terminal 3 of 4 terminals in a row or column.

### 5.13.1.9.1 Terminal Group to Body X-Offset (cont'd)

**Table 25 - Terminal Group to Package Body X-Offset**

Symbol	Description
X1	Left side of body to Left Side of Left Terminal Contact Area
X2	Left side of body to Center of Left Terminal Contact Area
X3	Left side of body to Right side of Left Terminal Contact Area
X4	Left side of body to Left Side of Center Terminal Contact Area
X5	Left side of body to Centerline of Terminal Group
X6	Left side of body to Right side of Center Terminal Contact Area
X7	Left side of body to Left side of Right Terminal Contact Area
X8	Left side of body to Center of Right Terminal Contact Area
X9	Left side of body to Right side of Right Terminal Contact Area
X10	Center of body to Left side of Left Terminal Contact Area
X11	Center of body to Center of Left Terminal Contact Area
X12	Center of body to Right side of Left Terminal Contact Area
X13	Center of body to Left side of Center Terminal Contact Area
X14	Center of body to Centerline of Terminal Group
X15	Center of body to Right side of Center Terminal Contact Area
X16	Center of body to Left side of Right Terminal Contact Area
X17	Center of body to Center of Right Terminal Contact Area
X18	Center of body to Right side of Right Terminal Contact Area
X19	Right side of body to Left side of Left Terminal Contact Area
X20	Right side of body to Center of Left Terminal Contact Area
X21	Right side of body to Right side of Left Terminal Contact Area
X22	Right side of body to Left side of Center Terminal Contact Area
X23	Right side of body to Centerline of Terminal Group
X24	Right side of body to Right side of Center Terminal Contact Area
X25	Right side of body to Left side of Right Terminal Contact Area
X26	Right side of body to Center of Right Terminal Contact Area
X27	Right side of body to Right side of Right Terminal Contact Area

#### 5.14.1.10.2 Terminal Group to Body Y-Offset

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroupToBodyRelationship/Y-Offset
diagram part 1 of 2	<pre> classDiagram     class TerminalGroupToBody_Y_OffsetType {         Y1-BackSideOfBody-to-BackSideOfBackTerminalContactArea JEP30-D10:ValueSetType         Y2-BackSideOfBody-to-CenterOfBackTerminalContactArea JEP30-D10:ValueSetType         Y3-BackSideOfBody-to-FrontSideOfBackTerminalContactArea JEP30-D10:ValueSetType         Y4-BackSideOfBody-to-BackSideOfCenterTerminalContactArea JEP30-D10:ValueSetType         Y5-BackSideOfBody-to-CenterlineOfTerminalGroup JEP30-D10:ValueSetType         Y6-BackSideOfBody-to-FrontSideOfCenterTerminalContactArea JEP30-D10:ValueSetType         Y7-BackSideOfBody-to-BackSideOfFrontTerminalContactArea JEP30-D10:ValueSetType         Y8-BackSideOfBody-to-CenterOfFrontTerminalContactArea JEP30-D10:ValueSetType         Y9-BackSideOfBody-to-FrontSideOfFrontTerminalContactArea JEP30-D10:ValueSetType         Y10-CenterOfBody-to-BackSideOfBackTerminalContactArea JEP30-D10:ValueSetType         Y11-CenterOfBody-to-CenterOfBackTerminalContactArea JEP30-D10:ValueSetType         Y12-CenterOfBody-to-FrontSideOfBackTerminalContactArea JEP30-D10:ValueSetType         Y13-CenterOfBody-to-BackSideOfCenterTerminalContactArea JEP30-D10:ValueSetType         Y14-CenterOfBody-to-CenterlineOfTerminalGroup JEP30-D10:ValueSetType     }     class Y_Offset {         type TerminalGroupToBody-Y-OffsetType     }     Y_Offset --&gt; Y1-BackSideOfBody-to-BackSideOfBackTerminalContactArea     </pre>

### 5.13.1.9.2 Terminal Group to Body Y-Offset (cont'd)

<p>diagram part 2 of 2</p>	
<p>type</p>	<p><b>TerminalGroupToBody-Y-OffsetType, JEP30-D10:ValueSetType.</b></p>

Table 26 - Terminal Group to Package Body Y-Offset lists out all the various dimensions that can be selected from any major point of the package body to any point of the terminal contact area in Y direction.

### 5.13.1.9.2 Terminal Group to Body Y-Offset (cont'd)

**Table 26 - Terminal Group to Package Body Y-Offset**

Symbol	Description
Y1	Back side of body to Back side of Back Terminal Contact Area
Y2	Back side of body to Center of Back Terminal Contact Area
Y3	Back side of body to Front side of Back Terminal Contact Area
Y4	Back side of body to Back side of Center Terminal Contact Area
Y5	Back side of body to Centerline of Terminal Group
Y6	Back side of body to Front side of Center Terminal Contact Area
Y7	Back side of body to Back side of Front Terminal Contact Area
Y8	Back side of body to Center of Front Terminal Contact Area
Y9	Back side of body to Front side of Front Terminal Contact Area
Y10	Center of body to Back side of Back Terminal Contact Area
Y11	Center of body to Center of Back Terminal Contact Area
Y12	Center of body to Front side of Back Terminal Contact Area
Y13	Center of body to Back side of Center Terminal Contact Area
Y14	Center of body to Centerline of Terminal Group
Y15	Center of body to Front side of Center Terminal Contact Area
Y16	Center of body to Back side of Front Terminal Contact Area
Y17	Center of body to Center of Front Terminal Contact Area
Y18	Center of body to Front side of Front Terminal Contact Area
Y19	Front side of body to Back side of Back Terminal Contact Area
Y20	Front side of body to Center of Back Terminal Contact Area
Y21	Front side of body to Front side of Back Terminal Contact Area
Y22	Front side of body to Back side of Center Terminal Contact Area
Y23	Front side of body to Centerline of Terminal Group
Y24	Front side of body to Front side of Center Terminal Contact Area
Y25	Front side of body to Back side of Front Terminal Contact Area
Y26	Front side of body to Center of Front Terminal Contact Area
Y27	Front side of body to Front side of Front Terminal Contact Area

### 5.14.1.10.3 Terminal Group to Body Z-Offset





### 5.13.1.9.3 Terminal Group to Body Z-Offset (cont'd)

<p>diagram part 2 of 2</p>	
<p>type</p>	<p><b>TerminalGroupToBody-Z-OffsetType, JEP30-D10:ValueSetType.</b></p>

Table 27 - Terminal Group to Package Body Z-Offset lists out all the various dimensions that can be selected from any major point of the package body to any point of the terminal contact area in Z direction.

### 5.13.1.9.3 Terminal Group to Body Z-Offset (cont'd)

**Table 27 - Terminal Group to Package Body Z-Offset**

Symbol	Description
Z1	Under side of Body to Lower side of Lower Terminal Contact Area
Z2	Under side of Body to Center of Lower Terminal Contact Area
Z3	Under side of Body to Upper side of Lower Terminal Contact Area
Z4	Under side of Body to Lower side of Center Terminal Contact Area
Z5	Under side of Body to Centerline of Terminal Group
Z6	Under side of Body to Upper side of Center Terminal Contact Area
Z7	Under side of Body to Lower side of Upper Terminal Contact Area
Z8	Under side of Body to Center of Upper Terminal Contact Area
Z9	Under side of Body to Upper side of Upper Terminal Contact Area
Z10	Center of Body to Lower side of Lower Terminal Contact Area
Z11	Center of Body to Center of Lower Terminal Contact Area
Z12	Center of Body to Upper side of Lower Terminal Contact Area
Z13	Center of Body to Lower side of Center Terminal Contact Area
Z14	Center of Body to Centerline of Terminal Group
Z15	Center of Body to Upper side of Center Terminal Contact Area
Z16	Center of Body to Lower side of Upper Terminal Contact Area
Z17	Center of Body to Center of Upper Terminal Contact Area
Z18	Center of Body to Upper side of Upper Terminal Contact Area
Z19	Upper side of Body to Lower side of Lower Terminal Contact Area
Z20	Upper side of Body to Center of Lower Terminal Contact Area
Z21	Upper side of Body to Upper side of Lower Terminal Contact Area
Z22	Upper side of Body to Lower side of Center Terminal Contact Area
Z23	Upper side of Body to Centerline of Terminal Group
Z24	Upper side of Body to Upper side of Center Terminal Contact Area
Z25	Upper side of Body to Lower side of Upper Terminal Contact Area
Z26	Upper side of Body to Center of Upper Terminal Contact Area
Z27	Upper side of Body to Upper side of Upper Terminal Contact Area

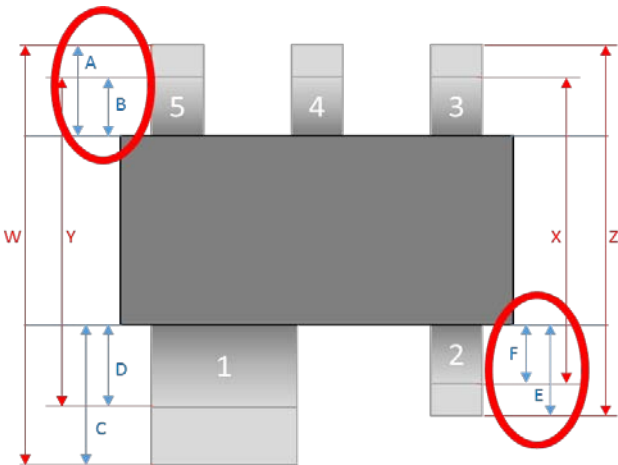
5.14.2 Terminal Group to Terminal Group Relationship Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroupToTerminalGroupRelationships-Array
diagram part 1 of 2	
diagram part 2 of 2	
type	TerminalGroupToTerminalGroupRelationships-ArrayType, TerminalGroupToTerminalGroupRelationship, TerminalSpanXYZType, TerminalSpacingXYZType.

This section is very similar to Terminal Span and Spacing but in this case, it is capturing the Terminal Span and Spacing between Terminal Contact Area that are not symmetrical around the Package-body shape, or between terminal groups whose terminal types are different.

Figure 7 - Terminal Group to Terminal Group Span and Spacing

Assuming that the only dimensions provided for the Part in Figure 7, were the dimensions (w, x, y, and z), but not dimensions (a, b, c, d, e or f), then the only way to establish the location of Terminal 1 contact area is to define terminals 2, 3, 4 and 5 first, and then leveraging off the terminal location for terminal 5, using dimension w and y, the dimension w therefore becomes the Terminal Span between terminal 1 and terminal 5. Dimension y becomes the Terminal Spacing between terminal 1 and terminal 5. Terminal 1 must be located in different terminal groups to terminals 2-5.



### 5.14.2 Terminal Group to Terminal Group Relationship Array (cont'd)

Terminal Span and Spacing values are more critical than Terminal length in the calculation of the most efficient land patterns for several Terminal Shapes. The requirements for the provision of the above data elements can be found in JESD30, under the “Terminal Span and Terminal Spacing” section which includes a table titled “Terminal Shapes requiring Terminal Span and Terminal Spacing dimensions”.

### 5.14.3 Terminal Detail Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array</a>
diagram	<pre> classDiagram     class TerminalDetailArray {         type TerminalDetail-ArrayType     }     class TerminalDetailArrayType {         TerminalDetail * 0..∞         FirstTerminalLocation         TerminalNumberPattern * 0..∞         TerminalDetailExceptions * 0..∞         constraints     }     TerminalDetailArray -- TerminalDetailArrayType     </pre>
type	<a href="#">TerminalDetail-ArrayType</a> , <a href="#">TerminalDetailType</a> , <a href="#">FirstTerminalLocationType</a> , <a href="#">TerminalNumberPatternType</a> , <a href="#">TerminalDetailExceptionsType</a> .

5.14.3.1 Terminal Detail

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalDetail
diagram	
type	TerminalDetailType, TerminalCenterType, JEP30-D10:MinIntegerOfOneType, TerminalStatusType, JEP30-D10:EmptyType.

This section identifies the relationship between the electrical *TerminalNumber* and the physical terminal location on the package. The physical terminal location can be defined by either the identification of the *TerminalGroup*, *RowTerminalIndex* and the *ColumnTerminalIndex* within that specific *TerminalGroup*, or by a set of x,y coordinates. If the *TerminalLocation* is a *CircularArray*, then the physical terminal location can be defined by either the identification of the *TerminalGroup*, *PolarTerminalIndex* within that specific *TerminalGroup*, or by polar coordinates.

The row and column terminal index are simply a grid index of the terminal locations, within a specific *TerminalGroup* and starts off with the number 1 in the lower left position. The *RowTerminalIndex* increments from left-to-right, whereas the *ColumnTerminalIndex* increments from back-to-front. The *TerminalGroupID*, *RowTerminalIndex* and the *ColumnTerminalIndex* combination provide a unique ID for every terminal on the package, irrespective of the status that is assigned to the specific terminal.

The *IgnoreForLandPattern* is when a specific terminal does not contribute to the generation of a land pattern.

5.14.3.1.1 Terminal Center

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalDetail/TerminalCenter
diagram	
type	TerminalCenterType, JEP30-D10:PointType

For [StandardArray](#) and for [RandomArray](#), xy coordinates can be used to capture the center of the terminals, whereas for [CircularArray](#), polar coordinates are a more convenient representation to capture the terminal centers.

### 5.13.3.1.1 Terminal Center (cont'd)

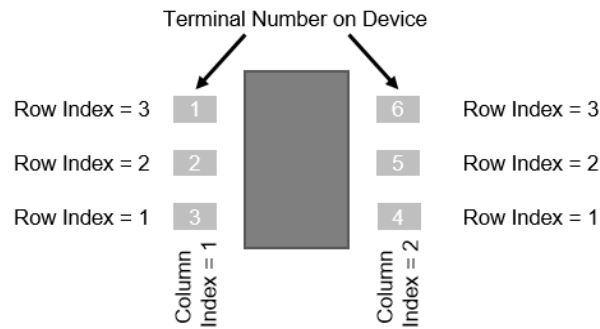


Figure 8 - SOIC

```

<TerminalDetail-Array>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>3</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>2</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>3</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>4</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>5</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>6</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>3</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
</TerminalDetail-Array>

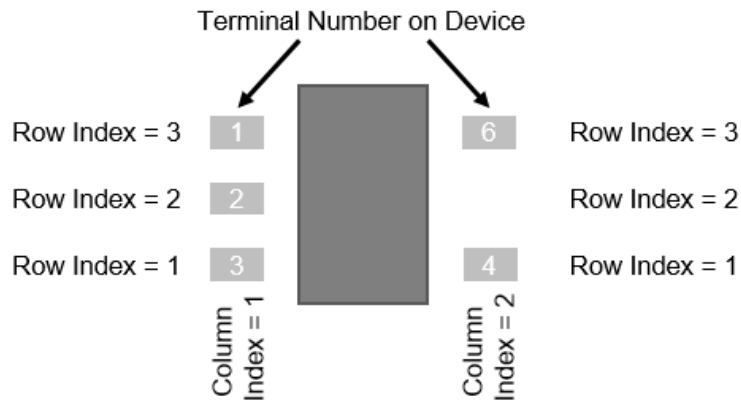
```

### 5.14.3.1.2 Terminal Status

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalDetail/TerminalStatus</a>
diagram	<p>The diagram illustrates the structure of the <b>TerminalStatusType</b>. It is a choice between four elements: <b>Missing</b> (type JEP30-D10:EmptyType), <b>Deleted</b> (type JEP30-D10:EmptyType), <b>Excluded</b> (type JEP30-D10:EmptyType), and <b>Associated</b> (type AssociatedTerminalStatusType). The <b>Associated</b> element is further detailed in the <b>AssociatedTerminalStatusType</b> diagram, which includes <b>ReferenceTerminalLocationID</b> (type xs:string), <b>ReferencePatternGroupID</b> (type xs:string), <b>ReferenceTerminalGroupID</b> (type xs:string), <b>TerminalCenter</b> (type TerminalCenterType), and a choice between <b>RowTerminalIndex</b> (type JEP30-D10:MinIntegerOfOneType, minIncl/maxIncl 1), <b>ColumnTerminalIndex</b> (type JEP30-D10:MinIntegerOfOneType, minIncl/maxIncl 1), and <b>PolarTerminalIndex</b> (type JEP30-D10:MinIntegerOfOneType, minIncl/maxIncl 1). Additionally, <b>LocationRelative-to-PackageCenter</b> (type LocationRelative-to-PackageCenterType) is shown as a separate element.</p>
type	<a href="#">TerminalStatusType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">AssociatedTerminalStatusType</a> , <a href="#">TerminalCenterType</a> , <a href="#">JEP30-D10:MinIntegerOfOneType</a> .

This section identifies the terminal status of each terminal. The status of [Missing](#), [Deleted](#) and [Excluded](#) are defined in section called “Terminal-count suffixes” of JESD30.

The [TerminalDetail](#) and associated [TerminalStatus](#) will now look like this in the xml file.



**Figure 9 - SOIC with Terminal 5 Missing**



### 5.13.3.1.2 Terminal Status (cont'd)

```
<TerminalDetail>
  <TerminalNumber>5</TerminalNumber>
  <TerminalGroupID>1</TerminalGroupID>
  <RowTerminalIndex>2</RowTerminalIndex>
  <ColumnTerminalIndex>2</ColumnTerminalIndex>
  <TerminalStatus>
    <Missing/>
  </TerminalStatus>
</TerminalDetail>
```

Note how *TerminalNumber* 5 is included in the xml file, and that the terminal at location of *RowTerminalIndex* = 3, and *ColumnTerminalIndex* = 2, that the *TerminalNumber* is 6.

When the Terminal is deleted, its index within the *TerminalGroup*, must still be defined. However, notice how the *TerminalNumber* element is not populated for this deleted position of *RowTerminalIndex* = 2, and *ColumnTerminalIndex* = 2. *TerminalNumber* 5 is now included in the xml file at location of *RowTerminalIndex* = 3, and *ColumnTerminalIndex* = 2.

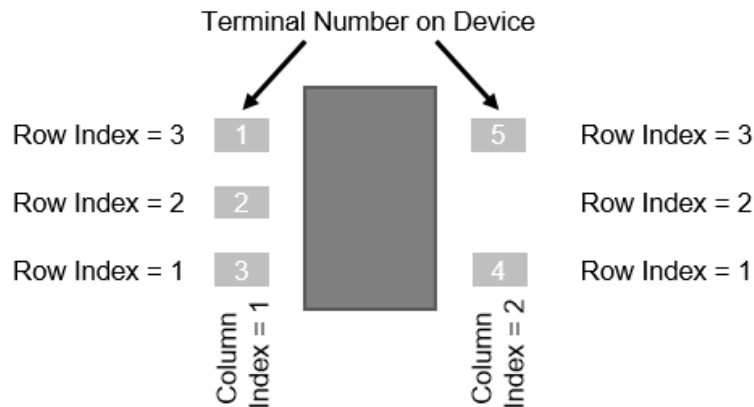


Figure 10 - SOIC with Terminal 5 Deleted

### 5.13.3.1.2 Terminal Status (cont'd)

```
<TerminalDetail-Array>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>3</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>2</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>3</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>4</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
    <TerminalStatus>
      <Deleted/>
    </TerminalStatus>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>5</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>3</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
</TerminalDetail-Array>
```

### 5.13.3.1.2 Terminal Status (cont'd)

*TerminalStatus* of *Associated* occurs when you have parts with physically connected terminals and where there is a special requirement to be implemented in the land pattern for the proper performance of the device. The device may be constructed in accordance with either option A or option B as shown in Figure 11.

This device has two *TerminalGroups*. Lets assume that *TerminalGroup* 1 represents a 2 rows of 4 columns, and that *TerminalGroup* 2 represent the actual shape of terminal 1 on the device.

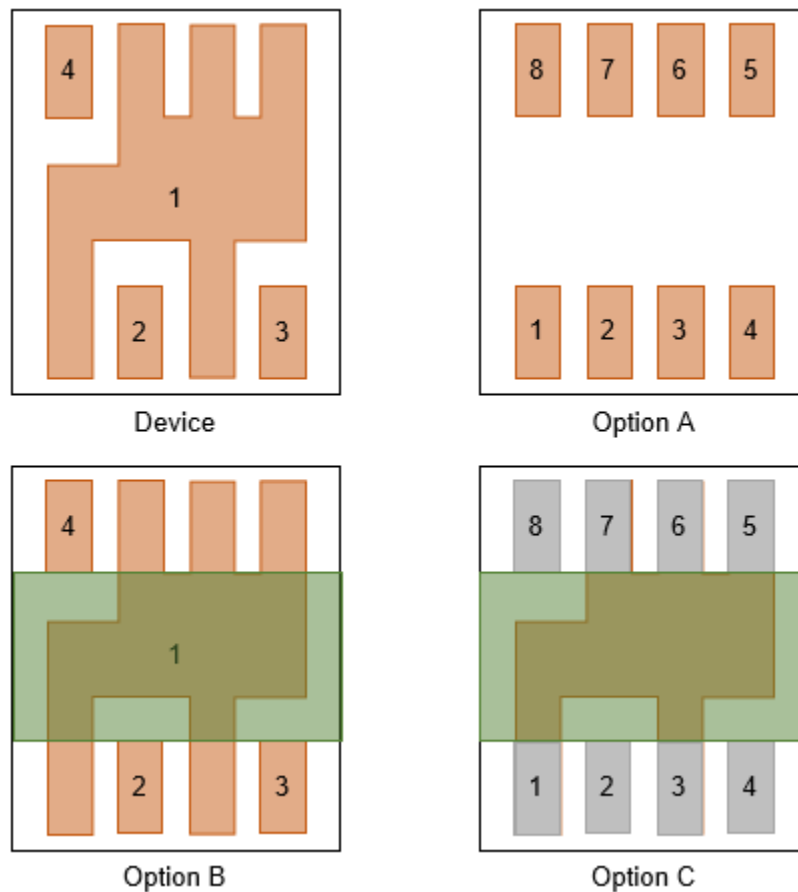


Figure 11 – Part with Special Terminal Pattern

### 5.13.3.1.2 Terminal Status (cont'd)

If option A land pattern is desired for the device, then the xml configuration should be as follows:

```
<TerminalDetail-Array>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>2</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>3</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>4</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>5</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>6</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>
</TerminalDetail-Array>
```

### 5.13.3.1.2 Terminal Status (cont'd)

```

<TerminalDetail>
  <TerminalNumber>7</TerminalNumber>
  <TerminalGroupID>1</TerminalGroupID>
  <RowTerminalIndex>2</RowTerminalIndex>
  <ColumnTerminalIndex>2</ColumnTerminalIndex>
  <TerminalStatus>
    <Associated>
      <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
    </Associated>
  </TerminalStatus>
</TerminalDetail>
<TerminalDetail>
  <TerminalNumber>8</TerminalNumber>
  <TerminalGroupID>1</TerminalGroupID>
  <RowTerminalIndex>2</RowTerminalIndex>
  <ColumnTerminalIndex>1</ColumnTerminalIndex>
</TerminalDetail>
<TerminalDetail>
  <TerminalGroupID>2</TerminalGroupID>
  <RowTerminalIndex>1</RowTerminalIndex>
  <ColumnTerminalIndex>1</ColumnTerminalIndex>
  <IgnoreForLandPattern/>
</TerminalDetail>
</TerminalDetail-Array>

```

In this scenario, the following are enforced:-

1. Terminal group 2 is set to *IgnoreForLandPattern*,
2. Positions as shown by the number sequence in option A of 1, 3, 5, 6 and 7 are associated with terminal Group 2.
3. Since there is only 1 terminal in TerminalGroup 2, then there is no requirement to reference the *RowTerminalIndex* and the *ColumnTerminalIndex*. If the *RowTerminalIndex* and the *ColumnTerminalIndex* are omitted, then 1 is assumed.

### 5.13.3.1.2 Terminal Status (cont'd)

If option B land pattern is desired for the device, then the xml configuration should be as follows.

```
<TerminalDetail-Array>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>2</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>3</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>

```

### 5.13.3.1.2 Terminal Status (cont'd)

```

        </TerminalStatus>
        <IgnoreForLandPattern/>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>1</TerminalNumber>
        <TerminalGroupID>1</TerminalGroupID>
        <RowTerminalIndex>2</RowTerminalIndex>
        <ColumnTerminalIndex>2</ColumnTerminalIndex>
        <TerminalStatus>
            <Associated>
                <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
            </Associated>
        </TerminalStatus>
        <IgnoreForLandPattern/>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>4</TerminalNumber>
        <TerminalGroupID>1</TerminalGroupID>
        <RowTerminalIndex>2</RowTerminalIndex>
        <ColumnTerminalIndex>1</ColumnTerminalIndex>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>1</TerminalNumber>
        <TerminalGroupID>2</TerminalGroupID>
        <RowTerminalIndex>1</RowTerminalIndex>
        <ColumnTerminalIndex>1</ColumnTerminalIndex>
    </TerminalDetail>
</TerminalDetail-Array>

```

In this scenario, the following are enforced:-

1. Terminal group 2 is not set to *IgnoreForLandPattern* and therefore drives the shape of the land pattern on the printed board.
2. Positions as shown by the number sequence in option A of 1, 3, 5, 6 and 7 are associated with terminal Group 2, and each of these are set to *IgnoreForLandPattern*.

### 5.13.3.1.2 Terminal Status (cont'd)

If option C land pattern is desired for the device, then the xml configuration should be as follows.

```
<TerminalDetail-Array>
  <TerminalDetail>
    <TerminalNumber>1</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>1</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>2</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>2</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>3</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>4</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>1</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>5</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>4</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
    <IgnoreForLandPattern/>
  </TerminalDetail>
  <TerminalDetail>
    <TerminalNumber>6</TerminalNumber>
    <TerminalGroupID>1</TerminalGroupID>
    <RowTerminalIndex>2</RowTerminalIndex>
    <ColumnTerminalIndex>3</ColumnTerminalIndex>
    <TerminalStatus>
      <Associated>
        <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
      </Associated>
    </TerminalStatus>
  </TerminalDetail>
</TerminalDetail-Array>
```



### 5.13.3.1.2 Terminal Status (cont'd)

```

        </TerminalStatus>
        <IgnoreForLandPattern/>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>7</TerminalNumber>
        <TerminalGroupID>1</TerminalGroupID>
        <RowTerminalIndex>2</RowTerminalIndex>
        <ColumnTerminalIndex>2</ColumnTerminalIndex>
        <TerminalStatus>
            <Associated>
                <ReferenceTerminalGroupID>2</ReferenceTerminalGroupID>
            </Associated>
        </TerminalStatus>
        <IgnoreForLandPattern/>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>8</TerminalNumber>
        <TerminalGroupID>1</TerminalGroupID>
        <RowTerminalIndex>2</RowTerminalIndex>
        <ColumnTerminalIndex>1</ColumnTerminalIndex>
    </TerminalDetail>
    <TerminalDetail>
        <TerminalNumber>1</TerminalNumber>
        <TerminalGroupID>2</TerminalGroupID>
        <RowTerminalIndex>1</RowTerminalIndex>
        <ColumnTerminalIndex>1</ColumnTerminalIndex>
    </TerminalDetail>
</TerminalDetail-Array>

```

In this scenario, the following are enforced:-

1. Terminal group 2 is not set to *IgnoreForLandPattern* and therefore drives the shape of the land pattern on the printed board.
2. Positions as shown by the number sequence in option A of 1, 3, 5, 6 and 7 are associated with terminal Group 2, and each of these are set to *IgnoreForLandPattern*.

### 5.14.3.2 First Terminal Location

path	<b>PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/FirstTerminalLocation</b>
diagram	<p>The diagram illustrates the structure of the <b>FirstTerminalLocationType</b> element. It is a complex type containing several child elements. The root element is <b>FirstTerminalLocationType</b>, which has a choice between <b>TerminalLocationID</b>, <b>PatternGroupID</b>, and <b>TerminalGroupID</b>. It also has a choice between <b>TerminalCenter</b> and a choice between <b>RowTerminalIndex</b> and <b>ColumnTerminalIndex</b>. Additionally, it has a choice between <b>PolarTerminalIndex</b> and a choice between <b>LocationRelative-to-PackageCenter</b> and a choice between <b>Code</b> and <b>Description</b>. The <b>LocationRelative-to-PackageCenter</b> element is further detailed with a choice between <b>Code</b> and <b>Description</b>.</p>
type	<b>FirstTerminalLocationType, TerminalCenterType, JEP30-D10:MinIntegerOfOneType, LocationRelative-to-PackageCenterType, LocationRelative-to-PackageCenterCodeType, LocationRelative-to-PackageCenterDescriptionType.</b>

The enumerated values for the *LocationRelative-to-PackageCenter* (Code and Description) are defined in Table 28.

### 5.13.3.2.1 First Terminal Location (cont'd)

**Table 28 - Location Relative to Package Center**

Code	Description
SW	Southwest
SE	Southeast
NE	Northeast
NW	Northwest
BL	Back-Left
BC	Back-Center
BR	Back-Right
FL	Front-Left
FC	Front-Center
FR	Front-Right
LB	Left-Bottom
LC	Left-Center
LT	Left-Top
RB	Right-Bottom
RC	Right-Center
RT	Right-Top
L	Left
B	Back
R	Right
F	Front

Reference the “Single Position” single terminal illustrations in JESD30, Annex A for graphical representations of some of the above descriptions.

### 5.14.3.3 Terminal Number Pattern

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern</a>
diagram	
type	<a href="#">TerminalNumberPatternType</a> , <a href="#">SequentialTerminalNumberOrderingType</a> , <a href="#">GridTerminalNumberOrderingType</a> .

If the terminal numbering on a Part has logical structure to its pattern, then this can be captured in this [TerminalNumberPattern](#) section. The section covers the capturing of the [FirstTerminalLocation](#) which is the start of the pattern numbering via the [Sequential](#) or [Grid](#) branches. Any terminals that are either deleted, missing or associated within the pattern are then covered via the [TerminalDetailExceptions](#).

5.14.3.3.1 Sequential

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern/Sequential
diagram	
type	SequentialTerminalNumberOrderingType, TerminalNumberOrderingCodeType, TerminalNumberOrderingDescriptionType, NumericalSequenceType, AlphabeticalSequenceType.

The enumerated values for the *Sequential* (*Code* and *Description*) are defined in Table 29.

#### 5.13.3.2.1 Sequential (cont'd)

**Table 29 - Terminal Number Sequential Pattern**

Code	Description
CW	Clockwise
CCW	Counter-Clockwise
L2R	Left-to-Right
B2F	Back-to-Front
R2L	Right-to-Left
F2B	Front-to-Back
S-H	Snake Horizontal
S-V	Snake Vertical
ZZ-H	Zig-Zag Horizontal
ZZ-V	Zig-Zag Vertical

Typically, the default sequence is a running number starting with the digit 1 and incrementing by 1. If this is the case, then the *NumericalSequence* or the *AlphabeticalSequence* do not have to be populated. However, in some cases, the Numerical or alphabetical sequence can override the default, in which case the sequence of terminal numbers is applied in the same order as the sequence pattern defined by the Code or Description as outlined in Table 29.






### 5.13.3.2.1 Sequential (cont'd)

**Table 30 - Terminal Number Patterns**

Pattern	Terminal Numbering Pattern	Sequential Code	Start Position	Towards Point 2	Towards Point 3	Towards Point 4	Towards Point 5	Last Point
1		CW	User Defined	Start + 1 CW	Start + 2 CW	Start + 3 CW	Start + 4 CW	Start - 1 CW
2		CCW	User Defined	Start + 1 CCW	Start + 2 CCW	Start + 3 CCW	Start + 4 CCW	Start - 1 CCW
3		ZZ-V	BL	TL	BL+1	TL+1	BL+2	TR
4		ZZ-V	BR	TR	BR-1	TR-1	BR-2	TL
5		ZZ-V	TR	BR	TR-1	BR-1	TR-2	BL
6		ZZ-V	TL	BL	TL+1	BL+1	TL+2	BR
7		ZZ-H	TL	TR	T-1L	T-1R	T-2L	BR
8		ZZ-H	BL	BR	B+1L	B+1R	B+2L	TR
9		ZZ-H	BR	BL	B+1R	B+1L	B+2R	TL
10		ZZ-H	TR	TL	T-1R	T-1L	T-2R	BL
11		S-V	TL	BL	BL+1	TL+1	TL+2	IF(Col # = odd, BR, TR)
12		S-V	BL	TL	TL+1	BL+1	BL+2	IF(Col # = odd, TR, BR)
13		S-V	TR	BR	BR-1	TR-1	TR-2	IF(Col # = odd, BL, TL)
14		S-V	BR	TR	TR-1	BR-1	BR-2	IF(Col # = odd, TL, BL)
15		S-H	TL	TR	T-1R	T-1L	T-2L	IF(Col # = odd, BR, BL)
16		S-H	BL	BR	B+1R	B+1L	B+2L	IF(Col # = odd, TR, TL)
17		S-H	TR	TL	R-1L	T-1R	T-2R	IF(Col # = odd, BL, BR)
18		S-H	BR	BL	B+1L	B+1R	B+2R	IF(Col # = odd, TL, TR)

### 5.13.3.2.1 Sequential (cont'd)

**Table 30 – Terminal Number Patterns (cont'd)**

Pattern	Terminal Numbering Pattern	Sequential Code	Start Position	Towards Point 2	Towards Point 3	Towards Point 4	Towards Point 5	Last Point
19		L2R	L	L+1	L+2	L+3	L+4	R
20		R2L	R	R-1	R-2	R-3	R-4	L
21		B2F	B	B+1	B+2	B+3	B+4	T
22		F2B	T	T-1	T-2	T-3	T-4	B
23		Assign	User Defined	N/A	N/A	N/A	N/A	N/A

By combining the start position of the terminal number via the [LocationRelative-to-PackageCenter](#) with the [Sequential](#) Pattern, we can achieve the patterns outlined in Table 30 - Terminal Number Patterns for terminal numbering.

For positions of BL, TL, BR, TR, the following is also valid for terminals positioned outside of the package body, namely.

1. The position of BL (Bottom Left) corresponds to
  - (a) For terminals with a position of Left, Right or Left-to-Right of the package body, then Left-bottom (LB),
  - (b) For terminals with a position of Back, Front or Back-to-Front of the package body, then Back-left (BL),
  - (c) For terminals with a position of Diagonal of the package body, then Southwest (SW),
2. The position of TL (Top Left) corresponds to
  - (a) For terminals with a position of Left, Right or Left-to-Right of the package body, then Left-top (LT),
  - (b) For terminals with a position of Back, Front or Back-to-Front of the package body, then Front-left (FL),
  - (c) For terminals with a position of Diagonal of the package body, then Northwest (NW),



### 5.13.3.2.1 Sequential (cont'd)

3. The position of BR (Bottom Right) corresponds to
  - (a) For terminals with a position of Left, Right or Left-to-Right of the package body, then Right-bottom (RB),
  - (b) For terminals with a position of Back, Front or Back-to-Front of the package body, then Back-right (BR),
  - (c) For terminals with a position of Diagonal of the package body, then Southeast (SE),
4. The position of TR (Top Right) corresponds to
  - (a) For terminals with a position of Left, Right or Left-to-Right of the package body, then Right-top (RT),
  - (b) For terminals with a position of Back, Front or Back-to-Front of the package body, then Front-right (FR),
  - (c) For terminals with a position of Diagonal of the package body, then Northeast (NE).



Figure 12 – Terminal Numbering Pattern Samples

### 5.14.3.3.2 Grid

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern/Grid
diagram	
type	GridTerminalNumberOrderingType, RowTerminalNumberOrderingType, NumericalSequenceType, AlphabeticalSequenceType, JEP30-D10:EmptyType, ColumnTerminalNumberOrderingType.

This section identifies the terminal numbering of each terminal in a grid array by providing both a *Row* and *Column* identification of the terminal location on the *Grid*. Each Row and Column can have either a *NumericalSequence* or an *AlphabeticalSequence*. The *NumericalSequence* is a running number beginning with the digit 1 and incrementing by 1, however that sequence can be over-ridden by the provision of a sequence of numbers defined in the *Numerical* element.

Similarly, the *AlphabeticalSequence* is a running number sequence of the letters of the English alphabet, however in the majority of cases, the characters I, O, Q, S, X, and Z are sometimes not used because it is hard to distinguish these either legibly or from their numerical equivalency from the legend. As such the Character element under the *AlphabeticalSequence* is an unbounded element that contains the used set of characters in an ordered sequence.

The *Prefix* is an optional integer that can be placed before the character to define the terminal numbering of a multi grid package, as shown in Figure 13 – A Multi grid Package.

5.13.3.2.3 Grid (cont'd)

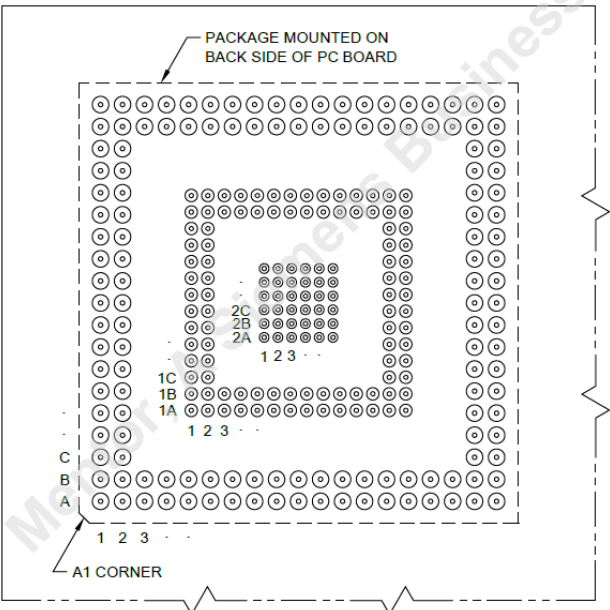
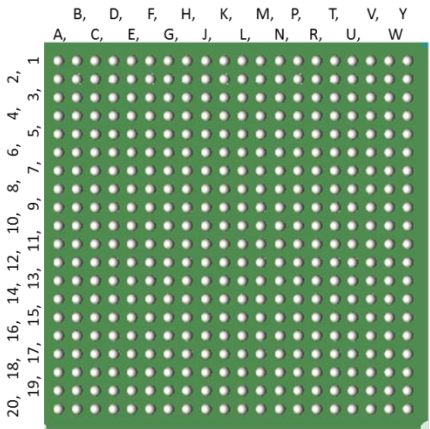


Figure 13 – A Multi grid Package

In Figure 14, the columns have a *NumericalSequence* from *Front-to-Back*. The rows have an *AlphabeticalSequence* going from *Left-to-Right*, with a specific set of alphabetical set of *Characters*. The xml structure below represents the detail terminal numbering in this example.

Figure 14 – Grid Array with Terminal Numbering



```
<TerminalDetail-Array>
  <TerminalNumberPattern>
    <FirstTerminalLocation>
      <TerminalGroupID> Terminal Group ID 1 </TerminalGroupID>
      <LocationRelative-to-PackageCenter>
        <Code>SW</Code>
      </LocationRelative-to-PackageCenter>
    </FirstTerminalLocation>
  </TerminalNumberPattern>
  <Grid>
    <Row>
      <AlphabeticalSequence>
        <Character>A</Character>
        <Character>B</Character>
        <Character>C</Character>
        <Character>D</Character>
        <Character>E</Character>
        <Character>F</Character>
        <Character>G</Character>
        <Character>H</Character>
        <Character>J</Character>
        <Character>K</Character>
        <Character>L</Character>
      </AlphabeticalSequence>
    </Row>
  </Grid>
</TerminalDetail-Array>
```

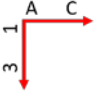
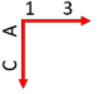

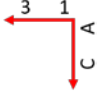



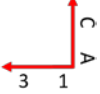
### 5.13.3.2.3 Grid (cont'd)

```


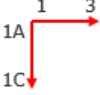

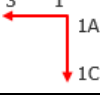

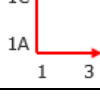
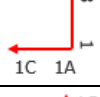

<Character>M</Character>
<Character>N</Character>
<Character>P</Character>
<Character>R</Character>
<Character>T</Character>
<Character>U</Character>
<Character>V</Character>
<Character>W</Character>
<Character>Y</Character>
</AlphabeticalSequence>
<Left-to-Right/>
</Row>
<Column>
<NumericalSequence></NumericalSequence>
<Front-to-Back/>
</Column>
</Grid>
</TerminalNumberPattern>
</TerminalDetail-Array>

```

**Table 31 – Grid Terminal Numbering Pattern**

Grid	Terminal Grid Pattern	First Terminal Location	Numerical Sequence	Alphabetical Sequence
1		NW	Front-to-Back	Left-to-Right
2		NW	Left-to-Right	Front-to-Back
3		NE	Front-to-Back	Right-to-Left
4		NE	Right-to-Left	Front-to-Back
5		SW	Back-to-Front	Left-to-Right
6		SW	Left-to-Right	Back-to-Front
7		SE	Back-to-Front	Right-to-Left
8		SE	Right-to-Left	Back-to-Front

**Table 31 - Grid Terminal Number Patterns (cont'd).**

Grid	Terminal Grid Pattern	First Terminal Location	Numerical Sequence	Alphabetical Sequence
9		NW	Front-to-Back	Left-to-Right
10		NW	Left-to-Right	Front-to-Back
11		NE	Front-to-Back	Right-to-Left
12		NE	Right-to-Left	Front-to-Back
13		SW	Back-to-Front	Left-to-Right
14		SW	Left-to-Right	Back-to-Front
15		SE	Back-to-Front	Right-to-Left
15		SE	Right-to-Left	Back-to-Front

5.14.3.4 Terminal Detail Exceptions

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalDetailExceptions
diagram	<p>The diagram illustrates the structure of the <b>TerminalDetailExceptionsType</b>. It is a complex type containing several elements:</p> <ul style="list-style-type: none"><li><b>TerminalLocationID</b>: type <code>xs:string</code></li><li><b>PatternGroupID</b>: type <code>xs:string</code></li><li><b>TerminalGroupID</b>: type <code>xs:string</code></li><li><b>TerminalCenter-Array</b>: type <code>TerminalCenter-ArrayType</code></li><li><b>TerminalIndex</b>: type <code>TerminalIndexType</code></li><li><b>TerminalNumber</b>: type <code>TerminalNumberType</code></li><li><b>TerminalStatus</b>: type <code>TerminalStatusType</code></li><li><b>IgnoreForLandPattern</b>: type <code>JEP30-D10:EmptyType</code></li></ul> <p>The <b>TerminalStatusType</b> is further detailed as follows:</p> <ul style="list-style-type: none"><li><b>Missing</b>: type <code>JEP30-D10:EmptyType</code></li><li><b>Deleted</b>: type <code>JEP30-D10:EmptyType</code></li><li><b>Excluded</b>: type <code>JEP30-D10:EmptyType</code></li><li><b>Associated</b>: type <code>AssociatedTerminalStatusType</code></li></ul> <p>The diagram also shows a dashed box for <b>TerminalDetailExceptions</b> with a cardinality of <code>0..∞</code>.</p>
type	TerminalDetailExceptionsType, TerminalCenterType, TerminalIndexType, TerminalNumberType, TerminalStatusType, JEP30-D10:EmptyType, AssociatedTerminalStatusType.

5.14.3.4.1.1      Terminal Center - Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern/TerminalDetailExceptions/TerminalCenter-Array
diagram	<p>The diagram illustrates the structure of the <b>TerminalCenter-Array</b> type. It is composed of a sequence of <b>TerminalCenter</b> types. Each <b>TerminalCenter</b> type contains a sequence of four attributes: <b>x</b> (xs:decimal), <b>y</b> (xs:decimal), <b>PitchRadius</b> (xs:decimal), and <b>Center</b> (JEP30-D10:PointXYType). The <b>Center</b> attribute is optional, indicated by a plus sign in the bottom right corner. Additionally, there is an optional <b>Angle</b> attribute (xs:integer) shown below the <b>Center</b> attribute. The <b>TerminalCenter</b> type is shown as a dashed box within the <b>TerminalCenter-Array</b> type, and the <b>TerminalCenter</b> type is also shown as a dashed box within the <b>TerminalCenter-Array</b> type.</p>
type	TerminalCenter-ArrayType, TerminalCenterType, JEP30-D10:PointXYType.

5.14.3.4.1.2 Terminal Index

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern/TerminalDetailExceptions/TerminalIndex
diagram	<p>The diagram illustrates the structure of the <b>TerminalIndexType</b>. It is a disjoint union of three main categories: <b>RowTerminalIndex</b>, <b>ColumnTerminalIndex</b>, and <b>PolarTerminalIndex</b>. Each of these categories is further divided into two sub-categories: <b>From</b> and <b>To</b>. All indices are of type <b>JEP30-D10:MinIntegerOfOneType</b> with <b>minIncl/maxIncl</b> of 1.</p> <pre>graph LR     TI[TerminalIndex] --&gt; RTI[RowTerminalIndex]     TI --&gt; CTI[ColumnTerminalIndex]     TI --&gt; PTI[PolarTerminalIndex]     RTI --&gt; FRTI[FromRowTerminalIndex]     RTI --&gt; TRTI[ToRowTerminalIndex]     CTI --&gt; FCTI[FromColumnTerminalIndex]     CTI --&gt; TCTI[ToColumnTerminalIndex]     PTI --&gt; FPTI[FromPolarTerminalIndex]     PTI --&gt; TPTI[ToPolarTerminalIndex]</pre> <p><b>TerminalIndex</b> type: TerminalIndexType</p> <p><b>TerminalIndexType</b></p> <p><b>RowTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>FromRowTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>ToRowTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>ColumnTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>FromColumnTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>ToColumnTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>PolarTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>FromPolarTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p> <p><b>ToPolarTerminalIndex</b> type: JEP30-D10:MinIntegerOfOneType minIncl/maxIncl: 1</p>
type	TerminalIndexType, JEP30-D10:MinIntegerOfOneType.



5.14.3.4.1.3 Terminal Number

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern/TerminalDetailExceptions/TerminalNumberType
diagram	
type	TerminalNumberType

If a pattern of terminals are missing from the device, then these missing patterns can be captured under the *TerminalDetailExceptions* section as shown below for the Peripheral Grid Array shown in Figure 15.

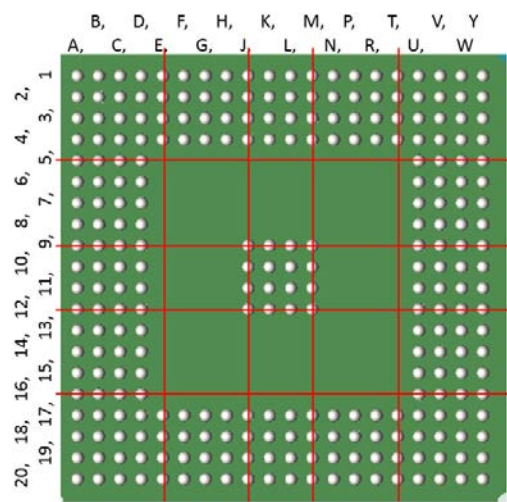


Figure 15 – Periphery Grid Array with Inner Array Matrix

#### 5.13.3.2.4.2 Terminal Number (cont'd)

```
<TerminalDetailExceptions>
  <TerminalNumber>
    <FromRow>5</FromRow>
    <ToRow>16</ToRow>
    <FromColumn>E</FromColumn>
    <ToColumn>H</ToColumn>
  </TerminalNumber>
  <TerminalStatus>
    <Missing></Missing>
  </TerminalStatus>
</TerminalDetailExceptions>
<TerminalDetailExceptions>
  <TerminalNumber>
    <FromRow>5</FromRow>
    <ToRow>16</ToRow>
    <FromColumn>N</FromColumn>
    <ToColumn>T</ToColumn>
  </TerminalNumber>
  <TerminalStatus>
    <Missing></Missing>
  </TerminalStatus>
</TerminalDetailExceptions>
<TerminalDetailExceptions>
  <TerminalNumber>
    <FromRow>5</FromRow>
    <ToRow>8</ToRow>
    <FromColumn>J</FromColumn>
    <ToColumn>M</ToColumn>
  </TerminalNumber>
  <TerminalStatus>
    <Missing></Missing>
  </TerminalStatus>
</TerminalDetailExceptions>
<TerminalDetailExceptions>
  <TerminalNumber>
    <FromRow>13</FromRow>
    <ToRow>16</ToRow>
    <FromColumn>J</FromColumn>
    <ToColumn>M</ToColumn>
  </TerminalNumber>
  <TerminalStatus>
    <Missing></Missing>
  </TerminalStatus>
</TerminalDetailExceptions>
```

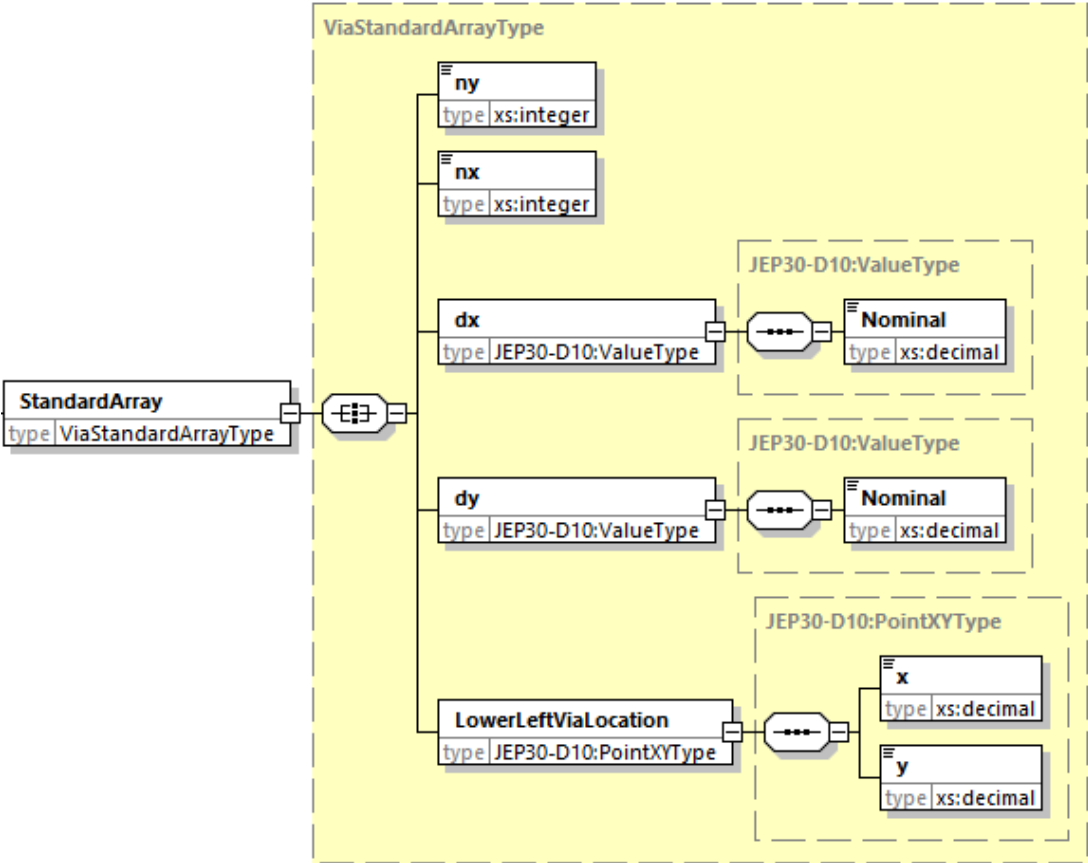
5.14.4 Via Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/Via-Array
diagram	<p>The diagram illustrates the structure of the <b>Via-Array</b> section. It shows a <b>Via-Array</b> type (type <b>Via-ArrayType</b>) which is a container for <b>ViaLocation</b> (type <b>ViaLocationType</b>) and <b>ViaDetail</b> (type <b>ViaDetailType</b>). The <b>ViaLocation</b> type is further detailed with attributes <b>ID</b> (type <b>xs:string</b>) and <b>TerminalGroupID</b> (type <b>xs:string</b>). The <b>ViaLocation</b> type is also associated with three array types: <b>StandardArray</b> (type <b>ViaStandardArrayType</b>), <b>CircularArray</b> (type <b>ViaCircularArrayType</b>), and <b>RandomArray</b> (type <b>ViaRandomArrayType</b>). The <b>ViaDetail</b> type is associated with a <b>constraints</b> type. The diagram uses dashed boxes to group related types and solid lines with arrows to show relationships and cardinalities (e.g., 1..∞ for <b>ViaLocation</b> and 0..∞ for <b>ViaDetail</b>).</p>
type	<b>Via-ArrayType</b> , <b>ViaLocationType</b> , <b>ViaStandardArrayType</b> , <b>ViaCircularArrayType</b> , <b>ViaRandomArrayType</b> , <b>ViaDetailType</b> .

Grounding Vias are sometimes required to support the operational functionality of the Part in a product design. This **Via-Array** section enables this detail to be captured. The location and pattern of these Vias can be captured via any combination of a **StandardArray**, **CircularArray** or **RandomArray** as shown in the diagram..

The dimensions and technology of the vias are not defined since this is determined by the Product on to which the Part is assembled.

5.14.4.1 Standard Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/Via-Array/ViaLocation/StandardArray
diagram	 <p>The diagram illustrates the structure of the <b>StandardArray</b> type. It is a class with a type <b>ViaStandardArrayType</b>. The structure is defined within a dashed box labeled <b>ViaStandardArrayType</b>. The structure includes the following elements:</p> <ul style="list-style-type: none"><li><b>ny</b>: type <b>xs:integer</b></li><li><b>nx</b>: type <b>xs:integer</b></li><li><b>dx</b>: type <b>JEP30-D10:ValueType</b>, which is further defined as <b>Nominal</b> (type <b>xs:decimal</b>) within a dashed box labeled <b>JEP30-D10:ValueType</b>.</li><li><b>dy</b>: type <b>JEP30-D10:ValueType</b>, which is further defined as <b>Nominal</b> (type <b>xs:decimal</b>) within a dashed box labeled <b>JEP30-D10:ValueType</b>.</li><li><b>LowerLeftViaLocation</b>: type <b>JEP30-D10:PointXYType</b>, which is further defined as a structure with <b>x</b> (type <b>xs:decimal</b>) and <b>y</b> (type <b>xs:decimal</b>) within a dashed box labeled <b>JEP30-D10:PointXYType</b>.</li></ul>
type	ViaStandardArrayType, JEP30-D10:ValueType, JEP30-D10:PointXYType.

5.14.4.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/Via-Array/ViaLocation/CircularArray
diagram	
type	ViaCircularArrayType, JEP30-D10:PointXYType.

5.14.4.3 Random Array

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/Via-Array/ViaLocation/RandomArray
diagram	
type	ViaRandomArrayType, JEP30-D10:PointXYType.

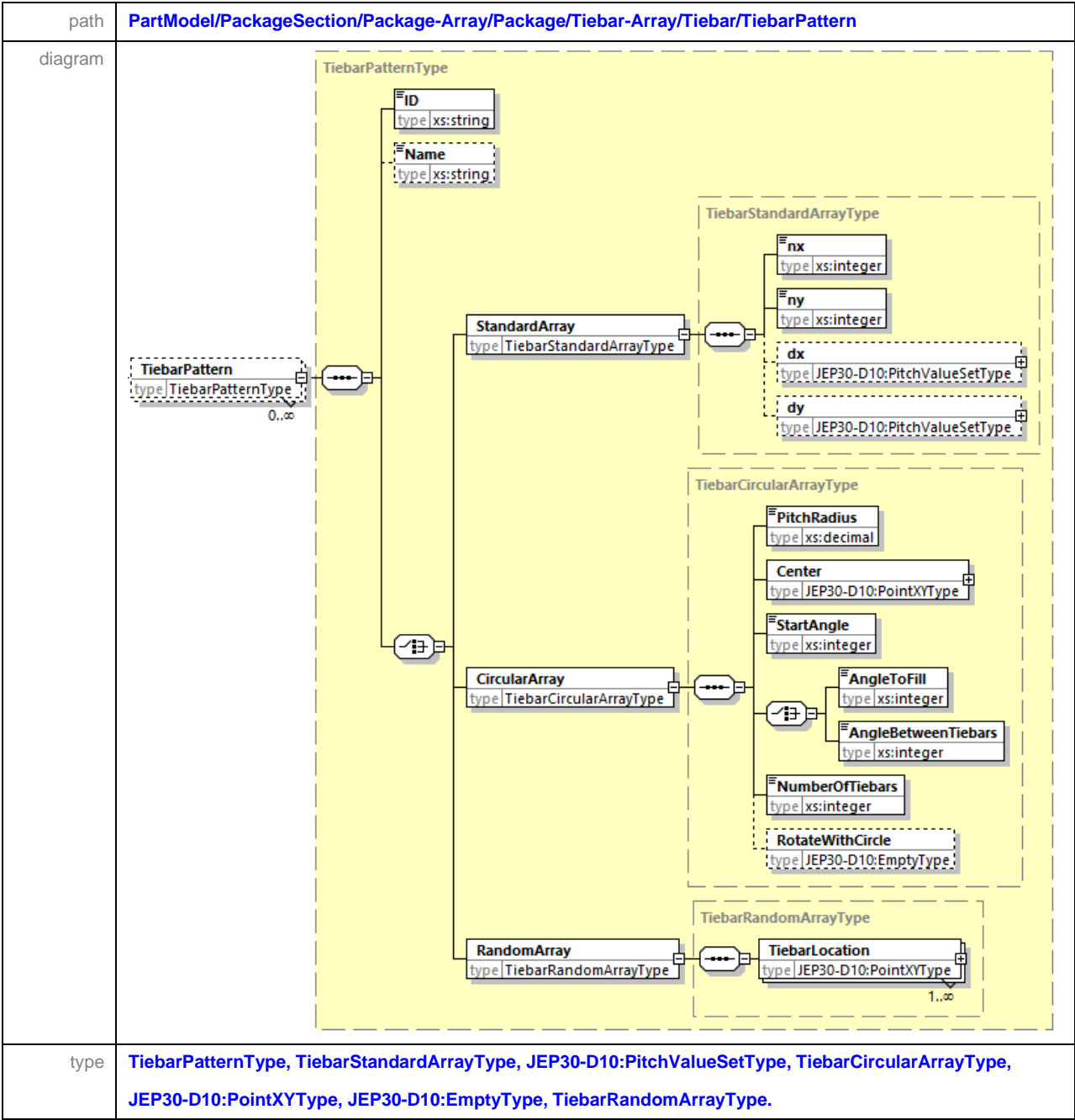
5.14.4.4 Via Detail

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/Via-Array/ViaDetail
diagram	
type	ViaDetailType, ViaCenterType, JEP30-D10:PointXYType, JEP30-D10:MinIntegerOfOneType, ViaStatusType, JEP30-D10:EmptyType.

