

PUBLICLY
AVAILABLE
SPECIFICATION

ISO/PAS
14306

First edition
2011-12-15

**Industrial automation systems and
integration — JT file format specification
for 3D visualization**

*Systèmes d'automatisation industrielle et intégration — Spécification de
format de fichier JT pour visualisation 3D*

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/PAS 14306:2011](https://standards.iteh.ai/catalog/standards/sist/7ddef245-fb1-477d-ad1b-765866d37fb/iso-pas-14306-2011)

<https://standards.iteh.ai/catalog/standards/sist/7ddef245-fb1-477d-ad1b-765866d37fb/iso-pas-14306-2011>



Reference number
ISO/PAS 14306:2011(E)

© ISO 2011

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/PAS 14306:2011

<https://standards.iteh.ai/catalog/standards/sist/7ddef245-fb1-477d-ad1b-765866d37fb/iso-pas-14306-2011>



COPYRIGHT PROTECTED DOCUMENT

© ISO 2011

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword	x
Introduction.....	xi
1 Scope	1
2 References and Additional Information.....	2
3 Definitions	3
3.1 Terms	3
3.2 Coordinate Systems.....	5
4 Acronyms and Abbreviations.....	5
5 Notational Conventions	6
5.1 Diagrams and Field Descriptions	6
5.2 Data Types	9
6 File Format.....	11
6.1 File Structure	12
6.1.1 File Header	12
6.1.2 TOC Segment.....	13
6.1.3 Data Segment.....	14
6.1.3.1 Segment Header.....	15
6.1.3.2 Data 16	15
6.2 Data Segments.....	19
6.2.1 LSG Segment	19
6.2.1.1 Graph Elements.....	19
6.2.1.1.1 Node Elements	20
6.2.1.1.1.1 Base Node Element	20
6.2.1.1.1.2 Partition Node Element.....	21
6.2.1.1.1.3 Group Node Element.....	24
6.2.1.1.1.4 Instance Node Element.....	25
6.2.1.1.1.5 Part Node Element.....	25
6.2.1.1.1.6 Meta Data Node Element	26
6.2.1.1.1.7 LOD Node Element.....	26
6.2.1.1.1.8 Range LOD Node Element.....	27
6.2.1.1.1.9 Switch Node Element.....	28
6.2.1.1.1.10 Shape Node Elements.....	29
6.2.1.1.2 Attribute Elements	39
6.2.1.1.2.1 Base Attribute Element.....	39
6.2.1.1.2.2 Material Attribute Element.....	41
6.2.1.1.2.3 Texture Image Attribute Element.....	44
6.2.1.1.2.4 Draw Style Attribute Element	58
6.2.1.1.2.5 Light Set Attribute Element	60
6.2.1.1.2.6 Infinite Light Attribute Element.....	61
6.2.1.1.2.7 Point Light Attribute Element.....	63
6.2.1.1.2.8 Linestyle Attribute Element.....	66
6.2.1.1.2.9 Pointstyle Attribute Element	67
6.2.1.1.2.10 Geometric Transform Attribute Element.....	68
6.2.1.1.2.11 Shader Effects Attribute Element	69
6.2.1.1.2.12 Vertex Shader Attribute Element	71
6.2.1.1.2.13 Fragment Shader Attribute Element.....	76
6.2.1.2 Property Atom Elements	77
6.2.1.2.1 Base Property Atom Element	77
6.2.1.2.2 String Property Atom Element	78