5.15 Tiebar - Array

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array
diagram	<p>The diagram illustrates the XSD structure for the Tiebar-Array. It shows a sequence of elements: Tiebar-Array (type Tiebar-ArrayType, N, NC), Tiebar (type TiebarType, N, NC), TiebarPattern (type TiebarPatternType, 0..∞), PatternGroup (type TiebarPatternGroupType, 0..∞), TiebarShape-Array (type TiebarShape-ArrayType), and TiebarSelectionToBodyRelationship-Array (type TiebarSelectionToBodyRelationship-ArrayType). The elements are nested within Tiebar-ArrayType and TiebarType. A constraints box is also present.</p>
type	Tiebar-ArrayType, TiebarType, TiebarPatternType, TiebarPatternGroupType, TiebarShape-ArrayType, TiebarSelectionToBodyRelationship-ArrayType.

5.15.1 Tiebar Pattern





5.15.2 Pattern Group

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/PatternGroup
diagram	<p>The diagram illustrates the structure of the TiebarPatternGroupType and its relationships. TiebarPatternGroupType is a container for a collection of TiebarPatternGroupRelationshipType objects (1..∞). TiebarPatternGroupRelationshipType is a container for a collection of TiebarPatternGroupRelationshipTransformationsType objects (1..∞). TiebarPatternGroupRelationshipType also contains attributes for Name, FromTiebarPatternID, and FromPatternGroupID.</p>
type	TiebarPatternGroupType, TiebarPatternGroupRelationshipType, TiebarPatternGroupRelationshipTransformationsType.

5.15.2.1 Relationship Transformations

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/PatternGroup/PatternRelationship/RelationshipTransformations
diagram	<p>The diagram illustrates the structure of the TiebarPatternGroupRelationshipTransformationsType. It shows a 1..∞ relationship from RelationshipTransformations (TiebarPatternGroupRelationshipTransformationsType) to TiebarPatternGroupRelationshipTransformationsType, which contains attributes for ToTiebarPatternID and ToPatternGroupID.</p>
type	TiebarPatternGroupRelationshipTransformationsType.

### 5.15.3 Tiebar Shape - Array

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array	
diagram	<p>The diagram illustrates the XSD structure for TiebarShape-Array. It is an array of TiebarShape elements (0..∞). Each TiebarShape element contains the following components:</p> <ul style="list-style-type: none"> <li><b>ID</b>: type xs:string</li> <li><b>TiebarPatternID</b>: type JEP30-D10:EmptyType</li> <li><b>TiebarPatternGroupID</b>: type JEP30-D10:EmptyType</li> <li><b>ViewPerspective</b>: type ViewPerspectiveType, 1..∞</li> <li><b>ViewPerspective-to-ViewPerspectiveAlignment-Array</b>: type ViewPerspective-to-ViewPerspectiveAlignment-ArrayType</li> <li><b>AssembledShape</b>: type AssembledShapeType</li> </ul>	
type	TiebarShape-ArrayType, TiebarShapeType, JEP30-D10:EmptyType, ViewPerspectiveType, ViewPerspective-to-ViewPerspectiveAlignment-ArrayType, AssembledShapeType.	

5.15.3.1 View Perspective

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective
diagram	<p>The diagram illustrates the XSD structure for <b>ViewPerspectiveType</b>. It is a sequence of elements:</p> <ul style="list-style-type: none"><li><b>ID</b> (type: xs:string)</li><li>A choice of <b>XYPlane</b> (type: JEP30-D10:EmptyType), <b>YZPlane</b> (type: JEP30-D10:EmptyType), or <b>XZPlane</b> (type: JEP30-D10:EmptyType).</li><li>Each plane element is followed by an optional offset: <b>ZOffset</b> (type: xs:decimal), <b>XOffset</b> (type: xs:decimal), or <b>YOffset</b> (type: xs:decimal).</li><li>Each offset is followed by a choice of optional elements: <b>Topside</b> and <b>Underside</b> (type: JEP30-D10:EmptyType) for ZOffset; <b>Left</b> and <b>Right</b> (type: JEP30-D10:EmptyType) for XOffset; <b>Back</b> and <b>Front</b> (type: JEP30-D10:EmptyType) for YOffset.</li><li>A choice of <b>RectangleShape</b> (type: RectangleShapeType), <b>ModifiedRectangleShape</b> (type: ModifiedRectangleShapeType), or <b>ContourShape</b> (type: ContourShapeType).</li></ul> <p>The entire structure is enclosed in a dashed box labeled <b>ViewPerspectiveType</b>.</p>
type	ViewPerspectiveType, JEP30-D10:EmptyType, RectangleShapeType, ModifiedRectangleShapeType, ContourShapeType.

### 5.15.3.1.1 Rectangle Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective/RectangleShape</a>
diagram	
type	<a href="#">RectangleShapeType</a> , <a href="#">JEP30-D10:DimensionalValueSetGroupType</a> .

### 5.15.3.1.2 Modified Rectangle Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective/ModifiedRectangleShape</a>
diagram	
type	<a href="#">ModifiedRectangleShapeType</a> , <a href="#">JEP30-D10:DimensionalValueSetGroupType</a> , <a href="#">ImpactedCornerUnspecifiedValueSetGroupType</a> .

### 5.15.3.1.2.1 Impacted Corner

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective/ModifiedRectangleShape/ImpactedCorner</a>
diagram	
type	<a href="#">ImpactedCornerUnspecifiedValueSetGroupType</a> , <a href="#">ExteriorCornerType</a> , <a href="#">UnspecifiedChamferedCornerType</a> , <a href="#">UnspecifiedRectangularInCornerType</a> , <a href="#">UnspecifiedConvexCornerType</a> .

### 5.15.3.1.2.2 Exterior Corner Type

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective/ModifiedRectangleShape/ImpactedCorner/...</a>
diagram	
type	<a href="#">ExteriorCornerType</a> , <a href="#">UnspecifiedChamferedCornerType</a> , <a href="#">JEP30-D10:UnspecifiedValueSetGroupType</a> , <a href="#">UnspecifiedRectangularInCornerType</a> , <a href="#">UnspecifiedConvexCornerType</a> , <a href="#">JEP30-D10:PointXYType</a> .

### 5.15.3.2 View Perspective – to – View Perspective Alignment - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective-to-ViewPerspectiveAlignment-Array</a>
diagram	
type	<a href="#">ViewPerspective-to-ViewPerspectiveAlignment-ArrayType</a> , <a href="#">ViewPerspective-to-ViewPerspectiveAlignmentType</a> .

#### 5.15.3.2.1 View Perspective – to – View Perspective Alignment

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective-to-ViewPerspectiveAlignment-Array/ViewPerspective-to-ViewPerspectiveAlignment</a>
diagram	
type	<a href="#">ViewPerspective-to-ViewPerspectiveAlignmentType</a> , <a href="#">ViewPerspectiveShapeIDReferenceType</a> .

5.15.3.2.1.1 From View Perspective Shape ID

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/ViewPerspective-to-ViewPerspectiveAlignment-Array/ViewPerspective-to-ViewPerspectiveAlignment/FromViewPerspectiveShapeID
diagram	<p>The diagram illustrates the structure of the <code>FromViewPerspectiveShapeID</code> element. It is defined as a sequence of elements within a dashed box labeled <code>ViewPerspectiveShapeIDReferenceType</code>. The sequence starts with <code>FromViewPerspectiveShapeID</code> (type <code>ViewPerspectiveShapeIDReferenceType</code>), followed by a choice of <code>ViewPerspectiveID</code> (type <code>xs:string</code>) and <code>ViewPerspective-to-ViewPerspectiveAlignmentID</code> (type <code>xs:string</code>). This is followed by a choice of <code>Underside</code> (type <code>JEP30-D10:EmptyType</code>), <code>Topside</code> (type <code>JEP30-D10:EmptyType</code>), <code>ZOffset</code> (type <code>xs:decimal</code>), <code>Back</code> (type <code>JEP30-D10:EmptyType</code>), <code>Front</code> (type <code>JEP30-D10:EmptyType</code>), <code>YOffset</code> (type <code>xs:decimal</code>), <code>Left</code> (type <code>JEP30-D10:EmptyType</code>), <code>Right</code> (type <code>JEP30-D10:EmptyType</code>), and <code>XOffset</code> (type <code>xs:decimal</code>).</p>
type	FromViewPerspectiveShapeIDType, JEP30-D10:EmptyType.

### 5.15.3.3 Assembled Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape</a>
diagram	<p>The diagram illustrates the structure of the <b>AssembledShapeType</b>. It is a complex type containing several elements:</p> <ul style="list-style-type: none"> <li><b>ViewPerspective-to-ViewPerspectiveAlignmentID</b>: A string element.</li> <li><b>ImpactedTiebar</b>: A complex type containing: <ul style="list-style-type: none"> <li><b>TiebarCenter</b>: A <b>TiebarCenterType</b>.</li> <li><b>RowTiebarIndex</b>: A <b>JEP30-D10:MinIntegerOfOneType</b> with minIncl/maxIncl of 1.</li> <li><b>ColumnTiebarIndex</b>: A <b>JEP30-D10:MinIntegerOfOneType</b> with minIncl/maxIncl of 1.</li> <li><b>PolarTiebarIndex</b>: A <b>JEP30-D10:MinIntegerOfOneType</b> with minIncl/maxIncl of 1.</li> </ul> </li> <li><b>Impact-to-TiebarGroup</b>: A complex type containing: <ul style="list-style-type: none"> <li><b>Apply-to-all-Tiebars</b>: An <b>Apply-to-all-TiebarsType</b>.</li> <li><b>Symmetry</b>: A <b>TiebarSymmetryType</b>.</li> </ul> </li> </ul> <p>The <b>AssembledShape</b> type is defined as <b>AssembledShapeType</b>.</p>
type	<b>TiebarPatternGroupRelationshipTransformationsType</b> .

#### 5.15.3.3.1 Tiebar Center

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/ImpactedTiebar/TiebarCenter</a>
diagram	<p>The diagram illustrates the structure of the <b>TiebarCenterType</b>. It is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>x</b>: A decimal element.</li> <li><b>y</b>: A decimal element.</li> <li><b>PitchRadius</b>: A decimal element.</li> <li><b>Center</b>: A <b>JEP30-D10:PointXYType</b>.</li> <li><b>Angle</b>: An integer element.</li> </ul> <p>The <b>TiebarCenter</b> type is defined as <b>TiebarCenterType</b>.</p>
type	<b>TiebarCenterType</b> , <b>JEP30-D10:PointXYType</b> .



### 5.15.3.3.2 Apply – to – all - Tiebars

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/Impact-to-TiebarGroup/Apply-to-all-Tiebars</a>
diagram	
type	<a href="#">Apply-to-all-TiebarsType</a> , <a href="#">JEP30-D10:CornerImpact-to-StandardArrayType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.15.3.3.3 Symmetry

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/Impact-to-TiebarGroup/Symmetry</a>
diagram	
type	<a href="#">TiebarSymmetryType</a> , <a href="#">TiebarSymmetryRotationType</a> , <a href="#">SymmetryRotationAxisType</a> , <a href="#">TiebarSymmetryRotationCenterType</a> , <a href="#">TiebarReflectionType</a> , <a href="#">ReflectionAxisType</a> , <a href="#">TiebarReflectionInversionCenterType</a> .

### 5.15.3.3.3.1 Center

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/Impact-to-TiebarGroup/Symmetry/Rotation/Center</a>
diagram	
type	<a href="#">TiebarSymmetryRotationCenterType</a> , <a href="#">JEP30-D10:EmptyType</a> .

### 5.15.3.3.3.2 Mirror Plane

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/Impact-to-TiebarGroup/Symmetry/Reflection/MirrorPlane</a>
diagram	
type	<a href="#">ReflectionAxisType</a> , <a href="#">JEP30-D10:EmptyType</a> .

### 5.15.3.3.3.3 Inversion Center

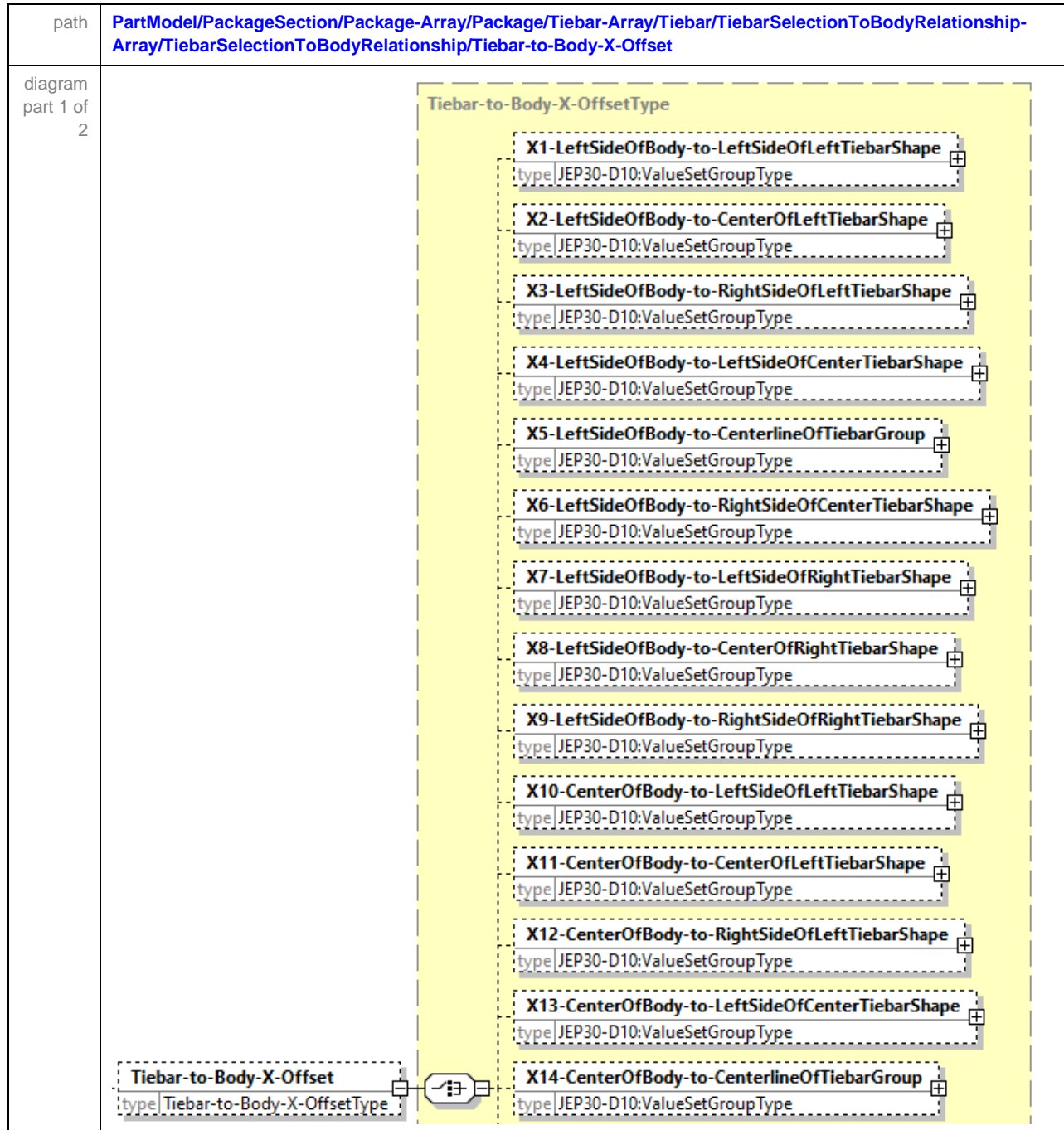
path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarShape-Array/TiebarShape/AssembledShape/Impact-to-TiebarGroup/Symmetry/Reflection/InversionCenter</a>
diagram	
type	<a href="#">TiebarReflectionInversionCenterType</a> , <a href="#">JEP30-D10:EmptyType</a> .

5.15.4 Tiebar Selection To Body Relationship - Array

path	PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarSelectionToBodyRelationship-Array
diagram	
type	TiebarSelectionToBodyRelationship-ArrayType, TiebarSelectionToBodyRelationshipType, Tiebar-to-Body-X-OffsetType, Tiebar-to-Body-Y-OffsetType, Tiebar-to-Body-Z-OffsetType.

For non-symmetrical Tiebars around the package center, dimensions may be referenced from the Tiebar shape to the Package body area. This data can be captured in the *TiebarSelectionToBodyRelationship* section.

#### 5.15.4.1 Tiebar Pattern or Tiebar Pattern Group to Body X-Offset



#### 5.15.4.1 Tiebar Pattern or Tiebar Pattern Group to Body X-Offset (cont'd)

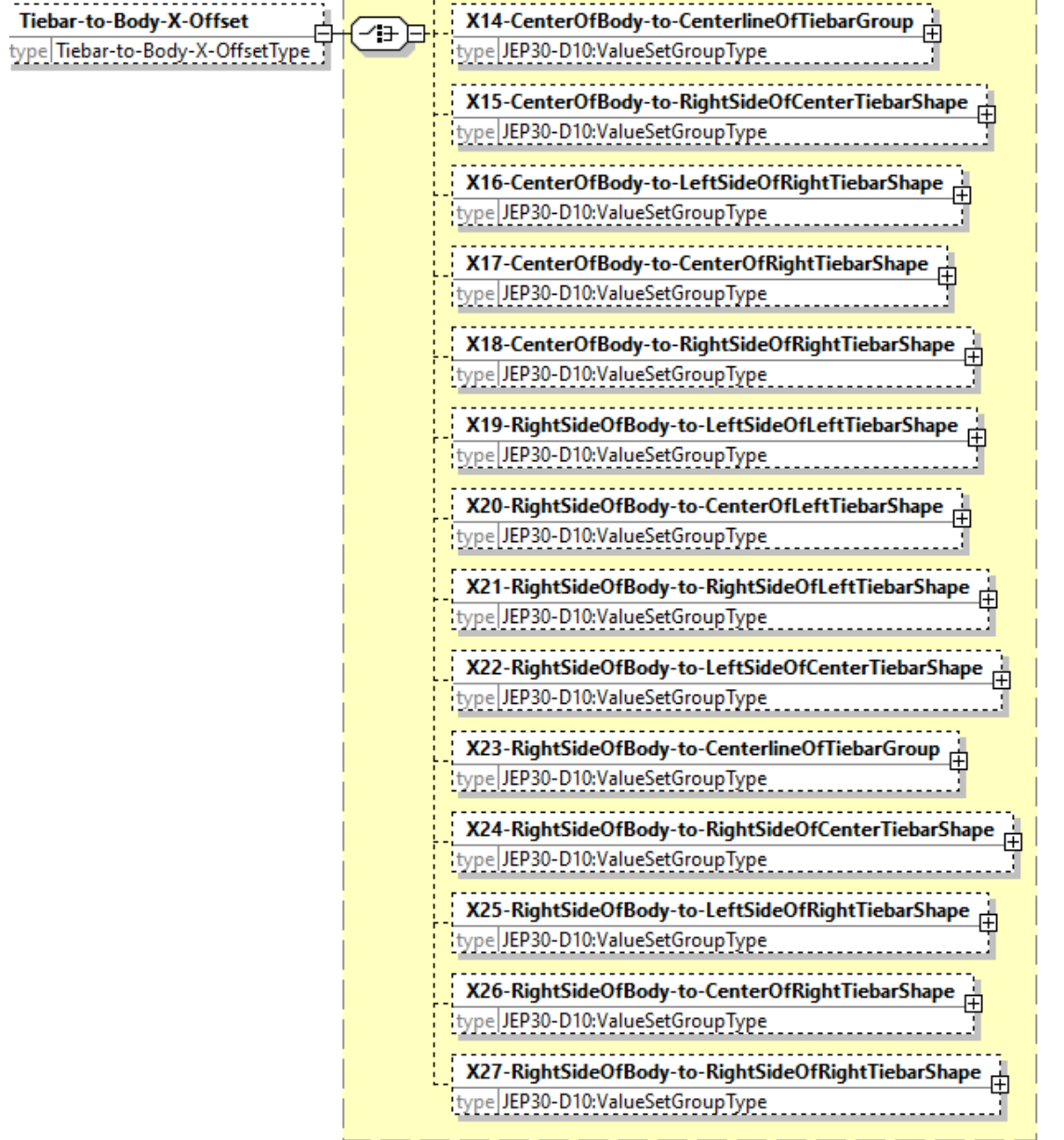
<p>diagram part 2 of 2</p>	 <p><b>Tiebar-to-Body-X-Offset</b> type Tiebar-to-Body-X-OffsetType</p> <p><b>X14-CenterOfBody-to-CenterlineOfTiebarGroup</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X15-CenterOfBody-to-RightSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X16-CenterOfBody-to-LeftSideOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X17-CenterOfBody-to-CenterOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X18-CenterOfBody-to-RightSideOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X19-RightSideOfBody-to-LeftSideOfLeftTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X20-RightSideOfBody-to-CenterOfLeftTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X21-RightSideOfBody-to-RightSideOfLeftTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X22-RightSideOfBody-to-LeftSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X23-RightSideOfBody-to-CenterlineOfTiebarGroup</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X24-RightSideOfBody-to-RightSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X25-RightSideOfBody-to-LeftSideOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X26-RightSideOfBody-to-CenterOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>X27-RightSideOfBody-to-RightSideOfRightTiebarShape</b> type JEP30-D10:ValueSetGroupType</p>
<p>type</p>	<p><b>Tiebar-to-Body-X-OffsetType, JEP30-D10:ValueSetGroupType.</b></p>

Table 32 - Tiebar to Package Body X-Offset lists out all the various dimensions that can be selected from any major point of the package body to any point of the tiebar shape in X direction.

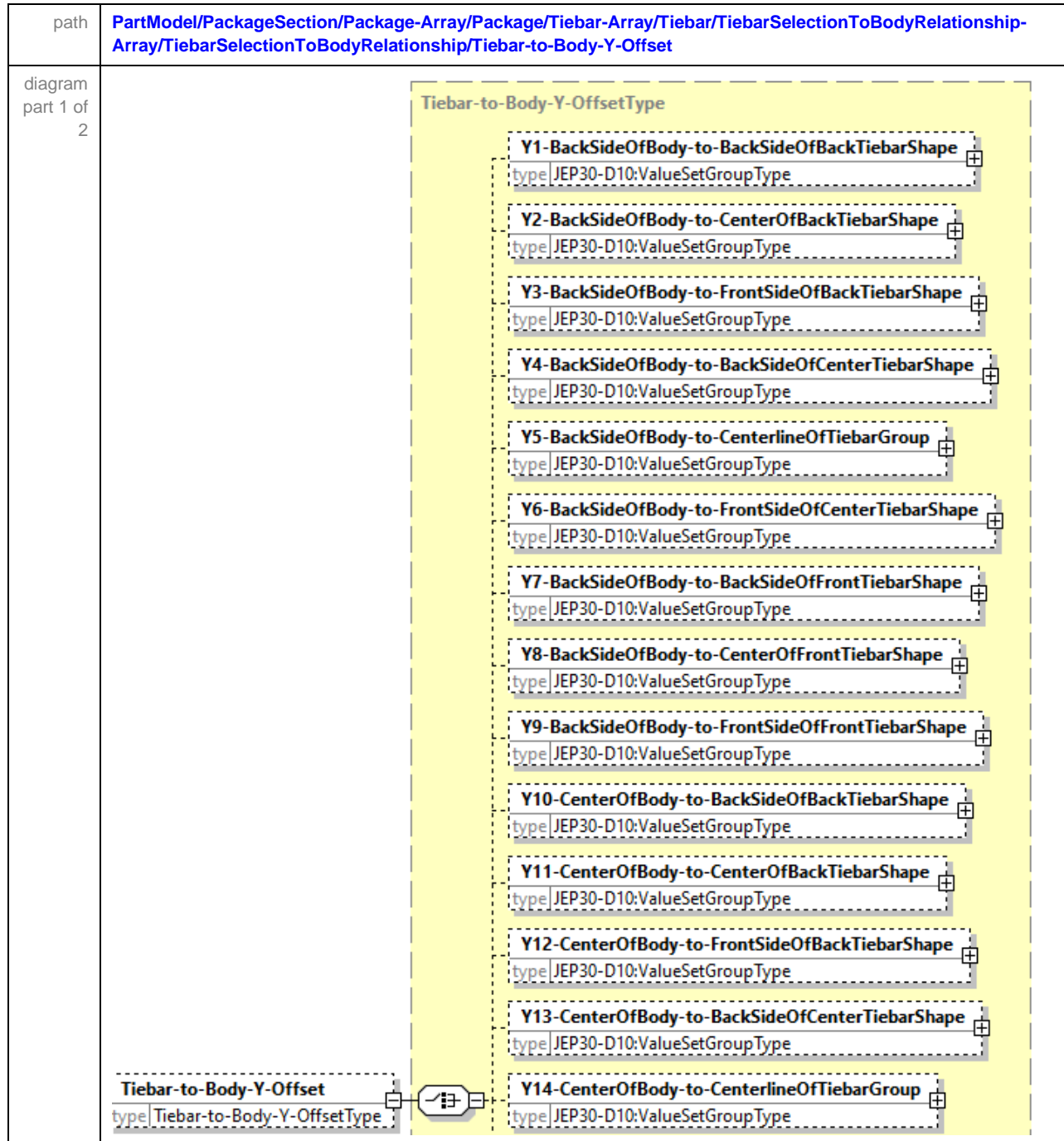
Center of Center Tiebar Group is the same as Centerline of Tiebar Group, however the use of the "...Centerline of Tiebar Group" is more encompassing since it also includes the centerline of a row or column of an even number of tiebars, i.e., midway between tiebar 2 and tiebar 3 of 4 tiebars in a row or column.

#### 5.15.4.1 Tiebar Pattern or Tiebar Pattern Group to Body X-Offset (cont'd)

**Table 32 - Tiebar to Package Body X-Offset**

Symbol	Description
X1	Left side of body to Left Side of Left Tiebar Shape
X2	Left side of body to Center of Left Tiebar Shape
X3	Left side of body to Right side of Left Tiebar Shape
X4	Left side of body to Left Side of Center Tiebar Shape
X5	Left side of body to Centerline of Tiebar Group
X6	Left side of body to Right side of Center Tiebar Shape
X7	Left side of body to Left side of Right Tiebar Shape
X8	Left side of body to Center of Right Tiebar Shape
X9	Left side of body to Right side of Right Tiebar Shape
X10	Center of body to Left side of Left Tiebar Shape
X11	Center of body to Center of Left Tiebar Shape
X12	Center of body to Right side of Left Tiebar Shape
X13	Center of body to Left side of Center Tiebar Shape
X14	Center of body to Centerline of Tiebar Group
X15	Center of body to Right side of Center Tiebar Shape
X16	Center of body to Left side of Right Tiebar Shape
X17	Center of body to Center of Right Tiebar Shape
X18	Center of body to Right side of Right Tiebar Shape
X19	Right side of body to Left side of Left Tiebar Shape
X20	Right side of body to Center of Left Tiebar Shape
X21	Right side of body to Right side of Left Tiebar Shape
X22	Right side of body to Left side of Center Tiebar Shape
X23	Right side of body to Centerline of Tiebar Group
X24	Right side of body to Right side of Center Tiebar Shape
X25	Right side of body to Left side of Right Tiebar Shape
X26	Right side of body to Center of Right Tiebar Shape
X27	Right side of body to Right side of Right Tiebar Shape

#### 5.15.4.2 Tiebar Pattern or Tiebar Pattern Group to Body Y-Offset



#### 5.15.4.2 Tiebar Pattern or Tiebar Pattern Group to Body Y-Offset (cont'd)

<p>diagram part 2 of 2</p>	 <p><b>Tiebar-to-Body-Y-Offset</b> type Tiebar-to-Body-Y-OffsetType</p> <p><b>Y14-CenterOfBody-to-CenterlineOfTiebarGroup</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y15-CenterOfBody-to-FrontSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y16-CenterOfBody-to-BackSideOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y17-CenterOfBody-to-CenterOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y18-CenterOfBody-to-FrontSideOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y19-FrontSideOfBody-to-BackSideOfBackTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y20-FrontSideOfBody-to-CenterOfBackTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y21-FrontSideOfBody-to-FrontSideOfBackTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y22-FrontSideOfBody-to-BackSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y23-FrontSideOfBody-to-CenterlineOfTiebarGroup</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y24-FrontSideOfBody-to-FrontSideOfCenterTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y25-FrontSideOfBody-to-BackSideOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y26-FrontSideOfBody-to-CenterOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p> <p><b>Y27-FrontSideOfBody-to-FrontSideOfFrontTiebarShape</b> type JEP30-D10:ValueSetGroupType</p>
<p>type</p>	<p><b>Tiebar-to-Body-Y-OffsetType, JEP30-D10:ValueSetGroupType.</b></p>

Table 33 - Tiebar to Package Body Y-Offset lists out all the various dimensions that can be selected from any major point of the package body to any point of the tiebar shape in Y direction.



#### 5.15.4.2 Tiebar Pattern or Tiebar Pattern Group to Body Y-Offset (cont'd)

**Table 33 - Tiebar to Package Body Y-Offset**

Symbol	Description
Y1	Back side of body to Back side of Back Tiebar Shape
Y2	Back side of body to Center of Back Tiebar Shape
Y3	Back side of body to Front side of Back Tiebar Shape
Y4	Back side of body to Back side of Center Tiebar Shape
Y5	Back side of body to Centerline of Tiebar Group
Y6	Back side of body to Front side of Center Tiebar Shape
Y7	Back side of body to Back side of Front Tiebar Shape
Y8	Back side of body to Center of Front Tiebar Shape
Y9	Back side of body to Front side of Front Tiebar Shape
Y10	Center of body to Back side of Back Tiebar Shape
Y11	Center of body to Center of Back Tiebar Shape
Y12	Center of body to Front side of Back Tiebar Shape
Y13	Center of body to Back side of Center Tiebar Shape
Y14	Center of body to Centerline of Tiebar Group
Y15	Center of body to Front side of Center Tiebar Shape
Y16	Center of body to Back side of Front Tiebar Shape
Y17	Center of body to Center of Front Tiebar Shape
Y18	Center of body to Front side of Front Tiebar Shape
Y19	Front side of body to Back side of Back Tiebar Shape
Y20	Front side of body to Center of Back Tiebar Shape
Y21	Front side of body to Front side of Back Tiebar Shape
Y22	Front side of body to Back side of Center Tiebar Shape
Y23	Front side of body to Centerline of Tiebar Group
Y24	Front side of body to Front side of Center Tiebar Shape
Y25	Front side of body to Back side of Front Tiebar Shape
Y26	Front side of body to Center of Front Tiebar Shape
Y27	Front side of body to Front side of Front Tiebar Shape

### 5.15.4.3 Tiebar Pattern or Tiebar Pattern Group to Body Z-Offset

path	<a href="#">PartModel/PackageSection/Package-Array/Package/Tiebar-Array/Tiebar/TiebarSelectionToBodyRelationship-Array/TiebarSelectionToBodyRelationship/Tiebar-to-Body-Z-Offset</a>
diagram	
type	<a href="#">Tiebar-to-Body-Z-OffsetType</a> , <a href="#">JEP30-D10:ValueSetGroupType</a> .

Table 34 lists out all the various dimensions that can be selected from any major point of the package body to any point of the terminal contact area in Z direction.

**Table 34 - Terminal Group to Package Body Z-Offset**

Symbol	Description
Z1	Seating Plane to Lower side of Tiebar Shape
Z2	Seating Plane to Center of Tiebar Shape
Z3	Seating Plane to Upper side of Tiebar Shape
Z4	Under side of Body to Lower side of Tiebar Shape
Z5	Under side of Body to Center of Tiebar Shape
Z6	Under side of Body to Upper side of Tiebar Shape
Z7	Upper side of Body to Lower side of Tiebar Shape
Z8	Upper side of Body to Center of Tiebar Shape
Z9	Upper side of Body to Upper side of Tiebar Shape

## 5.16 Fiducial Marking

path	<a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking</a>
diagram	
type	<a href="#">FiducialMarkingType</a> , <a href="#">FiducialShape-ArrayType</a> , <a href="#">FiducialShapeType</a> , <a href="#">JEP30-D10:ShapeOrderType</a> , <a href="#">JEP30-D10:GraphicalFormat-ArrayType</a>

A fiducial marker or fiducial is an object placed in the field of view of an imaging system that appears in the image produced, for use as a point of reference or a measure. It may be either something placed into or on the imaging subject, or a mark or set of marks in the reticle of an optical instrument.

### 5.16.1 Fiducial Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape</a>
diagram	
type	<a href="#">FiducialShapeType</a> , <a href="#">FiducialVertex-ArrayType</a> , <a href="#">PrimitiveFiducialShapeType</a> , <a href="#">JEP30-D10:GraphicalFormat-ArrayType</a>

### 5.14.1 Fiducial Shape (cont'd)

The Shape can be created from a choice of Vertices, or primitive shapes.

*GraphicalFormatID* serves as a reference ID for the *GraphicalFormat*, that is defined under the *GraphicalFormat-Array/GraphicalFormats* where a set of graphical formats can be defined that can be standardized across multiple *Shape* entries. The addition of the *GraphicalFormat* under *Shape* enables unique modification of a references Graphical Format for applying to this specific shape instance.

The *ShapeOrderSequence* can be used to define the order of shapes for a complex fiducial that may consist of various shapes. It works in tandem with the *ShapeOrder* that is defined under *FiducialMarking/FiducialShape-Array*.

### 5.16.1.1 Vertex - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/Vertex-Array</a>
diagram	
type	<a href="#">FiducialVertex-ArrayType</a> , <a href="#">FiducialVertexType</a> , <a href="#">JEP30-D10:FeatureControl-IDsType</a> , <a href="#">FiducialEdgeVertexControlType</a>

5.16.1.1.1 Edge

path	PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/Vertex-Array/Edge
diagram	<p>The diagram illustrates the structure of the <b>FiducialEdgeVertexControlType</b> and its relationship to other types. It is divided into two main sections: <b>FiducialEdgeVertexControlType</b> (left) and <b>FiducialLineVertexControlType</b> (right).</p> <p><b>FiducialEdgeVertexControlType</b> contains:</p> <ul style="list-style-type: none"><li><b>Edge</b> (type: <b>FiducialEdgeVertexControlType</b>): A dashed box containing the text "Default should be assumed to be line strip if this section is omitted".</li><li><b>Line</b> (type: <b>FiducialLineVertexControlType</b>): A box containing the text "Pairs of vertices interpreted as individual line segments".</li><li><b>Triangle</b> (type: <b>JEP30-D10:EmptyType</b>): A box containing the text "Same as strip, with a segment added between last and first vertices".</li></ul> <p><b>FiducialLineVertexControlType</b> contains:</p> <ul style="list-style-type: none"><li><b>Lines</b> (type: <b>JEP30-D10:EmptyType</b>): A box containing the text "Pairs of vertices interpreted as individual line segments".</li><li><b>Strip</b> (type: <b>JEP30-D10:EmptyType</b>): A box containing the text "Series of connected line segments".</li><li><b>Loop</b> (type: <b>JEP30-D10:EmptyType</b>): A box containing the text "Same as strip, with a segment added between last and first vertices".</li></ul> <p>Relationships are shown with arrows and connectors. A dashed line connects the <b>Edge</b> box to the <b>Line</b> box. A solid line connects the <b>Line</b> box to the <b>Triangle</b> box. A solid line connects the <b>Line</b> box to the <b>Lines</b> box. A solid line connects the <b>Line</b> box to the <b>Strip</b> box. A solid line connects the <b>Line</b> box to the <b>Loop</b> box.</p>
type	FiducialEdgeVertexControlType, FiducialLineVertexControlType, JEP30-D10:EmptyType

The default is assumed to be *Line/Strip* if the *Edge* element is omitted. The *Edge* structure governs how each of the vertices in the unbounded *Vertex* element is to be processed.

*Line/Lines* represent pairs of vertices interpreted as individual line segments. Vertices 1 and 2 make one line segment whereas vertices 3 and 4 makes the 2<sup>nd</sup> line segment. Vertices 2 and 3 are not connected with a line segment.

*Line/Strip* represent a series of connected line segments. Vertices 1 and 2 make one line segment whereas vertices 2 and 3 makes the 2<sup>nd</sup> line segment, and vertices 3 and 4 makes the 3<sup>rd</sup> line segment.

*Lines/Loop* is the same as strip, with a segment added between last and first vertices. From the previous example, vertices 4 and 1 makes the final line segment, creation a closed loop.

*Triangle* represent each triple set of vertices to be interpreted as an individual triangle. Vertices 1, 2, and 3 make one triangle whereas vertices 4, 5, and 6 makes the 2<sup>nd</sup> triangle.

### 5.16.1.2 Primitive Fiducial Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/PrimitiveFiducialShape</a>
diagram	
type	<a href="#">PrimitiveFiducialShapeType</a> , <a href="#">ReferenceRectangleType</a> , <a href="#">ReferenceCrosshairType</a> , <a href="#">JEP30-D10:DimensionalValueSetType</a> , <a href="#">ReferenceRegularPolygonType</a> , <a href="#">ReferenceCircleType</a> , <a href="#">JEP30-D10:PointXYType</a>

### 5.16.2 Graphical Format - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/GraphicalFormat-Array</a>
diagram	
type	<a href="#">JEP30-D10:GraphicalFormat-ArrayType</a> , <a href="#">GraphicalFormatsType</a> , <a href="#">GraphicalFormatType</a> .

5.16.2.1 Graphical Format

path	PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/GraphicalFormat
diagram	
type	GraphicalFormatType, StrokeWidthType, StrokeOpacityType, StrokeLineCapType, StrokeDash-ArrayType, ColorType, FillOpacityType

### 5.16.2.1.1 Color Type

path	<p><a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/GraphicalFormat/StrokeColor</a></p> <p><a href="#">PartModel/PackageSection/Package-Array/Package/FiducialMarking/FiducialShape-Array/FiducialShape/GraphicalFormat/FillColor</a></p>
diagram	
type	<p><a href="#">DefaultColorType</a>, <a href="#">DefaultColorNameType</a>, <a href="#">DefaultColorHexType</a>, <a href="#">DefaultColor-RType</a>, <a href="#">DefaultColor-GType</a>, <a href="#">DefaultColor-BType</a></p>

[ColorNameType](#) as defined in the [StrokeColor/Name](#) element list the names of the Scalable Vector Graphics (SVG) Colors as defined by the Scalable Vector Graphics (SVG) Specification.

[ColorHexType](#) as defined in the [StrokeColor/Hex](#) element uses a regular expression pattern to recognize the six-digit hexadecimal representation of the Scalable Vector Graphics (SVG) Colors.

[Color-RType](#), [Color-GType](#), [Color-BType](#) allows the construction of all the colors from the combination of the red, green, and blue colors in the RGB Color Space. The red, green, and blue use 8 bits each, which have integer values from 0 to 255.



## 5.17 Geometric Dimensioning and Tolerancing

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT</a> , <a href="#">PartModel/PackageSection/Package-Array/Package/GDandTDatum-to-ElementMap</a>
diagram	<pre> classDiagram     class JEP30-D10:GDAndTType {         +Datum-Array         +FeatureControl-Array         +FeatureControlPlacement-Array     }     class Datum-Array {         +type Datum-ArrayType     }     class FeatureControl-Array {         +type FeatureControl-ArrayType     }     class FeatureControlPlacement-Array {         +type FeatureControlPlacement-ArrayType     }     class GDandTDatum-to-ElementMap-ArrayType {         +GDandTDatum-to-ElementMap     }     class GDandTDatum-to-ElementMap {         +type GDandTDatum-to-ElementMapType     }     JEP30-D10:GDAndTType --&gt; Datum-Array     JEP30-D10:GDAndTType --&gt; FeatureControl-Array     JEP30-D10:GDAndTType --&gt; FeatureControlPlacement-Array     GDandTDatum-to-ElementMap-ArrayType --&gt; GDandTDatum-to-ElementMap     </pre>
type	<a href="#">JEP30-D10:GDAndTType</a> , <a href="#">Datum-ArrayType</a> , <a href="#">FeatureControl-ArrayType</a> , <a href="#">FeatureControlPlacement-ArrayType</a>

When applying GD&T, the first consideration is to establish a datum reference frame based on the function of the part in the assembly with its mating parts. After the datum reference plane is established, the form of the primary datum features is controlled, followed by the orientation and / or location of the secondary and tertiary datum features. After the datum features are related relative to each other, the remaining features are controlled for orientation and location relative to the datum reference framework.

As a result of increased decentralization of design and manufacturing, it is even more important that the design more precisely states the functional requirements. To accomplish this, it is becoming increasingly more important that the use of geometric and dimensioning replace the former limit dimensioning for form, orientation, location, and profile of part features. This section follows the ASME Y14.5-2009 Dimensioning and Tolerancing which establishes uniform practices for stating and interpreting dimensioning, Tolerancing and related requirements for use on engineering drawings and in related documents.

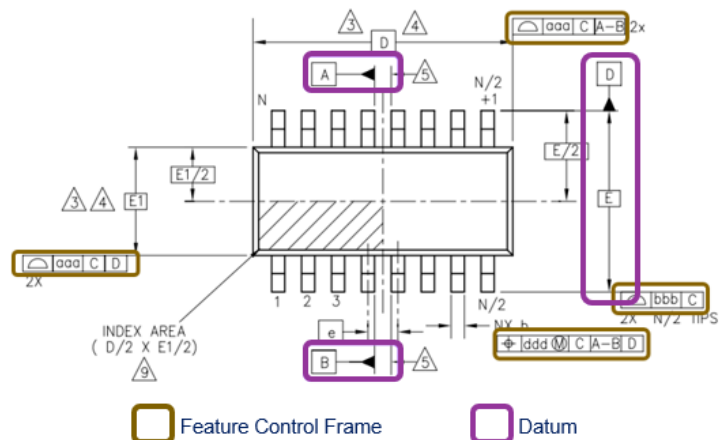
Refer to this ASME Y14.5-2009 Dimensioning and Tolerancing standard for a detailed explanation of the terminology used in association with this section.

### 5.17.1 Datum Array

path	PartModel/PackageSection/Package-Array/Package/GDAndT/Datum-Array
diagram	
type	Datum-ArrayType, DatumType, EmptyType.

A datum feature is selected based on its functional relationship to the tolerance feature and the requirements of the design.

**Figure 16 – Example showing Datum and Feature Control Frames**



### 5.15.1 Datum Array (cont'd)

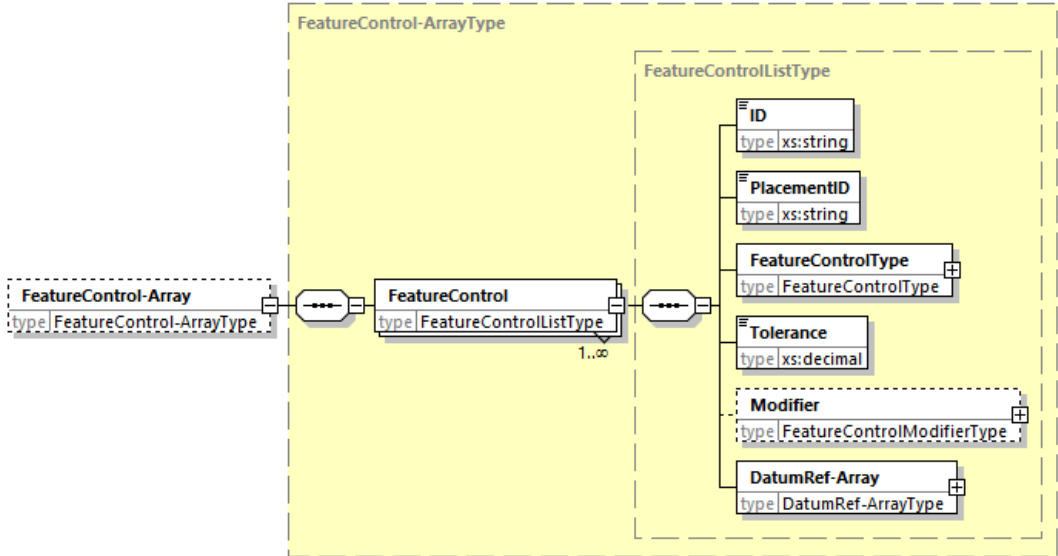
The diagram below from the ASME Y14.5-2009 Dimensioning and Tolerancing standard for a detailed explanation of the terminology used in association with this section.

FEATURE TYPE	ON THE DRAWING	DATUM FEATURE	DATUM AND DATUM FEATURE SIMULATOR	DATUM AND CONSTRAINING DEGREES OF FREEDOM
PLANAR (a)				
WIDTH (b)				
SPHERICAL (c)				
CYLINDRICAL (d)				
CONICAL (e)				
LINEAR EXTRUDED SHAPE (f)				
COMPLEX (g)				

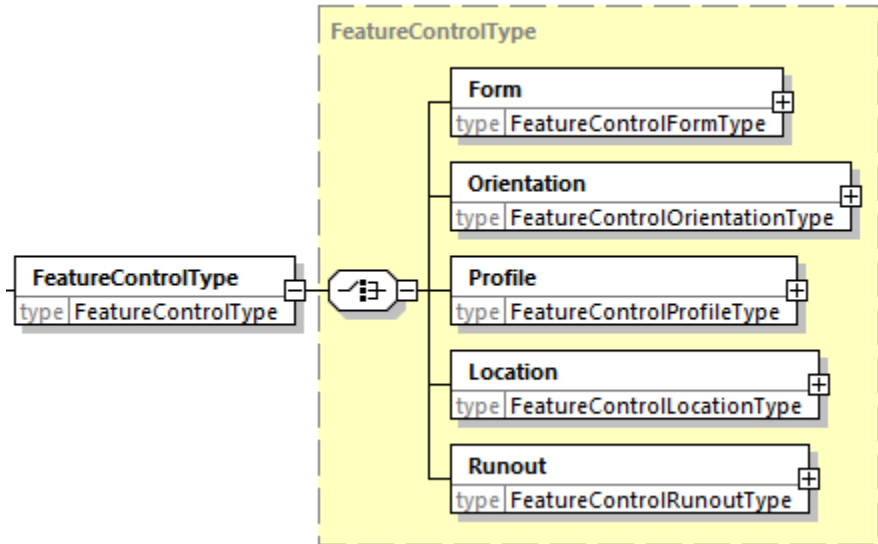
Source: ASME Y14-5 2009

Figure 17 – Constrained Degrees of Freedom for Primary Datum Features

## 5.17.2 Feature Control Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array</a>
diagram	 <p>The diagram illustrates the structure of the <b>FeatureControl-ArrayType</b>. It is a sequence of <b>FeatureControl</b> elements. Each <b>FeatureControl</b> element is of type <b>FeatureControlListType</b> and has a cardinality of 1..∞. The <b>FeatureControlListType</b> is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b>: type <b>xs:string</b></li> <li><b>PlacementID</b>: type <b>xs:string</b></li> <li><b>FeatureControlType</b>: type <b>FeatureControlType</b> (indicated with a '+' sign for one or more)</li> <li><b>Tolerance</b>: type <b>xs:decimal</b></li> <li><b>Modifier</b>: type <b>FeatureControlModifierType</b> (indicated with a '+' sign for one or more)</li> <li><b>DatumRef-Array</b>: type <b>DatumRef-ArrayType</b> (indicated with a '+' sign for one or more)</li> </ul>
type	<a href="#">FeatureControl-ArrayType</a> , <a href="#">FeatureControlListType</a> , <a href="#">FeatureControlType</a> , <a href="#">FeatureControlModifierType</a> , <a href="#">DatumRef-ArrayType</a> .

### 5.17.2.1 Feature Control Type

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType</a>
diagram	 <p>The diagram illustrates the structure of the <b>FeatureControlType</b>. It is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>Form</b>: type <b>FeatureControlFormType</b> (indicated with a '+' sign for one or more)</li> <li><b>Orientation</b>: type <b>FeatureControlOrientationType</b> (indicated with a '+' sign for one or more)</li> <li><b>Profile</b>: type <b>FeatureControlProfileType</b> (indicated with a '+' sign for one or more)</li> <li><b>Location</b>: type <b>FeatureControlLocationType</b> (indicated with a '+' sign for one or more)</li> <li><b>Runout</b>: type <b>FeatureControlRunoutType</b> (indicated with a '+' sign for one or more)</li> </ul>
type	<a href="#">FeatureControlTypeType</a> , <a href="#">FeatureControlFormType</a> , <a href="#">FeatureControlOrientationType</a> , <a href="#">FeatureControlProfileType</a> , <a href="#">FeatureControlLocationType</a> , <a href="#">FeatureControlRunoutType</a> .

### 5.17.2.1.1 Form

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType/Form</a>
diagram	
type	<a href="#">FeatureControlFormType</a> , <a href="#">EmptyType</a> .

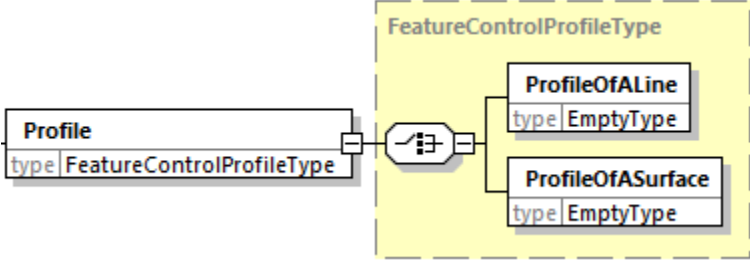
Form tolerances control straightness, flatness, circularity, and cylindricity. See section on “Tolerance of Form” in the ASME Y14-5 2009 for more details.

### 5.17.2.1.2 Orientation

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType/Orientation</a>
diagram	
type	<a href="#">FeatureControlOrientationType</a> , <a href="#">EmptyType</a> .

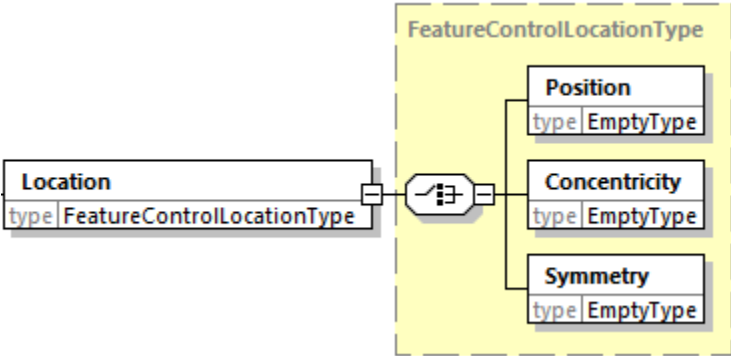
An orientation tolerance controls parallel, perpendicular, and all other angular relationships. See section on “Tolerance of Orientation” in the ASME Y14-5 2009 for more details.

### 5.17.2.1.3 Profile

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType/Profile</a>
diagram	 <p>The diagram shows a box labeled 'Profile' with 'type FeatureControlProfileType' below it. This box is connected to a dashed yellow box labeled 'FeatureControlProfileType'. Inside this dashed box, there is a central icon (a circle with a cross) connected to two sub-boxes: 'ProfileOfALine' and 'ProfileOfASurface'. Both sub-boxes have 'type EmptyType' below them.</p>
type	<a href="#">FeatureControlProfileType</a> , <a href="#">EmptyType</a> .

A profile is an outline of a surface, a shape made up of one or more features, or a two-dimensional element of one or more features. Profile tolerances are used to define a tolerance zone to control form or combinations of size, form, orientation, and location of a feature(s) relative to a true profile. See section on “Tolerance of Profile” in the ASME Y14-5 2009 for more details.

### 5.17.2.1.4 Location

path	<a href="#">PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType/Location</a>
diagram	 <p>The diagram shows a box labeled 'Location' with 'type FeatureControlLocationType' below it. This box is connected to a dashed yellow box labeled 'FeatureControlLocationType'. Inside this dashed box, there is a central icon (a circle with a cross) connected to three sub-boxes: 'Position', 'Concentricity', and 'Symmetry'. Each sub-box has 'type EmptyType' below it.</p>
type	<a href="#">FeatureControlLocationType</a> , <a href="#">EmptyType</a> .

Included in the principles of tolerances of location. are position, concentricity, and symmetry used to control the following relationships:

1. Center distance between features of size such as holes, slots, bosses, and tabs.
2. Location of features of size (such as in the previous bullet) as a group, from datum features, such as plane and cylindrical surfaces.
3. Coaxiality of features of size, and
4. Concentricity or symmetry of features of size—center distances of correspondingly located feature elements equally disposed about a datum axis or plane

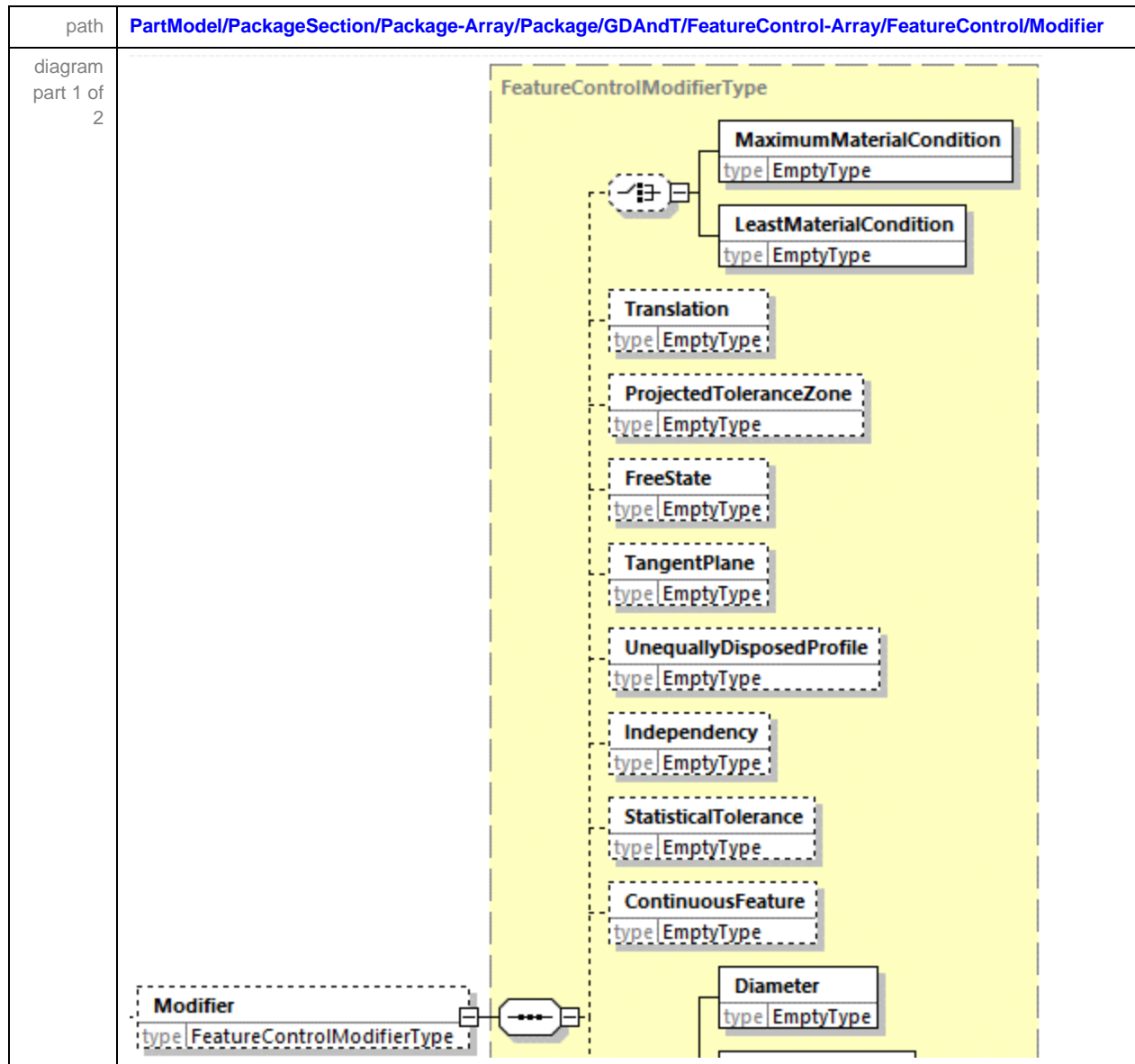
See section on “Tolerance of Location” in the ASME Y14-5 2009 for more details.

5.17.2.1.5     Runout

path	PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/FeatureControlType/Runout
diagram	<pre>classDiagram     class Runout {         type FeatureControlRunoutType     }     class FeatureControlRunoutType {         Runout         TotalRunout     }     Runout --&gt; FeatureControlRunoutType     FeatureControlRunoutType -- &gt; Runout     FeatureControlRunoutType -- &gt; TotalRunout     class RunoutSub {         type EmptyType     }     class TotalRunoutSub {         type EmptyType     }</pre>
type	FeatureControlRunoutType, EmptyType.

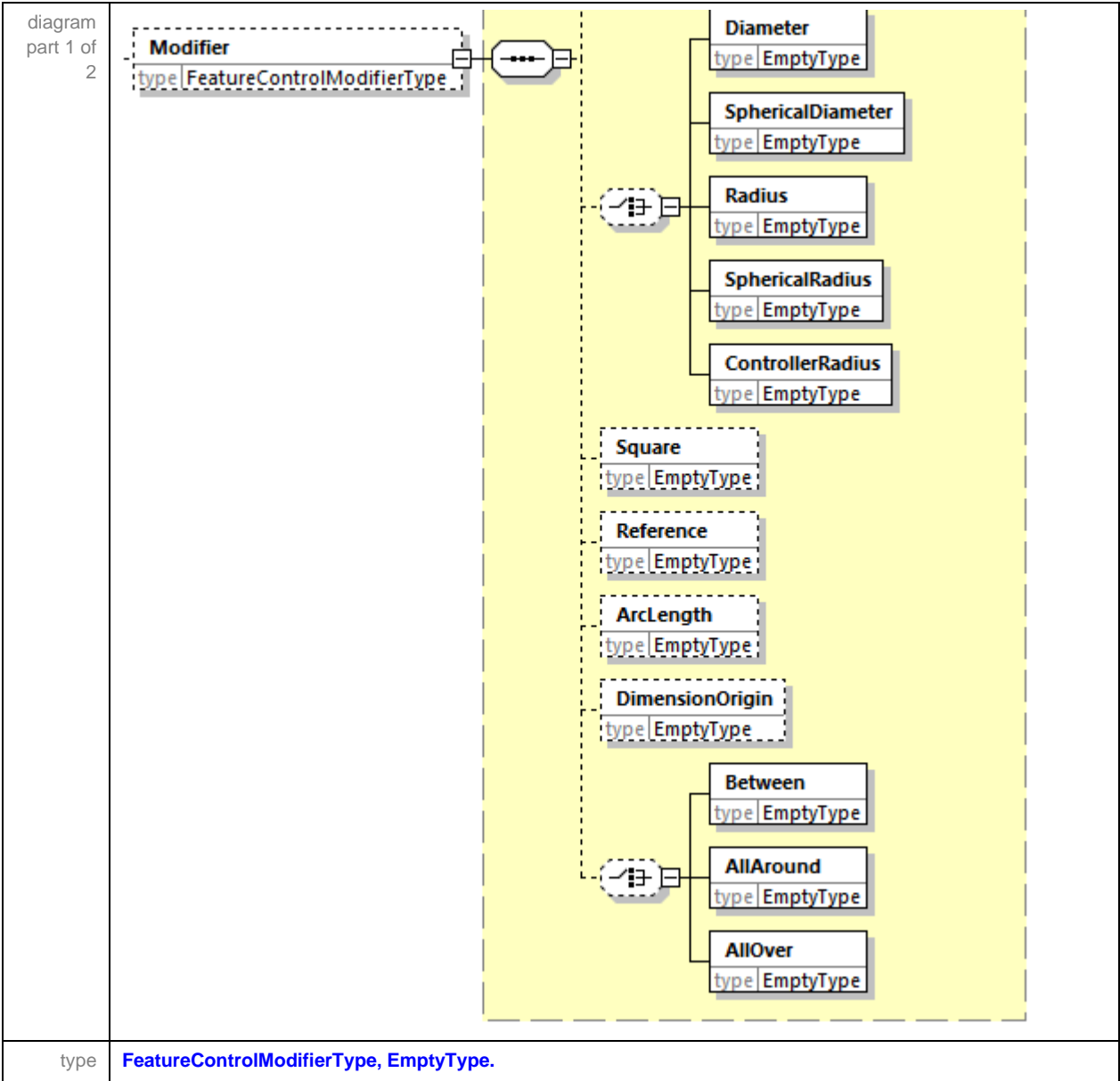
Runout is a tolerance used to control the functional relationship of one or more features to a datum axis established from a datum feature specified at RMB. The types of features controlled by runout tolerances include those surfaces constructed around a datum axis and those constructed at right angles to a datum axis. See section on “Tolerance of Runout” in the ASME Y14-5 2009 for more details.

### 5.17.3 Modifier



























5.15.3 Modifier (cont'd)



Modifiers establishes additional symbols for specifying geometric characteristics and other dimensional requirements on engineering drawings. The application of Modifiers such as Most Material Condition or Least Material Condition may be applied to geometric tolerance values on features of size. See section on “Applicability of Modifiers on Geometric Tolerance Values and Datum Feature References” in the ASME Y14-5 2009 for more details.

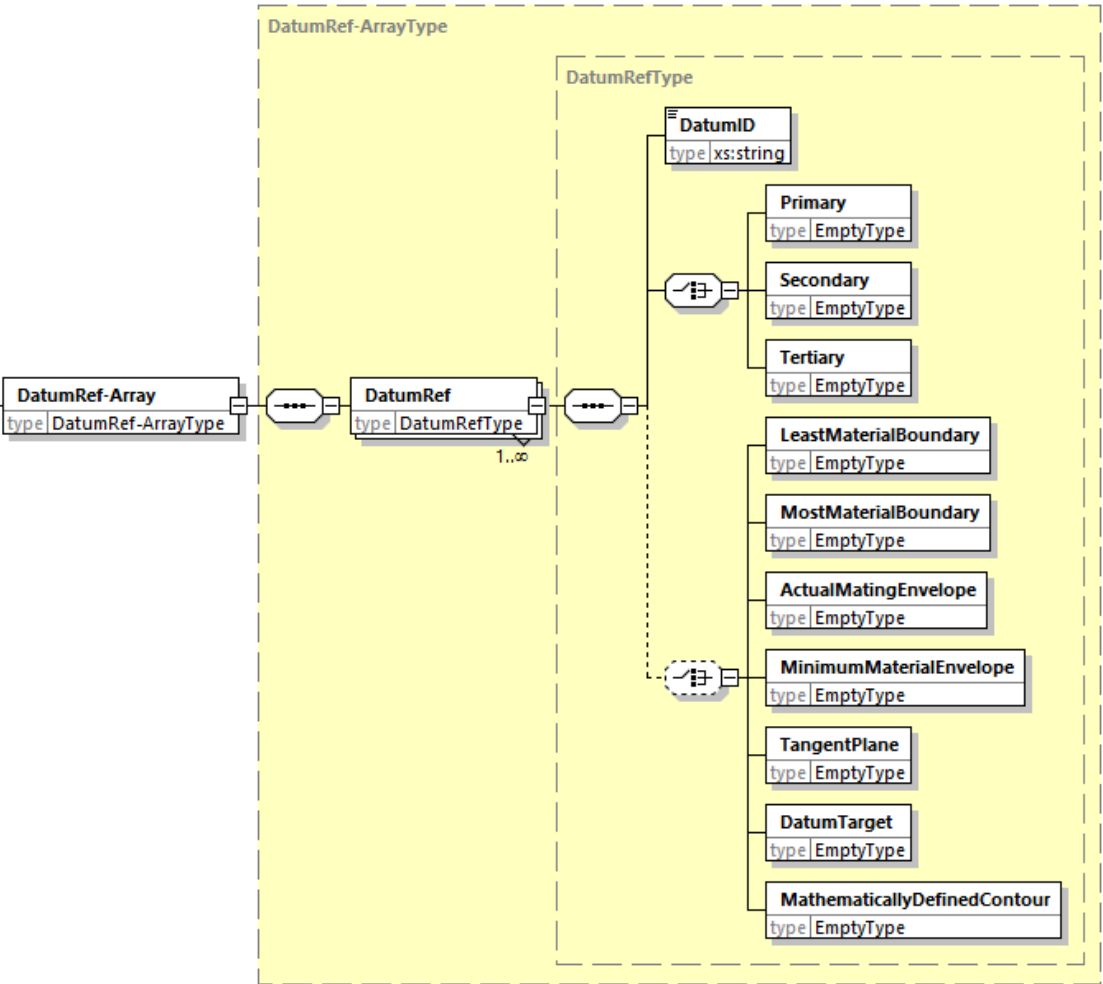
### 5.15.3 Modifier (cont'd)

TERM	SYMBOL	SEE:
AT MAXIMUM MATERIAL CONDITION (When applied to a tolerance value) AT MAXIMUM MATERIAL BOUNDARY (When applied to a datum reference)		3.3.5
AT LEAST MATERIAL CONDITION (When applied to a tolerance value) AT LEAST MATERIAL BOUNDARY (When applied to a datum reference)		3.3.5
TRANSLATION		3.3.26
PROJECTED TOLERANCE ZONE		3.3.6
FREE STATE		3.3.20
TANGENT PLANE		3.3.21
UNEQUALLY DISPOSED PROFILE		3.3.22
INDEPENDENCY		3.3.24
STATISTICAL TOLERANCE		3.3.10
CONTINUOUS FEATURE		3.3.23
DIAMETER		3.3.7
SPHERICAL DIAMETER		3.3.7
RADIUS		3.3.7
SPHERICAL RADIUS		3.3.7
CONTROLLED RADIUS		3.3.7
SQUARE		3.3.16
REFERENCE		3.3.8
ARC LENGTH		3.3.9
DIMENSION ORIGIN		3.3.17
BETWEEN		3.3.11
ALL AROUND		3.3.19
ALL OVER		3.3.25

Source: ASME Y14-5 2009

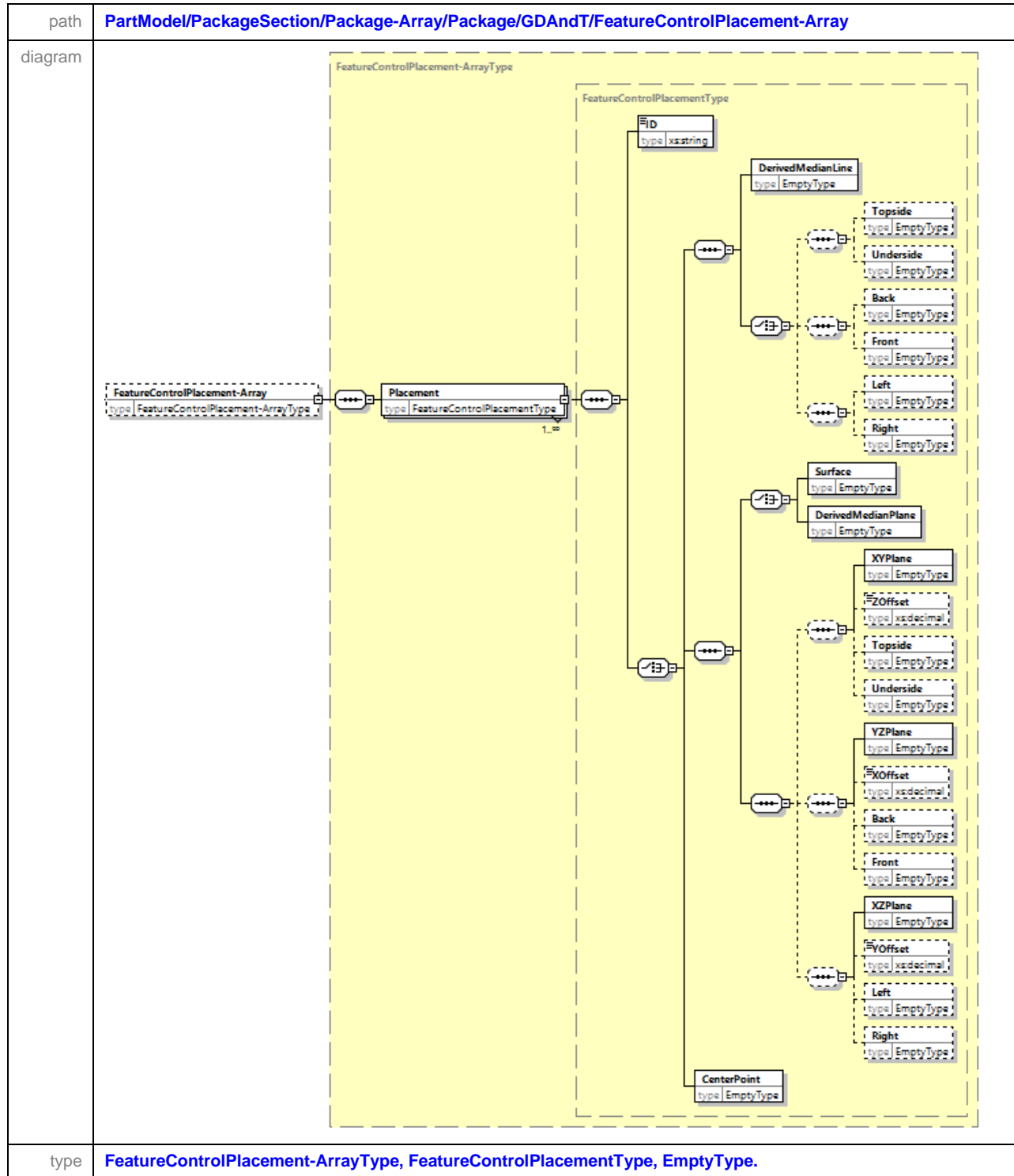
**Figure 18 – Modifying Symbols**

5.17.4 Datum Reference Array

path	PartModel/PackageSection/Package-Array/Package/GDAndT/FeatureControl-Array/FeatureControl/DatumRef-Array
diagram	 <p>The diagram illustrates the XSD structure for a Datum Reference Array. It shows a sequence of elements: <b>DatumRef-Array</b> (type <b>DatumRef-ArrayType</b>), followed by a sequence of <b>DatumRef</b> elements (type <b>DatumRefType</b>) with a cardinality of 1 to infinity. The <b>DatumRefType</b> is detailed within a dashed box and includes the following elements: <b>DatumID</b> (type <b>xs:string</b>), <b>Primary</b> (type <b>EmptyType</b>), <b>Secondary</b> (type <b>EmptyType</b>), <b>Tertiary</b> (type <b>EmptyType</b>), <b>LeastMaterialBoundary</b> (type <b>EmptyType</b>), <b>MostMaterialBoundary</b> (type <b>EmptyType</b>), <b>ActualMatingEnvelope</b> (type <b>EmptyType</b>), <b>MinimumMaterialEnvelope</b> (type <b>EmptyType</b>), <b>TangentPlane</b> (type <b>EmptyType</b>), <b>DatumTarget</b> (type <b>EmptyType</b>), and <b>MathematicallyDefinedContour</b> (type <b>EmptyType</b>).</p>
type	DatumRef-ArrayType, DatumRefType.

A datum reference frame is three mutually perpendicular intersecting datum planes. See section on “Datum Reference Frames” in the ASME Y14-5 2009 for more details.

### 5.17.5 Feature Control Placement Array



A feature control frame is related to a considered feature by one of the following methods

1. Locating the frame below or attached to a leader-directed note or dimension pertaining to the feature.

### 5.15.5 Feature Control Placement Array (cont'd)

2. Attaching a leader from the frame pointing to the feature.
3. Attaching a side, corner, or an end of the frame to an extension line from the feature, provided it is a plane surface,
4. Attaching a side, corner, or an end of the frame to an extension of the dimension line pertaining to a feature of size, and
5. Placing in a note, chart, or the general tolerance block.

See section on “Datum Reference Frames” in the ASME Y14-5 2009 for more details.

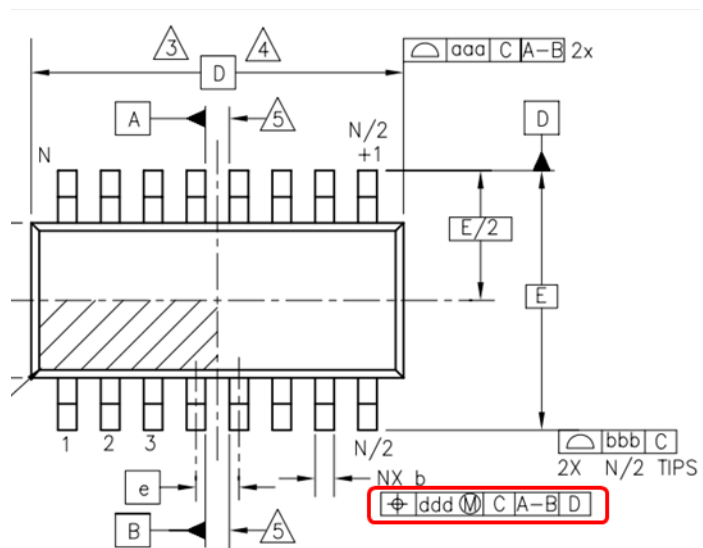
### 5.17.6 GD&T XML Example

Figure 19 represents an SOIC device with a Feature Control Frame highlighted that is represented in the xml example below.

```

<GDAndT>
  <FeatureControl-Array>
    <FeatureControl>
      <ID>Feature Control ID 1</ID>
      <PlacementID>Place ID X</PlacementID>
      <FeatureControlType>
        <Location>
          <Position>
            </Position>
          </Location>
        </FeatureControlType>
        <Tolerance>0.25</Tolerance>
        <Modifier>
          <MaximumMaterialBoundary/>
        </Modifier>
        <DatumRef-Array>
          <DatumRef>
            <DatumID>C</DatumID>
            <Primary/>
          </DatumRef>
          <DatumRef>
            <DatumID>A</DatumID>
            <Secondary/>
          </DatumRef>
          <DatumRef>
            <DatumID>B</DatumID>
            <Secondary/>
          </DatumRef>
          <DatumRef>
            <DatumID>D</DatumID>
            <Tertiary/>
          </DatumRef>
        </DatumRef-Array>
      </FeatureControl>
    </FeatureControl-Array>
    ...
    <FeatureControlPlacement-Array>
      <Placement>
        <ID>Place ID X</ID>
        <DerivedMedianPlane/>

```



**Figure 19 – Sample for Representation in an XML Structure**

### 5.15.6 GD&T XML Example (cont'd)

```

    <XZPlane/>
    <Left/>
    <Right/>
  </Placement>
</FeatureControlPlacement-Array>
</GDAndT>
...
<FeatureControl-ID>
  <FeatureControlID>456</FeatureControlID>
  <FeatureControlPlacementID>BBB</FeatureControlPlacementID>
</FeatureControl-ID>

```

### 5.17.7 GD and T Datum-to-Element Map

path	PartModel/PackageSection/Package-Array/Package/GDandTDatum-to-ElementMap-Array/GDandTDatum-to-ElementMap
diagram	<pre> sequenceDiagram     participant GDM as GDandTDatum-to-ElementMapType     GDM--&gt;&gt;ElementRef     GDM--&gt;&gt;Package     GDM--&gt;&gt;TerminalGroupID     GDM--&gt;&gt;TerminalDetailRef     </pre>
type	GDandTDatum-to-ElementMapType, JEP30-D10:EmptyType

## 5.18 Recommended Footprint - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array</a>
diagram	
type	<a href="#">RecommendedFootprint-ArrayType</a> , <a href="#">RecommendedFootprintType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">InterconnectTechnology-ArrayType</a> , <a href="#">RecommendedPadOrHole-ArrayType</a> , <a href="#">ThermalRelief-ArrayType</a> , <a href="#">AssemblyOutlineLayerType</a> , <a href="#">ConductiveArea-ArrayType</a> , <a href="#">PlacementOutlineType</a> , <a href="#">Keep-inLayer-ArrayType</a> , <a href="#">KeepoutRegion-ArrayType</a> , <a href="#">SoldermaskLayer-ArrayType</a> , <a href="#">PasteMaskLayer-ArrayType</a> .

The [FootprintName](#) and [FootprintExtendedName](#) is described in JESD30, “Descriptive Designation System for Electronic-device Packages and Footprints”.

### 5.18.1 Interconnect Technology - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/InterconnectTechnology-Array</a>
diagram	<p>The diagram illustrates the structure of the <code>InterconnectTechnology-ArrayType</code>. It consists of a sequence of elements: <code>InterconnectTechnology-Array</code> (type <code>InterconnectTechnology-ArrayType</code>), <code>InterconnectTechnology</code> (type <code>InterconnectTechnologyType</code>, 1..∞), and a choice of elements: <code>ID</code> (type <code>xs:string</code>), <code>FiniteSoldering</code> (type <code>JEP30-D10:EmptyType</code>), <code>InfiniteSoldering</code> (type <code>JEP30-D10:EmptyType</code>), <code>Pressfit</code> (type <code>JEP30-D10:EmptyType</code>), <code>AssemblyTechnologyID</code> (type <code>xs:string</code>), and <code>OtherTechnologyClass</code> (type <code>xs:string</code>).</p>
type	<a href="#">InterconnectTechnology-ArrayType</a> , <a href="#">InterconnectTechnologyType</a> , <a href="#">JEP30-D10:EmptyType</a>

### 5.18.2 Recommended Pad Or Hole Shape - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array</a>
diagram	<p>The diagram illustrates the structure of the <code>RecommendedPadOrHole-ArrayType</code>. It consists of a sequence of elements: <code>RecommendedPadOrHoleShape-Array</code> (type <code>RecommendedPadOrHole-ArrayType</code>), <code>RecommendedPadOrHoleShape</code> (type <code>RecommendedPadOrHoleType</code>, 0..∞), and <code>PadGroupToPadGroupRelationship</code> (type <code>PadGroupToPadGroupRelationshipType</code>, 0..∞).</p>
type	<a href="#">RecommendedPadOrHole-ArrayType</a> , <a href="#">RecommendedPadOrHoleType</a> , <a href="#">PadGroupToPadGroupRelationshipType</a>



5.18.2.1 Recommended Pad or Hole Shape

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape
diagram	<p>RecommendedPadOrHoleType</p> <ul style="list-style-type: none"><li>ID (type xs:string)</li><li>InterconnectTechnologyID (type xs:string)</li><li>SurfaceMount (type RecommendedSurfaceMountPadShapeType)</li><li>ThroughHole (type RecommendedThroughHoleType)</li><li>Location (type RecommendedPadOrHoleLocationType)</li><li>PatternGroup (type RecommendedPadOrHolePatternGroupType)</li><li>LandPatternSpan (type RecommendedLandPatternSpanType)</li><li>LandPatternSpacing (type RecommendedLandPatternSpacingType)</li></ul> <p>RecommendedPadOrHoleShape (type RecommendedPadOrHoleType) 0..∞</p>
type	RecommendedPadOrHoleShapeType, RecommendedSurfaceMountPadShapeType, RecommendedThroughHoleType, RecommendedPadOrHoleLocationType, RecommendedPadOrHolePatternGroupType, RecommendedLandPatternSpanType, RecommendedLandPatternSpacingType.

Although not recommended for most terminal types, the component manufacturer may provide their recommended land pattern or hole mounting requirement for the part. This is typically not good business practice as the component manufacturer can not necessarily control or know in advance all the possible use cases of the part, resulting in a one size fits all user environment. This seldom leads to a high yield impacting footprint. The exception to this rule is for the following terminal types since final hole sizes is critical to be properly matched to the dimensions of the pin or pressfit terminal. Recommended Pad or Hole Shape sizes are applicable to the following terminal types:

- Pressfit – All variations
- Column – Microspring solid core and air-core
- Pin – Press-in Solderable Terminal
- Pin – Swage Fastening Pin
- Pin – Press-in Non-Solderable

Although not recommended for most terminal types, the component manufacturer may provide

### 5.18.2.1.1 Surface Mount

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount</a>
diagram	<p>The diagram illustrates the structure of the <b>SurfaceMount</b> type. It is a choice type, indicated by a circle with a plus sign. The <b>SurfaceMount</b> type is defined as <code>type RecommendedSurfaceMountPadShapeType</code>. It can be one of the following shapes, each with its own type:</p> <ul style="list-style-type: none"> <li><b>Rectangle</b> (type: <code>RectangleValueType</code>)</li> <li><b>RoundedRectangle</b> (type: <code>RoundedRectangleValueType</code>)</li> <li><b>HalfRoundedRectangle</b> (type: <code>HalfRoundedRectangleValueType</code>)</li> <li><b>ModifiedRectangle</b> (type: <code>ModifiedRectangleValueType</code>)</li> <li><b>Circle</b> (type: <code>CircleValueType</code>)</li> <li><b>D-Shape</b> (type: <code>D-ShapeValueType</code>)</li> <li><b>RoundedRectangleD-Shape</b> (type: <code>D-ShapeRoundedRectangleValueType</code>)</li> <li><b>Double-D</b> (type: <code>Double-DValueType</code>)</li> <li><b>Contour</b> (type: <code>ContourShapeValueType</code>)</li> <li><b>PadVoid-Array</b> (type: <code>PadVoid-ArrayType</code>)</li> <li><b>ThermalReliefID</b> (type: <code>xs:string</code>)</li> </ul> <p>The shapes are grouped into a dashed box labeled <b>RecommendedSurfaceMountPadShapeType</b>.</p>
type	<a href="#">RecommendedSurfaceMountPadShapeType</a> , <a href="#">RectangleValueType</a> , <a href="#">RoundedRectangleValueType</a> , <a href="#">HalfRoundedRectangleValueType</a> , <a href="#">ModifiedRectangleValueType</a> , <a href="#">CircleValueType</a> , <a href="#">D-ShapeValueType</a> , <a href="#">D-ShapeRoundedRectangleValueType</a> , <a href="#">Double-DValueType</a> , <a href="#">ContourShapeValueType</a> , <a href="#">PadVoid-ArrayType</a> .

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions. Recommended land pattern only require the nominal defined and not any of the tolerances nor the min and max values.

### 5.18.2.1.1.1 Pad Void - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount/PadVoid-Array</a>
diagram	
type	<a href="#">PadVoid-ArrayType</a> , <a href="#">PadVoidType</a> , <a href="#">PadVoidShapeType</a> , <a href="#">PadVoidLocationType</a> .

### 5.18.2.1.1.1.1 Pad Void Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount/PadVoid-Array/PadVoid/PadVoidLocation/VoidStatus</a>
diagram	
type	<a href="#">PadVoidShapeType</a> , <a href="#">RectangleValueType</a> , <a href="#">RoundedRectangleValueType</a> , <a href="#">ModifiedRectangleValueType</a> , <a href="#">CircleValueType</a> , <a href="#">Double-DValueType</a> , <a href="#">ContourShapeValueType</a> .

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

### 5.18.2.1.1.2 Pad Void Location

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount/PadVoid-Array/PadVoid/PadVoidLocation
diagram	
type	PadVoidLocationType, VoidStandardArrayType, VoidCircularArrayType, DeletedStatusType, VoidRandomArrayType, JEP30-D10:EmptyType, JEP30-D10:PointXYType, JEP30-D10:MinIntegerOfOneType.

For *PadVoidLocation* defined via *StandardArray* or *CircularArray*, some voids in the array may be deleted. This is covered in the *VoidStatus* branch. Alternatively, *PadVoidLocation* can also be defined via *RandomArray* when there is just 1 Pad-or-Hole, or when there is no logical structure to the location of the Pads-or-Holes.

5.18.2.1.1.1.2.1      **Standard Array**

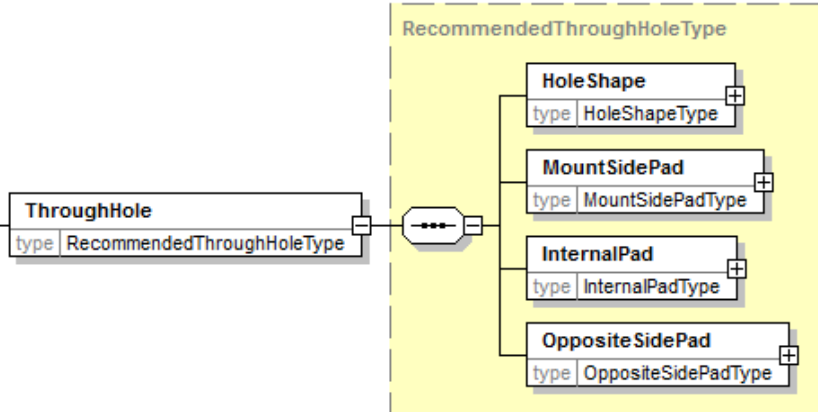
path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount/PadVoid-Array/PadVoid/PadVoidLocation/StandardArray
diagram	
type	VoidStandardArrayType, JEP30-D10:PitchValueSetType, JEP30-D10:PointXYType.

### 5.18.2.1.1.1.2.2 Circular Array

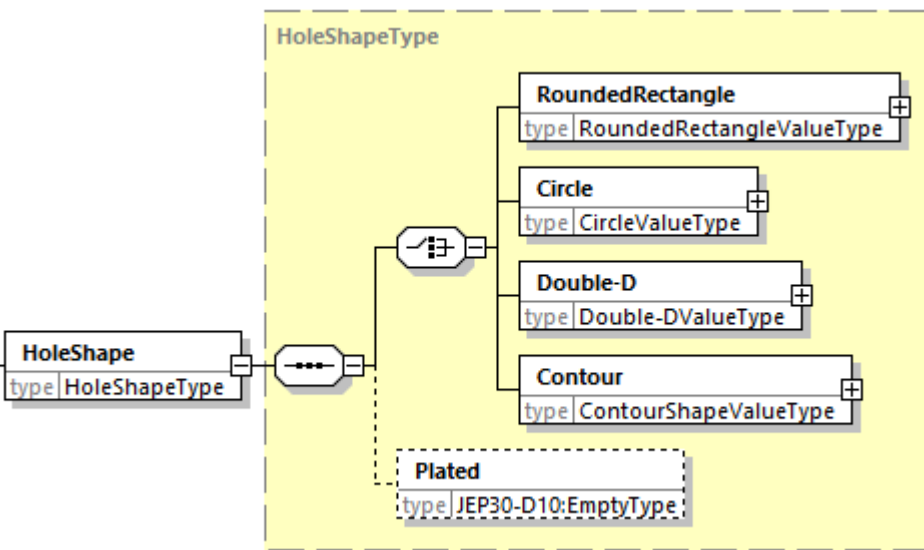
path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/SurfaceMount/PadVoid-Array/PadVoid/PadVoidLocation/CircularArray</a>
diagram	<pre> classDiagram     class CircularArray {         type VoidCircularArrayType     }     class PitchRadius {         type xs:decimal     }     class Center {         type JEP30-D10:PointXYType     }     class StartAngle {         type xs:integer     }     class AngleToFill {         type xs:integer     }     class AngleBetweenVoids {         type xs:integer     }     class NumberOfVoids {         type xs:integer     }     class Rotation {         type VoidRotationType     }     class RotateWithCircle {         type JEP30-D10:EmptyType     }     class RotationAngle {         type xs:integer     }      CircularArray "1" -- "*" PitchRadius     CircularArray "1" -- "*" Center     CircularArray "1" -- "*" StartAngle     CircularArray "1" -- "*" AngleToFill     CircularArray "1" -- "*" AngleBetweenVoids     CircularArray "1" -- "*" NumberOfVoids     CircularArray "1" -- "*" Rotation     CircularArray "1" -- "*" RotateWithCircle     CircularArray "1" -- "*" RotationAngle     </pre>
type	<a href="#">JEP30-D10:PointXYType</a> , <a href="#">VoidRotationType</a> .

See Table 24 - Void Circular Array Elements Definition for the definition of the [CircularArray](#) data elements.

### 5.18.2.1.2 Through Hole

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/ThroughHole</a>
diagram	 <p>The diagram shows a 'ThroughHole' block with a 'type' field set to 'RecommendedThroughHoleType'. This block is connected to a dashed box labeled 'RecommendedThroughHoleType'. Inside this dashed box, there are four sub-blocks: 'HoleShape' (type: HoleShapeType), 'MountSidePad' (type: MountSidePadType), 'InternalPad' (type: InternalPadType), and 'Opposite Side Pad' (type: OppositeSidePadType). Each sub-block has a '+' icon in its top right corner, indicating it is optional or a reference.</p>
type	<a href="#">RecommendedThroughHoleType</a> , <a href="#">HoleShapeType</a> , <a href="#">MountSidePadType</a> , <a href="#">InternalPadType</a> , <a href="#">OppositeSidePadType</a> .

### 5.18.2.1.2.1 Hole Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/ThroughHole/HoleShape</a>
diagram	 <p>The diagram shows a 'HoleShape' block with a 'type' field set to 'HoleShapeType'. This block is connected to a dashed box labeled 'HoleShapeType'. Inside this dashed box, there are four sub-blocks: 'RoundedRectangle' (type: RoundedRectangleValueType), 'Circle' (type: CircleValueType), 'Double-D' (type: Double-DValueType), and 'Contour' (type: ContourShapeValueType). Each sub-block has a '+' icon in its top right corner. Additionally, there is a 'Plated' block (type: JEP30-D10:EmptyType) shown in a dashed box below the main structure, connected to the 'HoleShapeType' dashed box by a dashed line.</p>
type	<a href="#">HoleShapeType</a> , <a href="#">RoundedRectangleValueType</a> , <a href="#">CircleValueType</a> , <a href="#">Double-DValueType</a> , <a href="#">ContourShapeValueType</a> , <a href="#">JEP30-D10:EmptyType</a> .

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

5.18.2.1.2.2 Mount Side Pad

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/ThroughHole/MountSidePad
diagram	
type	MountSidePadType, RectangleValueType, RoundedRectangleValueType, CircleValueType, D-ShapeValueType, Double-DValueType, ContourShapeValueType.

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.



5.18.2.1.2.3 Internal Pad

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/ThroughHole/InternalPad
diagram	
type	InternalPadType, RectangleValueType, RoundedRectangleValueType, CircleValueType, D-ShapeValueType, Double-DValueType, ContourShapeValueType.

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

5.18.2.1.2.4      Opposite Side Pad

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/ThroughHole/OppositeSidePad
diagram	<p>The diagram illustrates the structure of the <b>OppositeSidePadType</b>. It is a base type (indicated by a dashed line) for several other types: <b>Rectangle</b>, <b>RoundedRectangle</b>, <b>Circle</b>, <b>Double-D</b>, and <b>Contour</b>. Each of these types has a <b>type</b> attribute pointing to a specific value type: <b>RectangleValueType</b>, <b>RoundedRectangleValueType</b>, <b>CircleValueType</b>, <b>Double-DValueType</b>, and <b>ContourShapeValueType</b> respectively. Additionally, <b>OppositeSidePadType</b> has a reference (indicated by a solid line with an open circle) to a <b>Clearance</b> type, which has a <b>type</b> attribute pointing to <b>xs:decimal</b>.</p>
type	OppositeSideType, RectangleHoleType, RoundedRectangleHoleType, CircleHoleType, D-ShapeHoleType, Double-DHoleType, ContourShapeHoleType.

The definition of each shape is outlined in Annex A (informative) Shape Definitions & Dimensions.

### 5.18.2.1.3 Location

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/Location</a>
diagram	
type	<a href="#">RecommendedPadOrHoleLocationType</a> , <a href="#">PadStandardArrayType</a> , <a href="#">PadCircularArrayType</a> , <a href="#">DeletedStatusType</a> , <a href="#">Pad-or-HoleRandomArrayType</a> .

#### 5.18.2.1.3.1 Standard Array


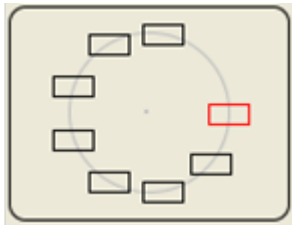
path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/Location/StandardArray</a>
diagram	
type	<a href="#">PadStandardArrayType</a> , <a href="#">JEP30-D10:PointType</a> .

5.18.2.1.3.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/Location/CircularArray
diagram	
type	PadCircularArrayType, PadRotationType, JEP30-D10:PointXYType.

### 5.16.2.1.3.2 Circular Array (cont'd)

**Table 35 - Pad-or-Hole Circular Array Elements Definition**

Element	Explanation
Pitch Radius	The radius of the circle of the Pads-or-Holes.
Center	The center of the circle of the Pads-or-Holes.
Start Angle	The angle at which to place the first Pad-or-Hole in the array. By default, the Pad-or-Hole is at the right-most point on the circle (i.e. the number 3 position on a clock face). This angle specifies a rotation clockwise around the circle from this point.
Angle to Fill	The angle through which the Pads-or-Holes are distributed, starting from the first Pad-or-Hole and continuing clockwise, until the last Pad-or-Hole is reached.
Angle between voids	Instead of specifying Angle to fill, you can specify the angle between the centers of each Pad-or-Hole.
Number of voids	The number of the Pads-or-Holes to be arrayed around the circle. The first Pad-or-Hole is displayed in red. This number includes any deleted the Pads-or-Holes in the array.
Rotate with Circle	 <p>If the Pads-or-Holes are rotated so that they are oriented perpendicular to the circle. This does not apply to circular or contour the Pad-or-Hole shapes.</p>
Rotation Angle	 <p>If the Pads-or-Holes are not oriented perpendicular to the circle, then the Pads-or-Holes could be rotated around the center of the Pad-or-Hole itself. In this image, each the Pad-or-Hole has a “0” degree rotation with respect to the “3 O’clock position”. This does not apply to circular or contour Pad-or-Hole shapes.</p>

### 5.18.2.1.3.3 Pad-or-Hole Status

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/Location/Pad-or-HoleStatus</a>
diagram	
type	<a href="#">DeletedStatusType</a> , <a href="#">JEP30-D10:PointXYType</a> , <a href="#">JEP30-D10:MinIntegerOfOneType</a> , <a href="#">JEP30-D10:EmptyType</a> .

For [Pad-or-HoleLocation](#) defined via [StandardArray](#) or [CircularArray](#), some Pads-or-Holes in the array may be deleted. This is covered in the [Pad-or-HoleStatus](#) branch.

#### 5.18.2.1.3.4 Random Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/Location/RandomArray</a>
diagram	
type	<a href="#">Pad-or-HoleRandomArrayType</a> , <a href="#">JEP30-D10:PointType</a> .

[Pad-or-HoleLocation](#) can also be defined via [RandomArray](#) when there is just 1 Pad-or-Hole, or when there is no logical structure to the location of the Pads-or-Holes.

#### 5.18.2.1.4 Pattern Groups

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/PatternGroup</a>
diagram	
type	<a href="#">RecommendedPadOrHolePatternGroupType</a> , <a href="#">RecommendedPadOrHolePatternRelationshipType</a> .

#### 5.18.2.1.4.1 Pattern Relationship

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/PatternGroup/PatternRelationship</a>
diagram	
type	<a href="#">RecommendedPadOrHolePatternRelationshipType</a> , <a href="#">RecommendedPadOrHoleRelationshipTransformationsType</a> , <a href="#">TransformDuplicateType</a>

#### 5.18.2.1.4.1.1 Relationship Transformations

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/PatternGroup/PatternRelationship/RelationshipTransformations</a>
diagram	
type	<a href="#">RecommendedPadOrHoleRelationshipTransformationsType</a> , <a href="#">TransformMirrorType</a> , <a href="#">TransformRotateType</a> , <a href="#">JEP30-D10:PointXYType</a> ,



### 5.18.2.1.5 Land Pattern Span

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/LandPatternSpan
diagram	
type	RecommendedLandPatternSpanType, JEP30-D10:ValueType.

### 5.18.2.1.6 Land Pattern Spacing

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/RecommendedPadOrHoleShape/LandPatternSpacing
diagram	
type	RecommendedLandPatternSpacingType, JEP30-D10:ValueType.

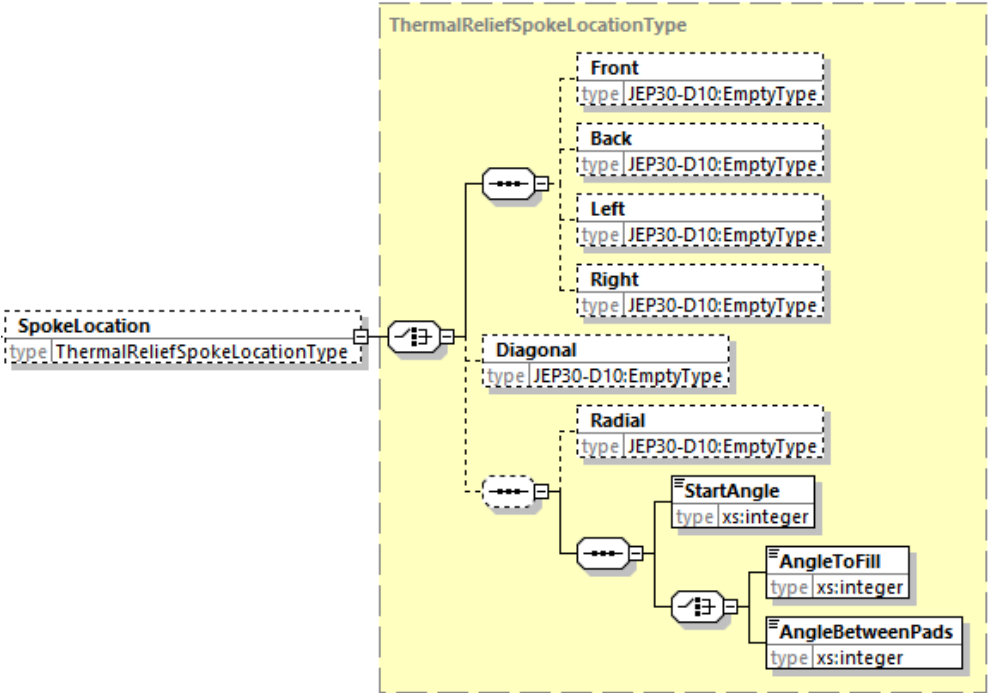
### 5.18.2.2 Pad Group To Pad Group Relationship

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/RecommendedPadOrHoleShape-Array/ PadGroupToPadGroupRelationship</a>
diagram	
type	<a href="#">PadGroupToPadGroupRelationshipType</a> , <a href="#">LandPatternSpanType</a> , <a href="#">LandPatternSpacingType</a> .

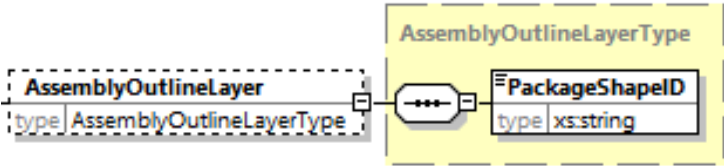
### 5.18.3 Thermal Relief - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ThermalRelief-Array</a>
diagram	
type	<a href="#">ThermalRelief-ArrayType</a> , <a href="#">ThermalReliefType</a> , <a href="#">ThermalReliefSpokeLocationType</a>

5.18.3.1 Spoke Location

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ThermalRelief-Array/ThermalRelief/SpokeLocation
diagram	 <p>The diagram illustrates the structure of the ThermalReliefSpokeLocationType. It shows a sequence of elements: SpokeLocation (type ThermalReliefSpokeLocationType), followed by a choice of Front, Back, Left, Right, Diagonal, and Radial (all type JEP30-D10:EmptyType:). The Radial element is further expanded to show StartAngle (type xs:integer), AngleToFill (type xs:integer), and AngleBetweenPads (type xs:integer).</p>
type	ThermalReliefSpokeLocationType, JEP30-D10:EmptyType.

5.18.4 Assembly Outline Layer

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/AssemblyOutlineLayer
diagram	 <p>The diagram illustrates the structure of the AssemblyOutlineLayerType. It shows a sequence of elements: AssemblyOutlineLayer (type AssemblyOutlineLayerType), followed by PackageShapeID (type xs:string).</p>
type	AssemblyOutlineLayerType.

### 5.18.5 Conductive Area - Array

path	<b>PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ConductiveArea-Array</b>
diagram	<p>The diagram illustrates the structure of the <b>ConductiveArea-Array</b>. It is an array of <b>ConductiveArea</b> objects. Each <b>ConductiveArea</b> object contains an <b>ID</b> (type: xs:string) and a choice of shapes: <b>Rectangle</b> (type: RectangleValueType), <b>ModifiedRectangle</b> (type: ModifiedRectangleValueType), <b>Circle</b> (type: CircleValueType), <b>ShapeCenter</b> (type: JEP30-D10:PointXYType), and <b>Contour</b> (type: ContourShapeValueType). Additionally, each <b>ConductiveArea</b> object contains a choice of <b>ConductiveAreaVoid-Array</b> (type: ConductiveAreaVoid-ArrayType).</p>
type	<b>ConductiveArea-ArrayType, ConductiveAreaType, RectangleValueType, ModifiedRectangleValueType, CircleValueType, JEP30-D10:PointXYType, ContourShapeValueType, ConductiveAreaVoid-ArrayType.</b>

#### 5.18.5.1 Conductive Area Void - Array

path	<b>PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ConductiveArea-Array/ConductiveAreaVoid-Array</b>
diagram	<p>The diagram illustrates the structure of the <b>ConductiveAreaVoid-Array</b>. It is an array of <b>ConductiveAreaVoid</b> objects. Each <b>ConductiveAreaVoid</b> object contains a choice of <b>ConductiveAreaVoidShape</b> (type: ConductiveAreaVoidShapeType) and <b>ConductiveAreaVoidLocation</b> (type: ConductiveAreaVoidLocationType).</p>
type	<b>ConductiveAreaVoid-ArrayType, ConductiveAreaVoidType, ConductiveAreaVoidShapeType, ConductiveAreaVoidLocationType.</b>

### 5.18.5.1.1 Conductive Area Void Shape

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ConductiveArea-Array/ConductiveAreaVoid-Array/ConductiveAreaVoid/ConductiveAreaVoidShape</a>
diagram	
type	<a href="#">ConductiveAreaVoidShapeType</a> , <a href="#">RectangleValueType</a> , <a href="#">RoundedRectangleValueType</a> , <a href="#">ModifiedRectangleValueType</a> , <a href="#">CircleValueType</a> , <a href="#">Double-DValueType</a> , <a href="#">ContourShapeValueType</a> .

### 5.18.5.1.2 Conductive Area Void Location

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/ConductiveArea-Array/ConductiveAreaVoid-Array/ConductiveAreaVoid/ConductiveAreaVoidLocation</a>
diagram	
type	<a href="#">ConductiveAreaVoidLocationType</a> , <a href="#">VoidStandardArrayType</a> , <a href="#">VoidCircularArrayType</a> , <a href="#">DeletedStatusType</a> , <a href="#">VoidRandomArrayType</a> .

5.18.6 Placement Outline

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PlacementOutline
diagram	
type	PlacementOutlineType, RectangleValueType, ModifiedRectangleValueType, CircleValueType, JEP30-D10:PointXYType, ContourShapeValueType

5.18.7 Keep-in Layer - Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/Keep-inLayer-Array
diagram	
type	Keep-inLayer-ArrayType, Keep-inLayerType, RectangleValueType, ModifiedRectangleValueType, CircleValueType, JEP30-D10:PointXYType, ContourShapeValueType, Keep-inRestrictiveLayerType, Keep-inLayerRestrictionType.

### 5.18.7.1 Restrictive Layer

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/Keep-inLayer-Array/Keep-inLayer/RestrictiveLayer</a>
diagram	
type	<a href="#">Keep-inRestrictiveLayerType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">Keep-inRestrictiveInnerLayerType</a> , <a href="#">RestrictiveInnerLayerFromOuterLayerType</a>

### 5.18.7.2 From Outer Layer

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/Keep-inLayer-Array/Keep-inLayer/RestrictiveLayer/Keep-inRestrictiveInnerLayer/FromOuterLayer</a>
diagram	
type	<a href="#">RestrictiveInnerLayerFromOuterLayerType</a> , <a href="#">JEP30-D10:EmptyType</a>



### 5.18.7.3 Restriction

path	PartModel/PackageSection/Package-Array/Package-ReferenceFootprint-Array/RecommendedFootprint/Keep-inLayer-Array/Keep-inLayer/Restriction
diagram	<pre> classDiagram     class Restriction {         type Keep-inLayerRestrictionType     }     class Keep-inLayerRestrictionType {         VialID type xs:string         GroundReference-TerminalName type xs:string         PositiveReference-TerminalName type xs:string         NegativeReference-TerminalName type xs:string         ConductiveArealID type xs:string     }     Restriction --&gt; Keep-inLayerRestrictionType   </pre>
type	Keep-inLayerRestrictionType.

### 5.18.8 Keepout Region - Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/KeepoutRegion-Array
diagram	<pre> classDiagram     class KeepoutRegionArray {         type KeepoutRegionArrayType     }     class KeepoutRegion {         type KeepoutRegionType     }     class Rectangle {         type RectangleValueType     }     class ModifiedRectangle {         type ModifiedRectangleValueType     }     class Circle {         type CircleValueType     }     class ShapeCenter {         type JEP30-D10:PointXYType     }     class Contour {         type ContourShapeValueType     }     class Component {         type xs:boolean     }     class AllCopper {         type xs:boolean     }     class Vias {         type xs:boolean     }     class Routes {         type xs:boolean     }      KeepoutRegionArray "1" -- "1..∞" KeepoutRegion     KeepoutRegion "1" -- "0..∞" Rectangle     KeepoutRegion "1" -- "0..∞" ModifiedRectangle     KeepoutRegion "1" -- "0..∞" Circle     KeepoutRegion "1" -- "0..∞" ShapeCenter     KeepoutRegion "1" -- "0..∞" Contour     KeepoutRegion "1" -- "0..∞" Component     KeepoutRegion "1" -- "0..∞" AllCopper     KeepoutRegion "1" -- "0..∞" Vias     KeepoutRegion "1" -- "0..∞" Routes     </pre> <p>The diagram illustrates the structure of the <b>KeepoutRegion-Array</b>. It is an array of <b>KeepoutRegion</b> objects (type <b>KeepoutRegionArrayType</b>). Each <b>KeepoutRegion</b> object (type <b>KeepoutRegionType</b>) contains several optional components (indicated by dashed boxes and optional multiplicity notations like 0..∞):</p> <ul style="list-style-type: none"> <li><b>Rectangle</b> (type <b>RectangleValueType</b>)</li> <li><b>ModifiedRectangle</b> (type <b>ModifiedRectangleValueType</b>)</li> <li><b>Circle</b> (type <b>CircleValueType</b>)</li> <li><b>ShapeCenter</b> (type <b>JEP30-D10:PointXYType</b>)</li> <li><b>Contour</b> (type <b>ContourShapeValueType</b>)</li> <li><b>Component</b> (type <b>xs:boolean</b>)</li> <li><b>All-Copper</b> (type <b>xs:boolean</b>)</li> <li><b>Vias</b> (type <b>xs:boolean</b>)</li> <li><b>Routes</b> (type <b>xs:boolean</b>)</li> </ul>
type	<b>KeepoutRegionArray, KeepoutRegionType, ContourShapeType, JEP30-D10:PointType.</b>

Keepouts are sometimes required around Parts to ensure the proper functionality of the Part. The definition of the contour shape is outlined in Annex A (informative) Shape Dimensions. The various type of Keepouts are:

- Component
- All-Copper
- Vias
- Routes

### 5.18.9 Soldermask Layer - Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array</a>
diagram	
type	<a href="#">SoldermaskLayer-ArrayType</a> , <a href="#">SoldermaskLayerType</a> , <a href="#">SoldermaskGangRelief-ArrayType</a> , <a href="#">SM-ShapeType</a> , <a href="#">SoldermaskLayerToSoldermaskLayerRelationshipType</a> .

#### 5.18.9.1 Soldermask Gang Relief- Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskGangRelief-Array</a>
diagram	
type	<a href="#">SoldermaskGangRelief-ArrayType</a> , <a href="#">SoldermaskGangReliefType</a> , <a href="#">TerminalCenter-ArrayType</a> , <a href="#">TerminalIndexType</a> , <a href="#">TerminalNumberType</a>

5.18.9.2      Soldermask Shape

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape
diagram	<p>The diagram illustrates the SoldermaskShape class and its associated types. The SoldermaskShape class (type SM-ShapeType) is shown with a multiplicity of 1..∞. It is connected to a dashed box labeled SM-ShapeType, which contains several subclasses and associated types. The subclasses include ID (type xs:string), Rectangle (type RectangleValueType), RoundedRectangle (type RoundedRectangleValueType), HalfRoundedRectangle (type HalfRoundedRectangleValueType), ModifiedRectangle (type ModifiedRectangleValueType), Circle (type CircleValueType), D-Shape (type D-ShapeValueType), RoundedRectangleD-Shape (type D-ShapeRoundedRectangleValueType), Double-D (type Double-DValueType), and Contour (type ContourShapeValueType). The associated types include SM-ShapeLocation (type SM-ShapeLocationType), PatternGroup (type SM-ShapePatternGroupType), SM-ShapePatternSpan (type SM-ShapePatternSpanType), and SM-ShapePatternSpacing (type SM-ShapePatternSpacingType). The SM-ShapeType box is highlighted in yellow.</p>
type	SM-ShapeType, RectangleValueType, RoundedRectangleValueType, HalfRoundedRectangleValueType, ModifiedRectangleValueType, CircleValueType, D-ShapeValueType, D-ShapeRoundedRectangleValueType, Double-DValueType, ContourShapeValueType, SM-ShapeLocationType, SM-ShapePatternGroupType, SM-ShapePatternSpanType, SM-ShapePatternSpacingType.

5.18.9.2.1 SM - Shape Location

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation
diagram	
type	SM-ShapeLocationType, SM-ShapeStandardArrayType, SM-ShapeCircularArrayType, SM-ShapeStatusType, SM-ShapeRandomArrayType.

5.18.9.2.1.1 Standard Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation
diagram	
type	SM-ShapeStandardArrayType, JEP30-D10:PointXYType.

### 5.18.9.2.1.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation/CircularArray
diagram	<p>The diagram illustrates the structure of the <b>SM-ShapeCircularArrayType</b>. It is a sequence of the following elements:</p> <ul style="list-style-type: none"> <li><b>PitchRadius</b>: type <code>xs:decimal</code></li> <li><b>Center</b>: type <code>JEP30-D10:PointXYType</code></li> <li><b>StartAngle</b>: type <code>xs:integer</code></li> <li><b>AngleToFill</b>: type <code>xs:integer</code></li> <li><b>AngleBetweenPads</b>: type <code>xs:integer</code></li> <li><b>NumberOfSM-Shapes</b>: type <code>xs:integer</code></li> </ul> <p>A dashed box labeled <b>SM-ShapeRotationType</b> contains a <b>Rotation</b> element (type <code>SM-ShapeRotationType</code>) which points to a <b>RotateWithCircle</b> element (type <code>JEP30-D10:EmptyType</code>) and a <b>RotationAngle</b> element (type <code>xs:integer</code>).</p>
type	SM-ShapeCircularArrayType, JEP30-D10:PointXYType, SM-ShapeRotationType, JEP30-D10:EmptyType.

### 5.18.9.2.1.3 SM - Shape Status

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation/ShapeStatus
diagram	<p>The diagram illustrates the structure of the <b>SM-ShapeStatusType</b>. It is a sequence of the following elements:</p> <ul style="list-style-type: none"> <li><b>SM-ShapeCenter-Array</b>: type <code>SM-ShapeCenter-ArrayType</code></li> <li><b>SM-ShapeIndex</b>: type <code>SM-ShapeIndexType</code></li> <li><b>Deleted</b>: type <code>JEP30-D10:EmptyType</code></li> </ul>
type	SM-ShapeStatusType, SM-ShapeCenter-ArrayType, SM-ShapeIndexType, JEP30-D10:EmptyType.

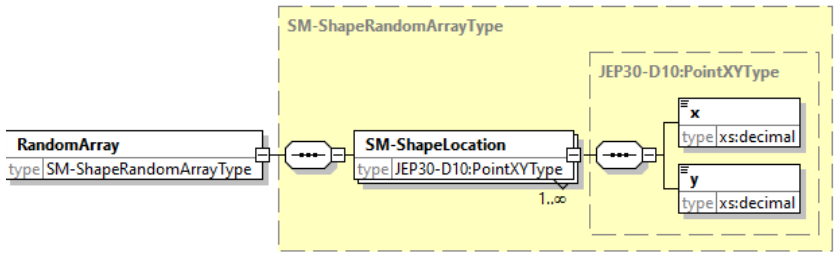
5.18.9.2.1.3.1 SM – Shape Center - Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation/ShapeStatus/SM-ShapeCenter-Array
diagram	
type	SM-ShapeCenter-ArrayType, SM-ShapeCenterType, JEP30-D10:PointXYType.

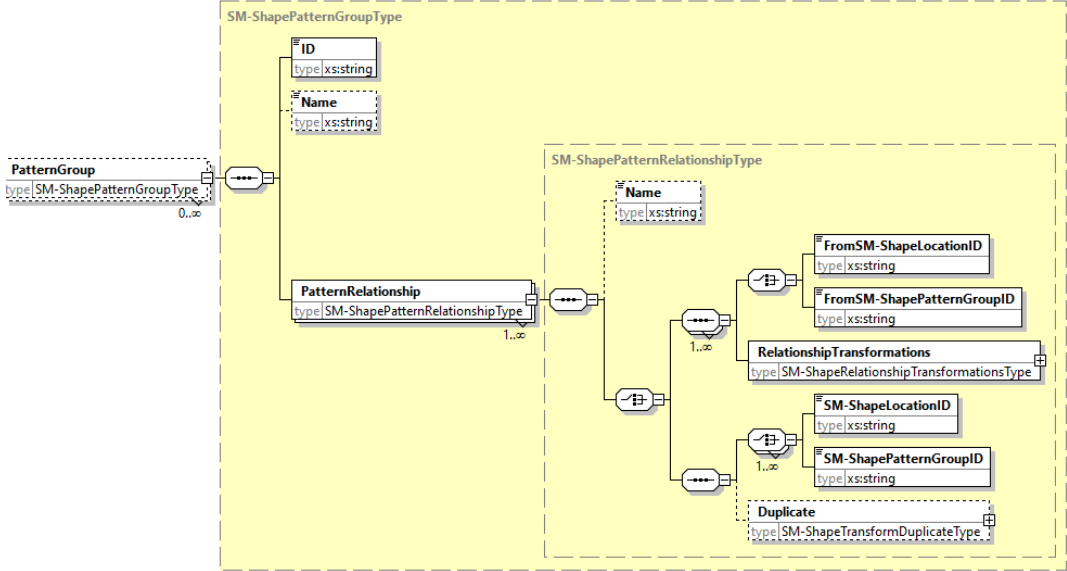
5.18.9.2.1.3.2 SM - Shape Index

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation/ShapeStatus/ShapeIndex
diagram	<p>The diagram illustrates the structure of the SM-ShapeIndexType. It features a central class <b>SM-ShapeIndex</b> with a type <b>SM-ShapeIndexType</b>. This class is linked to a dashed box labeled <b>SM-ShapeIndexType</b>, which contains several sub-classes and their relationships:</p> <ul style="list-style-type: none"><li><b>Center</b>: type <b>JEP30-D10:PointXYType</b>, multiplicity <b>1..∞</b>.</li><li><b>RowTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>FromRowTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>ToRowTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>ColumnTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>FromColumnTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>ToColumnTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>PolarTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>FromPolarTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li><li><b>ToPolarTerminalIndex</b>: type <b>JEP30-D10:MinIntegerOfOneType</b>, multiplicity <b>1..∞</b>.</li></ul>
type	SM-ShapeIndexType, JEP30-D10:PointXYType, JEP30-D10:MinIntegerOfOneType, JEP30-D10:EmptyType.

#### 5.18.9.2.1.4 Random Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapeLocation/RandomArray</a>
diagram	 <p>The diagram shows the structure of the <b>RandomArray</b> element. It is a sequence of <b>SM-ShapeLocation</b> elements. The <b>RandomArray</b> element has a type of <b>SM-ShapeRandomArrayType</b>. The <b>SM-ShapeLocation</b> element has a type of <b>JEP30-D10:PointXYType</b>. The <b>JEP30-D10:PointXYType</b> element contains two sub-elements: <b>x</b> (type <b>xs:decimal</b>) and <b>y</b> (type <b>xs:decimal</b>). The <b>SM-ShapeLocation</b> element is repeated 1..∞ times.</p>
type	<b>SM-ShapeRandomArrayType</b> , <b>JEP30-D10:PointXYType</b> .

#### 5.18.9.2.2 Pattern Group

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/PatternGroup</a>
diagram	 <p>The diagram shows the structure of the <b>PatternGroup</b> element. It is a sequence of <b>PatternRelationship</b> elements. The <b>PatternGroup</b> element has a type of <b>SM-ShapePatternGroupType</b>. The <b>PatternRelationship</b> element has a type of <b>SM-ShapePatternRelationshipType</b>. The <b>SM-ShapePatternRelationshipType</b> element contains several sub-elements: <b>ID</b> (type <b>xs:string</b>), <b>Name</b> (type <b>xs:string</b>), <b>FromSM-ShapeLocationID</b> (type <b>xs:string</b>), <b>FromSM-ShapePatternGroupID</b> (type <b>xs:string</b>), <b>RelationshipTransformations</b> (type <b>SM-ShapeRelationshipTransformationsType</b>), <b>SM-ShapeLocationID</b> (type <b>xs:string</b>), <b>SM-ShapePatternGroupID</b> (type <b>xs:string</b>), and <b>Duplicate</b> (type <b>SM-ShapeTransformDuplicateType</b>). The <b>PatternRelationship</b> element is repeated 1..∞ times.</p>
type	<b>SM-ShapePatternGroupType</b> , <b>SM-ShapePatternRelationshipType</b> , <b>SM-ShapeRelationshipTransformationsType</b> , <b>SM-ShapeTransformDuplicateType</b> .