6.2.1.2.3	Integer Property Atom Element	78
6.2.1.2.4	Floating Point Property Atom Element	79
6.2.1.2.5	JT Object Reference Property Atom Element	79
6.2.1.2.6	Date Property Atom Element	80
6.2.1.2.7	Late Loaded Property Atom Element	81
6.2.1.3	Property Table	82
6.2.1.3.1	Node Property Table	82
6.2.2	Shape LOD Segment	83
6.2.2.1	Shape LOD Element	83
6.2.2.1.1	Vertex Shape LOD Element	84
6.2.2.1.2	Tri-Strip Set Shape LOD Element	85
6.2.2.1.3	Polyline Set Shape LOD Element	86
6.2.2.1.4	Point Set Shape LOD Element	86
6.2.2.1.5	Polygon Set Shape LOD Element	87
6.2.2.1.6	Null Shape LOD Element	88
6.2.2.2	Primitive Set Shape Element	89
6.2.2.3	Wire Harness Set Shape Element	95
6.2.3	JT B-Rep Segment	108
6.2.3.1	JT B-Rep Element	109
6.2.4	XT B-Rep Segment	131
6.2.4.1	XT B-Rep Element	131
6.2.5	Wireframe Segment	132
6.2.5.1	Wireframe Rep Element	132
6.2.6	Meta Data Segment	134
6.2.6.1	Property Proxy Meta Data Element	135
6.2.6.2	PMI Manager Meta Data Element	137
6.2.6.2.1	PMI Entities	139
6.2.6.2.1.1	PMI Dimension Entities	139
6.2.6.2.1.2	PMI Note Entities	147
6.2.6.2.1.3	PMI Datum Feature Symbol Entities	148
6.2.6.2.1.4	PMI Datum Target Entities	148
6.2.6.2.1.5	PMI Feature Control Frame Entities	148
6.2.6.2.1.6	PMI Line Weld Entities	149
6.2.6.2.1.7	PMI Spot Weld Entities	149
6.2.6.2.1.8	PMI Surface Finish Entities	151
6.2.6.2.1.9	PMI Measurement Point Entities	152
6.2.6.2.1.10	PMI Locator Entities	153
6.2.6.2.1.11	PMI Reference Geometry Entities	154
6.2.6.2.1.12	PMI Design Group Entities	154
6.2.6.2.1.13	PMI Coordinate System Entities	157
6.2.6.2.2	PMI Associations	157
6.2.6.2.3	PMI User Attributes	160
6.2.6.2.4	PMI String Table	160
6.2.6.2.5	PMI Model Views	161
6.2.6.2.6	Generic PMI Entities	163
6.2.6.2.7	PMI CAD Tag Data	168
6.2.7	PMI Data Segment	168
7	Data Compression and Encoding	168
7.1	Common Compression Data Collection Formats	169
7.1.1	Int32 Compressed Data Packet	169
7.1.2	Float64 Compressed Data Packet	173
7.1.3	Vertex Based Shape Compressed Rep Data	176
7.1.4	Point Quantizer Data	184
7.1.5	Texture Quantizer Data	185
7.1.6	Color Quantizer Data	185
7.1.7	Uniform Quantizer Data	186
7.1.8	Compressed Entity List for Non-Trivial Knot Vector	187
7.1.9	Compressed Control Point Weights Data	189
7.1.10	Compressed Curve Data	189

ITIS STANDARD PREVIEW
 (standards.itech.ai)
 ISO/PAS 14306:2011
<https://standards.itech.ai/standards/sist/7ddef245-fb1-477d-ad1b-765866d37fb/iso-pas-14306-2011>

7.1.11	Compressed CAD Tag Data.....	192
7.2	Encoding Algorithms	194
7.2.1	Uniform Data Quantization	194
7.2.2	Bitlength CODEC	195
7.2.3	Huffman CODEC	196
7.2.4	Arithmetic CODEC.....	198
7.2.5	Deering Normal CODEC.....	202
7.3	ZLIB Compression.....	203
8	Usage Guide	204
8.1	Late-Loading Data	204
8.2	Bit Fields	204
8.3	Reserved Field	204
8.4	Metadata Conventions	204
8.4.1	CAD Properties	205
8.4.2	Tessellation Properties	206
8.4.3	Miscellaneous Properties	207
8.5	LSG Attribute Accumulation Semantics	208
8.6	LSG Part Structure	208
8.7	Range LOD Node Alternative Rep Selection	209
Annex A:	Object Type Identifiers	210
Annex B:	Semantic Value Class Shader Parameter Values	212
Annex C:	Decoding Algorithms – An Implementation.....	216
Annex D:	Parasolid XT Format Reference	236
	Introduction to the Parasolid XT Format.....	241
	Types of File Documented.....	241
	Text and Binary Formats	242
	Logical Layout	243
	Schema	244
	Physical Layout	250
	Model Structure	255
	Schema Definition	261
	Node Types	329
	Node Classes	332
	System Attribute Definitions	333

Tables

Table 1: Basic Data Types.....9
 Table 2: Composite Data Types10
 Table 3: Segment Types.....15
 Table 4: Object Base Types17
 Table 5: Primitive Set Primitive Data Elements91
 Table 6: Primitive Set “params#” Data Fields Interpretation.....91
 Table 7: Common Property Keys and Their Value Encoding formats 166
 Table 8: CAD Property Conventions.....205
 Table 9: CAD Optional Property Units206
 Table 10: Object Type Identifiers.....211
 Table 11: Semantic Value Class Shader Parameter Values212

Figures

Figure 1: File Structure12
 Figure 2: File Header data collection12
 Figure 3: TOC Segment data collection.....14
 Figure 4: TOC Entry data collection.....14
 Figure 5: Data Segment data collection.....15
 Figure 6: Segment Header data collection15
 Figure 7: Data data collection17
 Figure 8: Element Header data collection17
 Figure 9: Element Header ZLIB data collection18
 Figure 10: LSG Segment data collection19
 Figure 11: Base Node Element data collection20
 Figure 12: Base Node Data data collection20
 Figure 13: Partition Node Element data collection22
 Figure 14: Vertex Count Range data collection23
 Figure 15: Group Node Element data collection24
 Figure 16: Group Node Data data collection24
 Figure 17: Instance Node Element data collection25
 Figure 18: Part Node Element data collection25
 Figure 19: Meta Data Node Element data collection26
 Figure 20: Meta Data Node Data data collection26
 Figure 21: LOD Node Element data collection27
 Figure 22: LOD Node Data data collection27
 Figure 23: Range LOD Node Element data collection28
 Figure 24: Switch Node Element data collection29
 Figure 25: Base Shape Node Element data collection30
 Figure 26: Base Shape Data data collection30
 Figure 27: Vertex Count Range data collection31
 Figure 28: Vertex Shape Node Element data collection32
 Figure 29: Vertex Shape Data data collection33
 Figure 30: Quantization Parameters data collection34
 Figure 31: Tri-Strip Set Shape Node Element data collection34
 Figure 32: Polyline Set Shape Node Element data collection35
 Figure 33: Point Set Shape Node Element data collection35
 Figure 34: Polygon Set Shape Node Element data collection36
 Figure 35: NULL Shape Node Element data collection36
 Figure 36: Primitive Set Shape Node Element data collection37
 Figure 37: Primitive Set Quantization Parameters data collection38
 Figure 38: Wire Harness Set Shape Node Element data collection39
 Figure 39: Base Attribute Element data collection40
 Figure 40: Base Attribute Data data collection40
 Figure 41: Material Attribute Element data collection42
 Figure 42: Texture Image Attribute Element data collection45
 Figure 43: Texture Vers-1 Data data collection46
 Figure 44: Vers-1 Image Format Description data collection47

Figure 45: Vers-1 Texture Environment data collection	49
Figure 46: Texture Vers-2 Data data collection	51
Figure 47: Vers-2 Texture Environment data collection	53
Figure 48: Texture Coord Generation Parameters data collection	55
Figure 49: Inline Texture Image Data data collection	56
Figure 50: Vers-2 Image Format Description data collection	57
Figure 51: Draw Style Attribute Element data collection	59
Figure 52: Light Set Attribute Element data collection	61
Figure 53: Infinite Light Attribute Element data collection	62
Figure 54: Base Light Data data collection	63
Figure 55: Point Light Attribute Element data collection	64
Figure 56: Spread Angle value with respect to the light cone	65
Figure 57: Attenuation Coefficients data collection	65
Figure 58: Linestyle Attribute Element data collection	66
Figure 59: Pointstyle Attribute Element data collection	67
Figure 60: Geometric Transform Attribute Element data collection	68
Figure 61: Shader Effects Attribute Element data collection	70
Figure 62: Vertex Shader Attribute Element data collection	72
Figure 63: Base Shader Data data collection	72
Figure 64: Shader Parameter data collection	74
Figure 65: Fragment Shader Attribute Element data collection	77
Figure 66: Base Property Atom Element data collection	77
Figure 67: Base Property Atom Data data collection	78
Figure 68: String Property Atom Element data collection	78
Figure 69: Integer Property Atom Element data collection	79
Figure 70: Floating Point Property Atom Element data collection	79
Figure 71: JT Object Reference Property Atom Element data collection	80
Figure 72: Date Property Atom Element data collection	80
Figure 73: Late Loaded Property Atom Element data collection	81
Figure 74: Property Table data collection	82
Figure 75: Node Property Table data collection	83
Figure 76: Shape LOD Segment data collection	83
Figure 77: Vertex Shape LOD Element data collection	84
Figure 78: Vertex Shape LOD Data data collection	84
Figure 79: Tri-Strip Set Shape LOD Element data collection	85
Figure 80: Polyline Set Shape LOD Element data collection	86
Figure 81: Point Set Shape LOD Element data collection	87
Figure 82: Polygon Set Shape LOD Element data collection	88
Figure 83: Null Shape LOD Element data collection	88
Figure 84: Primitive Set Shape Element data collection	89
Figure 85: Lossless Compressed Primitive Set Data data collection	90
Figure 86: Lossy Quantized Primitive Set Data data collection	92
Figure 87: Compressed params1 data collection	94
Figure 88: Wire Harness Set Shape Element data collection	96
Figure 89: Wire Harness Set data collection	97
Figure 90: Entity Counts data collection	98
Figure 91: Topological Entities data collection	99
Figure 92: Harness data collection	100
Figure 93: Bundle data collection	101
Figure 94: Wire data collection	103
Figure 95: Wire Segment data collection	104
Figure 96: Branch Node data collection	105
Figure 97: Geometric data collection	105
Figure 98: Bundle Spine Curve data collection	106
Figure 99: NURBS XYZ Curve data collection	107
Figure 100: Entity Tag Counters data collection	108
Figure 101: JT B-Rep Segment data collection	109
Figure 102: JT B-Rep Element data collection	110
Figure 103: Topological Entity Counts data collection	111
Figure 104: Geometric Entity Counts data collection	112