### 5.18.9.2.2.1 RelationshipTransformations

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/PatternGroup/RelationshipTransformations</a>
diagram	
type	<a href="#">SM-ShapeRelationshipTransformationsType</a>

### 5.18.9.2.2.2 RelationshipTransformations

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/PatternGroup/Duplicate</a>
diagram	
type	<a href="#">SM-ShapeTransformDuplicateType.</a>

### 5.18.9.2.3 SM – Shape Pattern Span

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapePatternSpan</a>
diagram	<pre> classDiagram     class SMShapePatternSpan {         type SMShapePatternSpanType     }     class SMShapePatternSpanType {         X-Direction JEP30-D10:ValueType         Y-Direction JEP30-D10:ValueType     }     SMShapePatternSpan --&gt; SMShapePatternSpanType         </pre>
type	<a href="#">SM-ShapePatternSpanType</a> , <a href="#">JEP30-D10:ValueType</a> .

### 5.18.9.2.4 SM – Shape Pattern Spacing

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/SoldermaskLayer-Array/SoldermaskLayer/SoldermaskShape/SM-ShapePatternSpacing</a>
diagram	<pre> classDiagram     class SMShapePatternSpacing {         type SMShapePatternSpacingType     }     class SMShapePatternSpacingType {         X-Direction JEP30-D10:ValueType         Y-Direction JEP30-D10:ValueType     }     SMShapePatternSpacing --&gt; SMShapePatternSpacingType         </pre>
type	<a href="#">SM-ShapePatternSpacingType</a> , <a href="#">JEP30-D10:ValueType</a> .

5.18.10    Pastemask Layer- Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array
diagram	<p>The diagram illustrates the structure of the <b>PastemaskLayer-Array</b> type. It is composed of several nested and associated types:</p> <ul style="list-style-type: none"><li><b>PastemaskLayer-Array</b> (type: <i>PastemaskLayer-ArrayType</i>) is the root container, associated with a <b>PastemaskLayer</b> (type: <i>PastemaskLayerType</i>) via a 1..∞ relationship.</li><li><b>PastemaskLayer</b> (type: <i>PastemaskLayerType</i>) contains:<ul style="list-style-type: none"><li><b>RecommendedStencilThickness</b> (type: <i>RecommendedStencilThicknessType</i>) associated with a <b>RecommendedStencilThicknessType</b> (type: <i>RecommendedStencilThicknessType</i>) via a 1..∞ relationship.</li><li><b>PastemaskLayerToPastemaskLayerRelationship</b> (type: <i>PastemaskLayerToPastemaskLayerRelationshipType</i>) associated with a <b>PastemaskLayerToPastemaskLayerRelationshipType</b> (type: <i>PastemaskLayerToPastemaskLayerRelationshipType</i>) via a 0..∞ relationship.</li><li><b>PastemaskLayerType</b> (type: <i>PastemaskLayerType</i>) is a sub-container for:<ul style="list-style-type: none"><li><b>ID</b> (type: <i>xs:string</i>)</li><li><b>ApertureShape</b> (type: <i>ApertureShapeType</i>) associated with an <b>ApertureShapeType</b> (type: <i>ApertureShapeType</i>) via a 1..∞ relationship.</li></ul></li></ul></li><li><b>RecommendedStencilThicknessType</b> (type: <i>RecommendedStencilThicknessType</i>) contains:<ul style="list-style-type: none"><li><b>Thickness</b> (type: <i>JEP30-D10:MinNomMaxValueSetType</i>)</li><li><b>StencilThicknessUOM</b> (type: <i>StencilThicknessUOMType</i>)</li></ul></li></ul>
type	PasteMaskLayer-ArrayType, RecommendedStencilThicknessType, JEP30-D10:MinNomMaxValueSetType, StencilThicknessUOMType, PasteMaskLayerType, ApertureShapeType, PastemaskLayerToPastemaskLayerRelationshipType.

### 5.18.10.1 Aperture Shape

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape
diagram	
type	<p>ApertureShapeType, RectangleValueType, RoundedRectangleValueType, ModifiedApertureRectangleType, CircleValueType, SegmentedRingType, ContourShapeValueType, ApertureThicknessType, JEP30-D10:MinNomMaxValueSetType, StencilThicknessUOMType, ApertureLocationType, AperturePatternGroupType, AperturePatternSpanType, AperturePatternSpacingType, JEP30-D10:ValueType.</p>

All shapes are defined in Annex A.1

### 5.18.10.1.1 Aperture Location

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/ApertureLocation</a>
diagram	<pre> classDiagram     class ApertureLocation {         type ApertureLocationType     }     class ApertureLocationType {         ID xs:string         Name xs:string         StandardArray ApertureStandardArrayType         CircularArray ApertureCircularArrayType         RandomArray ApertureRandomArrayType     }     class ApertureStandardArrayType {         type ApertureStandardArrayType     }     class ApertureCircularArrayType {         type ApertureCircularArrayType     }     class ApertureRandomArrayType {         type ApertureRandomArrayType     }     class ApertureStatusType {         type ApertureStatusType     }      ApertureLocation "1..∞" -- "1" ApertureLocationType     ApertureLocationType -- "0..1" ApertureStandardArrayType     ApertureLocationType -- "0..1" ApertureCircularArrayType     ApertureLocationType -- "0..1" ApertureRandomArrayType     ApertureRandomArrayType -- "0..1" ApertureStatusType </pre>
type	<a href="#">ApertureLocationType</a> , <a href="#">ApertureStandardArrayType</a> , <a href="#">ApertureCircularArrayType</a> , <a href="#">ApertureStatusType</a> , <a href="#">ApertureRandomArrayType</a> .

### 5.18.10.1.1.1 Standard Array

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/ApertureLocation/StandardArray</a>
diagram	<pre> classDiagram     class StandardArray {         type ApertureStandardArrayType     }     class ApertureStandardArrayType {         nx xs:integer         ny xs:integer         dx xs:decimal         dy xs:decimal         Angle xs:integer         ApertureGroupLowerLeftApertureCenter JEP30-D10:PointXYType     }      StandardArray -- "1" ApertureStandardArrayType </pre>
type	<a href="#">ApertureStandardArrayType</a> , <a href="#">JEP30-D10:PointXYType</a> .

5.18.10.1.1.2 Circular Array

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/ApertureLocation/CircularArray
diagram	<p>The diagram illustrates the structure of the <b>CircularArray</b> type. It is defined as <code>type CircularArray ApertureCircularArrayType</code>. The <b>ApertureCircularArrayType</b> is a complex type containing several elements:</p> <ul style="list-style-type: none"><li><b>PitchRadius</b>: <code>type xs:decimal</code></li><li><b>Center</b>: <code>type JEP30-D10:PointXYType</code></li><li><b>StartAngle</b>: <code>type xs:integer</code></li><li><b>AngleToFill</b>: <code>type xs:integer</code></li><li><b>AngleBetweenPads</b>: <code>type xs:integer</code></li><li><b>NumberOfApertures</b>: <code>type xs:integer</code></li><li><b>ApertureRotation</b>: <code>type ApertureRotationType</code></li></ul> <p>The <b>ApertureRotationType</b> is a complex type containing:</p> <ul style="list-style-type: none"><li><b>RotateWithCircle</b>: <code>type JEP30-D10:EmptyType</code></li><li><b>RotationAngle</b>: <code>type xs:integer</code></li></ul> <p>The diagram uses dashed boxes to group the elements into <b>ApertureCircularArrayType</b> and <b>ApertureRotationType</b>. The <b>CircularArray</b> type is shown as a box on the left, connected to the <b>ApertureCircularArrayType</b> box.</p>
type	ApertureCircularArrayType, JEP30-D10:PointXYType, ApertureRotationType.

### 5.18.10.1.1.3 Status

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/ApertureLocation/Status
diagram	
type	ApertureStatusType, JEP30-D10:MinIntegerOfOneType, JEP30-D10:PointXYType, JEP30-D10:EmptyType.

### 5.18.10.1.1.4 RandomArray

path	PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/ApertureLocation/RandomArray
diagram	
type	ApertureRandomArrayType, JEP30-D10:PointXYType.

### 5.18.10.1.2 Pattern Groups

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/PatternGroup</a>
diagram	
type	<a href="#">AperturePatternGroupType</a> , <a href="#">AperturePatternRelationshipType</a> , <a href="#">AperturePatternRelationshipTransformationsType</a> , <a href="#">ApertureTransformDuplicateType</a> .

### 5.18.10.1.2.1 Relationship Transformations

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/PatternGroup/PatternRelationship/RelationshipTransformations</a>
diagram	
type	<a href="#">AperturePatternRelationshipTransformationsType</a> , <a href="#">TransformMirrorType</a> , <a href="#">TransformRotateType</a> , <a href="#">JEP30-D10:PointXYType</a>



### 5.18.10.1.2.2 Duplicate

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureShape/PatternGroup/PatternRelationship/Duplicate</a>
diagram	
type	<a href="#">ApertureTransformDuplicateType</a> .

### 5.18.10.2 Pastemask Layer To Pastemask Layer Relationship

path	<a href="#">PartModel/PackageSection/Package-Array/Package/RecommendedFootprint-Array/RecommendedFootprint/PastemaskLayer-Array/PastemaskLayer/ApertureGroupToApertureGroupRelationship</a>
diagram	
type	<a href="#">ApertureGroupToApertureGroupRelationshipType</a> , <a href="#">ApertureGroupSpanType</a> , <a href="#">ApertureGroupSpacingType</a> .

## 5.19 Physical Model

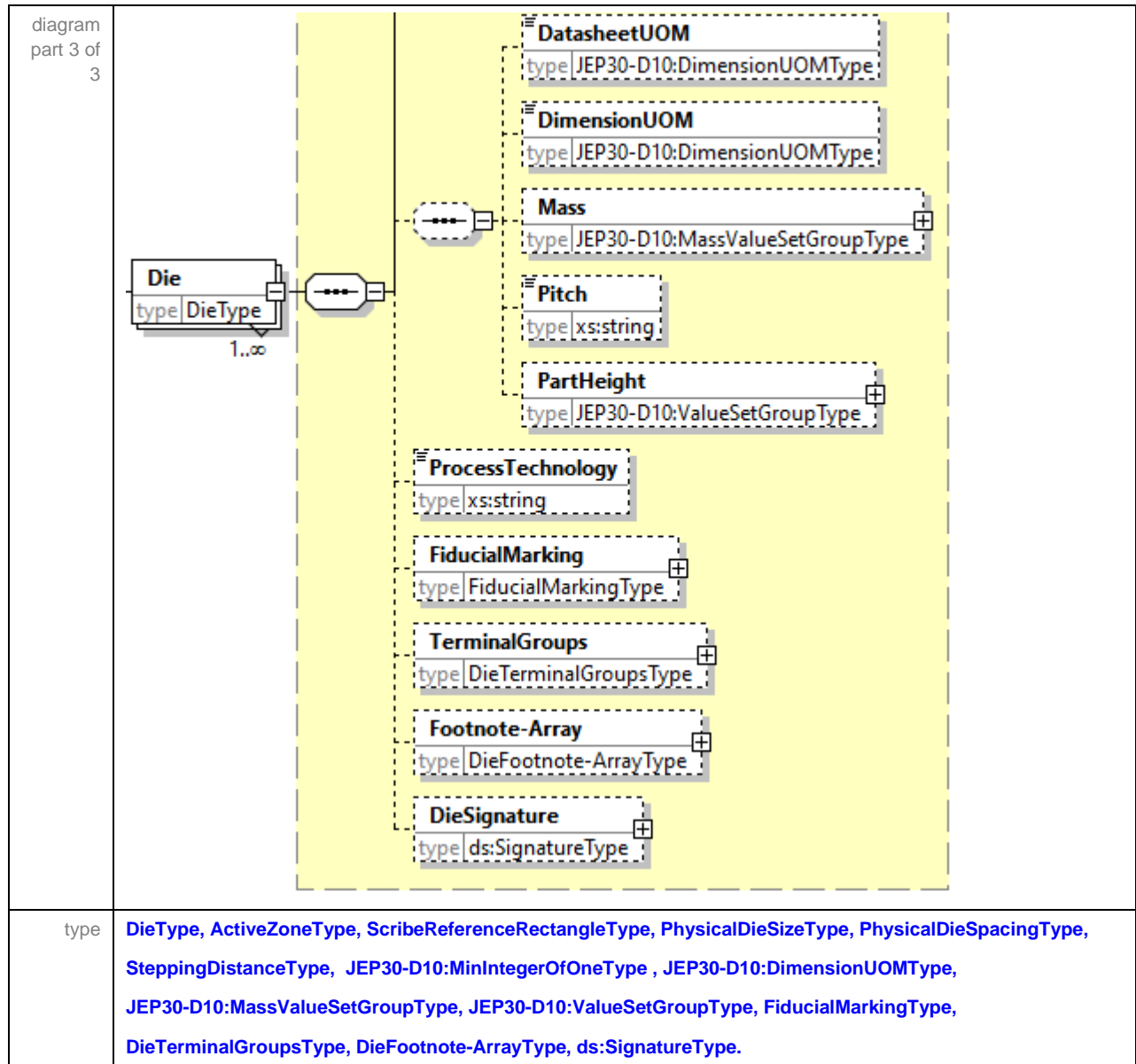
path	<a href="#">PartModel/PackageSection/Package-Array/Package/PhysicalModel</a>
diagram	<p>The diagram illustrates the structure of the <b>PhysicalModelType</b>. It is a sequence of the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b>: type <code>xs:string</code></li> <li>A choice between <b>Step</b> (type <code>JEP30-D10:EmptyType</code>) and <b>Other</b> (type <code>xs:string</code>).</li> <li><b>Model</b>: type <code>xs:string</code></li> <li><b>ModelDescription</b>: type <code>xs:string</code></li> <li><b>PhysicalModelSignature</b>: type <code>ds:SignatureType</code></li> </ul> <p>The <b>PhysicalModel</b> element is shown with a cardinality of <code>0..∞</code>.</p>
type	<a href="#">PhysicalModelType</a> , <a href="#">ModelType</a> , <a href="#">JEP30-D10:EmptyType</a> , <a href="#">ds:SignatureType</a> .

One or more references to external physical model files can be defined by the [PhysicalModel](#) section. The [Model](#) element can refer to either a file name of a file that is provided together with the corresponding JEP30 xml archive or a URL definition. An [Other](#) file format is accommodated to account for non-standard formats where it is advised that the optional [ModelDescription](#) element is used to describe the format and its intended usage.

6 Package Section - Die-Array

path	PartModel/PackageSection
diagram part 1 of 3	<p>This diagram illustrates the structure of the <b>JEP30-P101:PackageSectionType</b>. It shows a <b>PackageSection</b> element (type JEP30-P101:PackageSectionType) connected to a container that holds two optional elements: <b>Package-Array</b> (type Package-ArrayType) and <b>Die-Array</b> (type Die-ArrayType). The <b>Die-Array</b> element is further connected to a <b>Die-ArrayType</b> container, which contains a <b>Die</b> element (type DieType) with a cardinality of 1..∞ and a <b>constraints</b> element.</p>
type	PackageSectionType, Package-ArrayType, Die-ArrayType, DieType.
path	PartModel/PackageSection/Die-Array/Die
diagram part 2 of 3	<p>This diagram illustrates the structure of the <b>DieType</b>. It shows a <b>Die</b> element (type DieType) with a cardinality of 1..∞ connected to a container that holds several optional elements: <b>ID</b> (type xs:string), <b>ActiveZone</b> (type ActiveZoneType), <b>ScribeReferenceRectangle</b> (type ScribeReferenceRectangleType), <b>PhysicalDieSize</b> (type PhysicalDieSizeType), <b>PhysicalDieSpacing</b> (type PhysicalDieSpacingType), <b>SteppingDistance</b> (type SteppingDistanceType), <b>ExtendedTerminalCount</b> (type JEP30-D10:MinIntegerOfOneType, minIncl/maxIncl 1), <b>TerminalCount</b> (type JEP30-D10:MinIntegerOfOneType, minIncl/maxIncl 1), <b>DeletedTerminalCount</b> (type xs:integer), and <b>MissingTerminalCount</b> (type xs:integer).</p>

## 6 Package Section - Die-Array (cont'd)



The definitions of the [ExtendedTerminalCount](#), [TerminalCount](#), [DeletedTerminalCount](#) and [MissingTerminalCount](#) are described in the JESD30 document. The Die/Pitch follows the same rules as Package/Pitch as defined in the JESD30 document.

The enumerate values of the [JEP30-D10:DimensionUOMType](#) are [nm](#), [um](#), [mm](#), [m](#), [in](#) and [mil](#), while [nm](#), [um](#), [mm](#), are the units of measure used in the die's.

The component Manufacturer can provide details the [ProcessTechnology](#) that was used for the die fabrication, such as 5nm, or company specific internal code name.

## 6.1 Die Zones

path	<b>PartModel/PackageSection/Die-Array/Die/ActiveZone</b>	
diagram part 1 of 4		
type	<b>ActiveZoneType, ReferenceRectangleGroupType, ReferenceRegularPolygonGroupType, D10:PointXYType, ContourShapeGroupType.</b>	
path	<b>PartModel/PackageSection/Die-Array/Die/ScribeReferenceRectangle</b>	
diagram part 2 of 4		
type	<b>ScribeReferenceRectangleType, ReferenceRectangleGroupType, ReferenceRegularPolygonGroupType, JEP30-D10:PointXYType, ContourShapeGroupType, FromActiveZoneEdgesType.</b>	

## 6.1 Die Zones (cont'd)

path	PartModel/PackageSection/Die-Array/Die/PhysicalDieSize	
diagram part 3 of 4		
type	PhysicalDieSizeType, ReferenceRectangleGroupType, ReferenceRegularPolygonGroupType, JEP30-D10:PointXYType, ContourShapeGroupType, FromScribeReferenceRectangleEdgesType.	
path	PartModel/PackageSection/Die-Array/Die/PhysicalDieSizeSpacing	
diagram part 4 of 4		
type	PhysicalDieSizeSpacingType, JEP30-D10:DimensionalValueSetGroupType.	

The centers of the [ActiveZone](#), [ScribeReferenceRectangle](#), [PhysicalDieSize](#) do not have to be at the same point. Each can have different offsets with respect to the [ActiveZone](#) center. The center of the active die is the primary datum.

The distance between the edge of the [ActiveZone](#) to the edge of the [ScribeReferenceRectangle](#) is defined as the [SealRing](#) for that side. The [SealRing](#) is sometimes referred to as the “Etch Ring” or “Guard Ring”. The [SealRing](#) can be a different dimensions on each of the 4 sides, although in most cases, the right side equals the left side and the top side equals the bottom side, but the top/bottom side values does not typically equal to the right/left side values.

6.1 Die Zones (cont'd)

The distance between the edge of the *ScribeReferenceRectangle* to the edge of the *PhysicalDieSize* is defined as the *RemnantScribe* for that side. The *RemnantScribe* can also be a different dimensions on each of the 4 sides, similar to that of the *SealRing*.

The *PhysicalDieSizeSpacing* is the physical distance between each *PhysicalDieSize* on the wafer. This *PhysicalDieSizeSpacing* can be a different dimension in the X and Y directions.

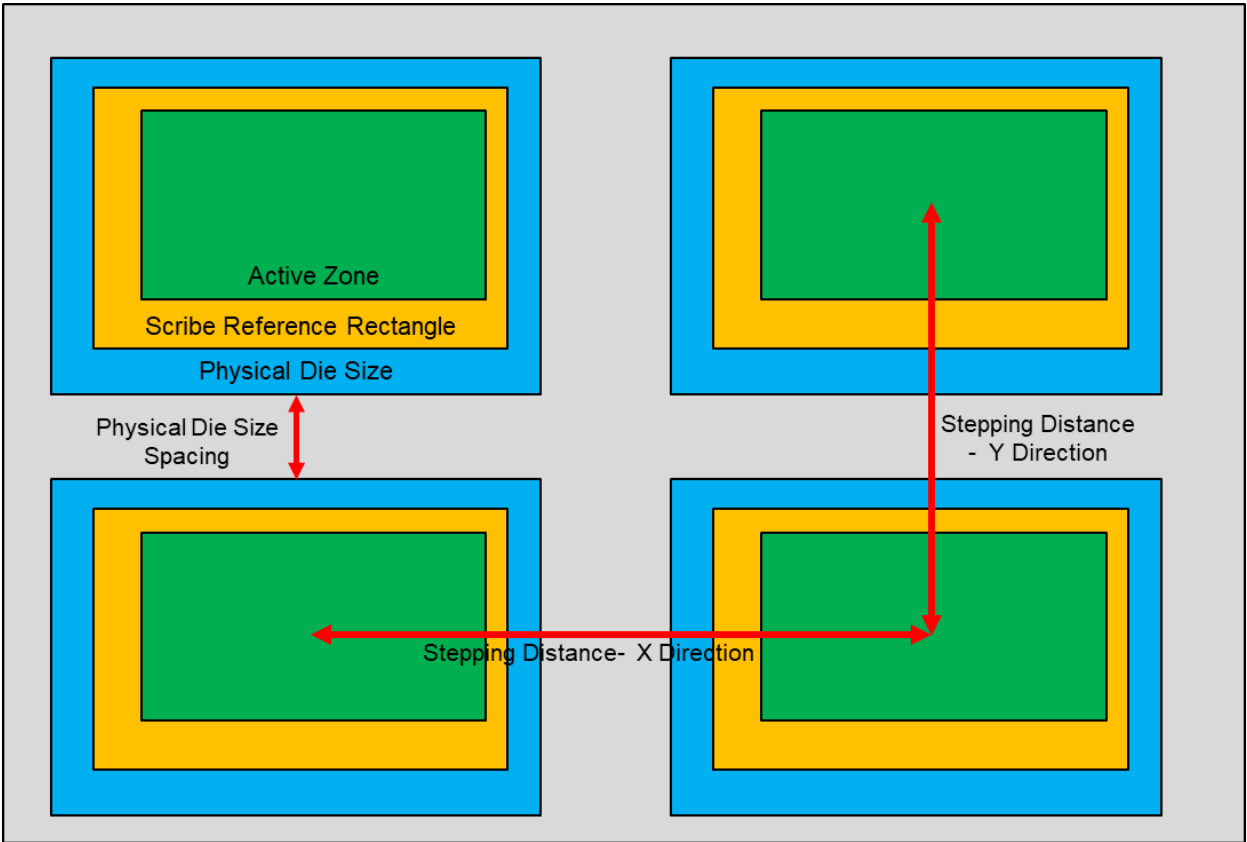


Figure 20 - Zone shapes for Physical Dies on a Wafer

6.2 Stepping Distance

path	PartModel/PackageSection/Die-Array/Die/SteppingDistance
diagram	
type	SteppingDistanceType.

As shown in Figure 20 above, the *SteppingDistance* can be a different value in the X and Y direction. This is sometimes referred to as the “Wafer Pitch”.

### 6.3 Mass

path	PartModel/PackageSection/Die-Array/Die/Mass
diagram	
type	JEP30-D10:MassValueSetGroupType, MassUOMType
group	ValueSetGroup

The *ValueSetGroup* is widely used throughout the Schema. It provides for all the following combinations in a structured way and enables the xml file to have all of the following elements directly underneath its parent element to which it is assigned to. :

- 1) Nominal,
- 2) Positive and Negative Tolerances, or Total tolerance based on the same UOM or based on a percentage of the nominal value, and
- 3) Minimum and or Maximum.

### 6.4 Part Height

path	PartModel/PackageSection/Die-Array/Die/Mass
diagram	
type	JEP30-D10:ValueSetGroupType.
group	ValueSetGroup

*PartHeight* for a *Die* represents the total height of the die part from its seating plane.



6.5 Terminal Groups

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups
diagram	
type	DieTerminalGroupsType, DieTerminalGroup-ArrayType, TerminalRegions-ArrayType, DieTerminalMaterial-ArrayType, CTE-ArrayType, DieTerminalShape-ArrayType, DieTerminalSpecification-ArrayType, DieTerminalDetail-ArrayType.

Let’s assume that we have a complex array of 12 instantiations of a UCle function organized in two columns and 6 rows as shown in Figure 21 below. Let’s define these 12 instantiations as Bank 1.

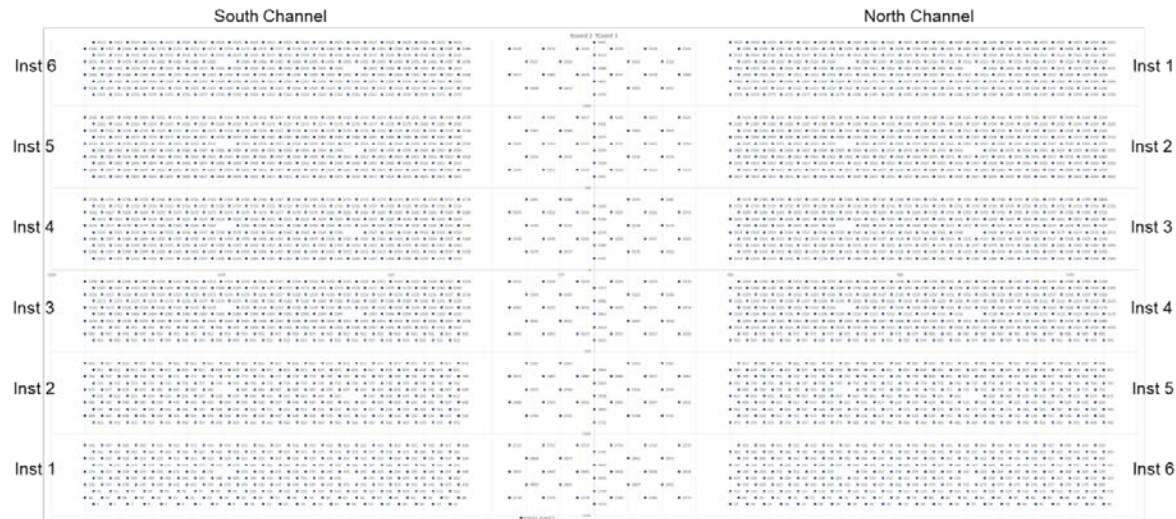


Figure 21 - Complex Array of 12 instantiations of a UCle Function

## 6.5 Terminal Groups (cont'd)

In the South channel, Inst 1 is duplicated 6 times in the Y direction. Then the entirety of the South channel is mirrored in both the x axis and y axis and offset to the right to create a duplicate North Channel.

There are many ways in which terminal numbering can be applied to this set of instantiations from maintaining a common numbering pattern for each instantiation or defining One contiguous sequential numbering pattern that covers all the terminals in Bank 1.

Let's also assume that each respective terminal location within each instantiation (the respective location is taken prior to any instantiations being mirrored), has the same StandardTerminalName, then, the assignment of properties, functions, terminal groupings to data/addresses busses, differential pairs, or any other logical groupings, only needs to be defined for one instantiation, and inherited by all other 11 instantiations.

The following sections will explore this example in more detail, to demonstrate the efficient compression of data, so that the resultant xml file remains as compact as possible.

### 6.5.1 Terminal Group - Array

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array	
diagram	<pre> classDiagram     class DieTerminalGroup-ArrayType {         TerminalGroup DieTerminalGroupType 1..∞         Relationship-to-ActiveZone Relationship-to-ActiveZoneType 0..∞         TerminalGroupToTerminalGroupRelationships DieTerminalGroupToTerminalGroupRelationshipsType 1..∞         Coplanarity xs:decimal     }     class constraints     DieTerminalGroup-ArrayType -- constraints     </pre>	
type	DieTerminalGroup-ArrayType, Relationship-to-ActiveZoneType, DieTerminalGroupToTerminalGroupRelationshipsType	

6.5.1.1 Terminal Group

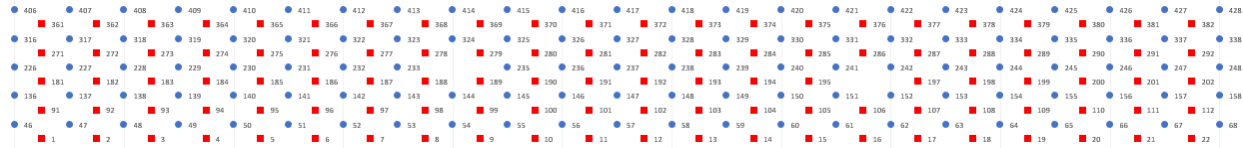
path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/TerminalGroup
diagram	<p>The diagram illustrates the structure of a TerminalGroup. It features a main class <b>TerminalGroup</b> (type <code>DieTerminalGroupType</code>) with a multiplicity of 1..∞. This class is associated with a <b>TerminalPosition</b> (type <code>DieTerminalPositionType</code>) and a <b>TerminalPattern-Array</b> (type <code>DieTerminalPattern-ArrayType</code>). The <b>TerminalPosition</b> class has an <b>ID</b> attribute (type <code>xs:string</code>) and is associated with a <b>DieTerminalPositionType</b> container. This container includes <b>Bottom</b> and <b>Upper</b> attributes, both of type <code>BottomUpperPositionConfigurationType</code>. The <b>TerminalPattern-Array</b> class is associated with a <b>DieTerminalPattern-ArrayType</b> container. This container includes a <b>TerminalPattern</b> (type <code>DieTerminalPatternType</code>) with a multiplicity of 1..∞, and a <b>PatternGroup</b> (type <code>DieTerminalPatternGroupType</code>) with a multiplicity of 0..∞. A <b>constraints</b> box is also present within the <b>DieTerminalPattern-ArrayType</b> container.</p>
type	DieTerminalGroupType, DieTerminalPositionType, BottomUpperPositionConfigurationType, DieTerminalPattern-ArrayType, DieTerminalPatternType, DieTerminalPatternGroupType.

6.5.1.1.1 Terminal Pattern

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/TerminalGroup/ TerminalPattern-Array/TerminalPattern
diagram	<p>The diagram illustrates the structure of the TerminalPattern class and its associated types. The TerminalPattern class is associated with the DieTerminalPatternType. It has a 1..∞ multiplicity. The DieTerminalPatternType is a base class for the StandardArray, CircularArray, and RandomArray classes. The StandardArray class is associated with the DieStandardArrayType. The DieStandardArrayType is a base class for the nx, ny, dx, dy, Angle, and TerminalPatternLowerLeftTerminalCenter classes. The nx, ny, dx, dy, and Angle classes are associated with the JEP30-D10:PointXYType. The TerminalPatternLowerLeftTerminalCenter class is associated with the JEP30-D10:PointXYType.</p> <pre>classDiagram     class TerminalPattern {         type DieTerminalPatternType     }     class DieTerminalPatternType {         ID xs:string         Name xs:string     }     class StandardArray {         type DieStandardArrayType     }     class CircularArray {         type CircularArrayType     }     class RandomArray {         type TerminalRandomArrayType     }     class DieStandardArrayType {         nx xs:integer         ny xs:integer         dx xs:decimal         dy xs:decimal         Angle xs:integer         TerminalPatternLowerLeftTerminalCenter JEP30-D10:PointXYType     }     class JEP30_D10_PointXYType {     }     TerminalPattern "1..∞" --&gt; "1" DieTerminalPatternType     DieTerminalPatternType -- &gt; StandardArray     DieTerminalPatternType -- &gt; CircularArray     DieTerminalPatternType -- &gt; RandomArray     StandardArray -- &gt; DieStandardArrayType     DieStandardArrayType -- &gt; nx     DieStandardArrayType -- &gt; ny     DieStandardArrayType -- &gt; dx     DieStandardArrayType -- &gt; dy     DieStandardArrayType -- &gt; Angle     DieStandardArrayType -- &gt; TerminalPatternLowerLeftTerminalCenter     nx -- &gt; JEP30_D10_PointXYType     ny -- &gt; JEP30_D10_PointXYType     dx -- &gt; JEP30_D10_PointXYType     dy -- &gt; JEP30_D10_PointXYType     Angle -- &gt; JEP30_D10_PointXYType     TerminalPatternLowerLeftTerminalCenter -- &gt; JEP30_D10_PointXYType</pre>
type	DieTerminalPatternType, DieStandardArrayType, JEP30-D10:PointXYType, CircularArrayType, TerminalRandomArrayType.

### 6.5.1.1.1 Terminal Pattern (cont'd)

A more detailed examination of a single UCle instantiation shows several distinct layout patterns of terminals, as shown by the various colored patterns in Figure 22 and Figure 23 below.



**Figure 22 - Single UCle Instantiation in South Channel Inst 1**



**Figure 23 - Power Nodes for South Channel Inst 1 & 2**

Each pattern can be represented by the standard array as shown in Table 36 below and in the subsequent xml structure.

**Table 36 - Pattern Details for the UCle Instantiation**

Terminal Pattern ID	nx	ny	dx	dy	Missing	
					Col Index	Row Index
1 (Blue circles)	22	5	50	80	16	3
2 (Red squares)	23	5	50	80	9	3
3 (Gold diamonds)	3	3	100	160	None	None
4 (Black triangles)	2	2	100	160	None	None
5 (Purple asterisks)	1	5	0	80	None	None
6 (Orange circles)	3	2	100	160	None	None
7 (Blue squares)	2	3	100	160	None	None

Note that some of the terminals are missing from the pattern ID's 1 and 2 which will be addressed in 6.5.7 Terminal Detail - Array below.

The VSS terminals as shown by Terminal pattern ID 5 (Purple asterisks) are shared between the South and North channels.

The VCCIO terminals as identified by Terminal patterns 3 (Gold diamonds) and 4 (Black triangles) for Instantiation 1 and Terminal patterns 6 (Orange circles) and 7 (Blue squares) for Instantiation 2 for the base of the repetitive pattern when applied to the rest of Bank 1.

### 6.5.1.1.1 Terminal Pattern (cont'd)

There may be additional VSS and VCCIO terminals intermingled throughout Terminal patterns 1 and 2, but there are treated as part of those patterns. Below is the xml representation of these Terminal Patterns.

```
<TerminalPattern-Array>
  <TerminalPattern>
    <ID>Terminal Pattern ID 1</ID>
    <Name>South Inst 1 Base</Name>
    <StandardArray>
      <nx>22</nx>
      <ny>5</ny>
      <dx>50.00</dx>
      <dy>80.00</dy>
      <TerminalPatternLowerLeftTerminalCenter>
        <x>-1475.00</x>
        <y>-1430.00</y>
      </TerminalPatternLowerLeftTerminalCenter>
    </StandardArray>
  </TerminalPattern>
  <TerminalPattern>
    <ID>Terminal Pattern ID 2</ID>
    <Name>South Inst 1 Staggered</Name>
    <StandardArray>
      <nx>23</nx>
      <ny>5</ny>
      <dx>50.00</dx>
      <dy>80.00</dy>
      <TerminalPatternLowerLeftTerminalCenter>
        <x>-1500.00</x>
        <y>-1390.00</y>
      </TerminalPatternLowerLeftTerminalCenter>
    </StandardArray>
  </TerminalPattern>
  <TerminalPattern>
    <ID>Terminal Pattern ID 3</ID>
    <Name>South Inst 1 VCCIO Base</Name>
    <StandardArray>
      <nx>3</nx>
      <ny>3</ny>
      <dx>100.00</dx>
      <dy>160.00</dy>
      <TerminalPatternLowerLeftTerminalCenter>
        <x>-250.00</x>
        <y>-1390.00</y>
      </TerminalPatternLowerLeftTerminalCenter>
    </StandardArray>
  </TerminalPattern>
  <TerminalPattern>
    <ID>Terminal Pattern ID 4</ID>
    <Name>South Inst 1 VCCIO Staggered</Name>
    <StandardArray>
      <nx>2</nx>
      <ny>2</ny>
      <dx>100.00</dx>
      <dy>160.00</dy>
      <TerminalPatternLowerLeftTerminalCenter>
        <x>-200.00</x>
        <y>-1310.00</y>
      </TerminalPatternLowerLeftTerminalCenter>
    </StandardArray>
  </TerminalPattern>
</TerminalPattern-Array>
```

### 6.5.1.1.1 Terminal Pattern (cont'd)

```

</TerminalPattern>
<TerminalPattern>
  <ID>Terminal Pattern ID 5</ID>
  <Name>VSS Center Inst 1</Name>
  <StandardArray>
    <nx>1</nx>
    <ny>5</ny>
    <dx>0</dx>
    <dy>80.00</dy>
    <TerminalPatternLowerLeftTerminalCenter>
      <x>0</x>
      <y>-1430.00</y>
    </TerminalPatternLowerLeftTerminalCenter>
  </StandardArray>
</TerminalPattern>
<TerminalPattern>
  <ID>Terminal Pattern ID 6</ID>
  <Name>South Inst 2 VCCIO Base</Name>
  <StandardArray>
    <nx>2</nx>
    <ny>3</ny>
    <dx>100.00</dx>
    <dy>160.00</dy>
    <TerminalPatternLowerLeftTerminalCenter>
      <x>-250.00</x>
      <y>-810.00</y>
    </TerminalPatternLowerLeftTerminalCenter>
  </StandardArray>
</TerminalPattern>
<TerminalPattern>
  <ID>Terminal Pattern ID 7</ID>
  <Name>South Inst 2 VCCIO Staggered</Name>
  <StandardArray>
    <nx>2</nx>
    <ny>3</ny>
    <dx>100.00</dx>
    <dy>160.00</dy>
    <TerminalPatternLowerLeftTerminalCenter>
      <x>-200.00</x>
      <y>-890.00</y>
    </TerminalPatternLowerLeftTerminalCenter>
  </StandardArray>
</TerminalPattern>
<TerminalPattern-Array>

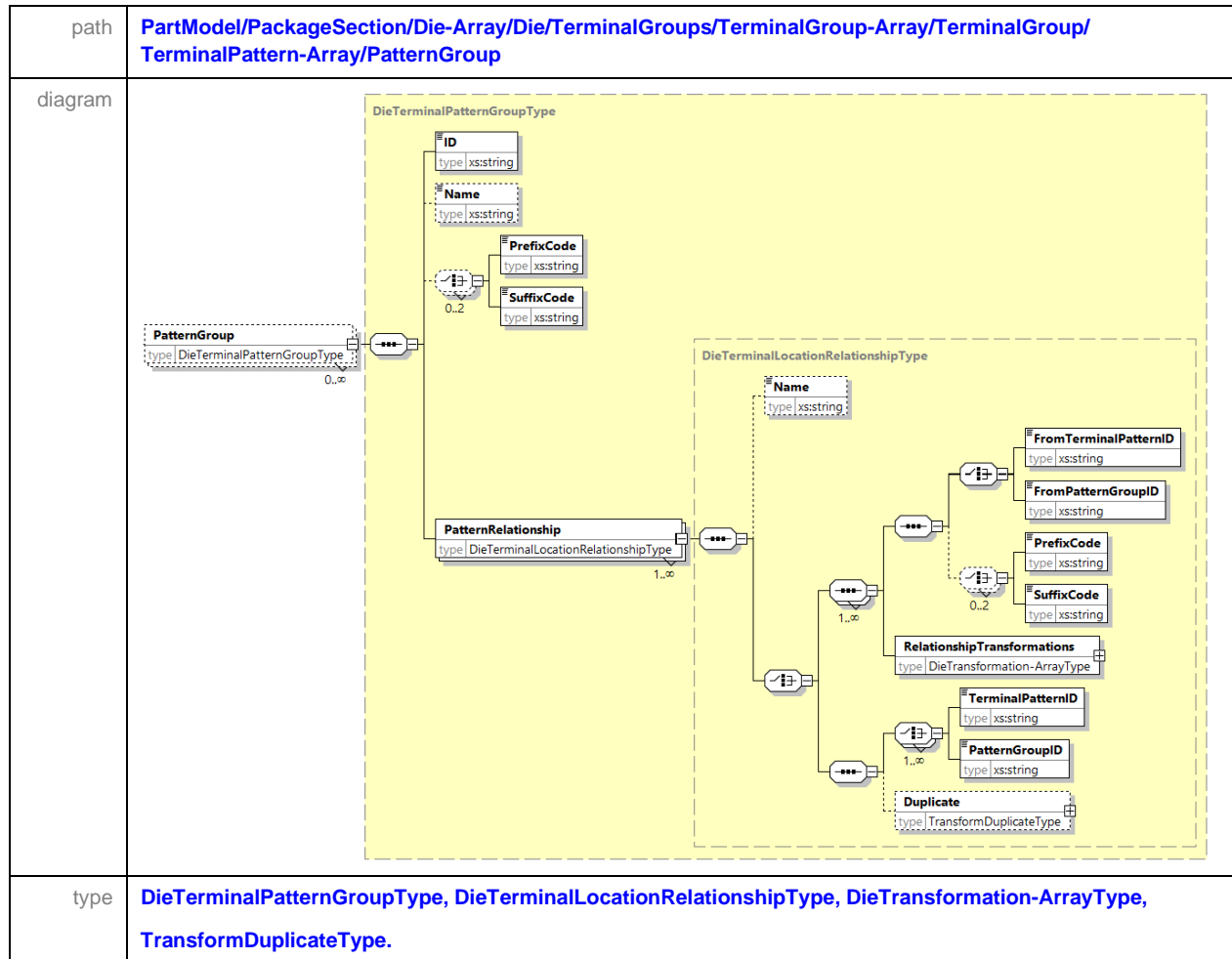
```

NOTE: The *TerminalPatternLowerLeftTerminalCenter* has been specified in each of these seven Terminal Patterns above, yet this is un-necessary, as the offset from one terminal pattern to the next terminal pattern could have been specified under section 6.5.1.2 Relationship – to – Active Zone below. The above *TerminalPatternLowerLeftTerminalCenter* coordinates are calculated based off the assumption that the entirety of Bank 1 is centered with the centered of the *ActiveZone*. In reality, Bank 1 may not be the final full configuration of all the die circuitry within the *ActiveZone*, and it may be just simply easier to adjust the final configuration with respect to the Active Zone instead of calculating all of the intermediate coordinates for every *TerminalPatternLowerLeftTerminalCenter*.

### 6.5.1.1.1 Terminal Pattern (cont'd)

Alternatively, *TerminalPatterns* can be grouped under a *PatternGroup* and their relationship between each other been determined via the *Offset* values. The *PatternGroup* could then be placed with respect to the *ActiveZone* under section 6.5.1.2 below.

### 6.5.1.1.2 Pattern Group



Before creating any *PatternGroup*, it is best to understand the most efficient form of defining the grouping to represent the duplication for the North and South channels as define in Figure 21 - Complex Array of 12 instantiations of a UCle Function above.

- Terminal Pattern ID 1 and 2 across all 12 instantiations of the UCle in the Bank 1 example have a contiguous sequential terminal numbering.
- Terminal Pattern ID 5 is a shared set of VSS terminals between the South and North channels and is not duplicated when a copy of the South channel is duplicated to form the North channel. For this reason, Terminal Pattern ID 5 does not form part of the *PatternGroup* that will be used later to create the North channel. However, Terminal Pattern ID 5 continues the terminal numbering from the Back to the Front for Bank 1 for that singular column of terminals.



### 6.5.1.1.2 Pattern Group (cont'd)

- The remaining terminals as defined within Terminal Pattern ID 3, 4, 6 and 7 across all instantiations in Bank 1 follow the S-H (Snake – Horizontal) pattern and continue the terminal numbering sequence from the shared VSS terminals.

path	<a href="#">PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalPattern-Array/PatternGroup/PatternRelationship/RelationshipTransformations</a>
diagram	<p>The diagram illustrates the XSD structure for <b>RelationshipTransformations</b>. The root element is <b>RelationshipTransformations</b> (type <b>DieTransformation-ArrayType</b>). It contains two main elements: <b>Mirror</b> (type <b>TransformMirrorType</b>) and <b>Rotate</b> (type <b>TransformRotateType</b>). The <b>Mirror</b> element has four attributes: <b>ToTerminalPatternID</b> (type <b>xs:string</b>), <b>ToPatternGroupID</b> (type <b>xs:string</b>), <b>PrefixCode</b> (type <b>xs:string</b>), and <b>SuffixCode</b> (type <b>xs:string</b>), with a cardinality of 0..2. The <b>Rotate</b> element has five attributes: <b>Origin</b> (type <b>JEP30-D10:EmptyType</b>), <b>SelectionCenter</b> (type <b>JEP30-D10:EmptyType</b>), <b>Horizontal</b> (type <b>JEP30-D10:EmptyType</b>), <b>Vertical</b> (type <b>JEP30-D10:EmptyType</b>), and <b>Coordinate</b> (type <b>JEP30-D10:PointXYType</b>). The <b>TransformMirrorType</b> and <b>TransformRotateType</b> elements are shown as dashed boxes containing their respective attributes. The <b>JEP30-D10:PointXYType</b> element is shown as a dashed box containing <b>x</b> (type <b>xs:decimal</b>) and <b>y</b> (type <b>xs:decimal</b>).</p>
type	<b>DieTransformation-ArrayType</b> , <b>TransformMirrorType</b> , <b>JEP30-D10:EmptyType</b> , <b>JEP30-D10:PointXYType</b> , <b>TransformRotateType</b> .

#### 6.5.1.1.2 Pattern Group (cont'd)

[illegible]

Option 1 – Leveraging Pattern Groups for the purpose of applying Terminal numbering.

- Group *TerminalPattern* ID 1 and 2 into a single *PatternGroup* ID 1 called “South\_Inst1\_TrmlPtrn1-2”.
- Create an array of *PatternGroup* ID 1 “South\_Inst1\_TrmlPtrn1-2” six times in the Y direction via a new *PatternGroup* ID 2 called “South\_Inst1-6\_PtrnGrp1”.
- Duplicate *PatternGroup* ID 2 “South\_Inst1-6\_TrmlPtrn1-2” one time in the X direction into a new *PatternGroup* ID 3 called “South-North\_Inst1-6\_PtrnGrp2”, ensuring that you mirror in both the horizontal and vertical axis for the North channel.
  - This enables the assignment of a unique sequence of terminal numbering to be applied specifically to the terminals defined within *PatternGroup* ID 3.
- Group *TerminalPattern* ID 3 and 4 into a single *PatternGroup* ID 4 called “South\_Inst1\_TrmlPtrn3-4”.
- Create an array of *PatternGroup* ID 4 “South\_Inst1, 3, 5\_PtrnGrp4” three times in the Y direction into a new *PatternGroup* ID 5 “South\_Inst1, 3, 5\_PtrnGrp4”.
- Group *TerminalPattern* ID 6 and 7 into a single *PatternGroup* ID 6 called “South\_Inst2\_TrmlPtrn6-7”.
- Create an array of *PatternGroup* ID 6 “South\_Inst2, 4, 6\_PtrnGrp6” three times in the Y direction into a new *PatternGroup* ID 7 “South\_Inst2, 4, 6\_PtrnGrp6”.
- Group *PatternGroup* ID 5 and 7 into a single *PatternGroup* ID 8 called “South\_Inst1-6\_PtrnGrp5, 7”
- Duplicate *PatternGroup* ID 8 “South\_Inst1-6\_PtrnGrp5, 7” one time in the X direction into a new *PatternGroup* ID 9 called “South-North\_Inst1-6\_PtrnGrp8”, with a straight offset and without rotation or mirroring.
  - This enables the assignment of a unique sequence of terminal numbering to be applied specifically to the terminals defined within *PatternGroup* ID 9.
- Create an array of *TerminalPattern* ID 5 “VSS Center Inst 1” six times in the Y direction via a new *PatternGroup* ID 10 called “VSS Center Inst 1-6”.
  - This enables the assignment of a unique sequence of terminal numbering to be applied specifically to the terminals defined within *PatternGroup* ID 10.
- Group *PatternGroup* ID 3, 9 and 10 into a single *PatternGroup* ID 11 called “Bank 1”.

### 6.5.1.1.2 Pattern Group (cont'd)

**Table 37 - Pattern Group Construction**

Pattern Group ID	Name	Pattern Relationship	nx	ny	Pitch (dx) / Offset (x)	Pitch (dy) / Offset (y)
1	South_Inst1_TrmlPtrn1-2	Terminal Pattern ID 1 Terminal Pattern ID 2				
2	South_Inst1-6_PtrnGrp1	Duplicate Pattern Group ID 1	1	6	0	500.00
3	South-North_Inst1-6_PtrnGrp2	Offset & Mirror (H & V) Pattern Group ID 2			1900.00	0
4	South_Inst1_TrmlPtrn3-4	Terminal Pattern ID 3 Terminal Pattern ID 4				
5	South_Inst1, 3, 5_PtrnGrp2	Duplicate Pattern Group ID 4	1	3	0	1000.00
6	South_Inst2_TrmlPtrn6-7	Terminal Pattern ID 6 Terminal Pattern ID 7				
7	South_Inst2, 4, 6_PtrnGrp6	Duplicate Pattern Group ID 6	1	3	0	1000.00
8	South_Inst1-6_PtrnGrp5, 7	Pattern Group ID 5 Pattern Group ID 7				
9	South-North_Inst1-6_PtrnGrp8	Offset & Mirror (H & V) Pattern Group ID 2			300.00	0
10	VSS Center Inst 1-6	Terminal Pattern ID 5	1	6	0	500.00
11	Bank 1	Pattern Group ID 3 Pattern Group ID 9 Pattern Group ID 10				

```

<PatternGroup>
  <ID>Pattern Group ID 1</ID>
  <Name>South_Inst1_TrmlPtrn1-2</Name>
  <PatternRelationship>
    <TerminalPatternID>Terminal Pattern ID 1</TerminalPatternID>
    <TerminalPatternID>Terminal Pattern ID 2</TerminalPatternID>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 2</ID>
  <Name>South_Inst1-6_PtrnGrp1</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 1</PatternGroupID>
    <Duplicate>
      <nx>1</nx>
      <ny>6</ny>
      <Y-PrefixCode>Inst 1</Y-PrefixCode>
      <Y-PrefixCode>Inst 2</Y-PrefixCode>
      <Y-PrefixCode>Inst 3</Y-PrefixCode>
      <Y-PrefixCode>Inst 4</Y-PrefixCode>
      <Y-PrefixCode>Inst 5</Y-PrefixCode>
      <Y-PrefixCode>Inst 6</Y-PrefixCode>
      <dy>500.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>

```

### 6.5.1.1.2 Pattern Group (cont'd)

```

<PatternGroup>
  <ID>Pattern Group ID 3</ID>
  <Name>South-North_Inst1-6_PtrnGrp2</Name>
  <PatternRelationship>
    <FromPatternGroupID>Pattern Group ID 2</FromPatternGroupID>
    <RelationshipTransformations>
      <ToPatternGroupID>Pattern Group ID 2</ToPatternGroupID>
      <Offset>
        <x>1900</x>
        <y>0</y>
      </Offset>
      <Mirror>
        <SelectionCenter/>
        <Horizontal/>
      </Mirror>
      <Mirror>
        <SelectionCenter/>
        <Vertical/>
      </Mirror>
    </RelationshipTransformations>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 4</ID>
  <Name>South_Inst1_Trm1Ptrn3-4</Name>
  <PatternRelationship>
    <TerminalPatternID>Terminal Pattern ID 3</TerminalPatternID>
    <TerminalPatternID>Terminal Pattern ID 4</TerminalPatternID>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 5</ID>
  <Name>South_Inst1, 3, 5_PtrnGrp4</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 4</PatternGroupID>
    <Duplicate>
      <nx>1</nx>
      <ny>3</ny>
      <Y-PrefixCode>Inst 1</Y-PrefixCode>
      <Y-PrefixCode>Inst 3</Y-PrefixCode>
      <Y-PrefixCode>Inst 5</Y-PrefixCode>
      <dy>1000.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 6</ID>
  <Name>South_Inst2_Trm1Ptrn6-7</Name>
  <PatternRelationship>
    <TerminalPatternID>Terminal Pattern ID 6</TerminalPatternID>
    <TerminalPatternID>Terminal Pattern ID 7</TerminalPatternID>
  </PatternRelationship>
</PatternGroup>

```

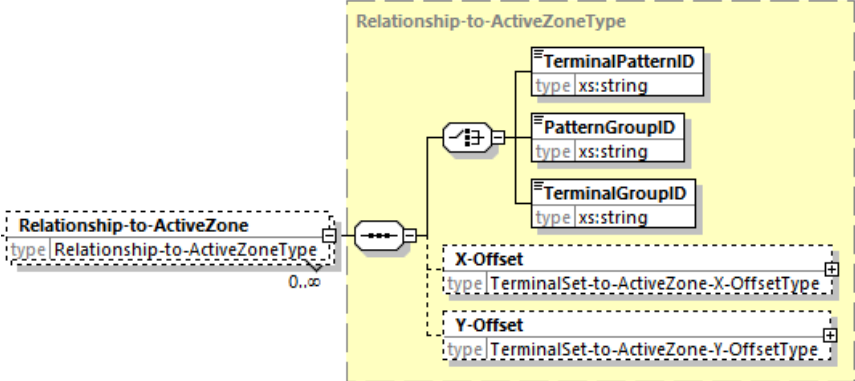
### 6.5.1.1.2 Pattern Group (cont'd)

```
<PatternGroup>
  <ID>Pattern Group ID 7</ID>
  <Name>South_Inst2, 4, 6_PtrnGrp6</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 6</PatternGroupID>
    <Duplicate>
      <nx>1</nx>
      <ny>3</ny>
      <Y-PrefixCode>Inst 2</Y-PrefixCode>
      <Y-PrefixCode>Inst 4</Y-PrefixCode>
      <Y-PrefixCode>Inst 6</Y-PrefixCode>
      <dy>1000.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 8</ID>
  <Name>South_Inst1-6_PtrnGrp5, 7</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 5</PatternGroupID>
    <PatternGroupID>Pattern Group ID 7</PatternGroupID>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 9</ID>
  <Name>South-North_Inst1-6_PtrnGrp8</Name>
  <PatternRelationship>
    <FromPatternGroupID>Pattern Group ID 8</FromPatternGroupID>
    <RelationshipTransformations>
      <ToPatternGroupID>Pattern Group ID 8</ToPatternGroupID>
      <Offset>
        <x>300</x>
        <y>0</y>
      </Offset>
    </RelationshipTransformations>
  </PatternRelationship>
</PatternGroup>
<PatternGroup>
  <ID>Pattern Group ID 10</ID>
  <Name>VSS Center Inst 1-6</Name>
  <PatternRelationship>
    <TerminalPatternID>Terminal Pattern ID 5</TerminalPatternID>
    <Duplicate>
      <nx>1</nx>
      <ny>6</ny>
      <dy>500.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>
```

6.5.1.1.2      Pattern Group (cont'd)

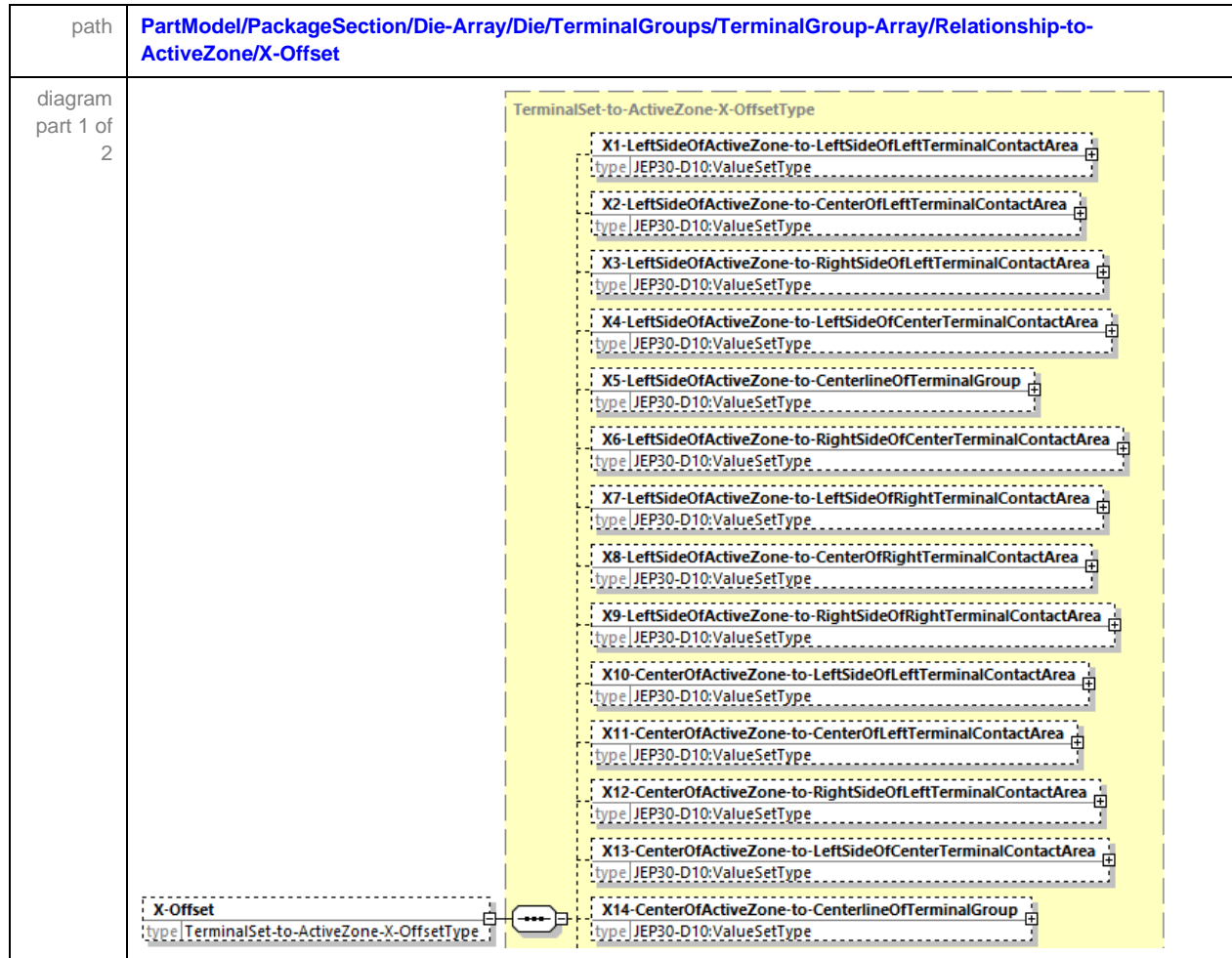
```
<PatternGroup>
  <ID>Pattern Group ID 11</ID>
  <Name>Bank 1</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 3</PatternGroupID>
    <PatternGroupID>Pattern Group ID 9</PatternGroupID>
    <PatternGroupID>Pattern Group ID 10</PatternGroupID>
  </PatternRelationship>
  :
  :
</PatternGroup>
```

6.5.1.2      Relationship – to – Active Zone

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/Relationship-to-ActiveZone
diagram	
type	Relationship-to-ActiveZoneType, TerminalSet-to-ActiveZone-X-OffsetType, TerminalSet-to-ActiveZone-Y-OffsetType

For non-symmetrical Terminals around the *ActiveZone* center, offset dimensions may be referenced from the Terminal Contact Area to the *ActiveZone* area. This data can be captured in the *Relationship-to-ActiveZone* section.