Figure 105: Topology Data data collection	113
Figure 106: Regions Topology Data data collection.....	114
Figure 107: Shells Topology Data data collection	114
Figure 108: Trim Loop example in parameter Space - One Face with 2 Holes	116
Figure 109: Faces Topology Data data collection	116
Figure 110: Loops Topology Data data collection	118
Figure 111: CoEdges Topology Data data collection	119
Figure 112: Edges Topology Data data collection.....	120
Figure 113: Vertices Topology Data data collection.....	120
Figure 114: Geometric Data data collection	121
Figure 115: Surfaces Geometric Data data collection.....	122
Figure 116: Non-Trivial Knot Vector NURBS Surface Indices data collection.....	123
Figure 117: NURBS Surface Degree data collection.....	123
Figure 118: NURBS Surface Control Point Counts data collection	124
Figure 119: NURBS Surface Control Point Weights data collection	124
Figure 120: NURBS Surface Control Points data collection.....	125
Figure 121: NURBS Surface Knot Vectors data collection.....	125
Figure 122: PCS Curves Geometric Data data collection	126
Figure 123: Trivial PCS Curves data collection	127
Figure 124: MCS Curves Geometric Data data collection.....	129
Figure 125: Point Geometric Data data collection	129
Figure 126: Topological Entity Tag Counters data collection	130
Figure 127: B-Rep CAD Tag Data data collection.....	131
Figure 128: XT B-Rep Element data collection	131
Figure 129: Wireframe Segment data collection	132
Figure 130: Wireframe Rep Element data collection.....	133
Figure 131: Wireframe MCS Curves Geometric Data data collection.....	134
Figure 132: Meta Data Segment data collection.....	134
Figure 133: Property Proxy Meta Data Element data collection.....	135
Figure 134: Date Property Value data collection	136
Figure 135: PMI Manager Meta Data Element data collection.....	138
Figure 136: PMI Entities data collection.....	139
Figure 137: PMI Dimension Entities data collection.....	140
Figure 138: PMI 2D Data data collection.....	140
Figure 139: PMI Base Data data collection	141
Figure 140: 2D-Reference Frame data collection.....	142
Figure 141: 2D Text Data data collection	142
Figure 142: Text Box data collection	143
Figure 143: Constructing Text Polylines from data arrays.....	144
Figure 144: Text Polyline Data data collection	145
Figure 145: Constructing Non-Text Polylines from packed 2D data arrays	146
Figure 146: Non-Text Polyline Data data collection	146
Figure 147: PMI Note Entities data collection.....	147
Figure 148: PMI Datum Feature Symbol Entities data collection	148
Figure 149: PMI Datum Target Entities data collection	148
Figure 150: PMI Feature Control Frame Entities data collection.....	149
Figure 151: PMI Line Weld Entities data collection	149
Figure 152: PMI Spot Weld Entities data collection.....	150
Figure 153: PMI 3D Data data collection	151
Figure 154: PMI Surface Finish Entities data collection	152
Figure 155: PMI Measurement Point Entities data collection	153
Figure 156: PMI Locator Entities data collection	154
Figure 157: PMI Reference Geometry Entities data collection.....	154
Figure 158: PMI Design Group Entities data collection	155
Figure 159: Design Group Attribute data collection.....	156
Figure 160: PMI Coordinate System Entities data collection.....	157
Figure 161: PMI Associations data collection	158
Figure 162: PMI User Attributes data collection	160
Figure 163: PMI String Table data collection.....	161
Figure 164: PMI Model Views data collection.....	162