### 6.5.1.2.1 X-Offset





6.5.1.2.1 X-Offset (cont'd)

diagram part 2 of 2	<div><div>X-Offset</div><div>type TerminalSet-to-ActiveZone-X-OffsetType</div><div><div>X14-CenterOfActiveZone-to-CenterlineOfTerminalGroup</div><div>type JEP30-D10:ValueSetType</div><div>X15-CenterOfActiveZone-to-RightSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X16-CenterOfActiveZone-to-LeftSideOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X17-CenterOfActiveZone-to-CenterOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X18-CenterOfActiveZone-to-RightSideOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X19-RightSideOfActiveZone-to-LeftSideOfLeftTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X20-RightSideOfActiveZone-to-CenterOfLeftTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X21-RightSideOfActiveZone-to-RightSideOfLeftTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X22-RightSideOfActiveZone-to-LeftSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X23-RightSideOfActiveZone-to-CenterlineOfTerminalGroup</div><div>type JEP30-D10:ValueSetType</div><div>X24-RightSideOfActiveZone-to-RightSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X25-RightSideOfActiveZone-to-LeftSideOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X26-RightSideOfActiveZone-to-CenterOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div><div>X27-RightSideOfActiveZone-to-RightSideOfRightTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div></div>
type	TerminalSet-to-ActiveZone-X-OffsetType, JEP30-D10:ValueSetType

Table 38 - Terminal Group to Active Zone X-Offset lists out all the various dimensions that can be selected from any major point of the *ActiveZone* to any point of the terminal contact area in X direction.

Center of Center Terminal Contact Area is the same as Centerline of Terminal Group, however the use of the “...Centerline of Terminal Group” is more encompassing since it also includes the centerline of a row or column of an even number of terminals, i.e., midway between terminal 2 and terminal 3 of 4 terminals in a row or column.

#### 6.5.1.2.1 X-Offset (cont'd)

**Table 38 - Terminal Group to Active Zone X-Offset**

Symbol	Description
X1	Left side of Active Zone to Left Side of Left Terminal Contact Area
X2	Left side of Active Zone to Center of Left Terminal Contact Area
X3	Left side of Active Zone to Right side of Left Terminal Contact Area
X4	Left side of Active Zone to Left Side of Center Terminal Contact Area
X5	Left side of Active Zone to Centerline of Terminal Group
X6	Left side of Active Zone to Right side of Center Terminal Contact Area
X7	Left side of Active Zone to Left side of Right Terminal Contact Area
X8	Left side of Active Zone to Center of Right Terminal Contact Area
X9	Left side of Active Zone to Right side of Right Terminal Contact Area
X10	Center of Active Zone to Left side of Left Terminal Contact Area
X11	Center of Active Zone to Center of Left Terminal Contact Area
X12	Center of Active Zone to Right side of Left Terminal Contact Area
X13	Center of Active Zone to Left side of Center Terminal Contact Area
X14	Center of Active Zone to Centerline of Terminal Group
X15	Center of Active Zone to Right side of Center Terminal Contact Area
X16	Center of Active Zone to Left side of Right Terminal Contact Area
X17	Center of Active Zone to Center of Right Terminal Contact Area
X18	Center of Active Zone to Right side of Right Terminal Contact Area
X19	Right side of Active Zone to Left side of Left Terminal Contact Area
X20	Right side of Active Zone to Center of Left Terminal Contact Area
X21	Right side of Active Zone to Right side of Left Terminal Contact Area
X22	Right side of Active Zone to Left side of Center Terminal Contact Area
X23	Right side of Active Zone to Centerline of Terminal Group
X24	Right side of Active Zone to Right side of Center Terminal Contact Area
X25	Right side of Active Zone to Left side of Right Terminal Contact Area
X26	Right side of Active Zone to Center of Right Terminal Contact Area
X27	Right side of Active Zone to Right side of Right Terminal Contact Area

6.5.1.2.2 Y-Offset

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/Relationship-to-ActiveZone/Y-Offset	
diagram part 1 of 2	<div><div>TerminalSet-to-ActiveZone-Y-OffsetType</div><div><div>Y1-BackSideOfActiveZone-to-BackSideOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y2-BackSideOfActiveZone-to-CenterOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y3-BackSideOfActiveZone-to-FrontSideOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y4-BackSideOfActiveZone-to-BackSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y5-BackSideOfActiveZone-to-CenterlineOfTerminalGroup</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y6-BackSideOfActiveZone-to-FrontSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y7-BackSideOfActiveZone-to-BackSideOfFrontTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y8-BackSideOfActiveZone-to-CenterOfFrontTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y9-BackSideOfActiveZone-to-FrontSideOfFrontTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y10-CenterOfActiveZone-to-BackSideOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y11-CenterOfActiveZone-to-CenterOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y12-CenterOfActiveZone-to-FrontSideOfBackTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y13-CenterOfActiveZone-to-BackSideOfCenterTerminalContactArea</div><div>type JEP30-D10:ValueSetType</div></div><div><div>Y14-CenterOfActiveZone-to-CenterlineOfTerminalGroup</div><div>type JEP30-D10:ValueSetType</div></div></div> <div><div>Y-Offset</div><div>type TerminalSet-to-ActiveZone-Y-OffsetType</div></div> <div><div></div><div></div></div>	

### 6.5.1.2.2 Y-Offset (cont'd)

<p>diagram part 2 of 2</p>	
<p>type</p>	<p><b>TerminalSet-to-ActiveZone-X-OffsetType, JEP30-D10:ValueSetType</b></p>

Table 39 - Terminal Group to Active Zone Y-Offset lists out all the various dimensions that can be selected from any major point of the *ActiveZone* to any point of the terminal contact area in X direction.

### 6.5.1.2.2 Y-Offset (cont'd)

**Table 39 - Terminal Group to Active Zone Y-Offset**

Symbol	Description
Y1	Back side of Active Zone to Back side of Back Terminal Contact Area
Y2	Back side of Active Zone to Center of Back Terminal Contact Area
Y3	Back side of Active Zone to Front side of Back Terminal Contact Area
Y4	Back side of Active Zone to Back side of Center Terminal Contact Area
Y5	Back side of Active Zone to Centerline of Terminal Group
Y6	Back side of Active Zone to Front side of Center Terminal Contact Area
Y7	Back side of Active Zone to Back side of Front Terminal Contact Area
Y8	Back side of Active Zone to Center of Front Terminal Contact Area
Y9	Back side of Active Zone to Front side of Front Terminal Contact Area
Y10	Center of Active Zone to Back side of Back Terminal Contact Area
Y11	Center of Active Zone to Center of Back Terminal Contact Area
Y12	Center of Active Zone to Front side of Back Terminal Contact Area
Y13	Center of Active Zone to Back side of Center Terminal Contact Area
Y14	Center of Active Zone to Centerline of Terminal Group
Y15	Center of Active Zone to Front side of Center Terminal Contact Area
Y16	Center of Active Zone to Back side of Front Terminal Contact Area
Y17	Center of Active Zone to Center of Front Terminal Contact Area
Y18	Center of Active Zone to Front side of Front Terminal Contact Area
Y19	Front side of Active Zone to Back side of Back Terminal Contact Area
Y20	Front side of Active Zone to Center of Back Terminal Contact Area
Y21	Front side of Active Zone to Front side of Back Terminal Contact Area
Y22	Front side of Active Zone to Back side of Center Terminal Contact Area
Y23	Front side of Active Zone to Centerline of Terminal Group
Y24	Front side of Active Zone to Front side of Center Terminal Contact Area
Y25	Front side of Active Zone to Back side of Front Terminal Contact Area
Y26	Front side of Active Zone to Center of Front Terminal Contact Area
Y27	Front side of Active Zone to Front side of Front Terminal Contact Area

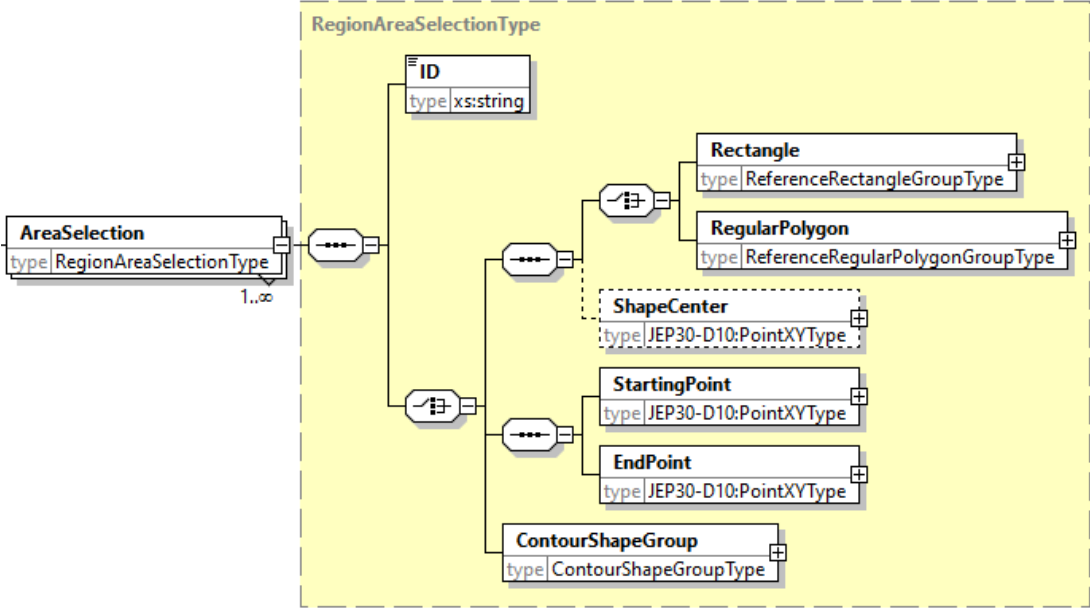
### 6.5.1.3 Terminal Group to Terminal Group Relationships

path	<a href="#">PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalGroup-Array/TerminalGroupToTerminalGroupRelationships</a>
diagram	
type	<a href="#">DieTerminalGroupToTerminalGroupRelationshipsType</a> , <a href="#">JEP30-D10:PointXYType</a>

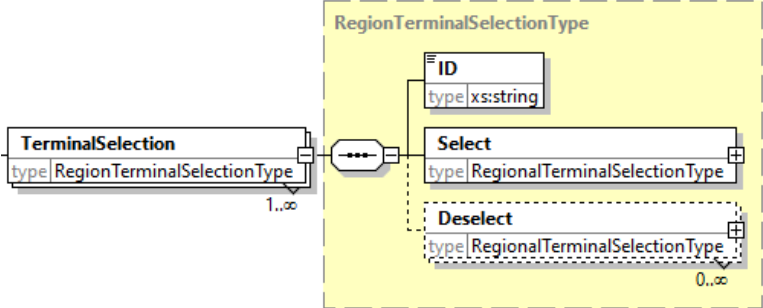
### 6.5.2 Terminal Regions - Array

path	<a href="#">PartModel/PackageSection/Die-Array/Die/TerminalGroups/Regions-Array</a>
diagram	
type	<a href="#">Region-ArrayType</a> , <a href="#">RegionType</a> , <a href="#">RegionAreaSelectionType</a> , <a href="#">RegionTerminalSelectionType</a> .

### 6.5.2.1 Area Selection

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/Regions-Array/Region/AreaSelection
diagram	 <p>The diagram illustrates the structure of the <b>AreaSelection</b> element. It is a container for one or more <b>RegionAreaSelectionType</b> elements (indicated by the <b>1..∞</b> cardinality). Each <b>RegionAreaSelectionType</b> is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b>: A required element of type <b>xs:string</b>.</li> <li><b>Rectangle</b>: An optional element of type <b>ReferenceRectangleGroupType</b>.</li> <li><b>RegularPolygon</b>: An optional element of type <b>ReferenceRegularPolygonGroupType</b>.</li> <li><b>ShapeCenter</b>: An optional element of type <b>JEP30-D10:PointXYType</b>.</li> <li><b>StartingPoint</b>: An optional element of type <b>JEP30-D10:PointXYType</b>.</li> <li><b>EndPoint</b>: An optional element of type <b>JEP30-D10:PointXYType</b>.</li> <li><b>ContourShapeGroup</b>: An optional element of type <b>ContourShapeGroupType</b>.</li> </ul>
type	<b>RegionAreaSelectionType</b> , <b>ReferenceRectangleGroupType</b> , <b>ReferenceRegularPolygonGroupType</b> , <b>JEP30-D10:PointXYType</b> , <b>ContourShapeGroupType</b> .

### 6.5.2.2 Terminal Selection

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/Regions-Array/Region/TerminalSelection
diagram	 <p>The diagram illustrates the structure of the <b>TerminalSelection</b> element. It is a container for one or more <b>RegionTerminalSelectionType</b> elements (indicated by the <b>1..∞</b> cardinality). Each <b>RegionTerminalSelectionType</b> is a complex type containing the following elements:</p> <ul style="list-style-type: none"> <li><b>ID</b>: A required element of type <b>xs:string</b>.</li> <li><b>Select</b>: An optional element of type <b>RegionalTerminalSelectionType</b>.</li> <li><b>Deselect</b>: An optional element of type <b>RegionalTerminalSelectionType</b>.</li> </ul>
type	<b>RegionTerminalSelectionType</b> , <b>RegionalTerminalSelectionType</b> .

6.5.2.2.1 Select

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/Regions-Array/Region/TerminalSelection/Select, PartModel/PackageSection/Die-Array/Die/TerminalGroups/Regions-Array/Region/TerminalSelection/Deselect
diagram	<p>The diagram illustrates the structure of the <code>RegionalTerminalSelectionType</code> XSD. It is a complex type with the following elements:</p> <ul style="list-style-type: none"><li><code>TerminalPatternID</code> (type: <code>xs:string</code>)</li><li><code>PatternGroupID</code> (type: <code>xs:string</code>)</li><li><code>TerminalGroupID</code> (type: <code>xs:string</code>)</li><li><code>RowTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>FromRowTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>ToRowTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>ColumnTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>FromColumnTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>ToColumnTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>PolarTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>FromPolarTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>ToPolarTerminalIndex</code> (type: <code>JEP30-D10:MinIntegerOfOneType</code>, minIncl/maxIncl: 1)</li><li><code>FromTerminalNumber</code> (type: <code>xs:integer</code>)</li><li><code>ToTerminalNumber</code> (type: <code>xs:integer</code>)</li></ul> <p>The diagram is enclosed in a dashed box labeled <code>RegionalTerminalSelectionType</code>. The <code>Select</code> element is shown with a type of <code>RegionalTerminalSelectionType</code>.</p>
type	<code>RegionalTerminalSelectionType</code> .



6.5.3 Terminal Materials - Array

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalMaterial-Array
diagram	<p>The diagram illustrates the XSD structure for the <code>TerminalMaterial-Array</code> element. It is contained within the <code>DieTerminalMaterial-ArrayType</code> namespace. The <code>TerminalMaterial-Array</code> element is of type <code>DieTerminalMaterial-ArrayType</code> and contains one or more <code>TerminalMaterial</code> elements (indicated by <code>1..∞</code>). The <code>TerminalMaterial</code> element is of type <code>DieTerminalMaterialType</code> and contains the following elements: <code>ID</code> (type <code>xs:string</code>), <code>Name</code> (type <code>xs:string</code>), <code>J-Std-609AlloyComposition</code> (type <code>J-Std-609AlloyCompositionType</code>), <code>J-Std-609e-code</code> (type <code>J-Std-609e-codeType</code>), <code>BaseMaterial</code> (type <code>BaseMaterialType</code>), <code>OtherBaseMaterial</code> (type <code>xs:string</code>), and <code>Plating-Array</code> (type <code>Plating-ArrayType</code>).</p>
type	<code>DieTerminalMaterial-ArrayType</code> , <code>DieTerminalMaterialType</code> , <code>J-Std-609AlloyCompositionType</code> , <code>J-Std-609e-codeType</code> , <code>BaseMaterialType</code> , <code>Plating-ArrayType</code> .

The *J-Std-609AlloyComposition* and *J-Std-609e-code* enumerated values are defined in “Annex A (informative) Example Alloys and Associated Material Codes” in the J-Std-609, “Marking, Symbols, and Labels of Leaded and Lead-Free Terminal Finished Materials Used in Electronic Assembly” standard.

The is *PlatingThicknessUOM* specified in *um*.

#### 6.5.4 CTE - Array

path	<a href="#">PartModel/PackageSection/Die-Array/Die/TerminalGroups/CTE-Array</a>
diagram	
type	<a href="#">ReferenceCTE-ArrayType</a> , <a href="#">ReferenceCTEType</a> , <a href="#">TemperatureConditionType</a> , <a href="#">JEP30-D10:CTE-UOMType</a> , <a href="#">CTE-GraphType</a> .

#### 6.5.5 Terminal Shape - Array

path	<a href="#">PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalShape-Array</a>
diagram	
type	<a href="#">DieTerminalShape-ArrayType</a> , <a href="#">DieTerminalShapeType</a> , <a href="#">DieTerminalStructuralType</a> , <a href="#">DieTerminalSoldermaskOpeningType</a> , <a href="#">DieTerminalPadType</a> .

The [TerminalShape-Array](#) for Die's is a subset of the shapes defined in section 5.13.1.7 Terminal Shape above.

6.5.5.1 Terminal

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalShape-Array/TerminalShape/Terminal
diagram	
type	DieTerminalStructuralType, JEP30-D10:EmptyType, ReferenceRectangleGroupType, ReferenceRoundedRectangleGroupType, ReferenceModifiedRectangleGroupType, ReferenceCircleGroupType, ReferenceDouble-DGroupType, ReferenceRegularPolygonType, ReferenceTerminalContourGroupType, JEP30-D10:UnspecifiedDimensionalValueSetGroupType.

6.5.5.2      Soldermask Opening

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalShape-Array/TerminalShape/SoldermaskOpening
diagram	<p>The diagram illustrates the structure of the <b>SoldermaskOpening</b> class. It is a class of type <b>DieTerminalSoldermaskOpeningType</b>. The class contains a dashed box representing a group of elements. Inside this box, there is a <b>SoldermaskThickness</b> element of type <b>JEP30-D10:UnspecifiedDimensionalValueSetGroupType</b>. Below this, there is a list of shape elements, each with a reference type: <b>Rectangle</b> (type <b>ReferenceRectangleGroupType</b>), <b>RoundedRectangle</b> (type <b>ReferenceRoundedRectangleGroupType</b>), <b>ModifiedRectangle</b> (type <b>ReferenceModifiedRectangleGroupType</b>), <b>Circle</b> (type <b>ReferenceCircleGroupType</b>), <b>Double-D</b> (type <b>ReferenceDouble-DGroupType</b>), <b>RegularPolygon</b> (type <b>ReferenceRegularPolygonType</b>), and <b>Contour</b> (type <b>ReferenceTerminalContourGroupType</b>). The diagram uses standard UML notation for class relationships and groupings.</p>
type	DieTerminalSoldermaskOpeningType, JEP30-D10:UnspecifiedDimensionalValueSetGroupType, ReferenceRectangleGroupType, ReferenceRoundedRectangleGroupType, ReferenceModifiedRectangleGroupType, ReferenceCircleGroupType, ReferenceDouble-DGroupType, ReferenceRegularPolygonType, ReferenceTerminalContourGroupType.

6.5.5.3 Pad

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalShape-Array/TerminalShape/Pad
diagram	
type	DieTerminalPadType, JEP30-D10:UnspecifiedDimensionalValueSetGroupType, ReferenceRectangleGroupType, ReferenceRoundedRectangleGroupType, ReferenceModifiedRectangleGroupType, ReferenceCircleGroupType, ReferenceDouble-DGroupType, ReferenceRegularPolygonType, ReferenceTerminalContourGroupType.

6.5.6 Terminal Specification - Array

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalSpecification-Array
diagram	
type	TerminalSpecification-ArrayType, TerminalSpecificationType, JEP30-D10:EmptyType

6.5.7 Terminal Detail - Array

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalDetail-Array
diagram	
type	DieTerminalDetail-ArrayType, DieTerminalDetailType, DieFirstTerminalLocationType, DieTerminalNumberPatternType, DieTerminalDetailExceptionsType.

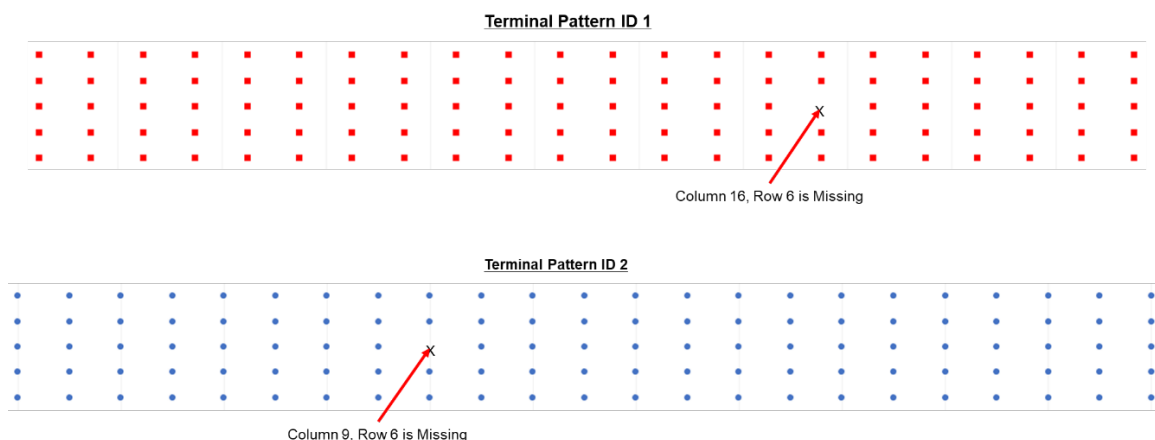
6.5.7.1 Terminal Detail

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalDetail-Array/TerminalDetail
diagram	
type	DieTerminalDetailType, TerminalCenterType, JEP30-D10:EmptyType, JEP30-D10:MinIntegerOfOneType, DieTerminalStatusType.

Similar to section 5.13.3.1 Terminal Detail above, this section identifies the relationship between the electrical *TerminalNumber* and the physical terminal location in the *ActiveZone*. The physical terminal location can be defined by a set of x,y coordinates, or by a combination of two choices, where the first choice is the identification of either the *TerminalPattern*, *PatternGroup* or the *TerminalGroup*, and the second choice is the combination of *RowTerminalIndex* and the *ColumnTerminalIndex* or the *PolarTerminalIndex*. TerminalDetail is primarily used when there is a need to specify the details for individual terminals that do not fit a pattern. This can be an excessive overhead for quantity of terminals, so it is preferred to use *TerminalNumberingPattern* as defined in section 6.5.7.2 below.

### 6.5.7.1 Terminal Detail (cont'd)

Based on Figure 22 - Single UCle Instantiation in South Channel Inst 1 above, *TerminalNumber* 196 is missing from *TerminalPatternID* 1, and *TerminalNumber* 234 is missing from *TerminalPatternID* 1. However, these Terminal numbers change from one instantiation to the next so a reference to the *TerminalNumber* for the purpose of specifying a missing status is incorrect, if you want to ensure that these relative locations are missing from every instantiation of these terminal patterns. In this case, reference to these missing terminals should be based their index location with respect to their terminal pattern, as shown in Figure 24.



**Figure 24 - Missing Terminal for Terminal Patterns**

Specifying the missing terminal from the Terminal Pattern by reference to its location as opposed to a specific terminal number, means that by duplicating this pattern to other locations, the same respective terminal location within that pattern ID would have the same terminal status.

The status of *Missing*, *Deleted* and *Excluded* are defined in section called “Terminal-count suffixes” of JESD30.

```
<TerminalDetail>
  <ID>1</ID>
  <TerminalPatternID>1</TerminalPatternID>
  <RowTerminalIndex>3</RowTerminalIndex>
  <ColumnTerminalIndex>16</ColumnTerminalIndex>
  <TerminalStatus>
    <Missing/>
  </TerminalStatus>
</TerminalDetail>
<TerminalDetail>
  <ID>2</ID>
  <TerminalPatternID>2</TerminalPatternID>
  <RowTerminalIndex>3</RowTerminalIndex>
  <ColumnTerminalIndex>9</ColumnTerminalIndex>
  <TerminalStatus>
    <Missing/>
  </TerminalStatus>
</TerminalDetail>
```



6.5.7.2 First Terminal Location

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalDetail-Array//FirstTerminalLocation
diagram	
type	DieFirstTerminalLocationType, JEP30-D10:EmptyType, TerminalCenterType, JEP30-D10:MinIntegerOfOneType, LocationRelative-to-SelectionCenterType, LocationRelative-to-SelectionCenterCodeType, LocationRelative-to-SelectionCenterDescriptionType.

The enumerated values for the *LocationRelative-to-SelectionCenter* (Code and Description) are defined in Table 40.

### 5.13.3.2.1 First Terminal Location (cont'd)

**Table 40 - Location Relative to Selection Center**

Code	Description
SW	Southwest
SE	Southeast
NE	Northeast
NW	Northwest
BL	Back-Left
BC	Back-Center
BR	Back-Right
FL	Front-Left
FC	Front-Center
FR	Front-Right
LB	Left-Bottom
LC	Left-Center
LT	Left-Top
RB	Right-Bottom
RC	Right-Center
RT	Right-Top
L	Left
B	Back
R	Right
F	Front

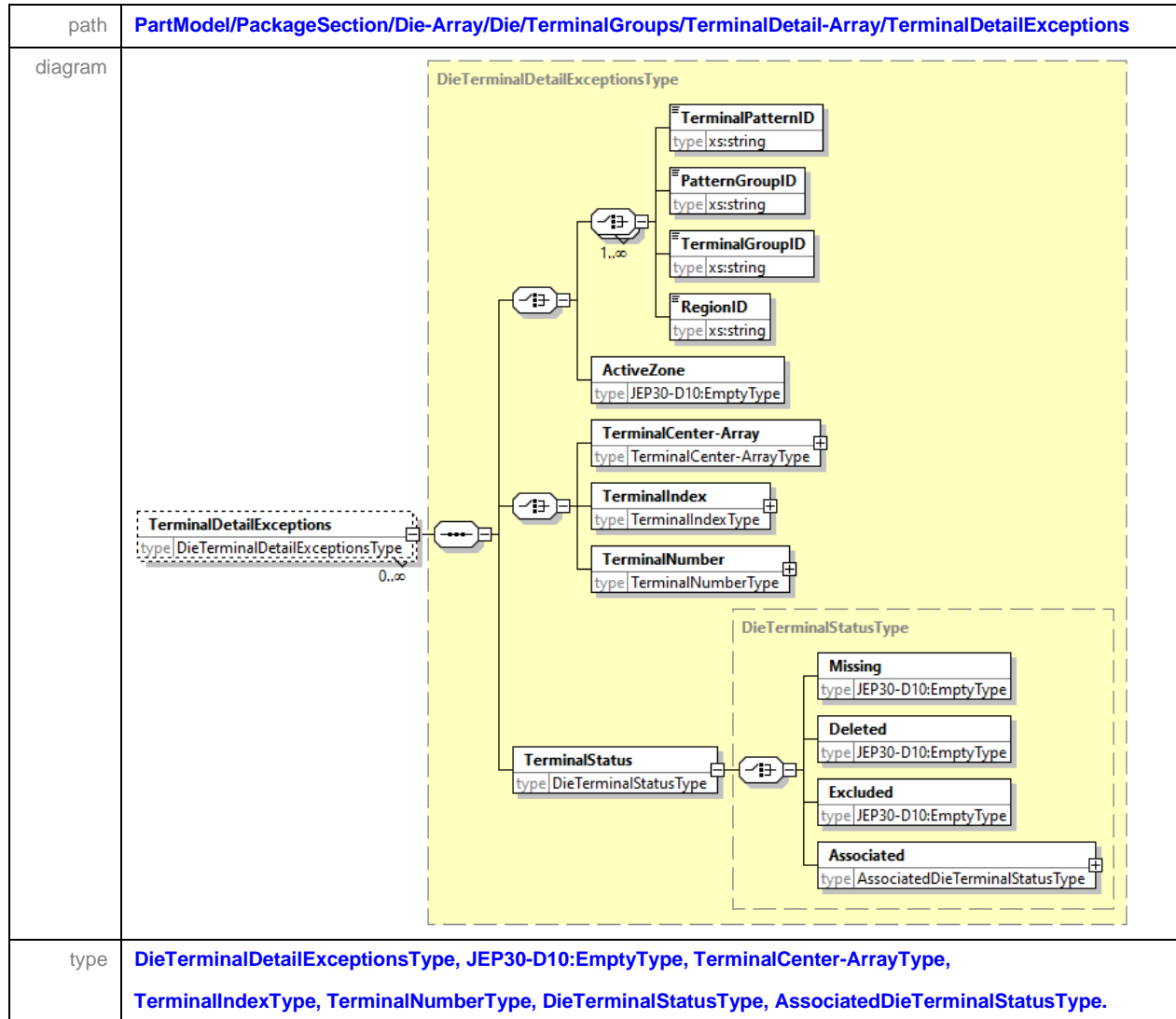
Reference the “Single Position” single terminal illustrations in JESD30, Annex A for graphical representations of some of the above descriptions.

6.5.7.3 Terminal Number Pattern

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalDetail-Array/TerminalNumberPattern
diagram	
type	DieTerminalNumberPatternType, SequentialTerminalNumberOrderingType, GridTerminalNumberOrderingType.
group	NumberingRestrictionGroup

The *TerminalNumberPattern* section that can be defined specifically to a *TerminalPattern*, *PatternGroup*, *TerminalGroup* or to all the terminals within a specific *TerminalRegion*, or finally to all the terminals within an *ActiveZone*.

#### 6.5.7.4 Terminal Details Exception

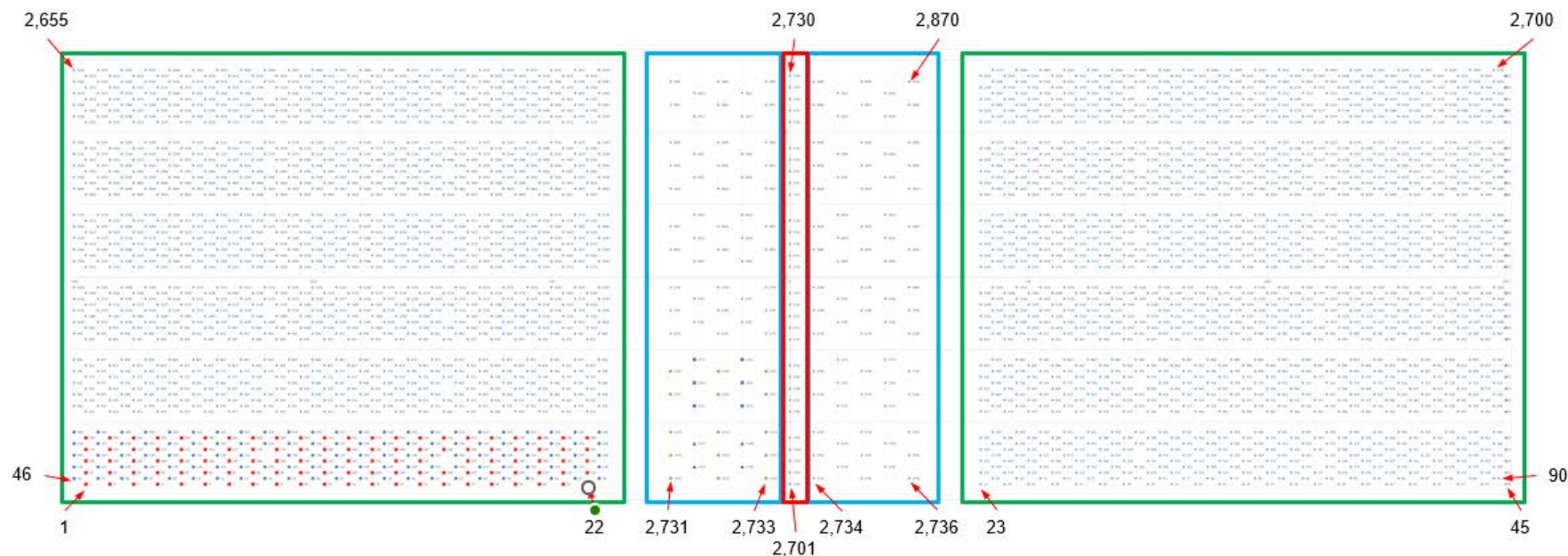


Referring to the example defined in section 6.5.1.1 Terminal Group above, Figure 25 shows the terminal numbering sequence for all the terminals that make up Bank 1. Here you can see 3 distinct patterns each with their own numbering sequence.

6.5.7.4.1 Associated

path	PartModel/PackageSection/Die-Array/Die/TerminalGroups/TerminalDetail-Array/TerminalDetailExceptions/TerminalStatus/Associated
diagram	<p>The diagram illustrates the structure of the <code>AssociatedDieTerminalStatusType</code>. It is a complex type with a choice between two main branches. The first branch is a sequence of four elements: <code>ReferenceTerminalPatternID</code>, <code>ReferencePatternGroupID</code>, <code>ReferenceTerminalGroupID</code>, and <code>ReferenceRegionID</code>, all of type <code>xs:string</code>. The second branch is a choice between <code>ActiveZone</code> (type <code>JEP30-D10:EmptyType</code>) and <code>TerminalCenter</code> (type <code>TerminalCenterType</code>). The <code>TerminalCenter</code> element is further divided into a choice between <code>RowTerminalIndex</code>, <code>ColumnTerminalIndex</code>, and <code>PolarTerminalIndex</code>, all of type <code>JEP30-D10:MinIntegerOfOneType</code> with a minimum inclusive value of 1. The <code>LocationRelative-to-PackageCenter</code> element is also present, with type <code>LocationRelative-to-PackageCenterType</code>.</p>
type	AssociatedDieTerminalStatusType, JEP30-D10:EmptyType, TerminalCenterType, JEP30-D10:MinIntegerOfOneType, LocationRelative-to-PackageCenterType.

#### 6.5.7.4.1 Associated (cont'd)



**Figure 25 - Terminal Numbering Sequence for Bank 1**

Pattern Group ID 3 is represented within the two extremity columns within the green outlines. Refer to Table 32 – Pattern Group Construction above for details of the Terminal patterns that makes the first pattern group which is duplicated 6 times to make the South channel before being duplicated again and mirrored to make the north channel. This pattern group has a terminal numbering sequence of Zig-Zag-Horizontal with a commencing location at the Back-left position starting at the number 1 and ends at 2700.

Pattern Group ID 10 is represented by the center red box and has a commencing location at the Back position starting at the number 2701 and ends at 2730 in the front position.

Pattern Group ID 9 is represented by the blue boxes on either side of pattern group 10 and has a commencing location at the Back-left (Southwest or Left-Bottom) position starting at the number 2731 and ends at 2870 in the Front-right (Northeast or Right-Top) position.

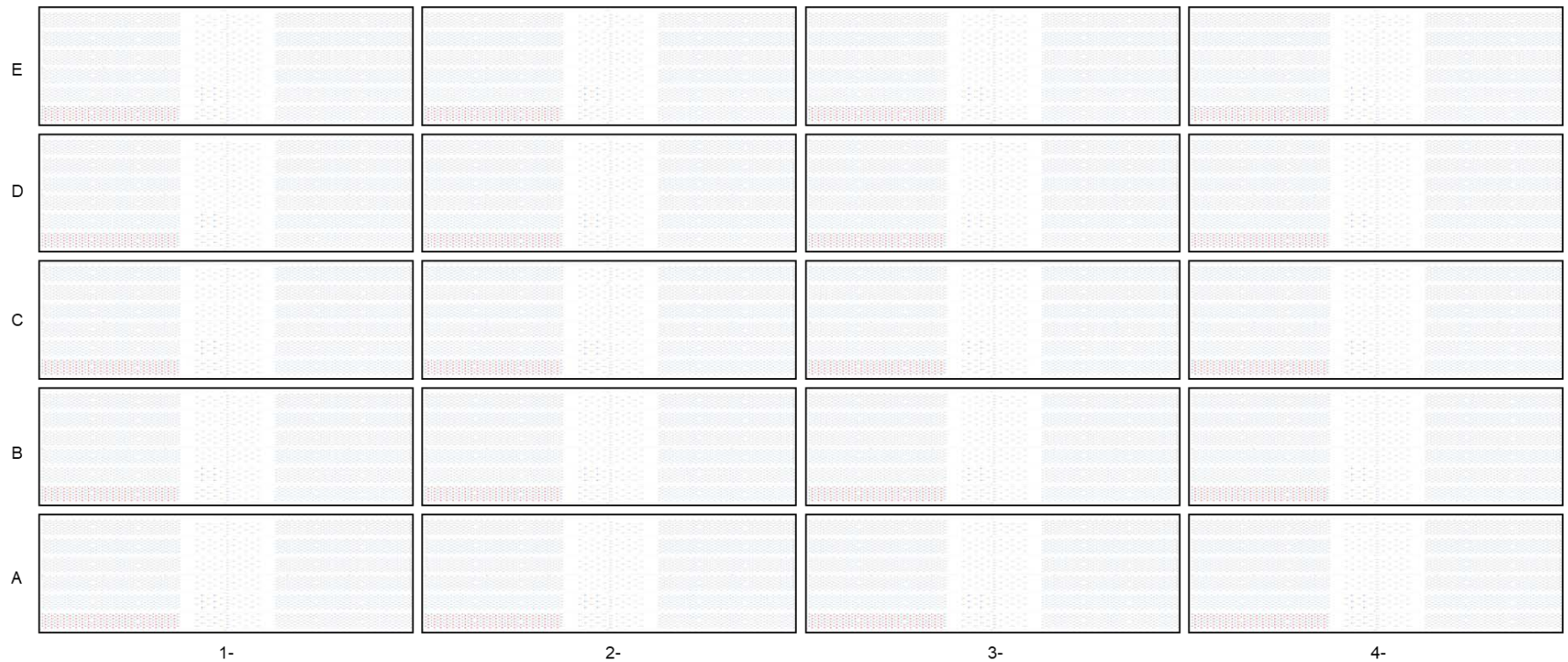
#### 6.5.7.4.1 Associated (cont'd)

The xml representation for this terminal numbering for Bank 1 is as follows:

```
<TerminalDetail-Array>
  <TerminalNumberPattern>
    <ID>Terminal Number Pattern ID 1</ID>
    <PatternGroupID>Pattern Group ID 3</PatternGroupID>
    <FirstTerminalLocation>
      <LocationRelative-to-SelectionCenter>
        <Description>Back-left</Description>
      </LocationRelative-to-SelectionCenter>
    </FirstTerminalLocation>
    <Sequential>
      <Description>Zig-Zag-Horizontal</Description>
    </Sequential>
  </TerminalNumberPattern>
  <TerminalNumberPattern>
    <ID>Terminal Number Pattern ID 2</ID>
    <PatternGroupID>Pattern Group ID 10</PatternGroupID>
    <FirstTerminalLocation>
      <LocationRelative-to-SelectionCenter>
        <Description>Back</Description>
      </LocationRelative-to-SelectionCenter>
    </FirstTerminalLocation>
    <Sequential>
      <Description>Back-to-Front</Description>
      <NumericalSequence>
        <Start>2701</Start>
      </NumericalSequence>
    </Sequential>
  </TerminalNumberPattern>
  <TerminalNumberPattern>
    <ID>Terminal Number Pattern ID 3</ID>
    <PatternGroupID>Pattern Group ID 9</PatternGroupID>
    <FirstTerminalLocation>
      <LocationRelative-to-SelectionCenter>
        <Description>Back</Description>
      </LocationRelative-to-SelectionCenter>
    </FirstTerminalLocation>
    <Sequential>
      <Description>Zig-Zag-Horizontal</Description>
      <NumericalSequence>
        <Start>2731</Start>
      </NumericalSequence>
    </Sequential>
  </TerminalNumberPattern>
</TerminalDetail-Array>
```

Assume now that Bank 1 is just one bank of an array of banks as shown in Figure 26 below

#### 6.5.7.4.1 Associated (cont'd)



**Figure 26 - An Array of Banks in a Zone**

Bank 1 as defined in Figure 25 - Terminal Numbering Sequence for Bank 1 above is replicated in an array of 5 rows and 4 columns. Assume that the user wants to maintain the terminal numbering sequence as defined in Bank 1 for each of these blocks within the array, with the only difference being the pattern location within the array as the prefix before the terminal number.



#### 6.5.7.4.1 Associated (cont'd)

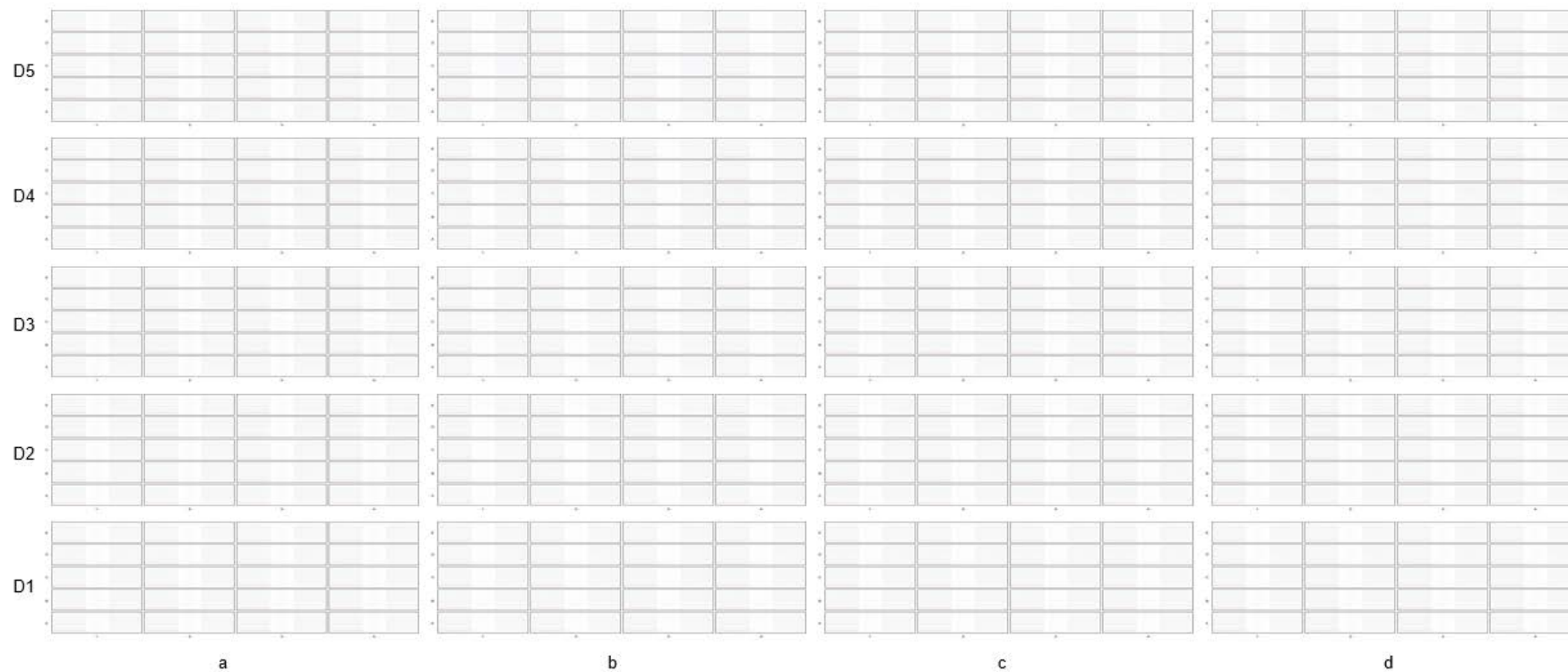
```
<PatternGroup>
  <ID>Pattern Group ID 12</ID>
  <Name>South_Inst1-6_PtrnGrp1</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 11</PatternGroupID>
    <Duplicate>
      <nx>4</nx>
      <ny>5</ny>
      <Y-PrefixCode>A</Y-PrefixCode>
      <Y-PrefixCode>B</Y-PrefixCode>
      <Y-PrefixCode>C</Y-PrefixCode>
      <Y-PrefixCode>D</Y-PrefixCode>
      <Y-PrefixCode>E</Y-PrefixCode>
      <X-PrefixCode>1</X-PrefixCode>
      <X-PrefixCode>2</X-PrefixCode>
      <X-PrefixCode>3</X-PrefixCode>
      <X-PrefixCode>4</X-PrefixCode>
      <dx>4000.00</dx>
      <dy>3500.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>
```

Since the Y-PrefixCode is entered into the xml file before the X-PrefixCode, then the first terminal number for terminal 1 in Bank 1 in the zone position A1 becomes A1-1, The same respective terminal in the A2-1, A3-1, A4-1, B1-1, B2-1, and so on.

If nx and/or ny is a large number, then the prefix and/or suffix codes can be auto generated in a similar way as the numbering sequence is defined.

```
<PatternGroup>
  <ID>Pattern Group ID 12</ID>
  <Name>South_Inst1-6_PtrnGrp1</Name>
  <PatternRelationship>
    <PatternGroupID>Pattern Group ID 11</PatternGroupID>
    <Duplicate>
      <nx>4</nx>
      <ny>5</ny>
      <Y-PrefixColumnCode>
        <AlphabeticalSequence/>
        <Back-to-Front/>
      </Y-PrefixColumnCode>
      <X-PrefixRowCode>
        <NumericalSequence>
          <Suffix>-</Suffix>
        </NumericalSequence>
        <Left-to-Right/>
      </X-PrefixRowCode>
      <dx>4000.00</dx>
      <dy>3500.00</dy>
    </Duplicate>
  </PatternRelationship>
</PatternGroup>
```

#### 6.5.7.4.1 Associated (cont'd)



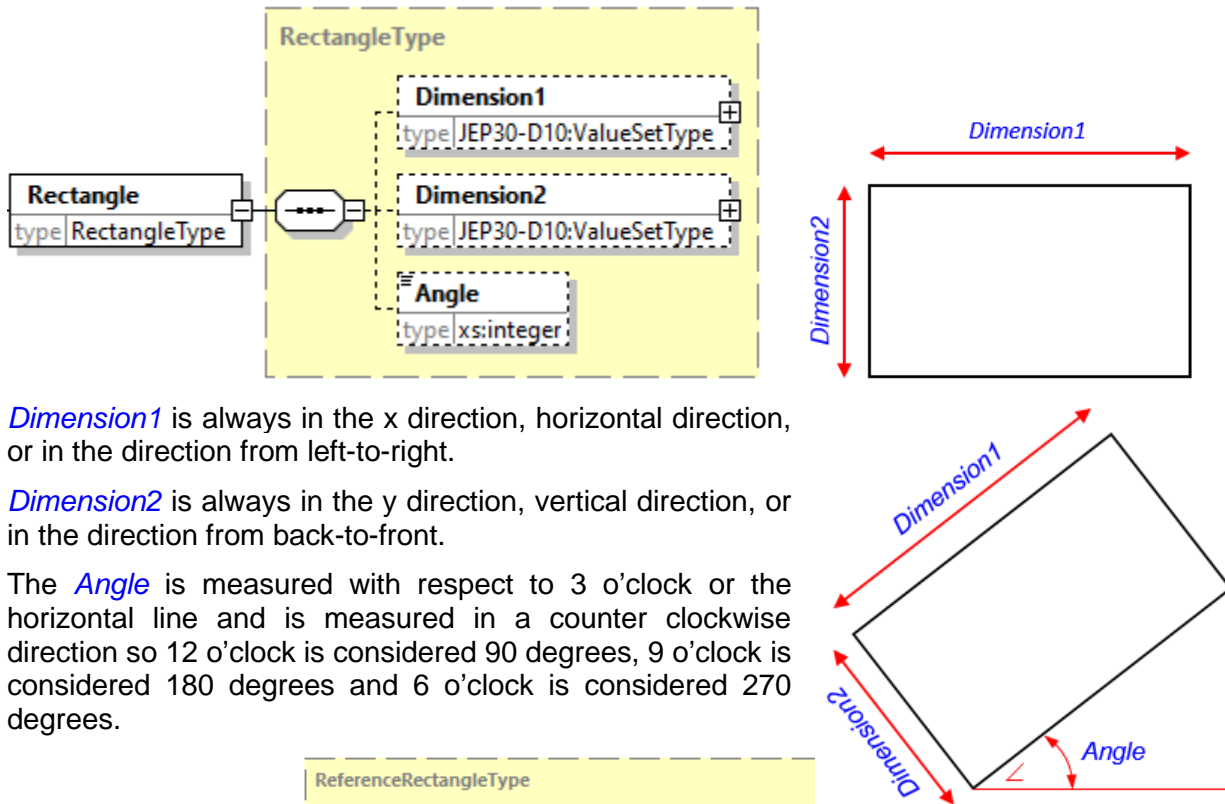
**Figure 27 - An Array of Zones in One District**

Similar to the concepts of Banks within a Zone, an array of Zones can be embedded within a District. Each hierarchical level in the nesting of pattern groups brings with it a corresponding prefix/suffix that gets appended to the reference label of the terminal number. Therefore, in District D1a, Zone A1- the first terminal now becomes D1aA1-1.

## Annex A (informative) Shape Definitions & Dimensions

### A.1 Shape Definitions

#### A.1.1 Rectangle

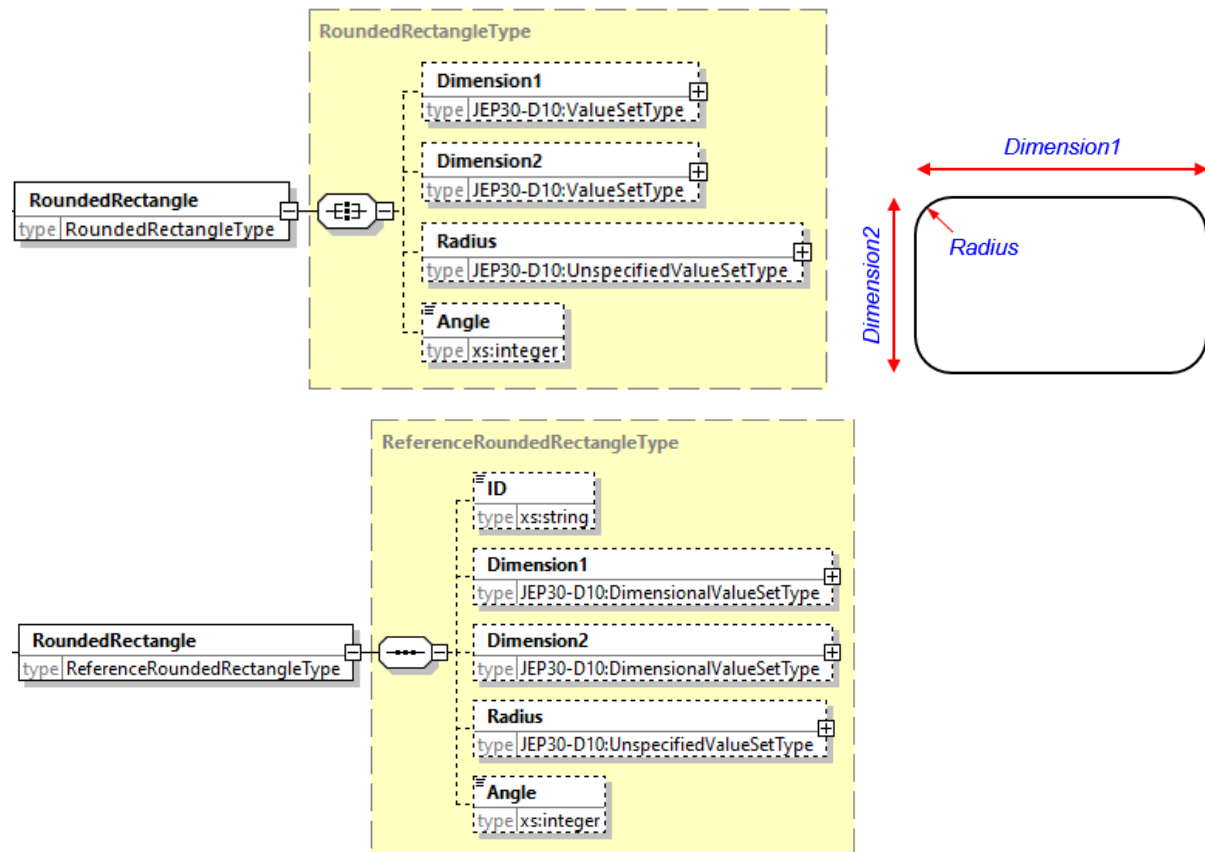


*Dimension1* is always in the x direction, horizontal direction, or in the direction from left-to-right.

*Dimension2* is always in the y direction, vertical direction, or in the direction from back-to-front.

The *Angle* is measured with respect to 3 o'clock or the horizontal line and is measured in a counter clockwise direction so 12 o'clock is considered 90 degrees, 9 o'clock is considered 180 degrees and 6 o'clock is considered 270 degrees.

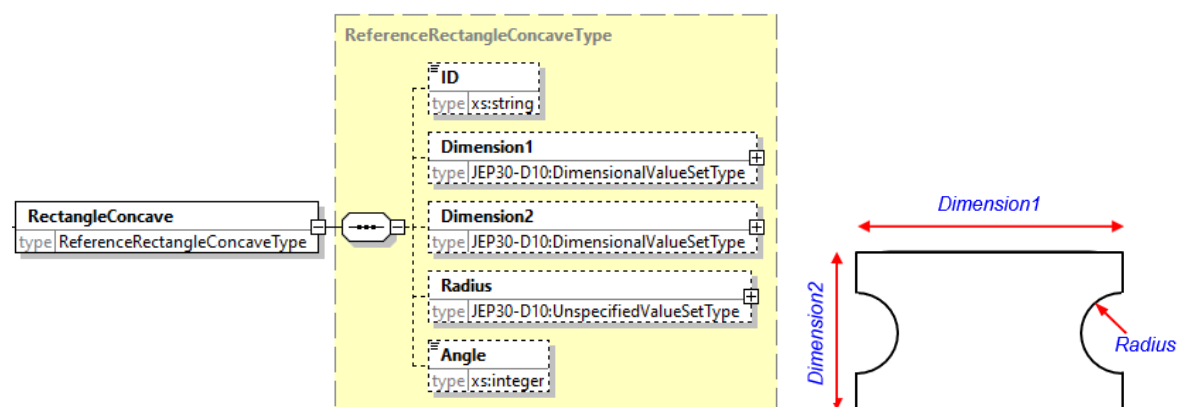
## A.1.2 Rounded Rectangle



*Dimension1*, *Dimension2* and *Angle* are the same as for *Rectangle*.

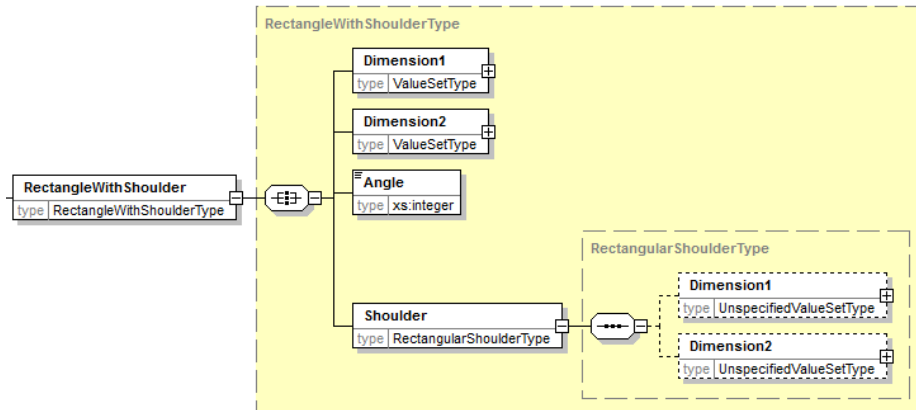
If all 4 corners have the same radius, then *RoundedRectangle* can be used. If the radius is different in any one corner, then *ModifiedRectangle* should be used.

## A.1.3 Rounded Concave



*Dimension1*, *Dimension2* and *Angle* are the same as for *Rectangle*.

#### A.1.4 Rectangle with Shoulder

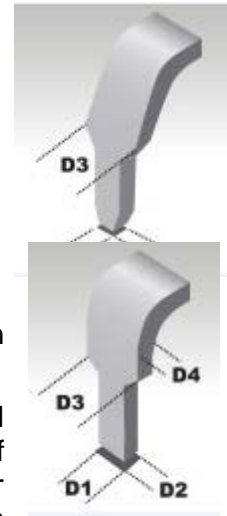


Some terminals have shoulders to control the depth of insertion into the printed board. Because hole diameter on the printed board is dependant upon many factors, one of which is the thickness of the printed board, due to aspect ratio due to ensure hole fill, some of these parts are unsuitable for thicker printed boards. If the hole size increases to be greater than the shoulder dimension, then the part can fall deeper into the hole. For this and similar reasons, shoulder dimensions should be captured.

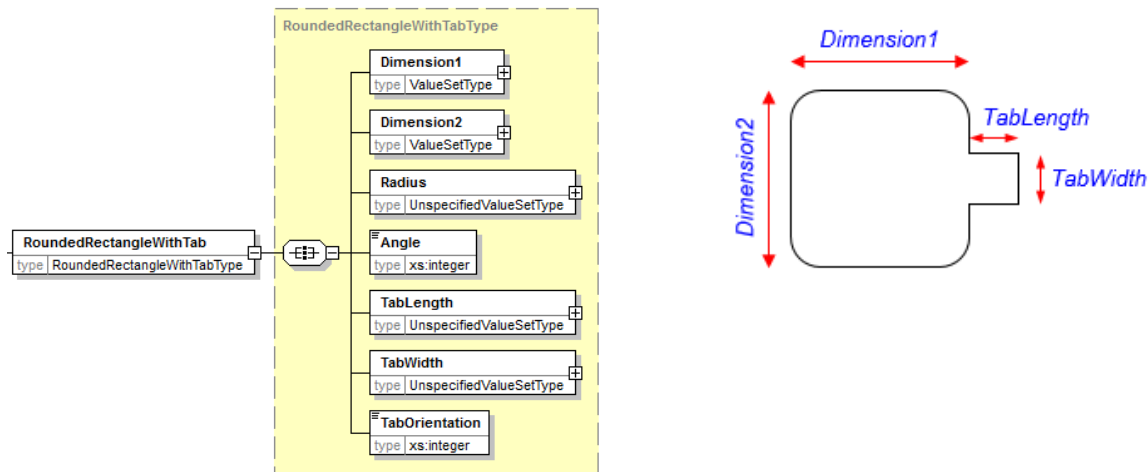
The *RectangleWithShoulder/Dimension1* (D1) is in the same direction as *RectangleWithShoulder/Shoulder/Dimension1* (D3).

Similarly the *RectangleWithShoulder/Dimension2* (D2) is in the same direction as *RectangleWithShoulder/Shoulder/Dimension2* (D4)

The *RectangleWithShoulder/Shoulder/Dimension1* and *RectangleWithShoulder/Shoulder/Dimension2* are optional, since either of these dimensions as *RectangleWithShoulder/Dimension1* or *RectangleWithShoulder/Dimension2*, as can be seen from the 2<sup>nd</sup> image where D2 = D4. However this is not always the case, and if different, but unspecified, then the *Unspecified* element should be set under the *UnspecifiedValueSetType* as opposed to not populating the branches under the *Shoulder*.

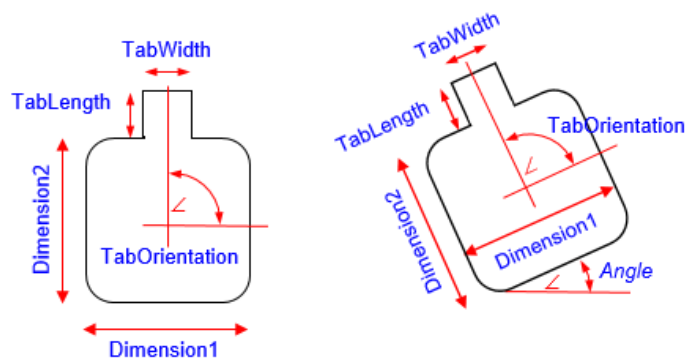


### A.1.5 Rounded Rectangle with Tab

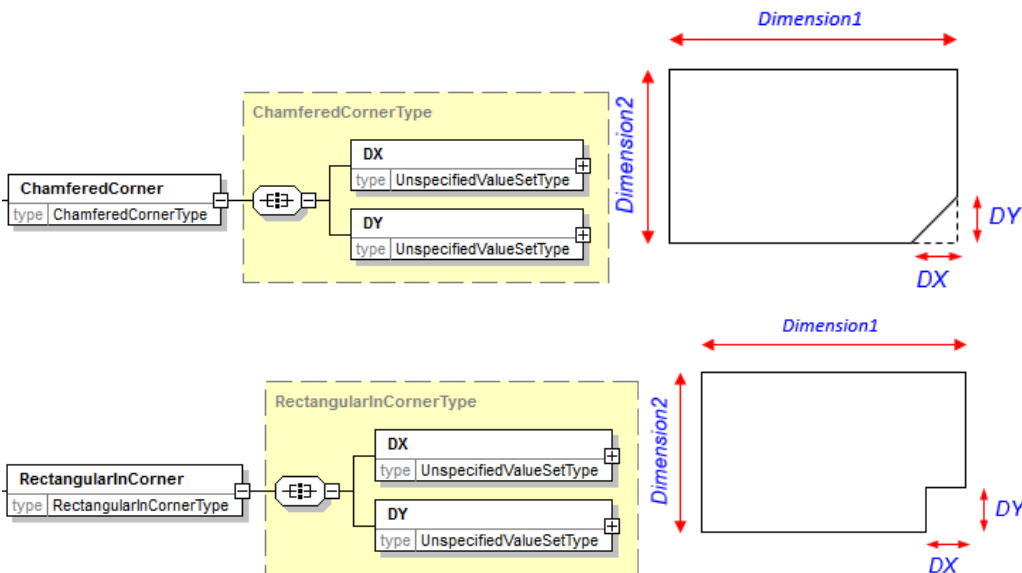
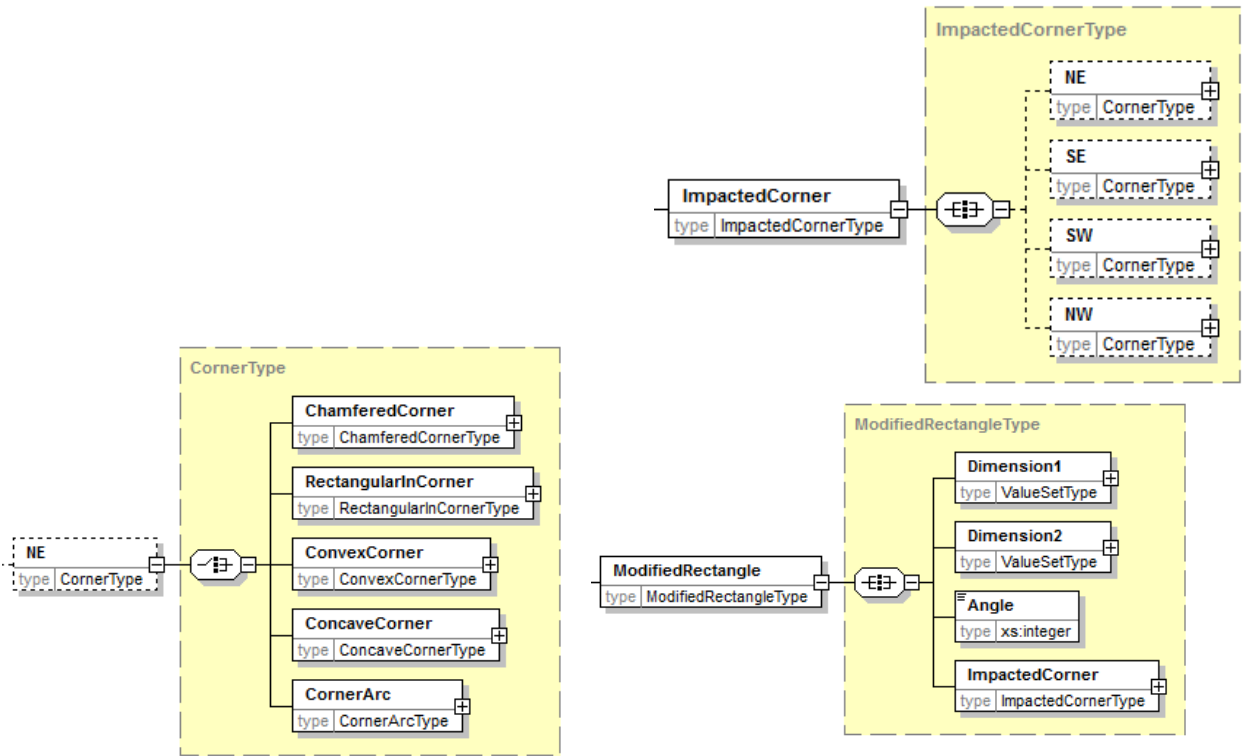


Irrespective of the orientation of the Tab, the **TabLength** is considered the direction away from the package body, while the **TabWidth** is considered the width of the tab parallel to the package body. The **TabOrientation** here is at 0 degrees, since the orientation follows the same rules as the **Angle**.

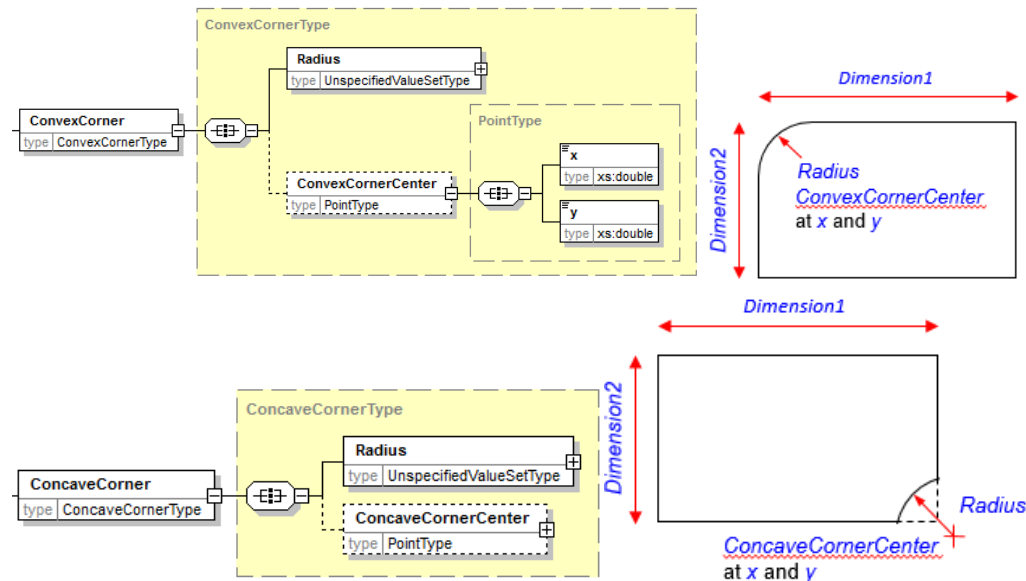
The **TabOrientation** is still captured with respect to the package body, In these 2 images, the **TabOrientation** is at 90° even if the package is rotated at an **Angle**.



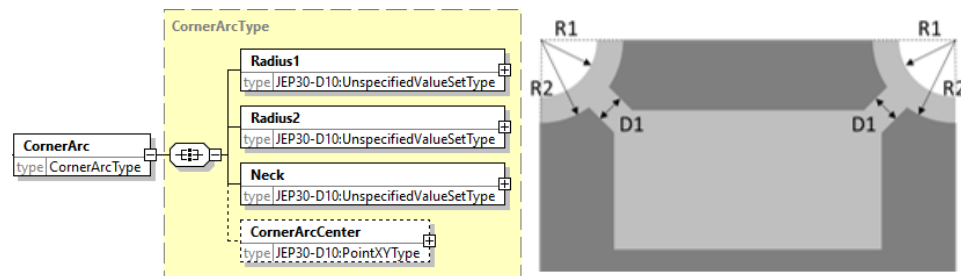
A.1.6 Modified Rectangle



## A.1.6 Modified Rectangle (cont'd)



Note that the **ConcaveCornerCenter** point of **x** and **y** is not necessary to be at the same point as the corner of the **ModifiedRectangle**. If unspecified, then it is defaulted to the corner of the **ModifiedRectangle**.



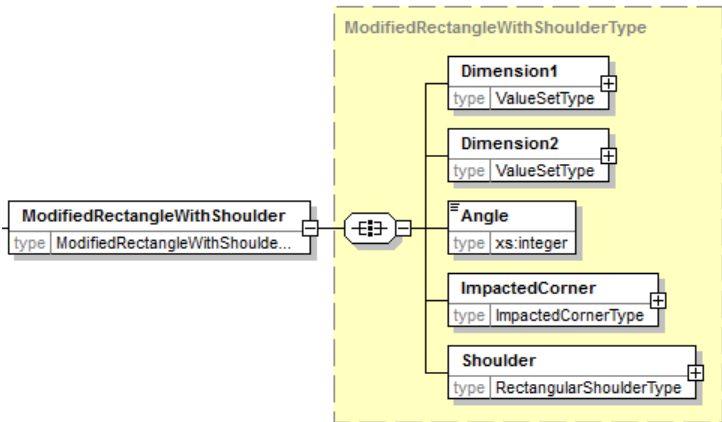
A **ModifiedRectangle** with **CornerArc** usually occurs when the Surface terminal is pulled back from the edge of the Package outline. In the majority of cases including those where the data is unspecified on the datasheet, we can assume that the center of the Arc from which (R1) **Radius1** and (R2) **Radius2** are defined, occurs at the linear intersection of the projection of the sides of the package body. This image shows that there are 2 **CornerArcs** connected to the same surface terminal in the **NW** and **NE** corners.

The trace **Neck** width (D1) connects the terminal shape to the arc along the line that would intersect with the center of the arc to the projected corner of the terminal shape.

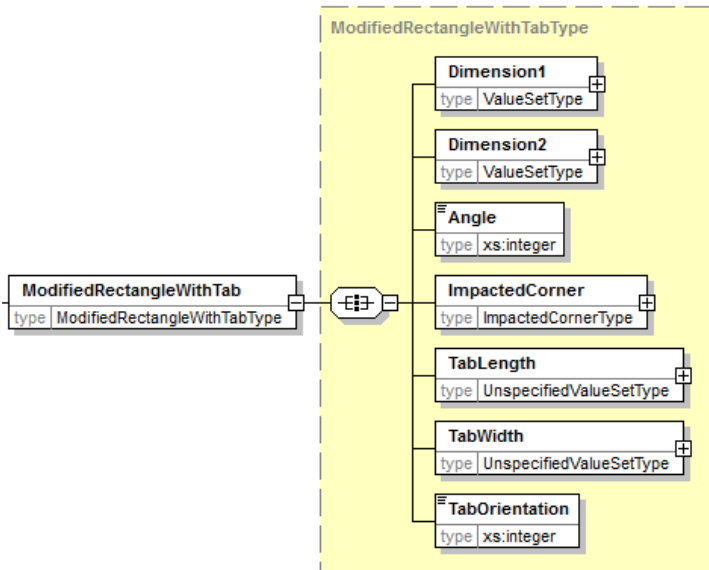
Note that the **ConcaveArcCenter** point of **x** and **y** is not necessary to be at the same point as the corner of the **ModifiedRectangle**. If unspecified, then it is defaulted to the corner of the **ModifiedRectangle**.



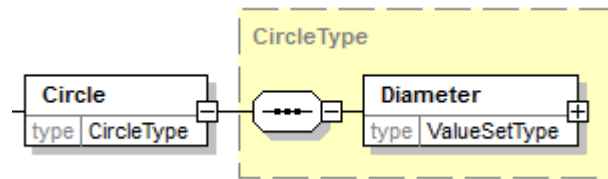
A.1.7 Modified Rectangle with Shoulder



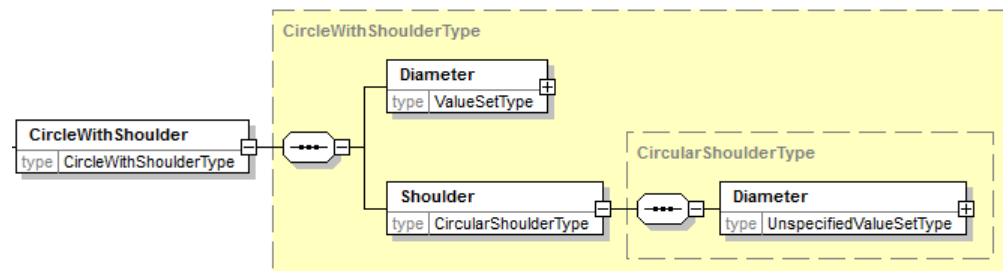
A.1.8 Modified Rectangle with Tab



### A.1.9 Circle



### A.1.10 Circle with Shoulder

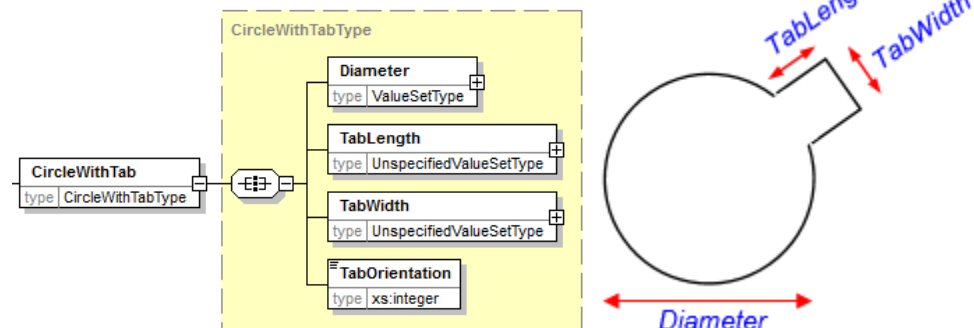


The **CircleWithShoulder/Diameter** (D1) is the diameter of the terminal that inserts into the printed board.

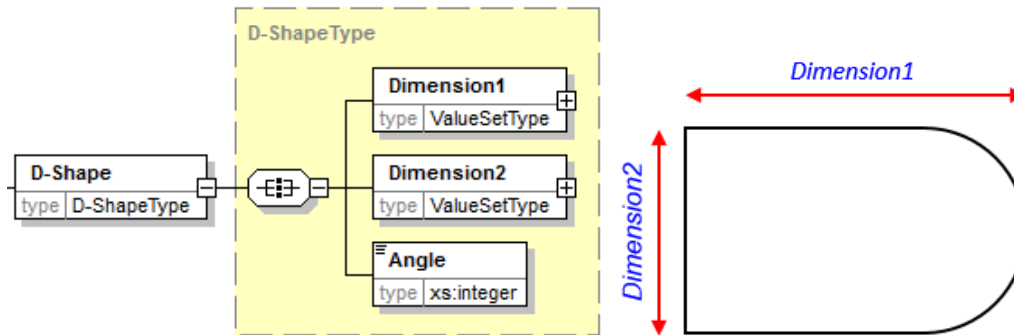
The **CircleWithShoulder/Shoulder/Diameter** (D2) is the diameter of the terminal shoulder that is typically not intended for insertion into the printed board.



### A.1.11 Circle with Tab



### A.1.12 D-Shape

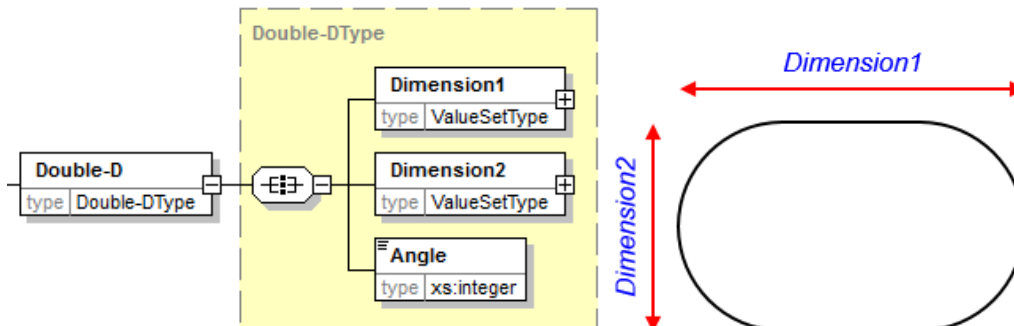


A *D-Shape* is where a semi-circle is attached to the end of a Rectangular shape. Note that radius is not required since it is 50% of *Dimension2*.

*Dimension1* is the distance from the end of the rectangle to the outside edge of the semicircle.

The side with the curved end is defaulted towards the package center. If the curve is on a different side, then Modified Rectangle should be used.

### A.1.13 Double-D

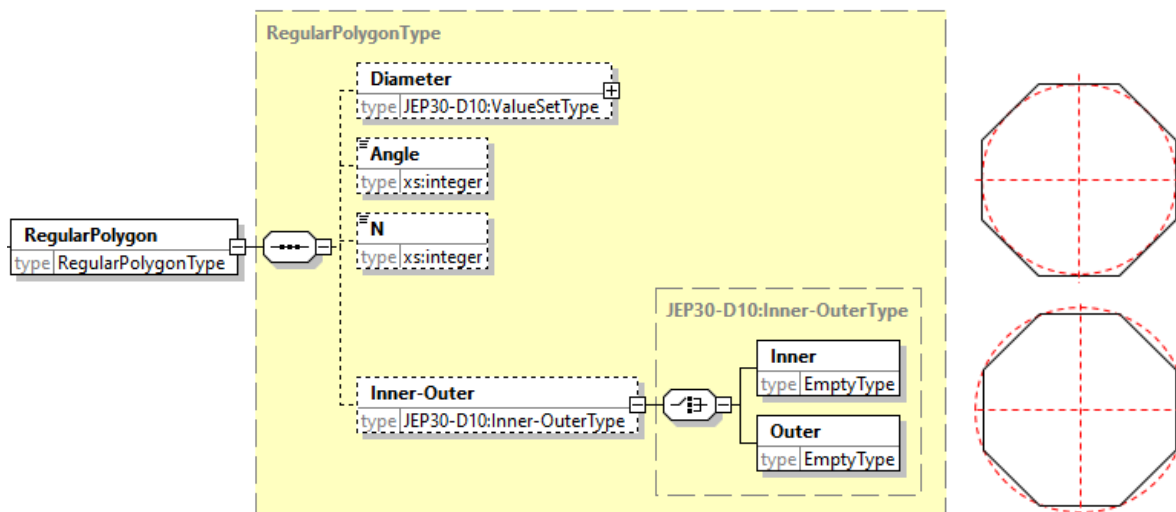


A *Double-D* shape is where a semi-circle is attached to the end of a Rectangular shape. Note that radius is not required since it is 50% of *Dimension2*.

*Dimension1* is the distance from the outside edges of each of the semicircle at opposite ends of the rectangle.

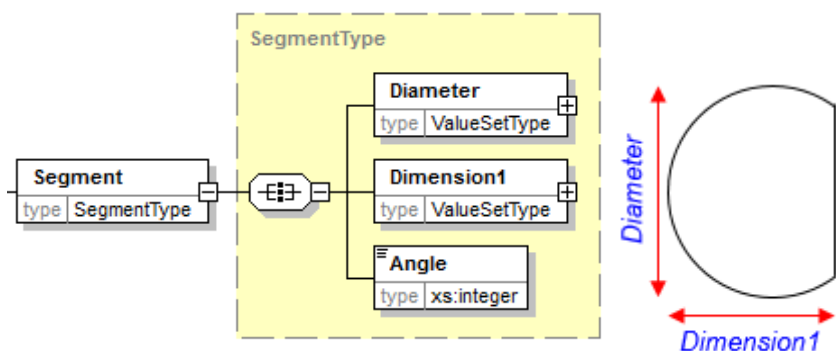
The curve sides occur on the shortest side of Dimension 1 or Dimension 2.

### A.1.14 Regular Polygon



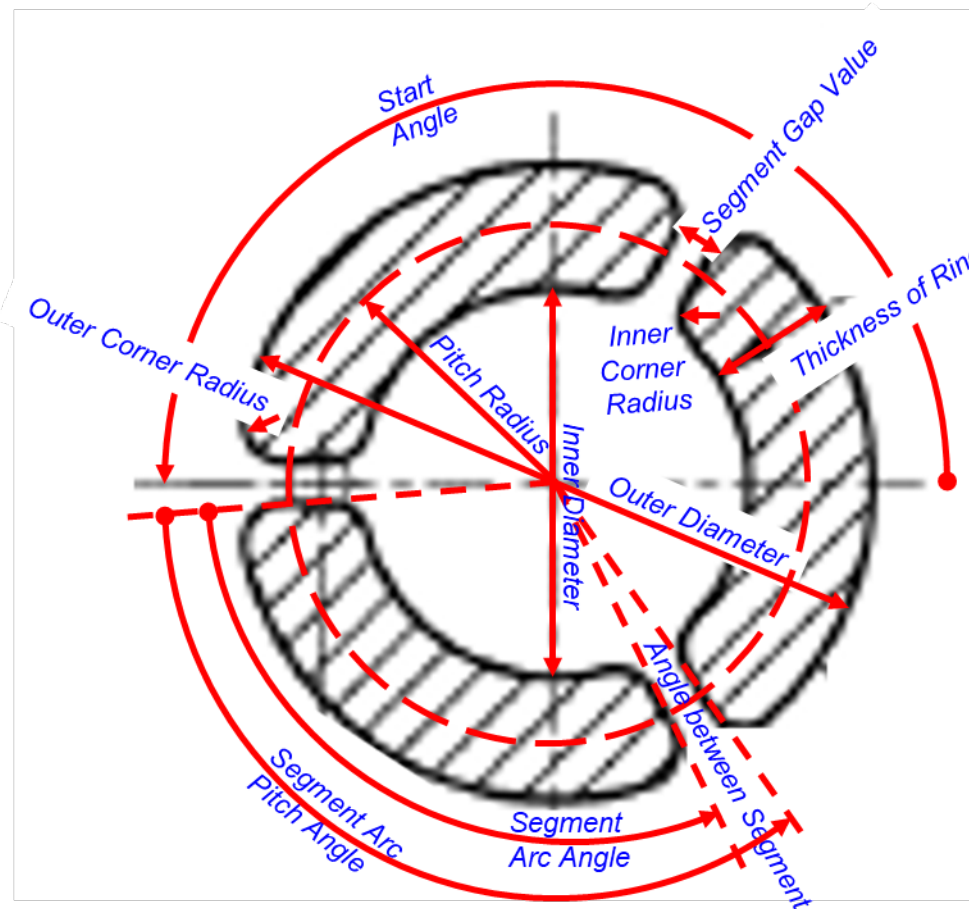
The most common Polygons required to represent package body shapes are [RegularPolygon](#), that are equiangular (all its corner angles are equal), Cyclic (all its corners lie on a single circle) and equilateral (all its edges are the same length). These Polygons are also tangential (all sides are tangent to an inscribed circle and are convex (all angles are < 180 degrees). Polygons dimensions can be captured by knowing the number of sides *N* and the *Diameter* of either the circumcircle (*Outer* circle that intersect with the corners of the polygon) or the inscribed circle (*Inner* circle which is tangential to all the sides within the polygon). All other forms of Polygons component shapes must be drawn via the [Contour](#) branch.

### A.1.15 Segment



The *Angle* here defines the rotation in the counter clockwise direction of the straight edge from the 3 o'clock position. The image here shows 90°.

### A.1.16 Segmented Ring

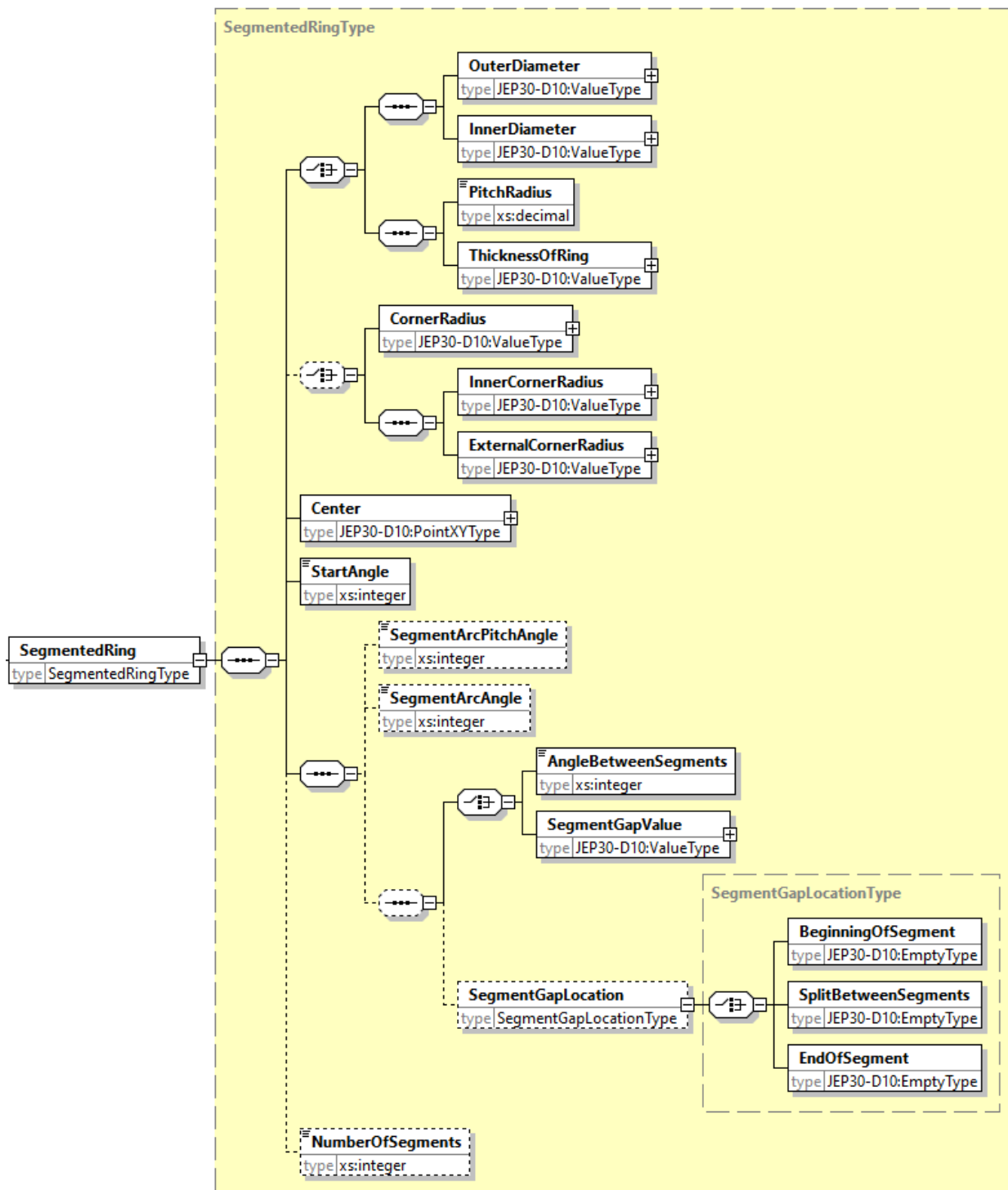


The thickness of the segment can be defined by either the combination of the *OuterDiameter* and the *InnerDiameter*, or the *PitchRadius* and the *ThicknessOfRing*. The *CornerRadius* of the segment can be optionally defined, however if the inner and outer corners are different, then this can be individually defined via the *InnerCornerRadius* and the *OuterCornerRadius*.

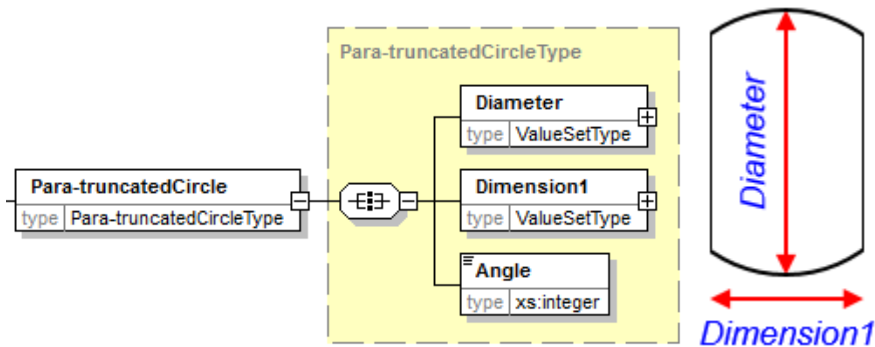
The *StartAngle* here defines the rotation in the counterclockwise direction of the straight edge from the 3 o'clock position. The image here shows 180°. The *SegmentArcPitchAngle* is the angle from any point in the Segment arc to the same point on the next segment. The *SegmentArcAngle* is the angle of one segment arc. When the segment is separated from the next segment, this can be defined by either the *AngleBetweenSegments* or the *SegmentGapValue*. If there are more than 1 segment arc in the *SegmentedRing*, then the *SegmentArcPitchAngle* equals the sum of the *SegmentArcAngle* plus the *AngleBetweenSegment*.

The *SegmentGapLocation* is dependent upon how the start angle is defined. Here in this diagram, the *StartAngle* is defined as 180° from the 3 o'clock position. The segment gap at that point shows that it is evenly split between the top and bottom segments. This is defined by the *SegmentGapLocation*, and is defined as a choice between *BeginningOfSegment*, *SplitBetweenSegments* or *EndOfSegment*.

### A.1.16 Segmented Ring (cont'd)

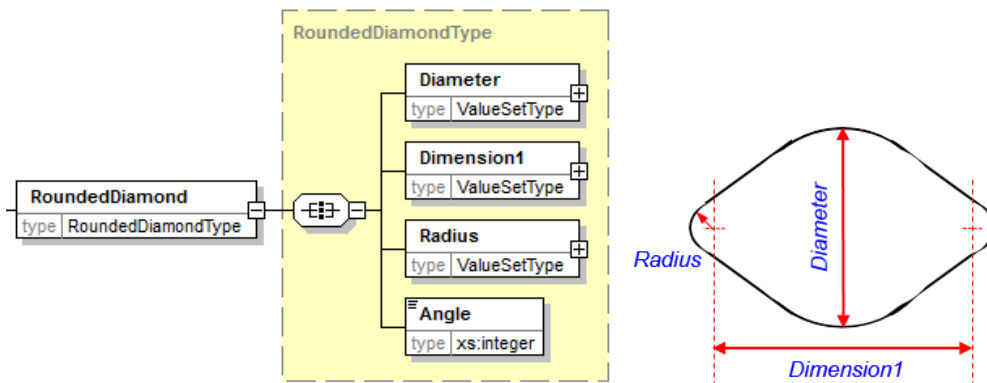


### A.1.17 Para-truncated Circle



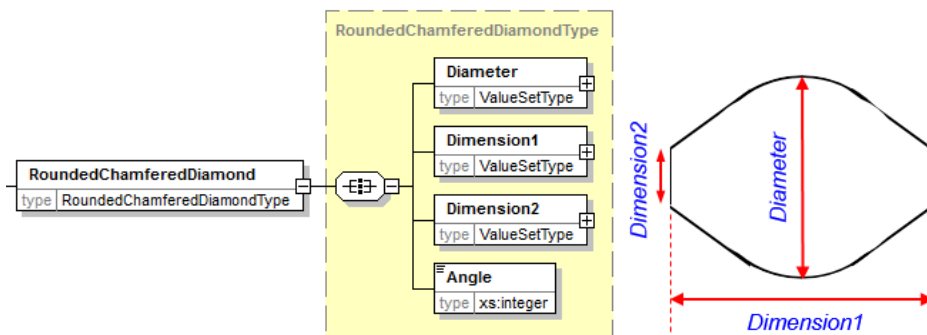
A **Para-truncatedCircle** is a circle with equal segments removed from opposite's sides.

### A.1.18 Rounded Diamond



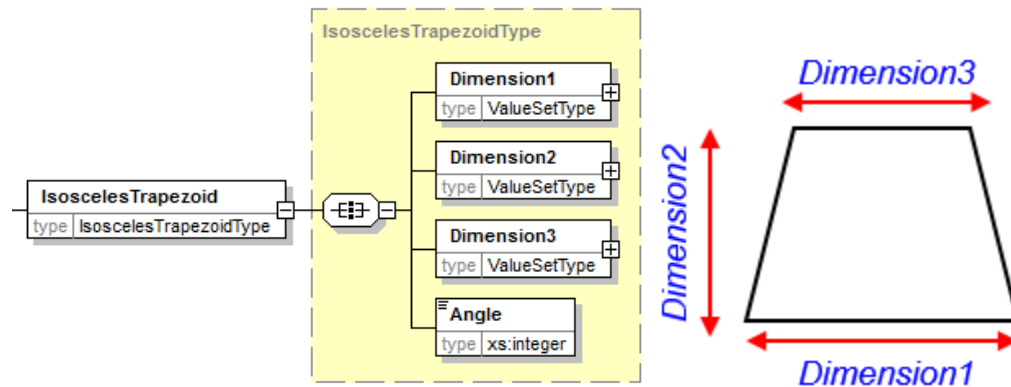
**Dimension1** specifies the distance between the centers of the 2 end circles, while the **Radius** specifies the curvature of the 2 end circles. The **Diameter** specifies the dimension of the center circle. The 4 straight sides intersect tangentially with their respective two circles.

### A.1.19 Rounded Chamfered Diamond

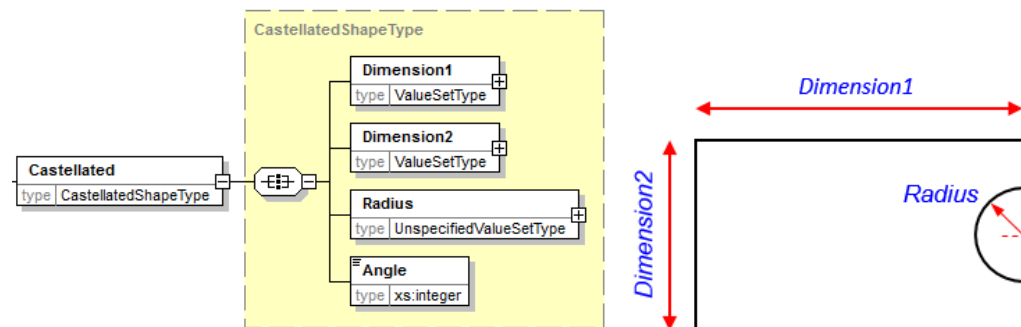


**Dimension1** specifies the distance between the end straight edges, while **Diameter** specifies the dimension of the center circle. The straight sides have a length of **Dimension2**.

### A.1.20 Isosceles Trapezoid

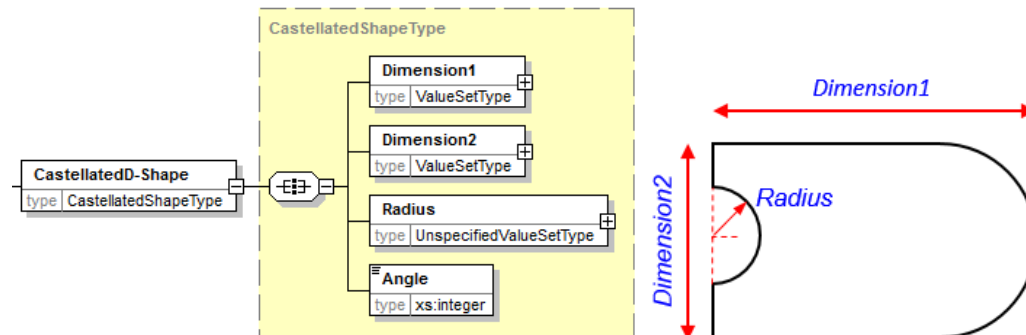


### A.1.21 Castellated Shape



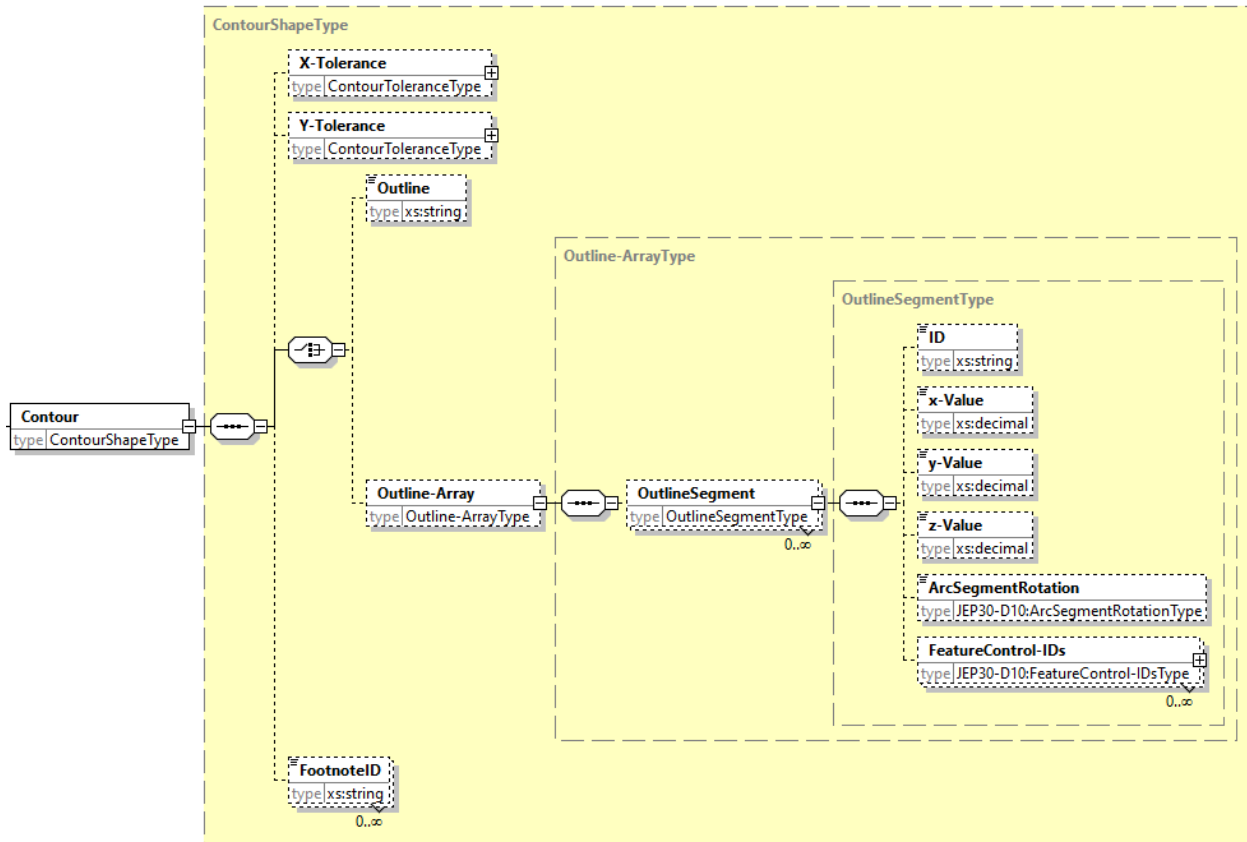
**Castellated** is shape with a recess in it. The **Radius** of this concave is less than the width of the shape side to which it is attached. The recess is also centered along that side of the shape.

### A.1.22 Castellated D-Shape Shape





### A.1.23 Contour



The **Outline** is a list of points representing the outline of the shape. In the XML it is defined as a string. This string must have a specific structure. The structure is  $(X_1, Y_1, R_1), (X_2, Y_2, R_2), \dots (X_n, Y_n, R_n)$ . The points are defined such that the (0, 0) location of the resulting shape is the reference point of the shape. Contour shaped terminal contacts that are positioned randomly will be positioned using the (0, 0) location of the terminal.

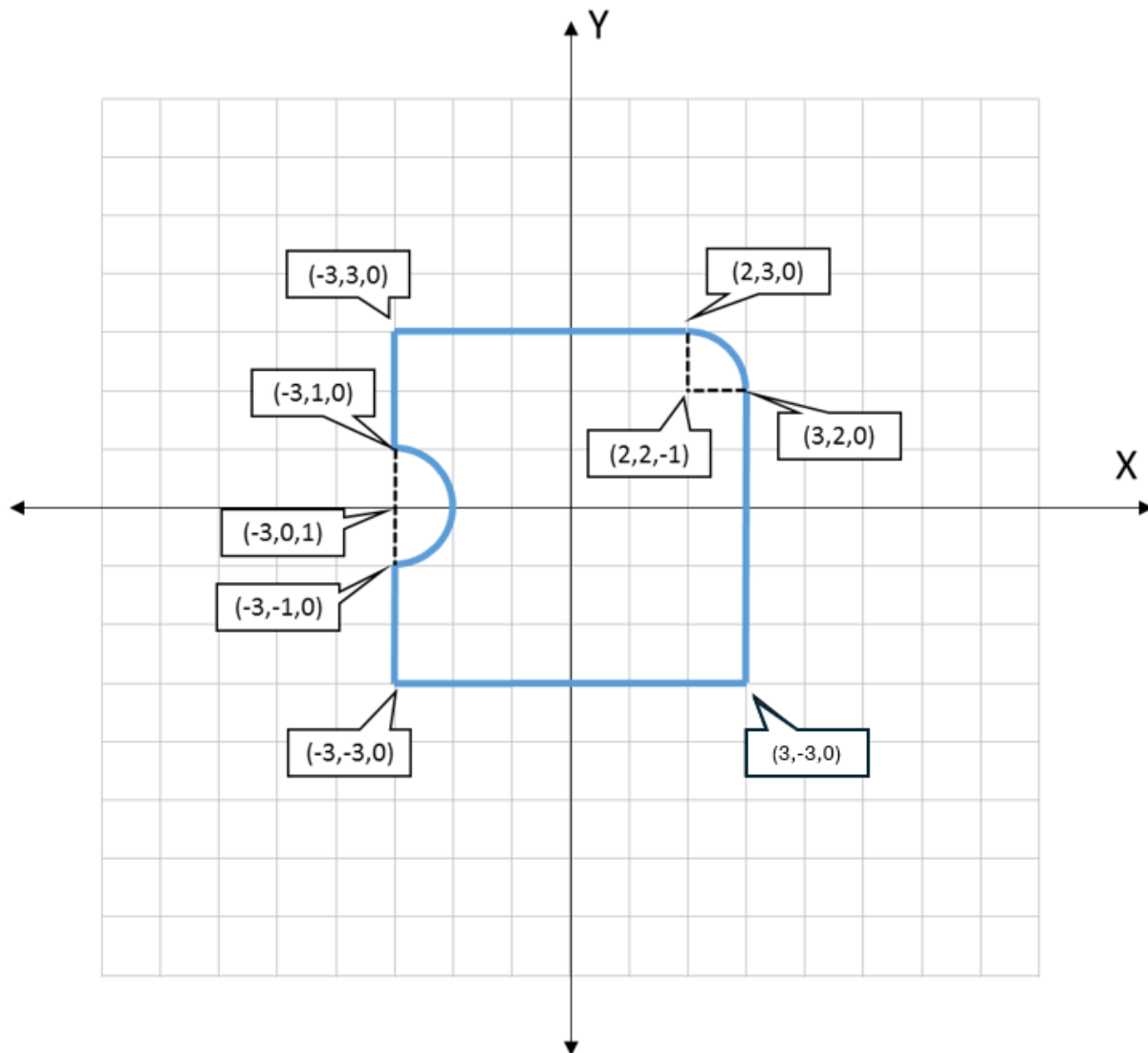
Non-arcs points are defined with an X, Y coordinate, The R value is always 0.

Arcs are defined with 3 points.

- The first point is the start point of the arc (R value is 0).
- The second point is the center point of the arc. The R value for this point is equal to the radius of the arc. If the R value is negative, a counter-clockwise arc is defined. If the R value is positive, a clockwise arc is defined.
- The third point is the end point of the arc (R value is 0).

The last point does not need to be repeated in the case of Segments as all are closed shapes. You may begin with any point except a radius point. The points must be order properly from that starting point.

### A.1.23 Contour (cont'd)



Example 1. The string value that will be present in *Outline* to represent this shape would be

- $(-3,-3,0),(3,-3,0),(3,2,0),(2,2,-1),(2,3,0),(-3,3,0),(-3,1,0),(-3,0,1),(-3,-1,0)$

### A.1.24 Reference Terminal Contour

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner
diagram	<pre> classDiagram     class Contour {         type ReferenceTerminalContourType     }     class ReferenceTerminalContourType {         ID xs:string         X-Tolerance ContourToleranceType         Y-Tolerance ContourToleranceType         OutlineReference OutlineReferenceType         ImpactedTerminal ImpactedTerminalType         Impact-to-TerminalGroup Impact-to-TerminalGroupType         FeatureControl-IDs JEP30-D10FeatureControl-IDsType     }     Contour --&gt; ReferenceTerminalContourType         </pre>
type	ModifiedCornerType, ImpactedTerminalType, TerminalCenterType, Impact-to-TerminalGroupType, Apply-to-all-TerminalsType, SymmetryType, ImpactedCornerType, CornerType.

### A.1.25 Modified Corner

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner
diagram	<pre> classDiagram     class ModifiedCorner {         type ModifiedCornerType     }     class ModifiedCornerType {         ImpactedTerminal ImpactedTerminalType         Impact-to-TerminalGroup Impact-to-TerminalGroupType         ShapeImpactedCorner ImpactedCornerType     }     class ImpactToTerminalGroupType {         Apply-to-all-Terminals Apply-to-all-TerminalsType         Symmetry SymmetryType     }     ModifiedCorner --&gt; ModifiedCornerType     ModifiedCornerType --&gt; ImpactToTerminalGroupType         </pre>
type	ModifiedCornerType, ImpactedTerminalType, TerminalCenterType, Impact-to-TerminalGroupType, Apply-to-all-TerminalsType, SymmetryType, ImpactedCornerType, CornerType.

### A.1.25.1 Impacted Terminal

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ImpactedTerminal
diagram	<pre> graph LR     subgraph ImpactedTerminalType         direction TB         subgraph TerminalCenterType             direction TB             x["x type xs:decimal"]             y["y type xs:decimal"]             PitchRadius["PitchRadius type xs:decimal"]             Center["Center type JEP30-D10:PointXYType"]             Angle["Angle type xs:integer"]         end         RowTerminalIndex["RowTerminalIndex type JEP30-D10:MinIntegerOfOneType minIncl/maxIncl 1"]         ColumnTerminalIndex["ColumnTerminalIndex type JEP30-D10:MinIntegerOfOneType minIncl/maxIncl 1"]         PolarTerminalIndex["PolarTerminalIndex type JEP30-D10:MinIntegerOfOneType minIncl/maxIncl 1"]     end     ImpactedTerminalType --&gt; TerminalCenterType     ImpactedTerminalType --&gt; RowTerminalIndex     ImpactedTerminalType --&gt; ColumnTerminalIndex     ImpactedTerminalType --&gt; PolarTerminalIndex </pre>
type	ImpactedTerminalType, TerminalCenterType, JEP30-D10:PointXYType.

### A.1.25.2 Impact – to – Terminal Group

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup
diagram	<pre> graph LR     subgraph Impact-to-TerminalGroupType         direction TB         ApplyToAllTerminals["Apply-to-all-Terminals type Apply-to-all-TerminalsType"]         subgraph SymmetryType             direction TB             Rotation["Rotation type SymmetryRotationType"]             Reflection["Reflection type ReflectionType"]         end     end     ImpactToTerminalGroupType["Impact-to-TerminalGroup type Impact-to-TerminalGroupType"] --&gt; ApplyToAllTerminals     ImpactToTerminalGroupType --&gt; SymmetryType </pre>
type	Impact-to-TerminalGroupType, Apply-to-all-TerminalsType, SymmetryType, SymmetryRotationType, ReflectionType,

#### A.1.25.2.1 Apply – to – all - Terminals

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner</a>
diagram	
type	<a href="#">Apply-to-all-TerminalsType</a> , <a href="#">CornerImpact-to-StandardArrayType</a> , <a href="#">JEP30-D10:EmptyType</a> .

#### A.1.25.2.2 Rotation

path	<a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup/Symmetry/Rotation</a>
diagram	
type	<a href="#">SymmetryRotationType</a> , <a href="#">SymmetryRotationAxisType</a> , <a href="#">SymmetryRotationCenterType</a> , <a href="#">JEP30-D10:EmptyType</a> .

A.1.25.2.3 Reflection

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/Impact-to-TerminalGroup/Symmetry/Reflection
diagram	<pre>classDiagram     class ReflectionType     class Reflection {         type ReflectionType     }     class MirrorPlane {         type ReflectionAxisType     }     class InversionCenter {         type ReflectionInversionCenterType     }     class ReflectionAxisType {         xyPlane {             type JEP30-D10:EmptyType         }         yzPlane {             type JEP30-D10:EmptyType         }         xzPlane {             type JEP30-D10:EmptyType         }     }     class ReflectionInversionCenterType {         TerminalCenter {             type JEP30-D10:EmptyType         }         PackageBodyCenter {             type JEP30-D10:EmptyType         }     }     ReflectionType &lt; -- Reflection     Reflection &lt; -- MirrorPlane     Reflection &lt; -- InversionCenter     MirrorPlane --&gt; ReflectionAxisType     InversionCenter --&gt; ReflectionInversionCenterType     ReflectionAxisType &lt; -- xyPlane     ReflectionAxisType &lt; -- yzPlane     ReflectionAxisType &lt; -- xzPlane     ReflectionInversionCenterType &lt; -- TerminalCenter     ReflectionInversionCenterType &lt; -- PackageBodyCenter</pre>
type	ReflectionType, ReflectionAxisType, JEP30-D10:EmptyType, ReflectionInversionCenterType.

A.1.25.3 Shape Impacted Corner

path	PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner
diagram	<p>The diagram illustrates the structure of the Shape Impacted Corner. It features a central <b>ShapeImpactedCorner</b> class (solid box) with a type <b>ImpactedCornerType</b>. This class is connected via a composition relationship (indicated by a solid line with a filled diamond) to a large yellow-shaded container labeled <b>ImpactedCornerType</b>. Inside this container, there are several components: <ul style="list-style-type: none"><li>A dashed box labeled <b>CornerType</b> containing five classes: <b>ChamferedCorner</b> (type <b>ChamferedCornerType</b>), <b>RectangularInCorner</b> (type <b>RectangularInCornerType</b>), <b>ConvexCorner</b> (type <b>ConvexCornerType</b>), <b>ConcaveCorner</b> (type <b>ConcaveCornerType</b>), and <b>CornerArc</b> (type <b>CornerArcType</b>). Each of these classes has a composition relationship with the <b>CornerType</b> container.</li><li>A dashed box containing three classes: <b>NE</b> (type <b>CornerType</b>), <b>SE</b> (type <b>CornerType</b>), <b>SW</b> (type <b>CornerType</b>), and <b>NW</b> (type <b>CornerType</b>). Each of these classes has a composition relationship with the <b>CornerType</b> container.</li></ul></p>
type	ImpactedCornerType, CornerType, ChamferedCornerType, RectangularInCornerType, ConvexCornerType, ConcaveCornerType, CornerArcType.

#### A.1.25.4 Corner Type

path	<p><a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/NE</a></p> <p><a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/SE</a></p> <p><a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/SW</a></p> <p><a href="#">PartModel/PackageSection/Package-Array/Package/TerminalGroups/TerminalGroup-Array/TerminalGroup/TerminalShape/ModifiedCorner/ShapeImpactedCorner/NW</a></p>
diagram	<pre> classDiagram     class CornerType {         &lt;&lt;abstract&gt;&gt;     }     class ChamferedCorner {         DX         DY     }     class RectangularInCorner {         DX         DY     }     class ConvexCorner {         Radius         ConvexCornerCenter     }     class ConcaveCorner {         Radius         ConcaveCornerCenter     }     class CornerArc {         Radius1         Radius2         Neck         CornerArcCenter     }     CornerType &lt; -- ChamferedCorner     CornerType &lt; -- RectangularInCorner     CornerType &lt; -- ConvexCorner     CornerType &lt; -- ConcaveCorner     CornerType &lt; -- CornerArc </pre>
type	<p>ImpactedCornerType, CornerType, ChamferedCornerType, RectangularInCornerType, ConvexCornerType, ConcaveCornerType, CornerArcType.</p>



## A.2 Terminal Dimensions

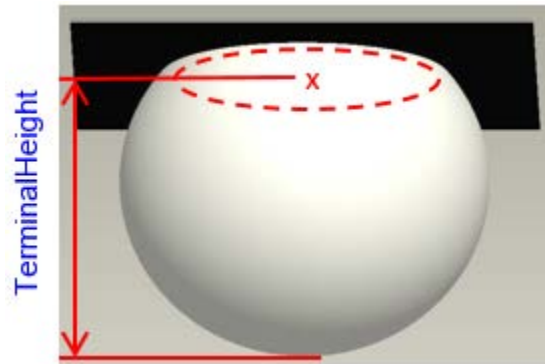
The following sections outlines the vertical dimensions that are required for each terminal.

### A.2.1 Ball

The Height of the Ball is dependent upon the *Ball* type and is mandatory.

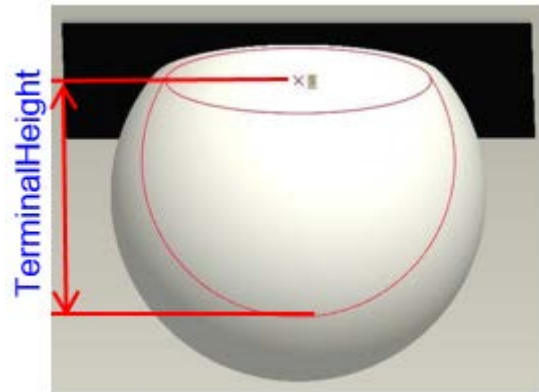
#### A.2.1.1 Collapsing Ball

For the *Ball/Collapsing* and for the *Ball/Bump*, the *TerminalHeight* is given as the dimension from the Seating Plane to the underside of the Package.



#### A.2.1.2 Non-collapsing Ball

For the *Ball/Non-collapsing*, the *TerminalHeight* is given as the dimension from the Seating Plane to the underside of the Package after the Package is soldered to the PCB Substrate. A Non-collapsing Ball has a high temperature ball within a ball, so that during reflow, only the outer ball reflows and the inner ball does not. This inner ball holds the package up from the printed board and prevents it from collapsing any further. This is the height that needs to be captured.

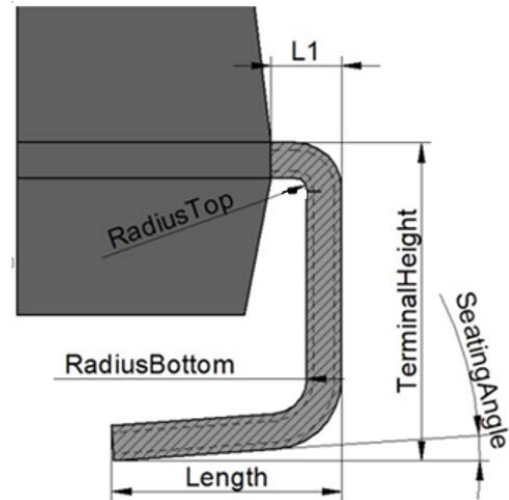


### A.2.2 C-Bend

C-bend compliant terminal extend from the sides of the body, bend down, and form a flat contact area with the board under the package body. It is similar to a J-bend terminal except that the bottom of the J-terminal is flat, and not rounded up under the part body.

When terminals are in a Dual position on the device, then the dimension *Length* can be derived from alternative dimensions provided, such as

- 1)  $(\text{Terminal Span} - \text{Terminal Spacing})/2$ ,
- 2)  $\text{Terminal Span} - \text{Terminal Span Pitch}$ ,
- 3) etc.

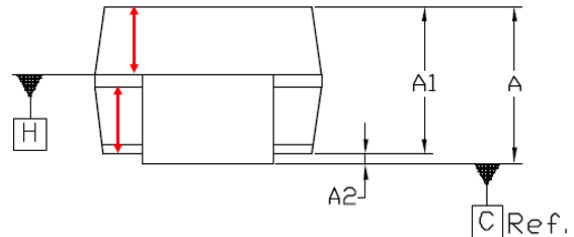


Similarly, the *PackageEdge-to-end-of-Terminal* can be derived from alternative dimensions provided if the terminals are in a Dual position, such as

- 1)  $(\text{Terminal Span} - \text{Package dimension})/2$ ,
- 2) etc.