ITC STANDARD PREVIEW
 (standards.iteh.ai)
 ISO/PAS:14306:2011
<https://standards.iteh.ai/catalog/standards/sist/7ddef245-fb1-477d-ad1b-7c5f66d37fb/iso-pas-14306-2011>

Figure 165: Generic PMI Entities data collection	164
Figure 166: PMI Property data collection	166
Figure 167: PMI Property Atom data collection	167
Figure 168: PMI CAD Tag Data data collection	168
Figure 169: Int32 Compressed Data Packet data collection	170
Figure 170: Int32 Probability Contexts data collection	171
Figure 171: Int32 Probability Context Table Entry data collection	172
Figure 172: Float64 Compressed Data Packet data collection	174
Figure 173: Float64 Probability Contexts data collection	175
Figure 174: Float64 Probability Context Table Entry data collection	176
Figure 175: Vertex Based Shape Compressed Rep Data data collection	177
Figure 176: Lossless Compressed Raw Vertex Data data collection	178
Figure 177: Lossy Quantized Raw Vertex Data data collection	179
Figure 178: Quantized Vertex Coord Array data collection	180
Figure 179: Quantized Vertex Normal Array data collection	181
Figure 180: Quantized Vertex Texture Coord Array data collection	182
Figure 181: Quantized Vertex Color Array data collection	183
Figure 182: Point Quantizer Data data collection	184
Figure 183: Texture Quantizer Data data collection	185
Figure 184: Color Quantizer Data data collection	186
Figure 185: Uniform Quantizer Data data collection	187
Figure 186: Compressed Entity List for Non-Trivial Knot Vector data collection	188
Figure 187: Compressed Control Point Weights Data data collection	189
Figure 188: Compressed Curve Data data collection	190
Figure 189: Non-Trivial Knot Vector NURBS Curve Indices data collection	192
Figure 190: NURBS Curve Control Point Weights data collection	192
Figure 191: NURBS Curve Control Points data collection	192
Figure 192: Compressed CAD Tag Data data collection	193
Figure 193: Compressed CAD Tag Type-2 Data data collection	194
Figure 194: Huffman Tree	197
Figure 195: Sphere divided into eight octants and octant divided into six sextants with each sextant assigned an identifying three bit code.	203
Figure 196: JT Format Convention for Modeling each Part in LSG	208

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

In other circumstances, particularly when there is an urgent market requirement for such documents, a technical committee may decide to publish other types of document:

- an ISO Publicly Available Specification (ISO/PAS) represents an agreement between technical experts in an ISO working group and is accepted for publication if it is approved by more than 50 % of the members of the parent committee casting a vote;
- an ISO Technical Specification (ISO/TS) represents an agreement between the members of a technical committee and is accepted for publication if it is approved by 2/3 of the members of the committee casting a vote.

<https://standards.itec.ai/catalog/standards/sist/7ddc245-fb1-477d-ad1b-765865df378f/iso-pas-14306-2011>

An ISO/PAS or ISO/TS is reviewed after three years in order to decide whether it will be confirmed for a further three years, revised to become an International Standard, or withdrawn. If the ISO/PAS or ISO/TS is confirmed, it is reviewed again after a further three years, at which time it must either be transformed into an International Standard or be withdrawn.

ISO/PAS 14306 was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

ISO/PAS 14306 is based on Siemens JT File Format Reference Version 8.1 Rev-C.

Introduction

This Publicly Available Specification was transposed by an ad hoc committee focused on industrial requirements for 3D product data visualization under the ISO/TC 184/SC 4 Harvesting Process, as defined in SC4 Standing Document (SC4N1198), Procedures for Transposing Externally Developed Specifications into ISO Deliverables.

The ad hoc committee was formed by members of ISO/TC 184/SC 4 in response to requests from the global industrial community for information on visualization formats. The group assessed several 3D visualization formats including COLLADA, JT, U3D and X3D against a list of 36 requirements. The final results concluded that these candidate formats are complementary to the ISO 10303 "STEP" series of standards concerning visualization data exchange. These formats are not intended for use for CAx data exchange or product data exchange.