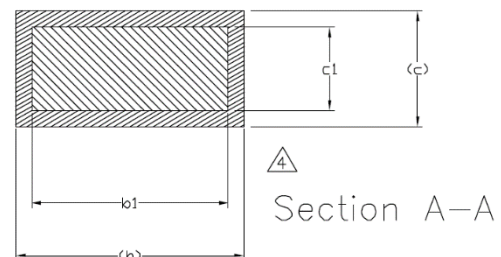
The *Width* (b) of the terminal is necessary to accurately calculate the width of the land pattern and is therefore mandatory to be captured.

*TerminalHeight* is the dimension from the Seating Plane to the highest point where the Terminal exits the Package Body as defined by the datum H in this drawing. The provision of this dimension will facilitate a more accurate representation of a 3D model created from this content. This data is mandatory to be provided, but if unavailable, it can be assumed that the portion of the package height above the terminal frame is also the same height of the package body below the terminal frame.



- 1)  $\text{Terminal Height} = (\text{Package Body Height (A1)} - \text{Terminal Thickness})/2 + \text{Terminal Thickness} + \text{Standoff (A2)}$

*TerminalThickness* is defined as the thickness of the terminal (c). It is the same value on the horizontal portion as it is on the vertical portion. This is the dimension that drives the Toe of the Land Pattern in the land pattern calculation and is therefore mandatory. The Plating thickness is the value of the outer dimensions – base metal dimensions (i.e., b-b1 or c-c1 which should result in the same value).



The *RadiusTop* and *RadiusBottom* of the terminal is optional but preferred to accurately represent the 3D model of the part. When unspecified, it can be assumed to be equal to the terminal thickness.

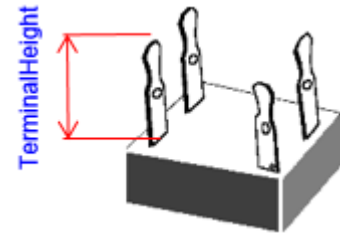
### A.2.2 C-Bend (cont'd)

The *SeatingAngle* of the terminal is optional but preferred to accurately represent the 3D model of the part. When unspecified, it can be assumed to be 4 degrees, with a min of 0 degrees and a max of 8 degrees.

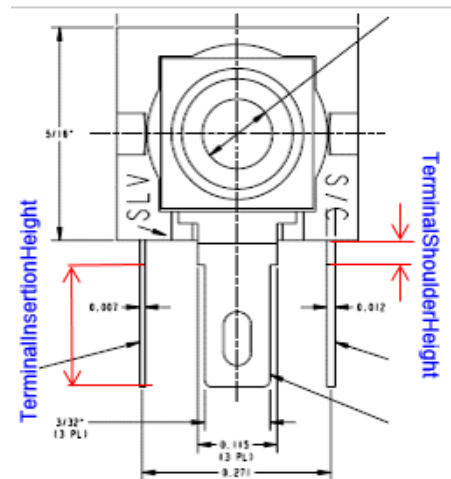
The *RiseAngle* of the terminal is optional but preferred to accurately represent the 3D model of the part. When unspecified, it can be assumed to be 90 degrees for a c-bend terminal.

### A.2.3 Lug

Lugs typically are not soldered to the PCB. Instead they have mating connectors or wires connected to them. However some are soldered to the printed board and some penetrate the printed board to be connected on the other side. Therefore if the Lug is vertical off the upperside of the Package, then the *TerminalHeight* is captured as shown here.



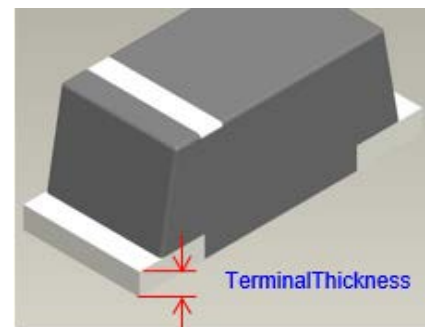
When the Terminals exit from the underside of the body, with the intention of penetrating the PCB, then the *TerminalInsertionLength* and the *TerminalShoulderHeight* are required. If either of the data is unavailable, then the User should state *Unspecified* for that respective dimension that is unavailable.



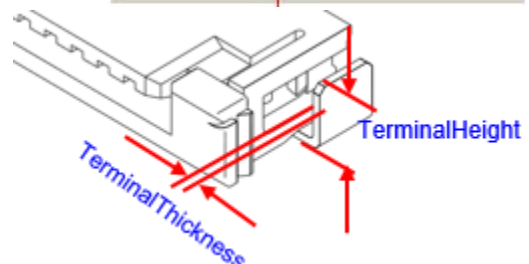
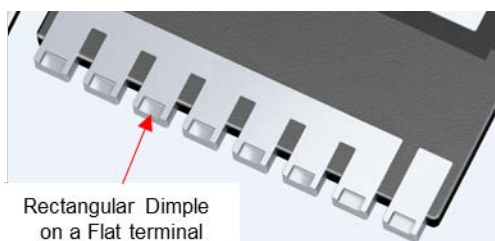
### A.2.4 Flat Terminal

*TerminalThickness* is required for *Flat/Elevated*, *Flat/Hole*, *Flat/With-opening* terminals or for a normal *Flat* terminal.

The variant *Flat-L-bend* terminal however does have both the *TerminalThickness* plus the *TerminalHeight* requirements. The thickness of the Terminal will determine the side fillets outside the Body outline, whereas the height of the outer L-Bend will determine the Toe.



*Flat* terminals can also have horizontal castellations *Rectangular Dimple*

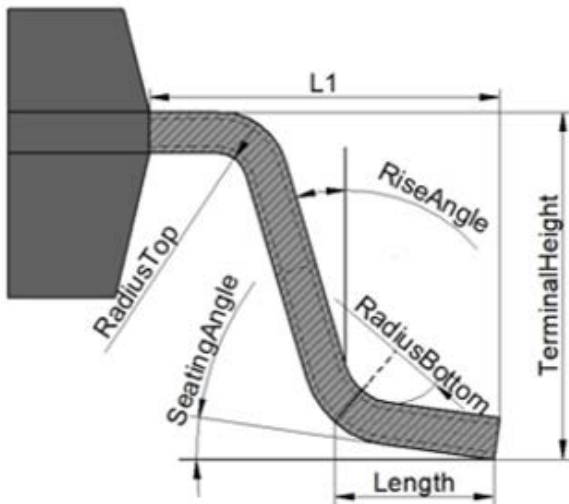


## A.2.5 Gull-wing Terminal

Gull-wing compliant terminal extend from the sides of the body, bend down, and form a flat contact area with the board outside the package body. It is similar to a C-terminal except that the bottom of the J-terminal points away from the package center and not towards the package center..

As with the C-bend, when terminals are in a Dual position on the device, then the dimension *Length* can be derived from alternative dimensions provided, such as

- 1)  $(\text{Terminal Span} - \text{Terminal Spacing})/2$ ,
- 2)  $\text{Terminal Span} - \text{Terminal Span Pitch}$ ,
- 3) etc.

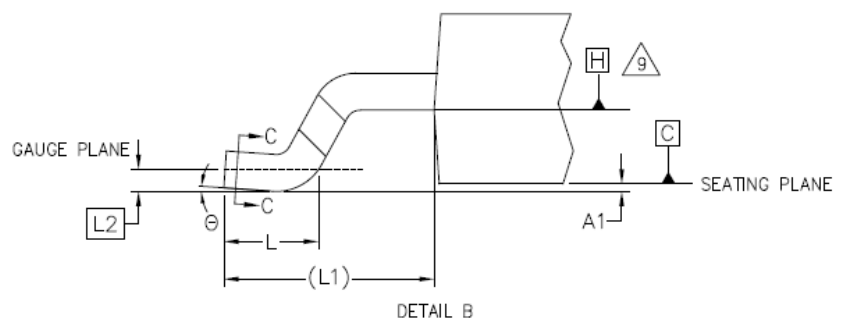
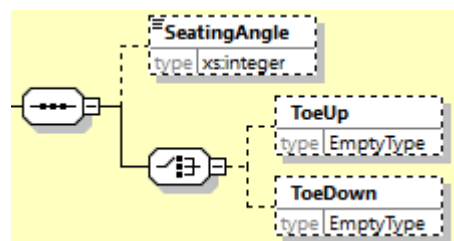


Similarly, the *PackageEdge-to-end-of-Terminal* can be derived from alternative dimensions provided if the terminals are in a Dual position, such as

- 1)  $(\text{Terminal Span} - \text{Package dimension})/2$ ,
- 2) etc.

The elements *Width*, *TerminalHeight*, *TerminalThickness*, *RadiusTop*, *RadiusBottom*, *SeatingAngle*, and *RiseAngle*, follow the same definitions as that provided for the C-bend.

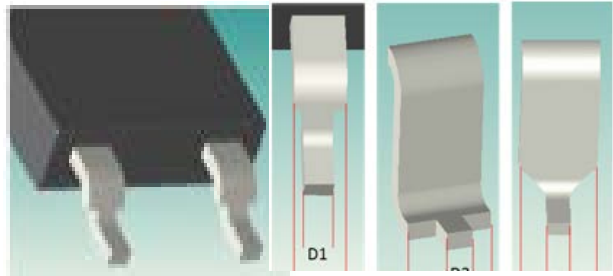
The above image shows that the Gull-wing terminal toe is pointing downwards, but an option is available for Gull-wing terminals whose Toe is pointing upwards.



The above concepts apply to all variants types of the Gull Wing Terminal with Modifications. There are several basis types of gull-wing modifications as defined herein.

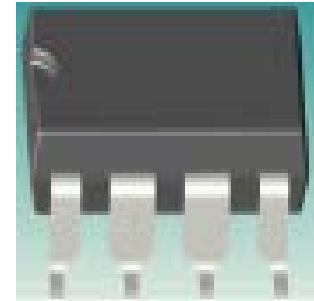
## A.2.5 Gull-wing Terminal (cont'd)

When a Gull-wing has a shoulder, this is typically centered across the terminal contact area with the printed board. Notice that the variations can have the transition from the narrow portion of the terminal to the wider portion of the terminal at right angles or at a taper. The schema has the option to capture the details for either options. The schema provides 4 options to capture the dimensional details, namely:



- 1) From the end of terminal to the Start/End of the Shoulder taper transition,
- 2) From the Package edge to the Start/End of the Shoulder taper transition,
- 3) From the Seating Plane to the Start/End of the Shoulder taper transition,
- 4) From the top of the Terminal Exit from the Package to the Start/End of the Shoulder taper transition.

A DIP TH terminal can be prepped to be surface mounted as opposed to be TH mounted. In this situation, the element *TH-PreppedShoulder* should be selected.



For Inner and Outer configurations, we can assume that the edge of the terminal contact area aligns perfectly with the edge of the neck (or Shoulder). No Offset needs to be specified.

When the edge of the terminal contact area does not align perfectly with the edge of the neck (or Shoulder), then the *Offset* can be set under the *Configuration*.



Configuration = Inward



Configuration = Center

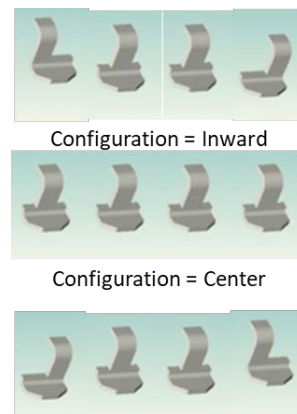


Configuration = Outward

## A.2.5 Gull-wing Terminal (cont'd)

A Neck is very similar to a Shoulder except that the width of the neck is narrower than the width of the terminal contact area.

Typically, these Dimensions are “Unspecified Value Set type”, meaning that if the datasheet does not provide these values, we will set them to defaults.



The final gull-wing modification is a Dambar as shown here. The Dambar protrusion is a remnant from tie bar cutting.

The terminal width dimension “b” does not include dambar protrusion, but most manufacturers will specify the allowable dambar protrusion as not causing the terminal width to exceed the maximum terminal width dimension by a specified amount.

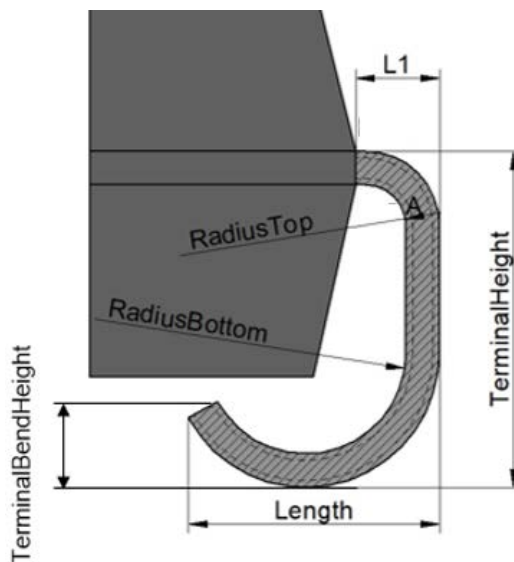


## A.2.6 J-Bend

J-bend compliant terminal extend from the sides of the body, bend down, and form a curved contact area with the board under the package body. It is similar to a C-bend terminal except that the bottom of the J-terminal is curved and rounded up under the part body.

When terminals are in a Dual position on the device, then the dimension *Pitch* in the direction across the package body for the terminals in the dual position is normally provided, in addition to the terminal *Length*.

The elements *PackageEdge-to-end-of-Terminal*, *Width*, *TerminalHeight*, *TerminalThickness*, *RadiusTop*, *RadiusBottom*, and *RiseAngle*, follow the same definitions as that provided for the C-bend. *TerminalBendHeight* is useful for the purpose of generating more accurate 3D models and is therefore optional.



The applicable shoulder elements for the *ShoulderTransition/Tapered* shoulder are as follows:

- 1) *SeatingPlane-to-start-of-ShoulderTaperTransition*, and *SeatingPlane-to-start-of-ShoulderTaperTransition*,



### A.2.6 J-Bend (cont'd)

or

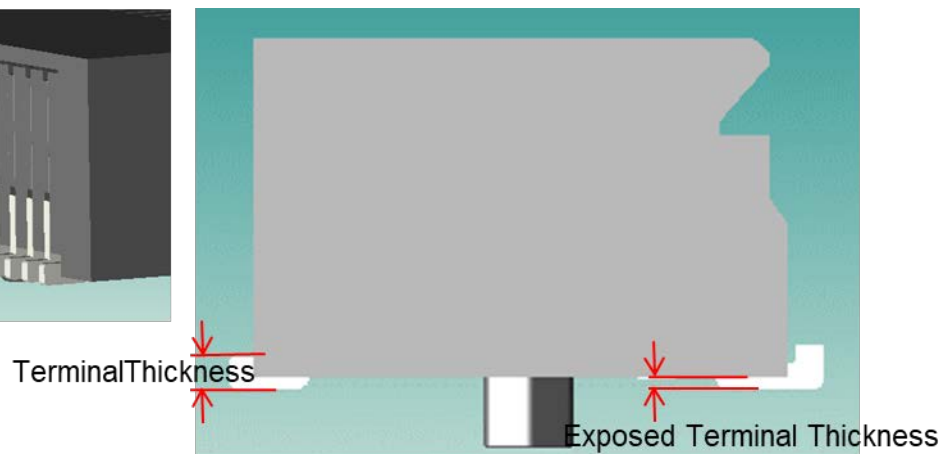
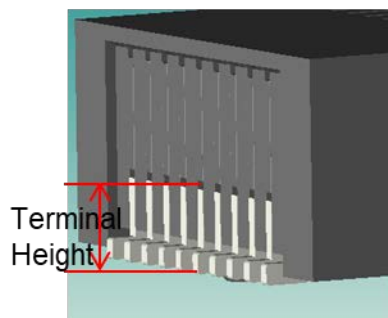
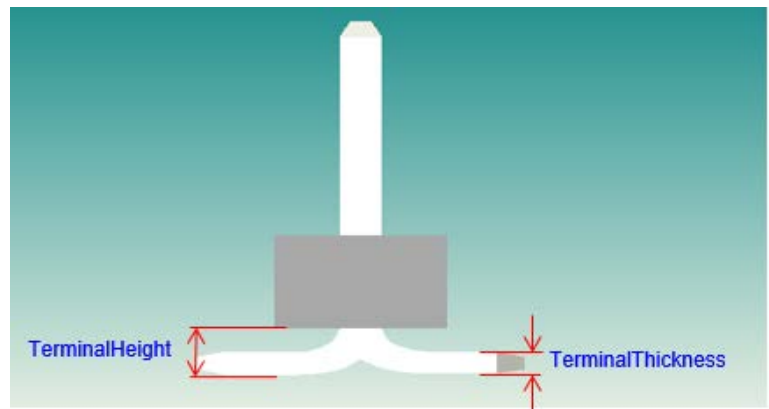
- 2) *Top-of-TerminalExit-from-Package-to-start-of-ShoulderTaperTransition*, and *Top-of-TerminalExit-from-Package-to-start-of-ShoulderTaperTransition*.

The applicable shoulder elements for the *ShoulderTransition/RightAngled* shoulder are the

- 1) *SeatingPlane-to-ShoulderTransition*, or *Top-of-TerminalExit-from-Package-to-ShoulderTransition*.

### A.2.7 L-bend

The *L-bend* terminals exits vertically from the bottom of the package and then turns horizontal. Irrespective of whether, the terminal variant is *Inward*, *Outward*, *SideInward*, *SideOutward* or it is a normal *L-bend* in which the position is one of *Single*, *Dual*, *Triple* or *Quad*, the *TerminalThickness* drives the Toe calculation of the land pattern, while the *TerminalHeight* drives the heel portion of the calculation.

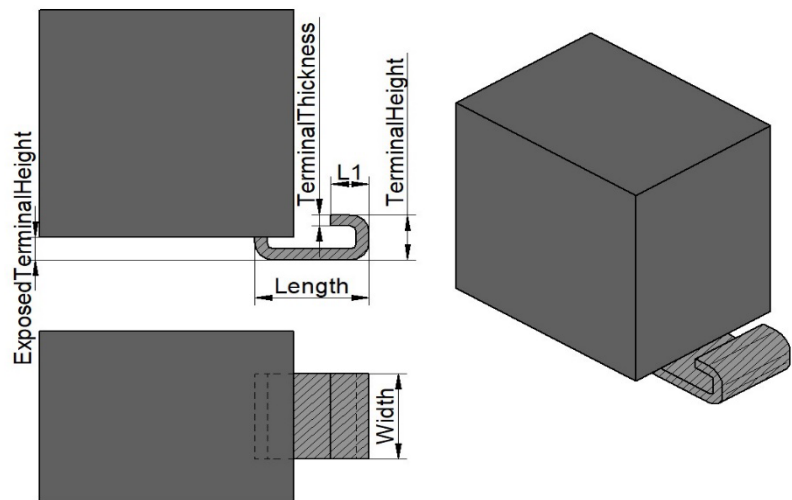


Sometimes as shown here in the 2nd image, the package body sits on the top surface of the horizontal portion of the L-Bend, in which case the *TerminalHeight* is reduced to the *TerminalThickness*.

## A.2.7 L-Bend (cont'd)

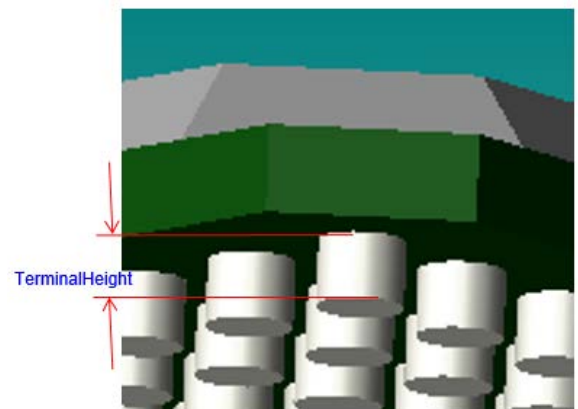
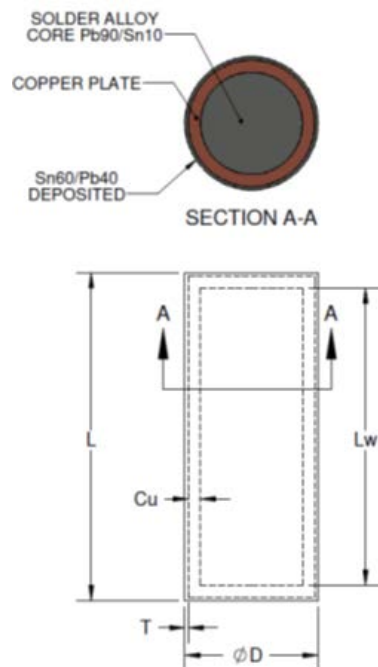
For Side-L-Bends, the terminal is partially immersed into the package body, in which case the *ExposedTerminalThickness* may be reduced to less than the *TerminalThickness*. In this case, the *ExposedTerminalThickness* is mandatory, while the *TerminalThickness* is optional.

*LC-bends* follows the same rule as the Side-L-Bend in which the *ExposedTerminalThickness* under the package body is the critical measurement.



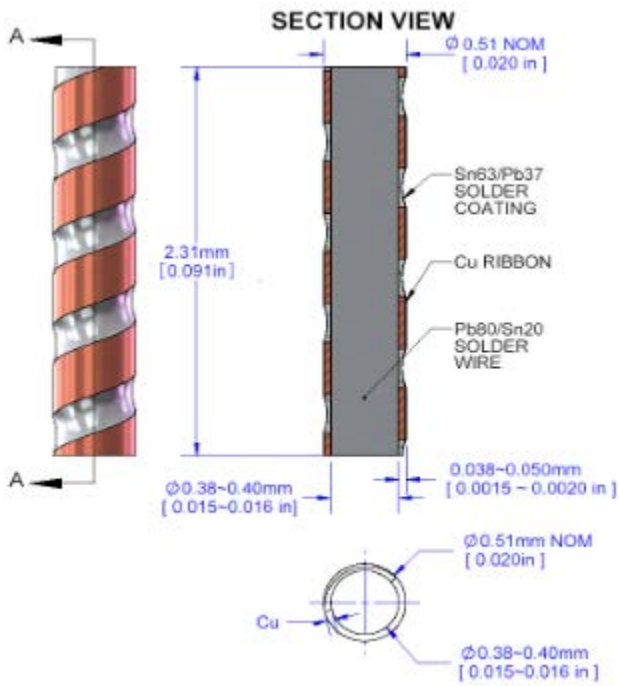
## A.2.8 Column

The Column *TerminalHeight* is the same as the Standoff for the Part, provided that there are no bumps on the package body.





### A.2.8 Column (cont'd)

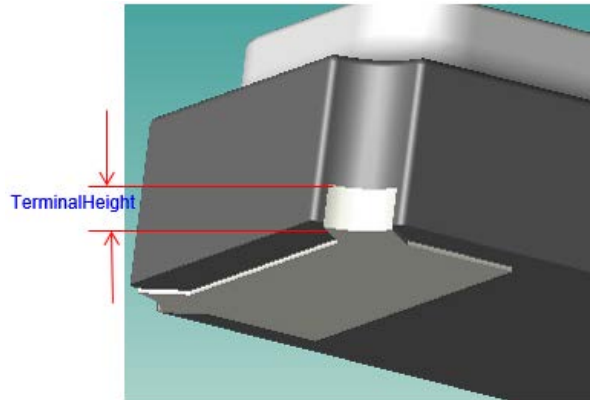


### A.2.9 Surface Terminal

The majority of *SurfaceTerminals* are under the bottom of the Package where the Terminal Contact Area is Inside the package outline. Such Terminals have no Terminal Thickness or Terminal Height as these 3 images indicate.

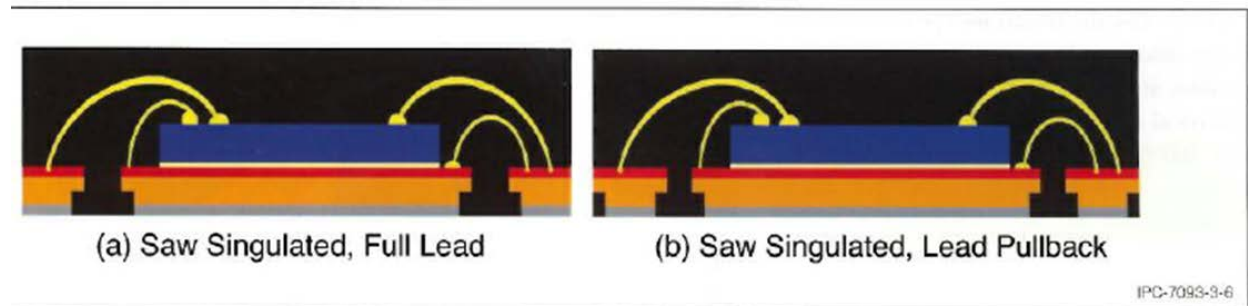
However there are variants where the edge of the *SurfaceTerminal* is touching the edge of the Body and sometime wrap up the side of the body. In such cases, *TerminalHeight* shall be captured.

Refer to the *ModifiedRectangle* and the *CornerArc* within the Modified Rectangle structure to capture the details of this terminal shape in this image to the right.

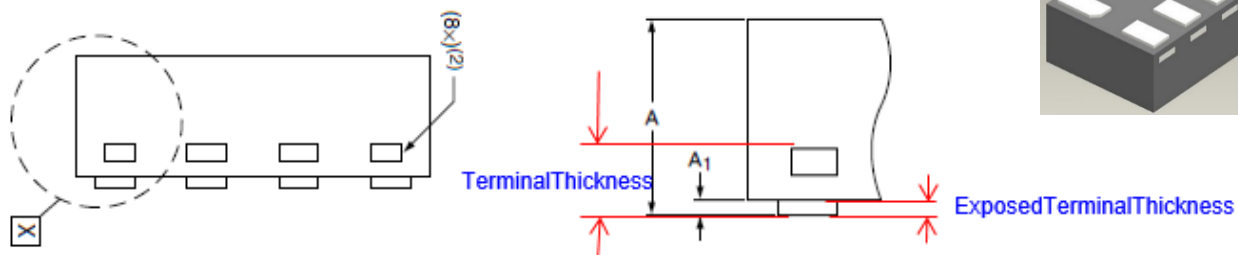


There are also some variants in which the side wrap is the same height as the package body. This is almost considered a *Wraparound* terminal, but if the terminal does not wrap around the top surface, then it is still a *SurfaceTerminal*. In these cases, the *TerminalHeight* is equal to the Body Height.

*SurfaceTerminal* variants that are *Pullback*, have an exposed thickness of the terminal under the bottom surface of the package body.



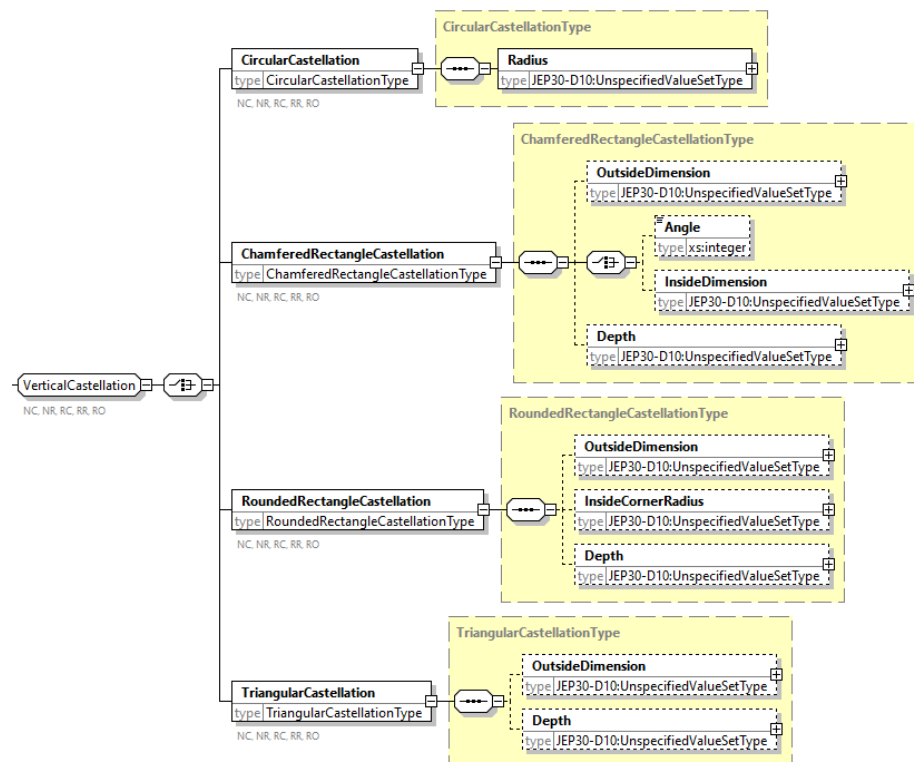
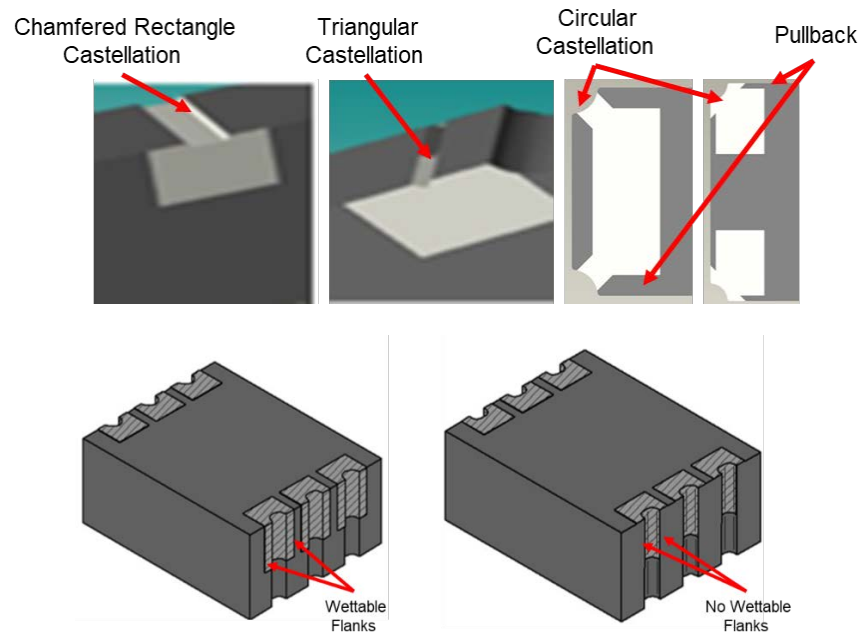
The *ExposedTerminalThickness* is critical for the calculation of Land Patterns and should be captured. The *TerminalThickness* as defined in this image is not used for land pattern generation, since the exposed piece on the side is typically not plated.



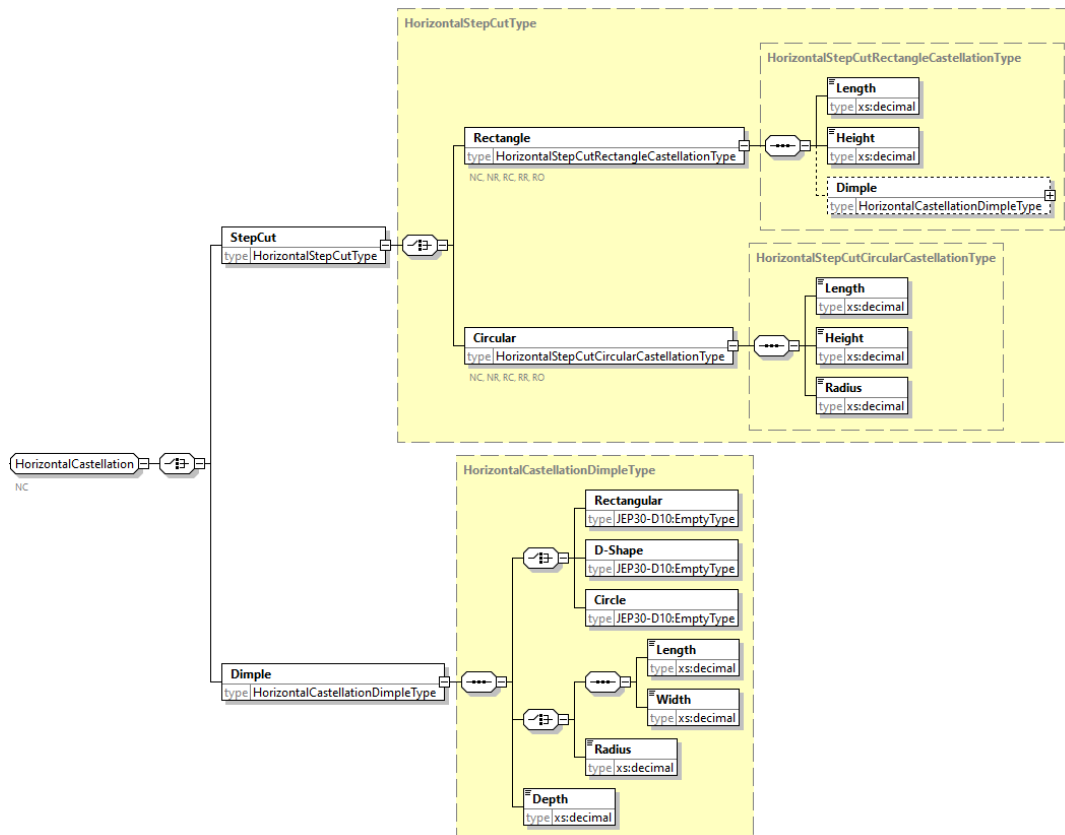
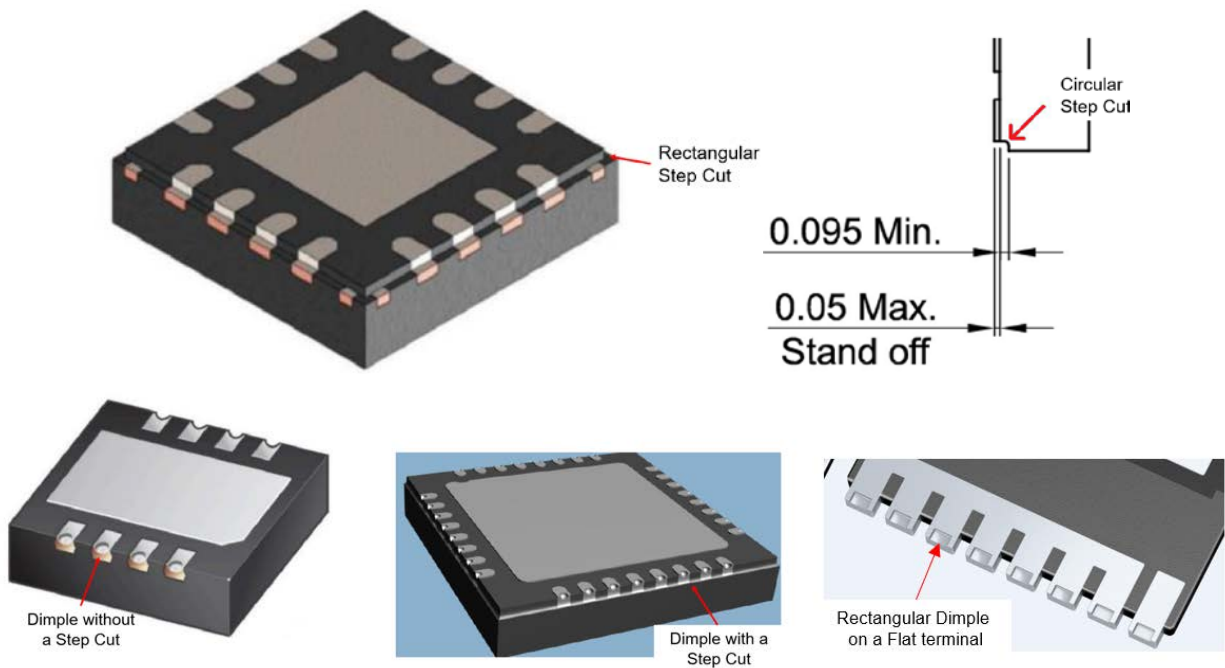
### A.2.9.1 Vertical Castellation

There are several different kinds of vertical castellations such as

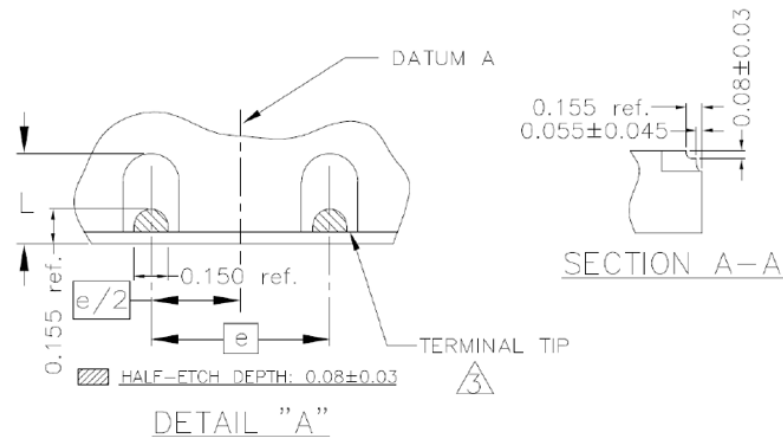
- 1) Triangular Castellation,
- 2) Chamfered Rectangle Castellation,
- 3) Rounded Rectangle Castellation, and
- 4) Circular Castellation.



## A.2.9.2 Horizontal Castellation



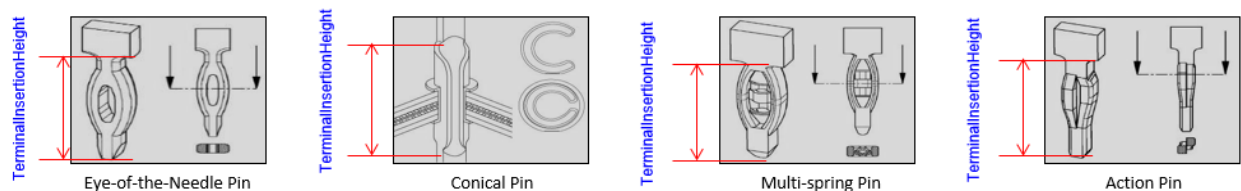
### A.2.9.2 Horizontal Castellation (cont'd)



The above example shows a Rectangular Step Cut of Length =  $0.555 \pm 0.045$ . The Dimple is a D-Shape where the Length is measured from the package body edge and therefore is 0.155. The Width of the Dimple is 0.150. While the radius of the dimple is half of 0.150 ref (i.e. 0.075), at nominal dimensions, this provides a straight edge of the D-Shape of  $0.155 - 0.075 = 0.08$ . When the length of the Step Cut is at its maximum value of  $0.055 + 0.045 = 0.1$ , then this means that the length of the dimple D-Shape is reduced to 0.055, which is less than the diameter of the D-Shape and thus a circle. However, this is ignored since the shape determination is defined at nominal dimensions.

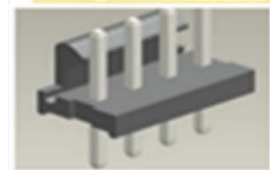
### A.2.10 Pressfit

There are 4 types of *Pressfit/Compliant* terminals plus a Non-Compliant terminal. In all cases, the dimension to be captured is the length of the terminal from the end point to the point of maximum insertion of the terminal through the hole. This stopping point can be a shoulder on the pin, or it can be the package body. This is defined as the *TerminalInsertionHeight*.

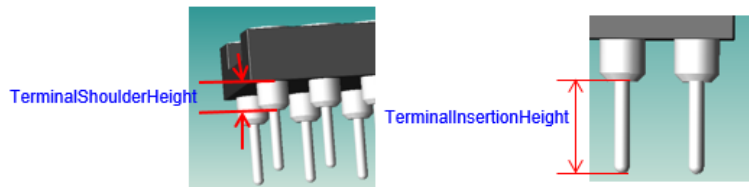


## A.2.11 Pin

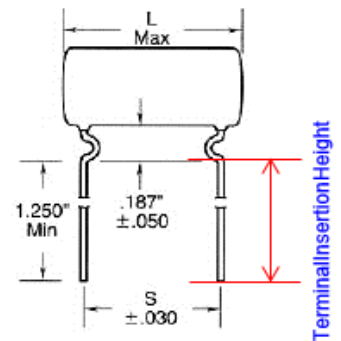
A *Pin* can be straight, *Kinked* or come with a *Shoulder*. There is also 3 additional variants, notably a *Press-inSolderable*, *Press-inNon-Solderable*, or a *SwageFasteningPin*.



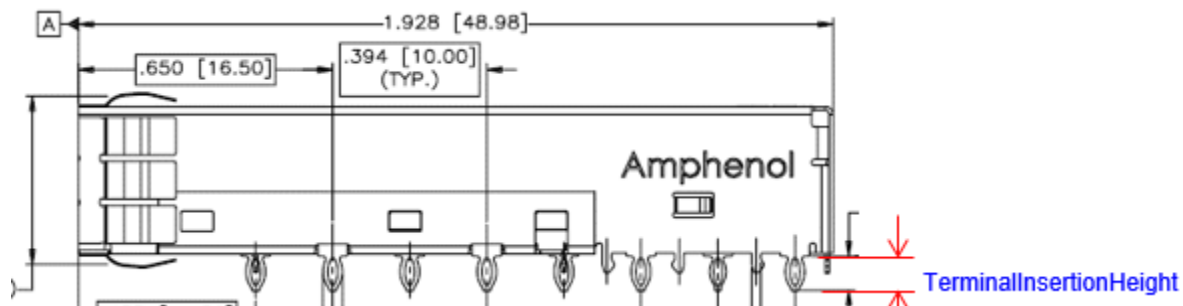
A *Pin* with a *Shoulder* will have a *TerminalInsertionHeight* that goes through the printed board. Plus a *TerminalShoulderHeight* that is typically the same as the Stand-off height.



The Pin with the variant *Kinked* acts in the same way as the Pin with the *Shoulder*. The dimension of the terminal that can penetrate the printed board (i.e. the section of the Pin below the Kink” is considered the *TerminalInsertionHeight*. The distance from the bottom of the “kink” to the underside of the package body is the considered the standoff.

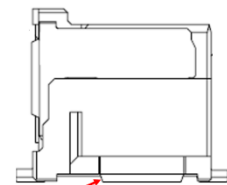


For *Press-inSolderable* terminals, the *TerminalInsertionHeight* is considered the section of the terminal below the shoulder or below the Package body in the absence of a shoulder.



The same concept applies to the *Press-inNon-Solderable* terminals. The *TerminalInsertionHeight* is the length of the terminal that could penetrate the printed board.

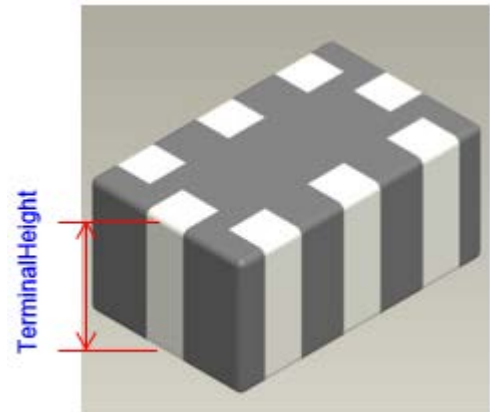
Some *Pins* are designed to be *Surface-mount* where the base of the pin is tapered.



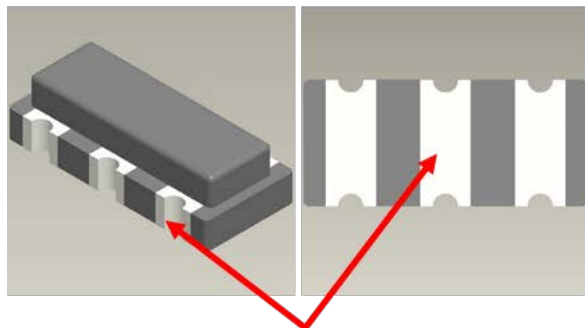
## A.2.12 Wraparound

*Wraparound* terminals are like *SurfaceTerminals* in that the surface of the terminal is on the same plane as the surface of the package body, hence the term surface terminal. *TerminalThickness* may be “0” but the *TerminalHeight* is the height of the terminal when looking at the side elevation.

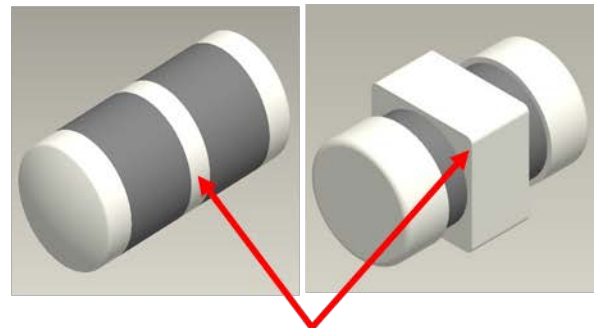
Some Parts can have a wraparound terminal that does not cover the same height as the Package Body. In this case the Terminal Height is still captured and is mandatory.



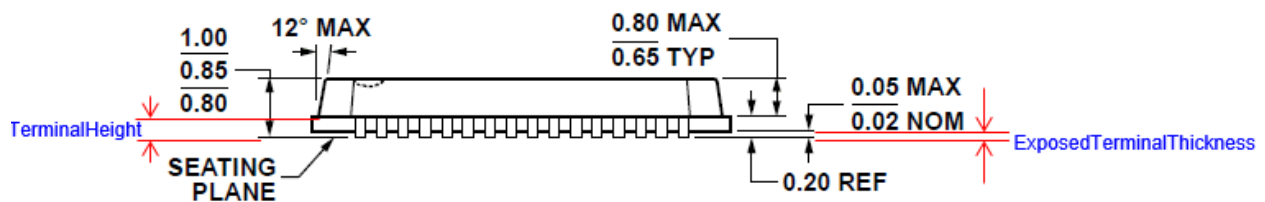
There are various options for Wraparound terminals. The castellation structure for *Wraparound* terminals is the same as for *Surface-terminals*.



Open Ring

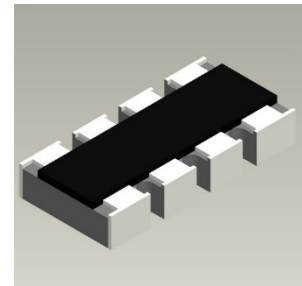


Ring



It is also possible that the surface of the wraparound terminal is not flush with the surface of the package body. This becomes particularly important on the underside of the part that connects with the land pattern. Since these terminals are embedded into the Package body material, only a portion of the terminal thickness is exposed. This *ExposedTerminalThickness* is a mandatory dimension to be collected.

Convex-S has the end Terminals larger than the terminal in between the ends.





### A.2.13 S-Bend

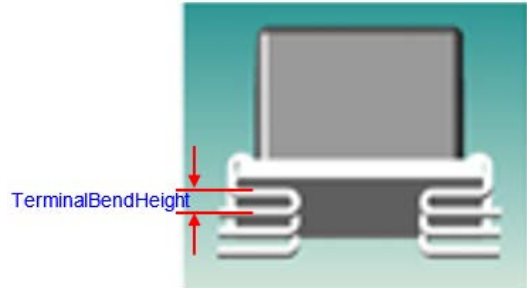
*S-bend* has a terminal contact area similar to a *C-bend*, or a *J-bend*.

The critical dimension for footprints is the *TerminalThickness* as shown here.

Similar to the J-bend, other dimensions can also be captured to facilitate more accurate 3D representation.

The *TerminalHeight* is useful only for the purpose of generating more accurate 3D models, and is therefore optional to be captured.

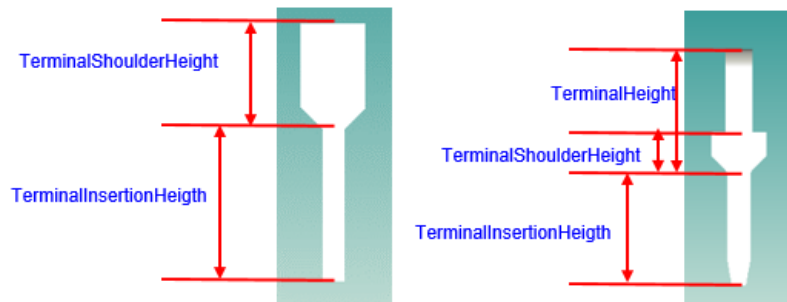
The *TerminalBendHeight* is also only useful only for the purpose of generating more accurate 3D models. It is also assumed that for S-bends, both the *TerminalHeight* and the *TerminalBendHeight* are well in excess of the max height that the solder will creep up the terminals, from which the projection to the land pattern for the toe or heel is defined. Therefore the *TerminalHeight* and the *TerminalBendHeight* are optional data to be captured.



### A.2.14 Through-Hole

Through-hole terminals can exit from the side of the package or from the underside of the Package body.

As can be seen from the different types, the terminal shoulder can travel up all the way to reach the Package body. The *TerminalShoulderHeight* is the distance from the bottom of the Shoulder to the top of the terminal shoulder, irrespective of whether or not the Terminal exits from the side of the body or from the underside of the package body.



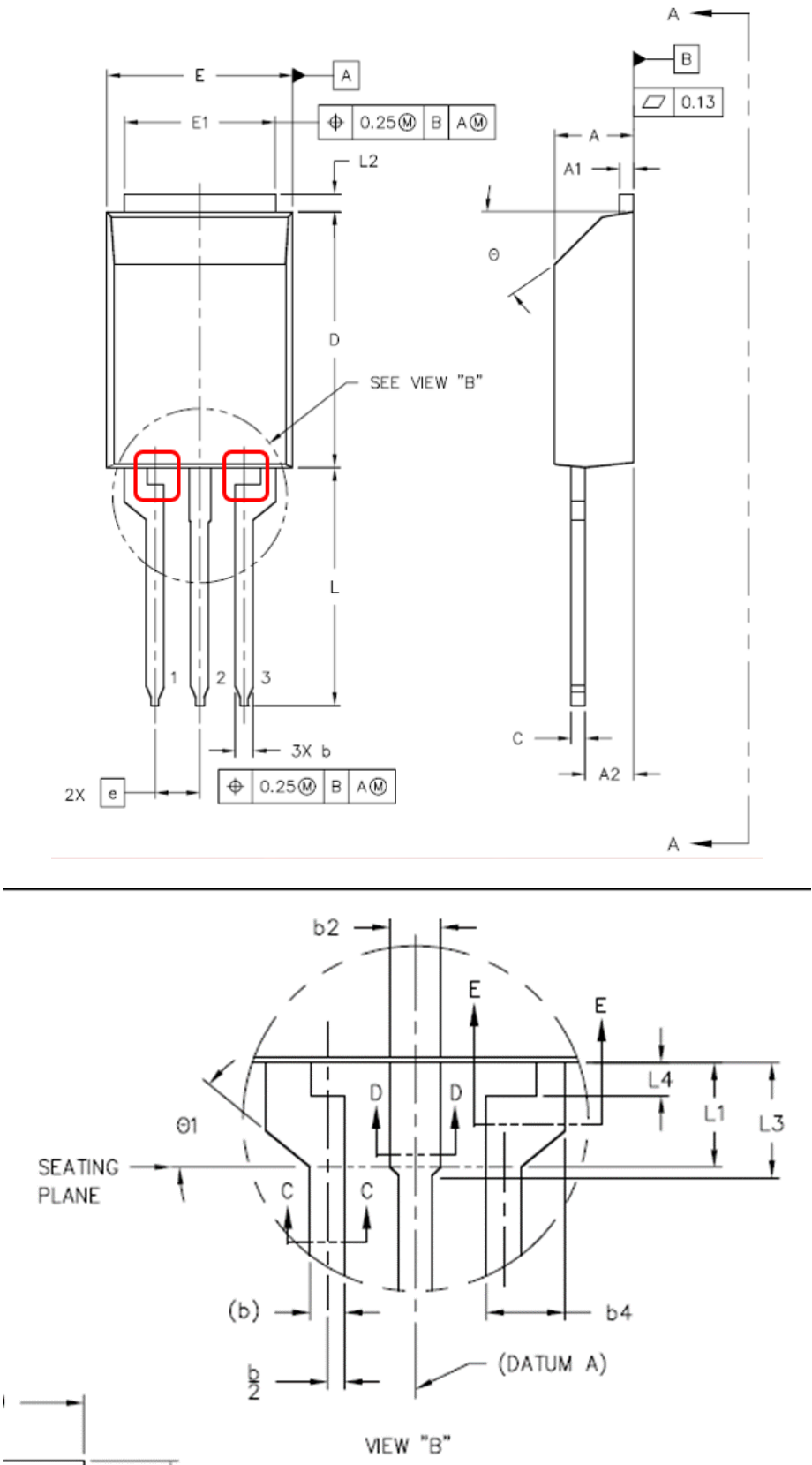
However in 2<sup>nd</sup> image, the shoulder does not reach the Package body. In this case the dimension from the bottom of the Shoulder to the top of the Terminal is called the *TerminalHeight*, whereas the distance from the bottom of the Shoulder to the top of the Shoulder remains the *TerminalShoulderHeight*.

In all types, the *TerminalInsertionHeight* is the distance from the bottom of the Shoulder to the end of the Terminal. This dimension is critical.

*Through-Hole* terminals such as those on Flange Mount devices can have cutouts at the top of the terminals. The details of these cutouts can be captured via the *TH-ShoulderCutout* section of the schema. An example of cutouts is shown below.



A.3.4 Through Hole (cont'd)

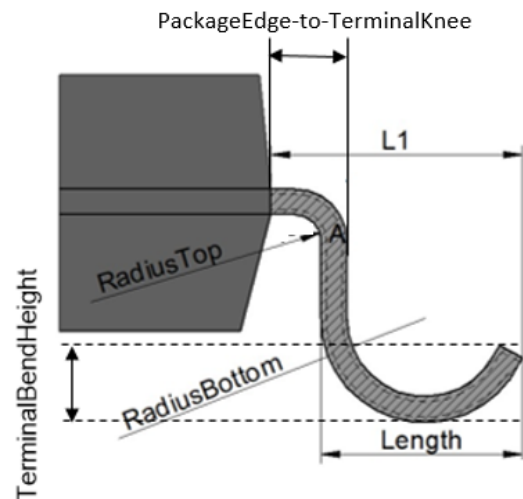


## A.2.15 J-Inverted

J-inverted compliant terminal extends from the sides of the body, bend down, and form a curved contact area with the board outside the package body. It is similar to a J-bend terminal except that the bottom of the J-terminal is outside the package as opposed to under the package.

When terminals are in a Dual position on the device, then the dimension *Pitch* in the direction across the package body for the terminals in the dual position is normally provided, in addition to the terminal *Length*.

The element the *PackageEdge-to-TerminalKnee* measures the distance from the edge of the package to the intersection of the projection of the vertical portion of the terminal as defined by the rise angle. If the rise angle is 90 degrees, then this *PackageEdge-to-TerminalKnee* dimension equals the span  $Pitch - Length + TerminalThickness \cdot 2$



The elements *Width*, *TerminalHeight*, *TerminalThickness*, *RadiusTop*, *RadiusBottom*, and *RiseAngle*, follow the same definitions as that provided for the C-bend. *TerminalBendHeight* is useful for the purpose of generating more accurate 3D models and is therefore optional.

The applicable shoulder elements for the *ShoulderTransition/Tapered* shoulder are as follows:

- 1) *SeatingPlane-to-start-of-ShoulderTaperTransition*, and *SeatingPlane-to-start-of-ShoulderTaperTransition*,
- or
- 2) *Top-of-TerminalExit-from-Package-to-start-of-ShoulderTaperTransition*, and *Top-of-TerminalExit-from-Package-to-start-of-ShoulderTaperTransition*.

The applicable shoulder elements for the *ShoulderTransition/RightAngled* shoulder are the

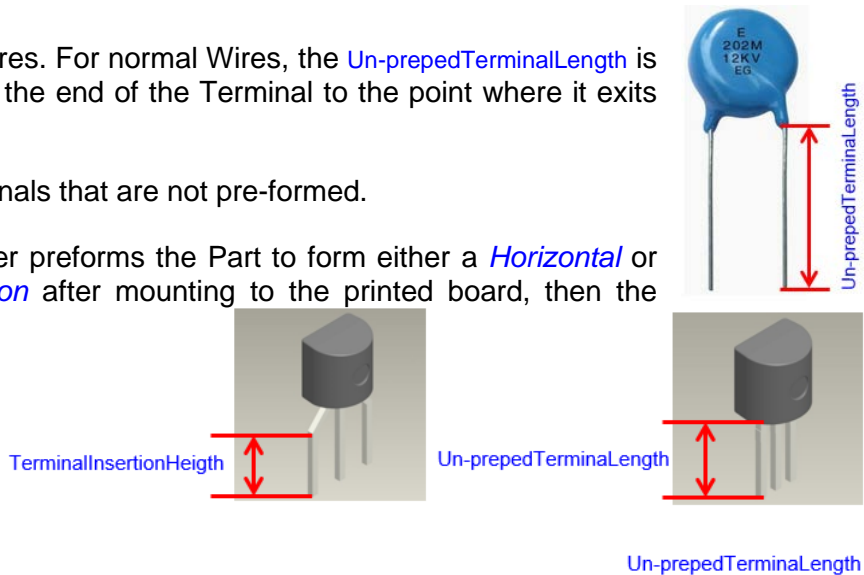
- 2) *SeatingPlane-to-ShoulderTransition*, or *Top-of-TerminalExit-from-Package-to-ShoulderTransition*.

## A.2.16 Wire

There are several types of wires. For normal Wires, the **Un-preppedTerminalLength** is defined as the distance from the end of the Terminal to the point where it exits from the Package body.

This applies to all Wire Terminals that are not pre-formed.

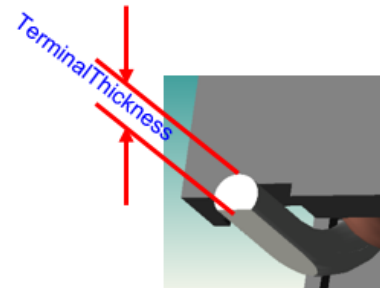
Where the Product Assembler preforms the Part to form either a **Horizontal** or **Vertical PackageBodyDirection** after mounting to the printed board, then the dimensions captured should reflect the prepped part. After prepping, the distance from the end of the Terminal to the Kink in the Terminal is defined as the **TerminalInsertionHeight**.



**Un-preppedTerminalLength** dimension is required to enable the various Terminal Insertion Spans to be calculated.



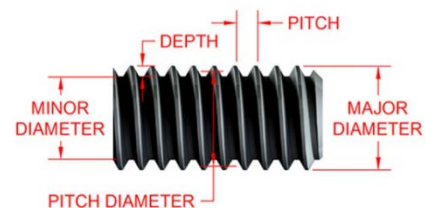
A **Wire** can also be surface mounted to the printed board, as in this image. Most **SM-Wires** are flattened so that the terminal does not roll off the pad, and this results in a lower vertical **TerminalThickness** dimension than the terminal diameter.



## A.2.17 Screw

Major diameter (outside diameter) is the largest material diameter of the thread feature. Outside or external thread diameter means the diameter of a screw and includes a raised helix height around the thread.

Minor diameter can be defined as an imaginary cylinder that can touch the roots of the thread crests.



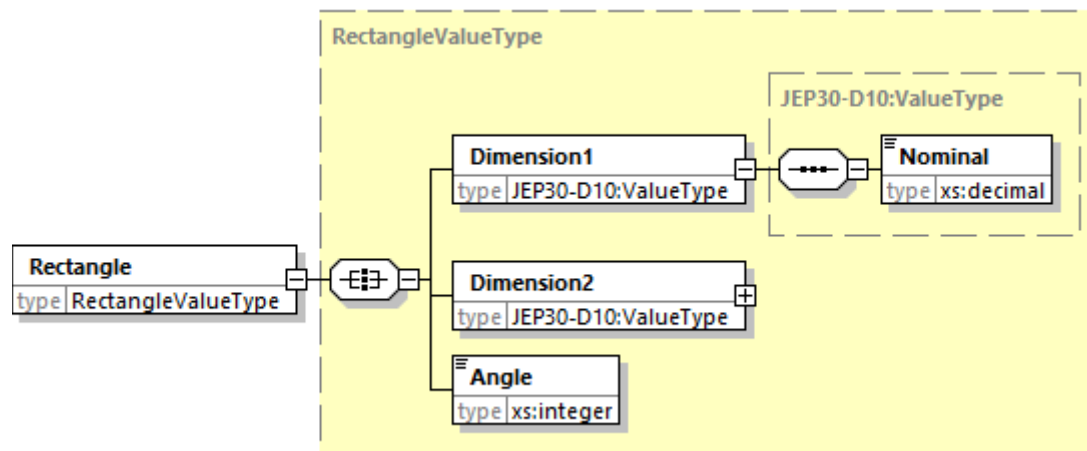
Effective or pitch diameter is the diameter of an imaginary co-axial cylinder that can intercept the surface of the thread. Pitch diameter can cross so that the intercept on a cylinder generator at the point where pitch diameter meets the opposite flanks of the thread groove. The thread groove is equal to half of the nominal pitch of the screw thread.

The Thread Pitch is a distance from a point on the same screw thread to a corresponding threshold.

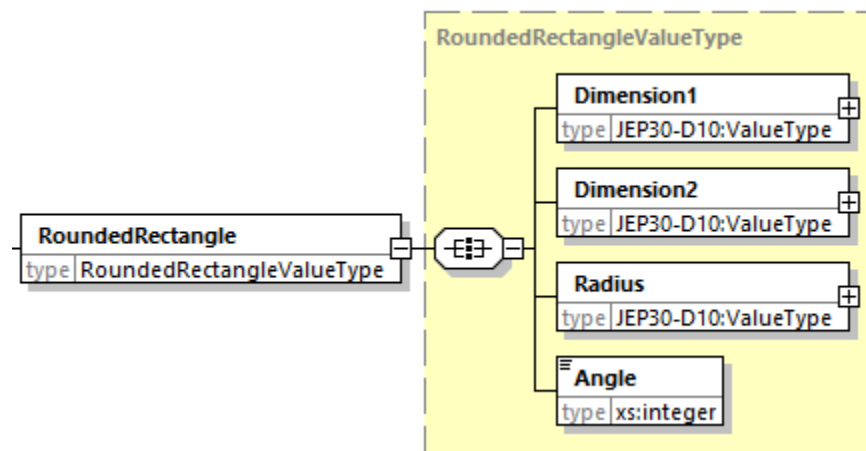
### A.3 Land Pattern Shapes

The following sections outlines the recommended land pattern shapes. All shapes defined for design tools will only contain the Nominal value, since this is the target value that is defined in the software tools for which the design is intended.

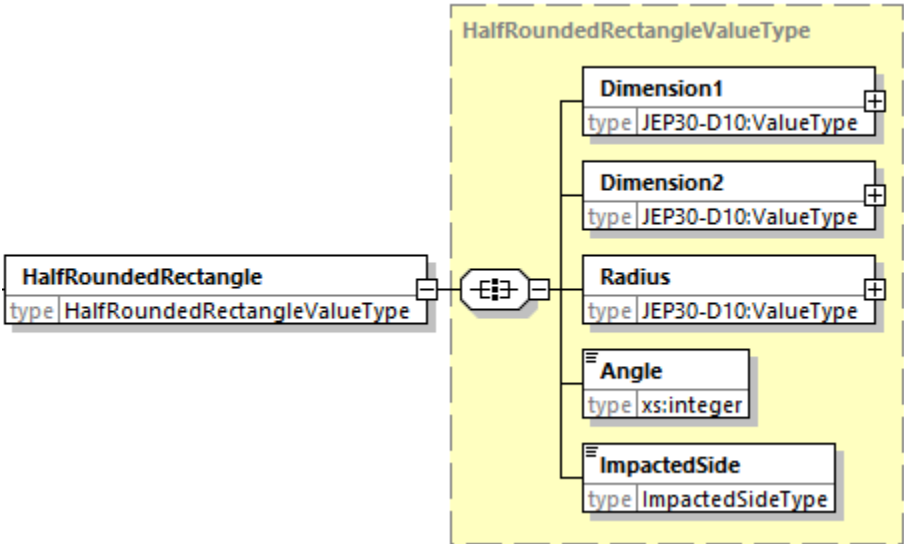
#### A.3.1 Rectangle



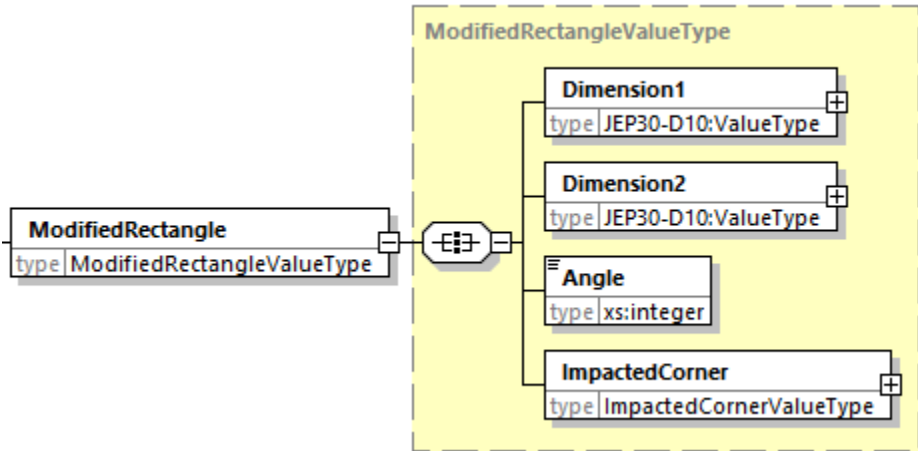
#### A.3.2 Rounded Rectangle



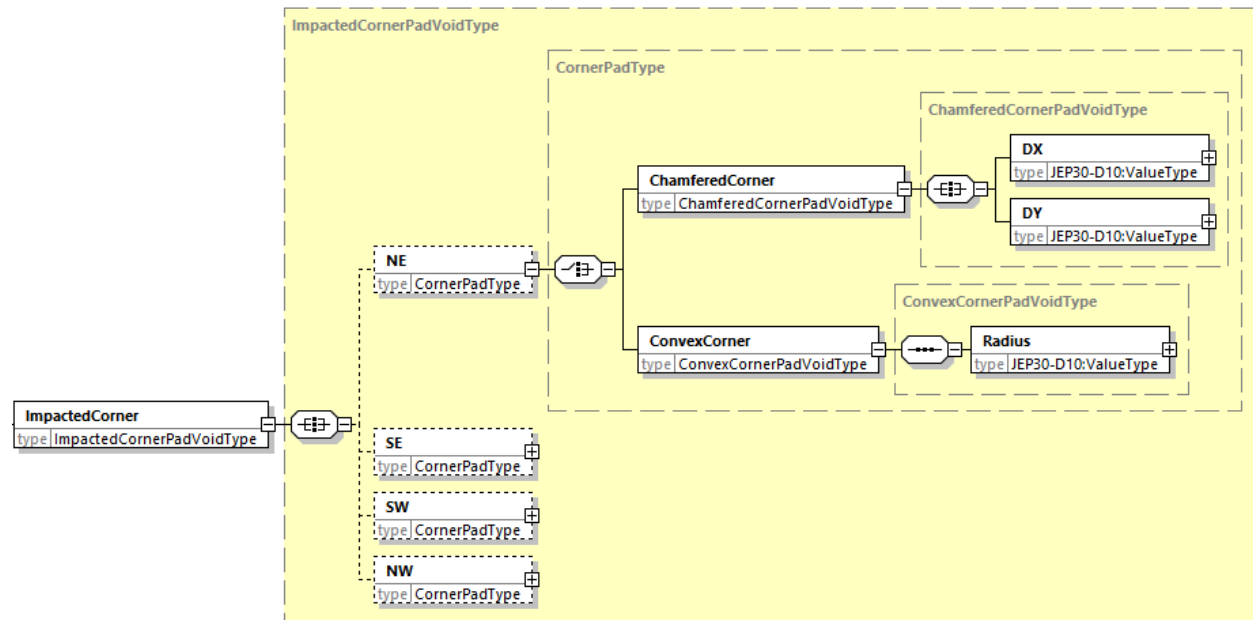
A.3.3 Half Rounded Rectangle



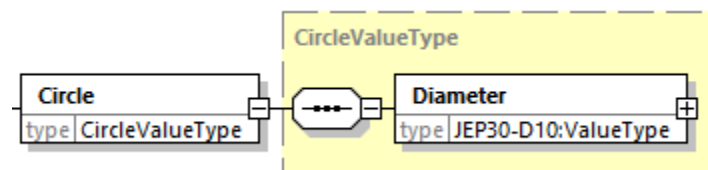
A.3.4 Modified Rectangle



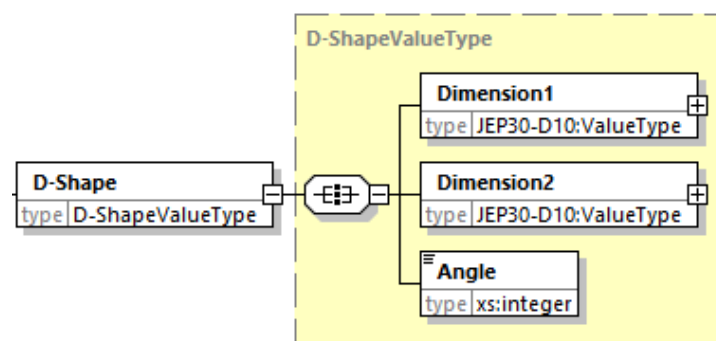
### A.3.4 Modified Rectangle (cont'd)



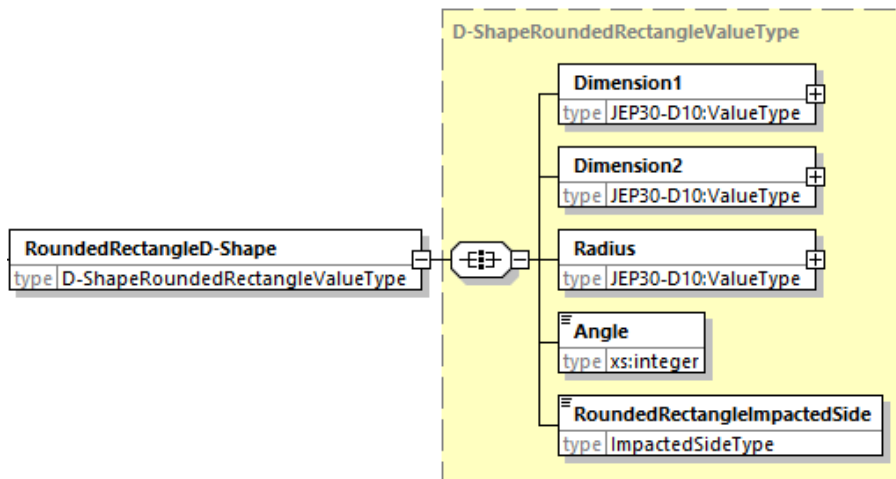
### A.3.5 Circle



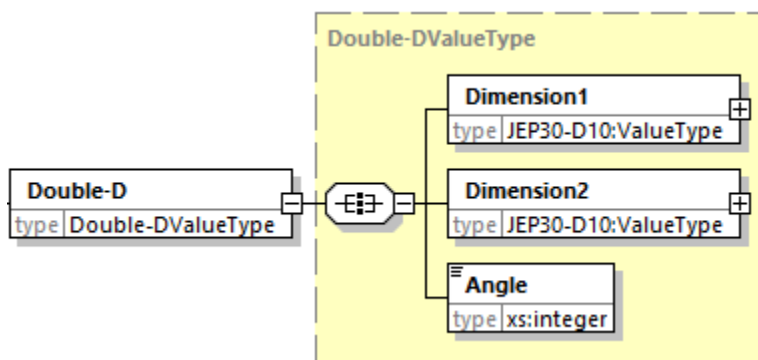
### A.3.6 D-Shape



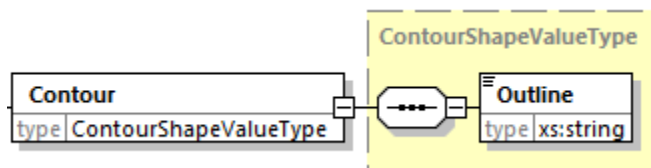
### A.3.7 Rounded Rectangle D-Shape



### A.3.8 Double-D

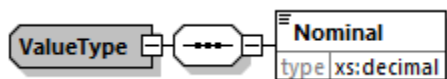


### A.3.9 Contour



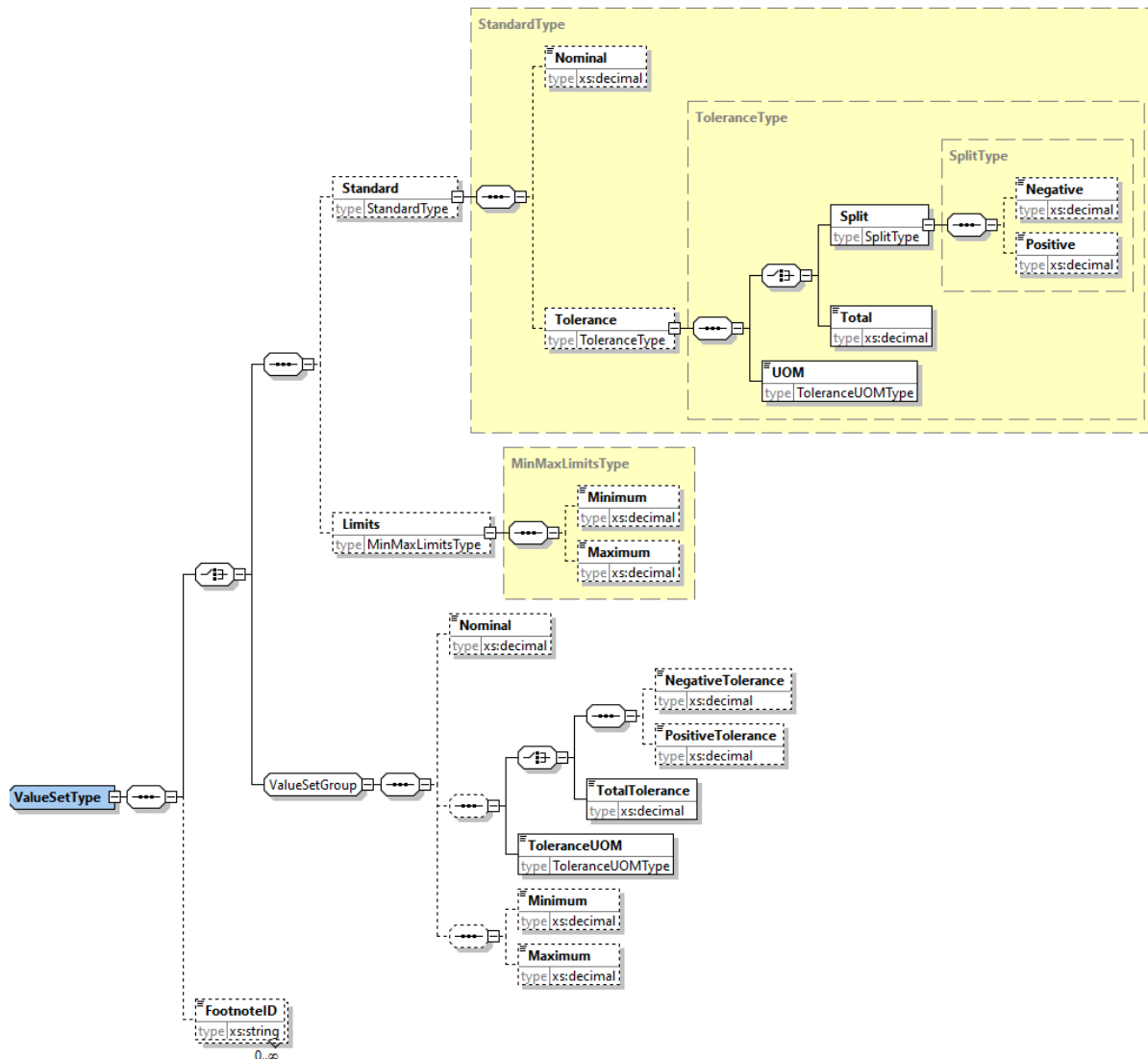
## A.4 Value Set Types

### A.4.1 Value Type



Recommended land pattern shapes, only requires the nominal to be defined and not any of the tolerances nor the min and max values

## A.4.2 Value Set Type



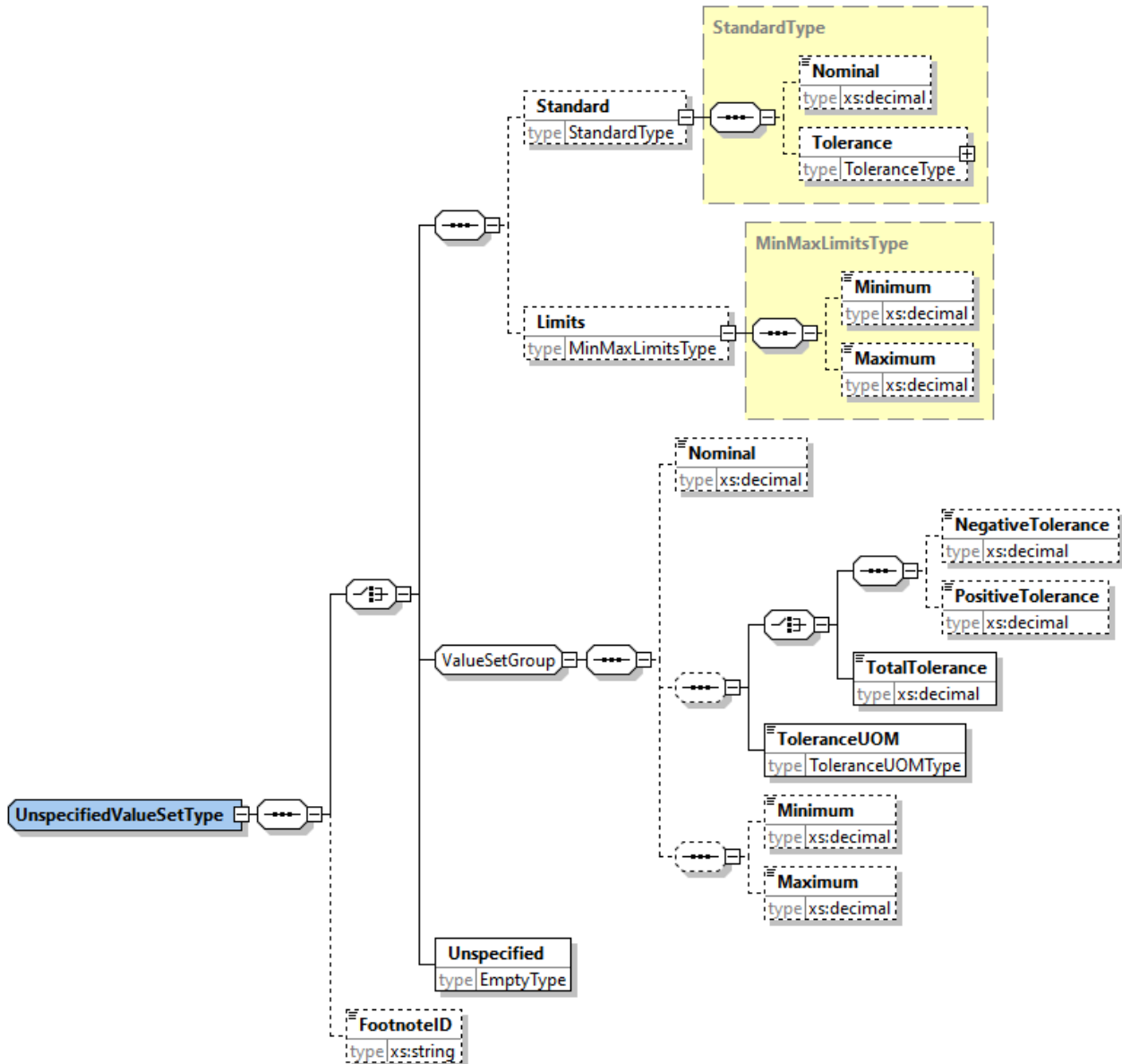
The following set of data can be captured for each dimension via the *ValueSetType*:

- *Nominal*,
- *Negative* and *Positive* Tolerance, or *Total* Tolerance. The *UOM* (*ToleranceUOMType*), defines whether the tolerance value is a percentage of the Nominal or in absolute values, according to the *DimensionUOM* selected.
- *Minimum*,
- *Maximum*.

It is highly recommended to utilize the *ValueSetGroup* as opposed to the upper *Standard* and *Limit* structure, as this reduces the size of the xml file and flattens the hierarchy within the xml file. The existing *Standard* and *Limit* structures are maintained within the schema for backward compatibility.

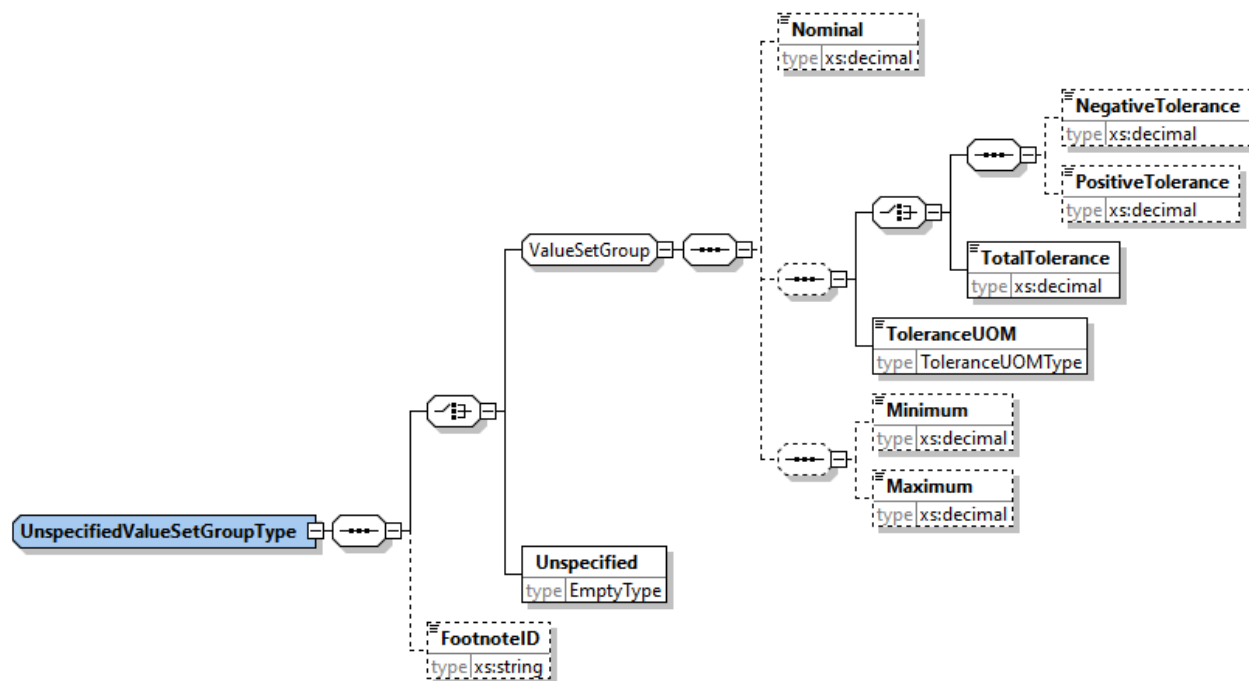


### A.4.3 Unspecified Value Set Type



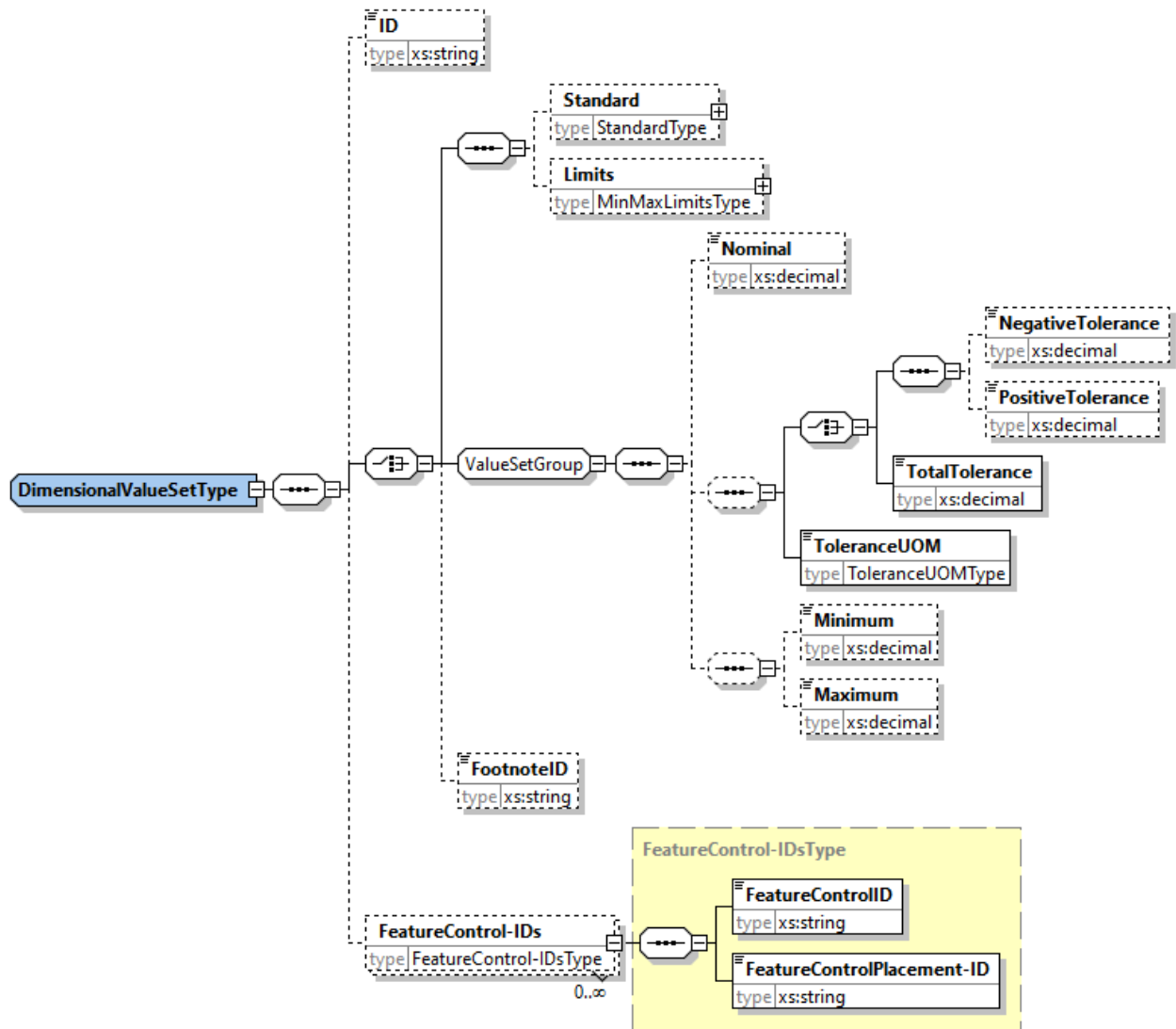
This is the same as the *ValueSetType* with the exception that *Unspecified* can be chosen in the event that that particular dimension is not specified. This is only applied to non-critical dimensions.

#### A.4.4 Unspecified Value Set Group Type



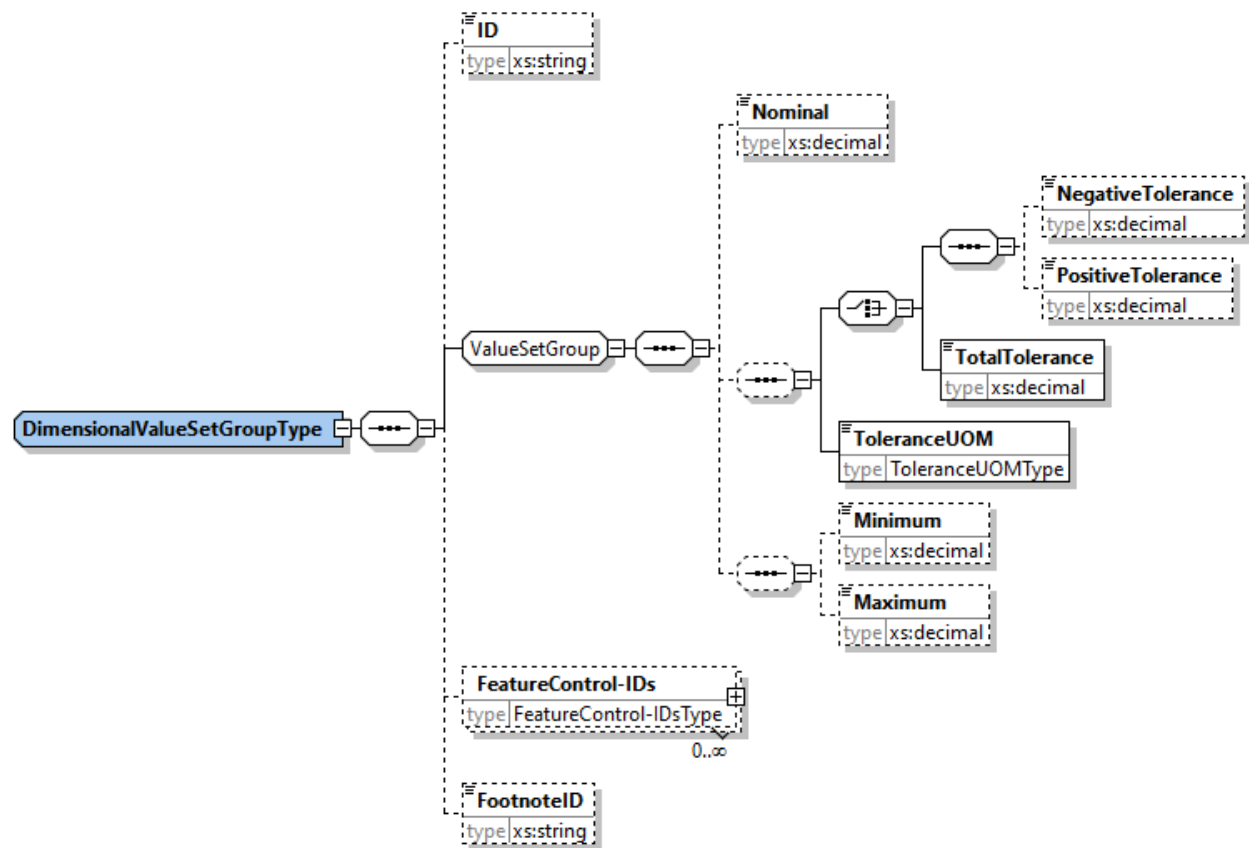
This is the same as the [UnspecifiedValueSetType](#) with the exception that the dimensional structure is limited to the [ValueSetGroup](#).

#### A.4.5 Dimensional Value Set Type



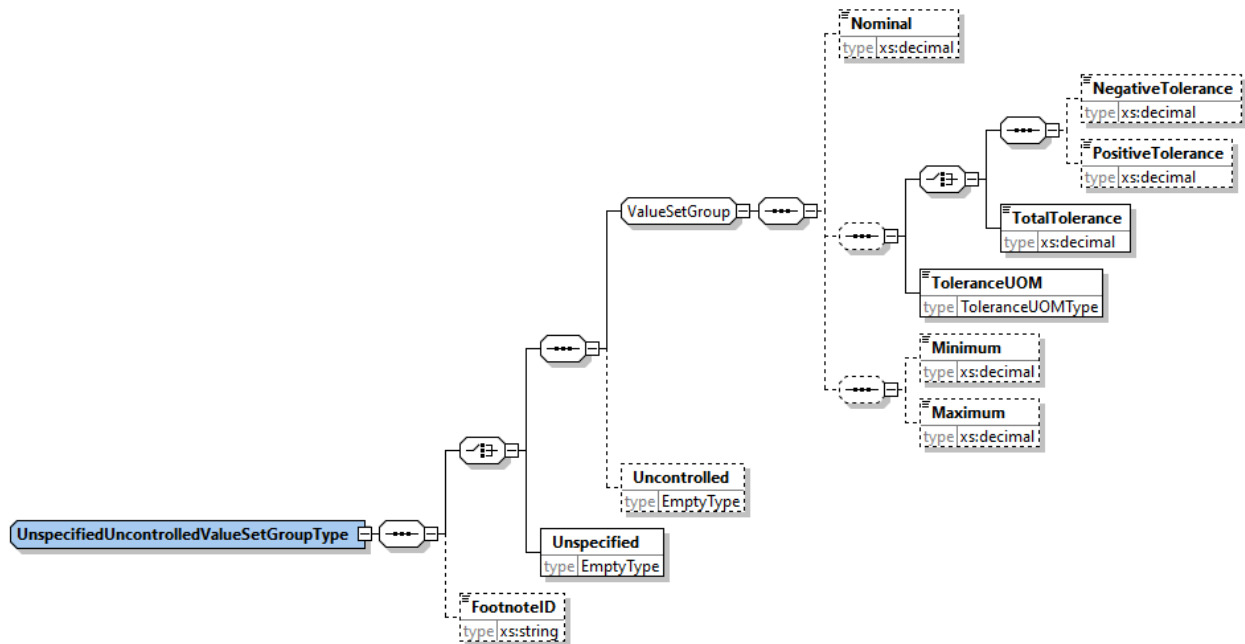
This is the same as the [ValueSetType](#) with the exception that [FeatureControl-IDs](#) is an added structure to support GD&T.

#### A.4.6 Dimensional Value Set Group Type



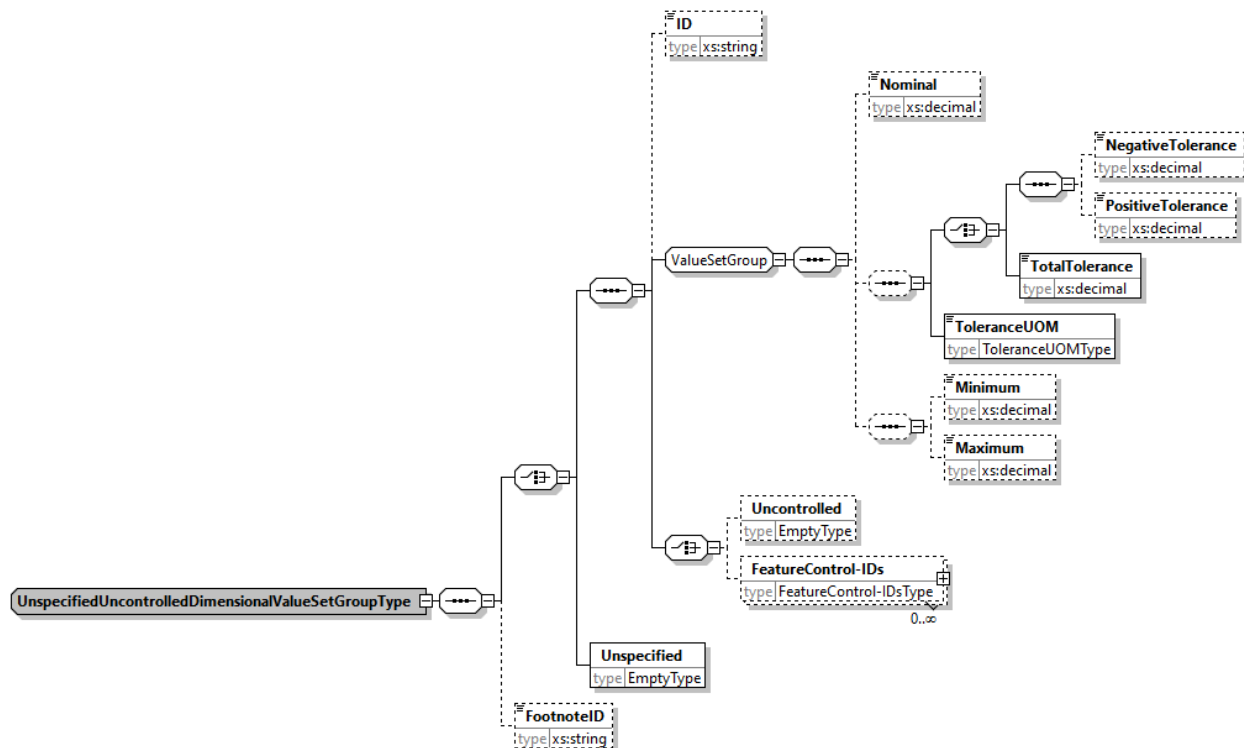
This is the same as the [DimensionalValueSetType](#) with the exception that the dimensional structure is limited to the [ValueSetGroup](#).

#### A.4.7 Unspecified and Uncontrolled Value Set Type



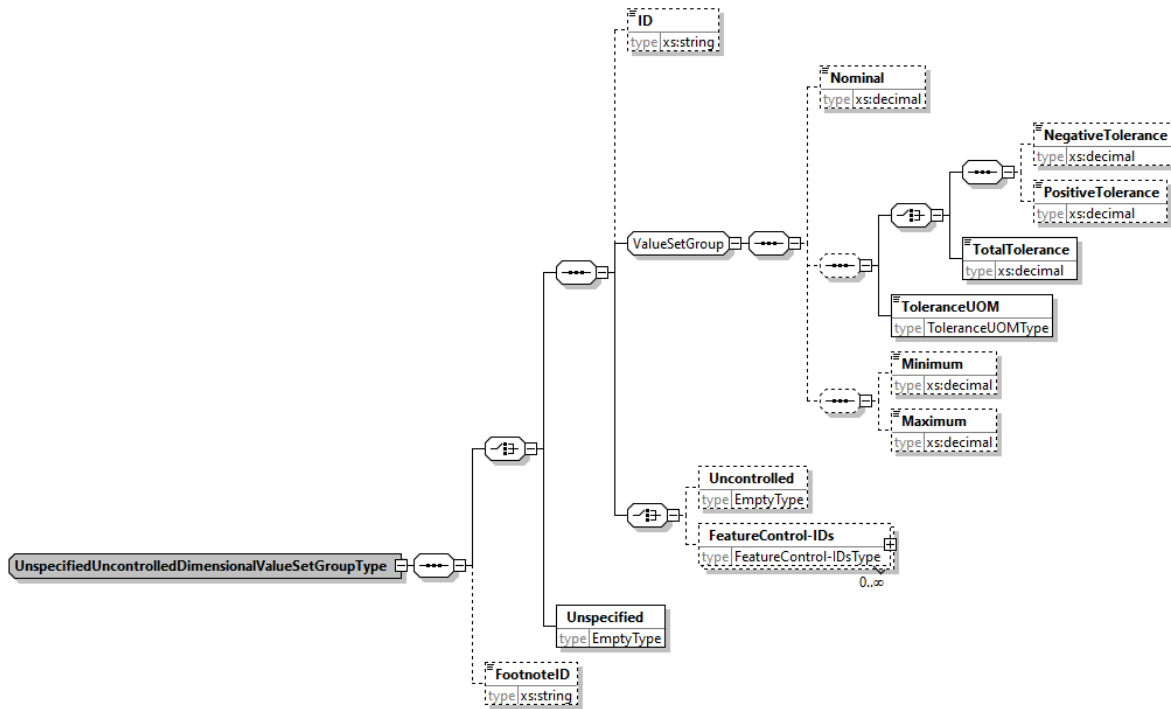
This is the same as the *ValueSetType* with the exception that even when dimensions are provided, that these dimensions are un-controlled. Often, these dimensions are omitted from the drawing, hence *Unspecified* can be chosen in the event that that particular dimension is not specified. This is only applied to non-critical dimensions.

#### A.4.8 Unspecified Dimensional Value Set Type



This is the same as the [DimensionalValueSetType](#) with the exception that *Unspecified* can be chosen in the event that that particular dimension is not specified. This often occurs on terminal shoulders where some situations these critical shoulders definitions require GD&T references, and in other in other situations, the shoulder is not critical and can be left unspecified.

#### A.4.9 Unspecified and Uncontrolled Dimensional Value Set Type



This is the same as the [UnspecifiedDimensionalValueSetType](#) with the exception of the addition of the *Uncontrolled* element as a choice to the *FeatureControl-IDs*. Often, these dimensions are omitted from the drawing, hence *Unspecified* can be chosen in the event that that particular dimension is not specified. This is only applied to non-critical dimensions.

## Annex B (informative) Differences between JEP30-P100 and its predecessors

This table briefly describes most of the changes made to entries that appear in this standard, JEP30-P100, compared to its predecessor; Punctuation changes may or may not be included.

Initial Issue:	Date: May 2018	Item Number: 11.2-938
----------------	----------------	-----------------------

### Change Record History

Issue: A	Date: March 2023	Item Number: 11.2-839S
Section 2.1 Applicable Documents JEDEC – Added in reference to new document SPP-010		
Section 2.1 Applicable Documents ASME – Added in reference to new document ASME Y14.5-2009		
Section 4.4 Package-Array – Updated Package Type to include <i>GDA</i> and <i>T</i> section.		
Updated all diagrams in all sections that had elements with no type to now have an JEP30-D10:EmptyType.		
Section 4.5 Package Terminal Position – Added Internal position to align with JESD30J Also changed MultiPosition to MixedPosition.		
Section 4.6.1 Microelectronics Assembly: Added new extended package outline option for “Chiplets”		
Section 4.6.2 Post Mount: Changed “Sequence” to “Choice”		
Removed Section 4.6.3 <i>Switch</i> , since the extended package outline section for Switch is moved to Part Classification under JEP30-E101		
Section 4.7.13 Through-Hole: Added missing diagram for Through-Hole		
Section 4.14 Keepout Region – Make Component Optional		
Section 4.19 Terminal Function – Added Thermal as an additional function.		
Sections 4.20.2, 4.21.2.2 and 4.22.3.2 Circular Array – Updated <i>RotateWithCircle</i> and <i>RotationAngle</i> types		
Section 4.22.3.3 Pad-or-Hole Status is updated to enable the center and Index to be unbounded.		
Section 4.25 Updated Table 8 - Terminal Group to Package Body Relationship Dimension Variations to replace “Lower” and “Upper” with “Back” and “Front in Y1-Y27 Offsets. Also added Offsets for the Z-direction		
Section 4.26 Added in the <i>Z-Direction</i> to <i>TerminalSpan</i> and <i>TerminalSpacing</i> into <i>TerminalGroupToTerminalGroupRelationships-Array</i>		
Section 4.27 Changed <i>TerminalDetails-Array</i> from a Choice to a Sequence		
Section 4.27.1 Terminal Details - Added " <i>PolarTerminalIndex</i> " to " <i>TerminalDetailsType</i> " and to " <i>FirstTerminalLocationType</i> " in Section 4.27.2.1		
Section 4.27.1.1 – Added new section for Terminal Center to include polar coordinates		
Section 4.27.1.2 – Added " <i>Excluded</i> " to " <i>TerminalStatus</i> "		
Section 4.27.2.1 First Terminal Location		
Section 4.27.2.2 Terminal Numbering Pattern		
Section 4.27.2.3 Added Grid Terminal Numbering Pattern		



## Annex B (cont'd)

Section 4.27.2.4 Terminal Details Exceptions - Added " <i>Excluded</i> " to " <i>TerminalStatus</i> "
Section 4.27.2.4.1 Terminal Index - Added " <i>PolarTerminalIndex</i> " to " <i>AssociatedTerminalStatus</i> "
Section 4.27.2.4.2 Terminal Number – Added " <i>FromNumber</i> " and " <i>ToNumber</i> " to " <i>TerminalNumberType</i> "
Section 4.28 Updated Via Array
Section 4.29 Added a new section for 4.29 Geometric Dimensioning and Tolerancing
Section 4.30 Added a new section for Physical Model
Annex A Section A.1 – Update Contour example to fix one of the coordinates
Revised Table of Contents

Issue: B	Date: Jun 2023	Item Number: 11.2-1032
Section 2.1 Applicable Documents: Update reference to JESD30K		
Section 2.3 JEDEC/IPC: Added new reference to J-Std-609 Standard		
Section 4.4 Package: Added <i>AssemblyTechnologyType</i> , <i>CTE-ArrayType</i> , and <i>YoungsModulus-ArrayType</i> to <i>PackageType</i> .		
Section 4.5 Package Terminal Position: Updated Package Terminal Position / Upper position to be consistent with Bottom position to support Chiplets		
Section 4.5.8 Upper: Added new section for upper terminal position		
Section 4.7.4 Gull-wing: Renamed extended terminal code from Shoulder to Modified to align with JESD30K		
Section 4.11 Package Shape: Add in a new primitive shape <i>RectangleConcave</i>		
Section 4.15.1 Terminal Position: Updated Terminal Group Terminal Position / Upper position to be consistent with Bottom position to support Chiplets		
Added new section 4.16 for Assembly Technology		
Added new section 4.17 for CTE - Array		
Added new section 4.17 for Young Modulus		
Section 4.18.1 Terminal Group – Array: Added Terminal Material and CTE-Array to Terminal Group Type		
Section 4.18.3.1 Terminal Detail: Added ID key under terminal Details array for reference by the Fiducial Array		
Section 4.19 Fiducial: Added new section for Fiducial Marking		
Section 4.20 : Changed GD&T Datum-to-Element Map to unbounded under GD&T Datum-to-Element Map - Array		
Section 4.20.2.1 Recommended Pad Or Hole Shape: Moved this structure from under Terminal Groups to an array under Recommended Footprints		
Section 4.20.8 Keepout Region -Array: Moved this structure from under Package to under Recommended Footprints		

## Annex B (cont'd)

Issue: C	Date: November 2023	Item Number: 11.2-1040
Section 2.1 Applicable Documents: Update reference to JESD30L		
Section 4.5.2 Dual: Added “Bottom-to-Upper” as a new Dual position to support Compressed Mount Technology terminal types.,		
Section 4.5.2.2 Dual Bottom-to-Upper: Added new structure for Bottom-to-Upper position		
Section 4.7 Package Terminal Code: Updated diagram to include “Compressed Mount Technology” as a new Terminal type.		
Section 4.17.1.2 Terminal: Added Compressed Mount Technology as a new terminal type		

Issue: D	Date: February 2024	Item Number: 11.2-1053
Description of Change		
Section 4.3 Linking the MPN to a specific Package Family Data set: Update section to include reference from Part Number to Die		
Section 4.3.2 Linking the Manufacturing Part Number to Physical Model Content: Corrected the PhysicalModelKey assignment to PhysicalModel/ID,		
Added new Section 4.3.3 “Linking the Manufacturing Part Number to Die Content”		
Section 5 Package Section - Package: Updated diagram to include the addition of Die-Array		
Updated images in sections 5.6, 5.8, and A.4.2 thru A.4.7 to include ValueSetGroup		
Updated image in section 5.7.1 to include Outline-Array		
Section 5.13.3 Terminal Detail – Array: “Terminal Number Pattern” is made unbounded		
Section 5.13.3.2 Terminal Number Pattern: Added Prefix and Suffix under Terminal Number Pattern		
Section 5.13.3.2.2 Sequential: Added Prefix, Start and Suffix under both Numerical and Alphabetical Sequence.		
Section 5.13.3.2.3 Grid: Updated image.		
Section 5.13.3.2.1 First Terminal Location: Updated elements in “Location Relative-to-Package Center” type from optional to mandatory under choice.		
Section 5.13.3.2.2 Sequential: Updated Numerical and Alphabetical Sequence		
Added new Section 6 “Package Section – Die-Array”		

## Annex B (cont'd)

Issue: E	Date: August 2024	Item Number: 11.2-1059
Description of Change		
Section 4.1, and section 4.2: Update sections to align with modifications performed at the JEP30 parent structure		
Section 5 Package Section – Package: Updated name of the JESD30 document. Added Package Status. Changed xs:integer with Min of 1 to		
Changed xs:integer with Min of 1 to “Min Integer Of One Type” throughout the document		
Section 5.13.1 Terminal Group – Array: Make Terminal Location unbounded. Added Pattern Group section.		
Section 5.13.1.5 Terminal Location: Updated Terminal location to make it unbounded.		
Section 5.13.1.5.1 Standard Array: Make Angle optional		
Section 5.13.1.6: Added new section for Pattern Groups		
Section 5.13.1.7.22.3: Updated image for Void Status		
Section 5.13.3.2.1: Update image to include choice of Terminal Pattern ID or Pattern Group ID.		
Section 5.13.3.2.4.1: Update Image to include Terminal Location ID, and Pattern Group ID		
Section 5.13.3.2.4.2: Updated image to make various branches unbounded		
Section 5.16 Recommended Footprint – Array: Updated section with Footprint Name, Footprint Extended Name, Polarity. Made Soldermask and Pastemask Layer arrays optional.		
Section 5.16.2 Recommended Pad Or Hole Shape – Array: Added new element for Pad Group To Pad Group Relationship		
Section 5.16.2.1 Recommended Pad Or Hole Shape: Make Location unbounded. Added Pattern Group.		
Section 5.16.2.1.1.1.1.1 Standard Array: Make dx, dy, Angle and VoidGroupLowerLeftVoidCenter optional.		
Section 5.16.2.1.3 Location: Updated Location image to make it unbounded.		
Section 5.16.2.1.3.1 Standard Array: Updated image to make dx, dy, Angle and LandGroupLowerLeftPad-or-HoleCenter optional		
Section 5.16.2.1.4: Added new section for Pattern Groups		
Section 5.16.2.1.5: Added new section for Land Pattern Span		
Section 5.16.2.1.6: Added new section for Land Pattern Spacing		
Section 5.16.2.2: Added new section for Pad Group To Pad Group Relationship		
Section 5.16.7.1 Restrictive layer: Update element name to InnerLayer		
Section 5.16.9 Soldermask Layer – Array: Make Soldermask Layer – Array optional		
Section 5.16.10 Pastemask Layer – Array: Revised section to align with previous sections		
Section 6 Package Section – Die-Array: Added Process Technology		

## Annex B (cont'd)

Issue: F	Date: November 2024	Item Number: 11.2-1070
Description of Change		
Section 5.3.8 Surface Terminal: Added choice for Horizontal versus Vertical Castellation		
Section 5.7 Package Shape: Add recommendation of how to represent the package outline shape when the shape is a modified rectangle and when the corner dimensions are unspecified.		
Section 5.13.1 Terminal Group – Array: Change schema type from “All” to “Sequence”.		
Section 5.13.1.7 Terminal Shape: <ol style="list-style-type: none"> <li>1. Update diagram 1 of 9 to create structure to capture the dimensions for the Pad-on-Package, Soldermask Opening-on-Package, Ball and Column. Added diagrams 1.1 to 1.4 that defines each structure</li> <li>2. Update diagram 3 of 9 to change Rise Angle and Seating Angle from xs:integer to JEP30-D10:UnspecifiedValueSetGroupType and all respective tables.</li> <li>3. Update diagram 9 of 9 to insert Castellation Array</li> </ol>		
Section 5.13.1.7.1 Ball Types: Update Ball Dimension Table to add Regular Polygon, and added notes to highlight the preferred process of gathering the relevant information for this terminal.		
Section 5.13.1.7.9 Column Types: Update Column Dimension Table to add UpperSide-of-Column and LowerSide-of-Column structures. Also added notes to highlight the preferred process of gathering the relevant information for this terminal.		
Section 5.13.1.7.22 Castellation – Array: Added new section to define the vertical and horizontal castellations.		
Section 5.13.1.7.23 Terminal Void – Array: Added Cavity Depth for embedded cavity within void.		
Section 5.13.3 Terminal Detail Array: Relocate First terminal location and Terminal Detail Exceptions directly under Terminal Detail Array.		
Section 5.13.3.1.2 Terminal Status: Extend “Associated” to have a choice of “Terminal Pattern ID”, “Pattern Group ID”, or “Terminal Group ID” for the association to apply to.		
Section 5.13.3.3 Terminal Number Pattern: Update image based on changes to section 5.13.3, and extend to reference a choice of “Terminal Pattern ID”, “Pattern Group ID”, or “Terminal Group ID”.		
Section 5.13.3.3.1 Sequential: Add a choice under Row and Column for both “Numerical Sequence” and “Alphabetical Sequence” between included and excluded Numbers/Characters.		
Section 5.13.3.3.2 Grid: Updated image to correct Numerical Sequence Type and Alphabetical Sequence Type.		
Section 5.16.10.1 Aperture Shape: Added Segmented Ring to Aperture Shape.		
Section 6.5.7 Terminal Detail Array: Relocate First terminal location and Terminal Detail Exceptions directly under Terminal Detail Array.		
Section 6.5.7.1 Terminal Detail, Section 6.5.7.2 First Terminal Location and Section 6.5.7.4 Terminal Detail Exceptions: Extend structures to have a choice of “Terminal Pattern ID”, “Pattern Group ID”, “Terminal Group ID”, RegionID or Active Zone.		
Section 6.5.7.3 Terminal Number Pattern: Update image based on changes to section 6.5.7, and extend to reference a choice of “Terminal Pattern ID”, “Pattern Group ID”, “Terminal Group ID”, RegionID or Active Zone.		

## Annex B (cont'd)

Section A.1.16 Segmented Ring: Added new shape called Segmented Ring to support Pastemask Apertures
Section A.2.9 Surface Terminal: Enhanced section for the vertical and horizontal castellations

Issue: G	Date: February 2025	Item Number: 11.2-1073
Description of Change		
Change xs:all to xs:sequence throughout schema to facilitate lower level extensions with Choices and to simplify file comparisons.		
Section 5 Package Section: Add Attributes to Package Section in diagram 1 of 4. Update diagram part 4 of 4 to move CTE Array and Youngs Modulus Array to JEP30-D10 dictionary, and to add Heat Capacity, and Footnote Array.		
Section 5.3.10: Pin: Updated image to include Surface-mount as a new type of pin		
Sections A.4.X – All ValueSetType(s): Updated images to include Footnote, Added the word Tolerance to Negative, Positive, Total and UOM under the ValueSetGroup		
Section 5.7.1 Contour: Update image to make the contour tolerance type align with the value set group. Add Arc Segment Rotation and Footnote ID.		
Section 5.8 Part Height: Update image for Value Set Type		
Section 5.9 Clearance Region – Array: Update image to move Unspecified in under Standoff via Unspecified Value Set Type. Add Footnote ID to Standoff.		
Section 5.11 CTE - Array: Update entire section to align the CTE structure with that used in the Material Design Kit under JEP30-K101.		
Section 5.12 Young Modulus – Array: Update entire section to align the Young Modulus structure with that used in the Material Design Kit under JEP30-K101.		
Section 5.14.1 Terminal Group – Array: Update image to move CTE Array to Dictionary.		
Section 5.14.1.7 Terminal Shape: Update image part 2, 5 and 8 of 9, to add in PM code to the relevant terminal shape dimensions.		
Section 5.14.1.7.5.2.1 Reference Terminal Contour Type: Update Contour structure to align with other Contour structures. Add Footnotes to structure.		
Section 5.14.1.7.11 Pin: Update table 14 to support new Surface-mount Pin.		
Section 5.15: Add new section for Tiebar - Array		
Section 5.18.10.1.2.2 Duplicate: Added Footnote ID to the duplicate structure for aperture shapes.		
Section 6 Package Section – Die-Array: Update diagram part 3 of 3 to add Footnote Array		
Section A.1.1 Rectangle: Update image to change xml type from All to Sequence		
Section A.1.23 Contour: Updated Contour structure to capture the Arc direction and corrected the sample image with the contour points		
Section A.2.4 Flat Terminal: Add Flat Terminals can also have rectangular Dimples via horizontal castellations.		
Section A.2.9.2 Horizontal Castellations: Update image to add Rectangular Dimples		
Section A.2.11 Pin: Update section to add Terminal End Shape		

**Annex B (cont'd)**

Section A.4.1 Value Type: Do we need to add Footnotes & Rules to structure
Section A.4.2 Value Set Type: Update to add Footnotes. Also update the images in section 5.6, A.4.3, A.4.4, A.4.5, A.4.6, A.4.7.
Section A.4.4: Added Unspecified Value Set Group Type.
Section A.4.6: Added Dimensional Value Set Group Type



---

**Standard Improvement Form****JEDEC Standard No. JEP30-P100G**

---

The purpose of this form is to provide the Technical Committees of JEDEC with input from the industry regarding usage of the subject standard. Individuals or companies are invited to submit comments to JEDEC. All comments will be collected and dispersed to the appropriate committee(s).

If you can provide input, please complete this form and return to:

JEDEC  
Attn: Publications Department  
3103 10th Street North  
Suite 240S  
Arlington, VA 22201

Email: [angies@jedec.org](mailto:angies@jedec.org)

---

1. I recommend changes to the following:

☐ Requirement, clause number \_\_\_\_\_

☐ Test method number \_\_\_\_\_ Clause number \_\_\_\_\_

The referenced clause number has proven to be:

☐ Unclear ☐ Too Rigid ☐ In Error

☐ Other \_\_\_\_\_

---

2. Recommendations for correction:


---

3. Other suggestions for document improvement:


---

Submitted by

Name: \_\_\_\_\_

Phone: \_\_\_\_\_

Company: \_\_\_\_\_

E-mail: \_\_\_\_\_

Address: \_\_\_\_\_

City/State/Zip: \_\_\_\_\_

Date \_\_\_\_\_

---