The JT file format presented in this Publicly Available Specification is intended to provide data that can be used for further engineering activities in a PLM domain. The other formats were found to support product documentation (U3D) and visualisation data exchange based on XML (COLLADA, X3D) in a similar domain.

The International Organization for Standardization (ISO) draws attention to the fact that it is claimed that compliance with this document may involve the use of patent USA 20110199382.

ISO takes no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured ISO that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with ISO. Information may be obtained from:

Siemens PLM Software Product Development
 345 Woodcliff Drive
 Fairport
 NY 14450
 USA

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. ISO shall not be held responsible for identifying any or all such patent rights.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/PAS 14306:2011

<https://standards.iteh.ai/catalog/standards/sist/7ddef245-fb1-477d-ad1b-765866d37fb/iso-pas-14306-2011>

Industrial automation systems and integration — JT file format specification for 3D visualization

1 Scope

This Publicly Available Specification defines the syntax and semantics of the JT Version 8.1 file format.

The JT format is an industry focused, high-performance, lightweight, flexible file format for capturing and repurposing 3D Product Definition data that enables collaboration, validation and visualization throughout the extended enterprise. JT format is the de-facto standard 3D Visualization format in the automotive industry, and the single most dominant 3D visualization format in Aerospace, Heavy Equipment and other mechanical CAD domains.

The JT format is both robust, and streamable, and contains best-in-class compression for compact and efficient representation. The JT format was designed to be easily integrated into enterprise translation solutions, producing a single set of 3D digital assets that support a full range of downstream processes from lightweight web-based viewing to full product digital mockups.

At its core the JT format is a scene graph with CAD specific node and attributes support. Facet information (triangles), is stored with sophisticated geometry compression techniques. Visual attributes such as lights, textures, materials and shaders (Cg and OGLSL) are supported. Product and Manufacturing Information (PMI), Precise Part definitions (B-Rep) and Metadata as well as a variety of representation configurations are supported by the format. The JT format is also structured to enable support for various delivery methods including asynchronous streaming of content.

Some of the highlights of the JT format include:

- Built-in support for assemblies, sub-assemblies and part constructs
- Flexible partitioning scheme, supporting single or multiple files
- B-Rep, including integrated support for industry standard Parasolid® (XT) format
- Product Manufacturing Information in support of paperless manufacturing initiatives
- Precise and imprecise wireframe
- Discrete purpose-built Levels of Detail
- Wire harness information
- Triangle sets, Polygon sets, Point sets, Line sets and Implicit Primitive sets (cylinder, cone, sphere, etc...)
- Full array of visual attributes: Materials, Textures, Lights, Shaders
- Hierarchical Bounding Box and Bounding Spheres
- Advanced data compression that allows producers of JT files to fine tune the trade off between compression ratio and fidelity of the data.

Beyond the data contents description of the JT Format, the overall physical structure/organization of the format is also designed to support operations such as:

- Offline optimizations of the data contents
- File granularity and flexibility optimized to meet the needs of Enterprise Data Translation Solutions
- Asynchronous streaming of content
- Viewing optimizations such as view frustum and occlusion culling and fixed-framerate display modes.
- Layers, and Layer Filters.

Along with the pure syntactical definition of the JT Format, there is also series of conventions which although not required to have a reference compliant JT file, have become commonplace within JT format translators. These conventions have been documented in the “Best Practices” section of this JT format reference.

This JT format reference does not specifically address implementation of, nor define, a run-time architecture for viewing and/or processing JT data. This is because although the JT format is closely aligned with a run-time data representation for fast and efficient loading/unloading of data, no interaction behavior is defined within the format itself, either in the form of specific viewer controls, viewport information, animation behavior or other event-based interactivity. This exclusion of interaction behavior from the JT format makes the format more easily reusable for dissimilar application interoperation and also facilitates incremental update, without losing downstream authored data, as the original CAD asset revises.

2 References and Additional Information

- [1] *JT Open Program* (<http://www.jtopen.com>) --- A program to help members leverage the benefits of open collaboration across the extended enterprise through the adoption of the JT format, a technology that makes it possible to view and share product information throughout the product lifecycle. Membership in the JT Open Program provides access to the JT Open Toolkit library, which among other things, provides read and write access to JT data and enforces certain JT conventions to ensure data compatibility with other JT-enabled applications.
- [2] *JT2Go download* (<http://www.jt2go.com>) --- JT2Go is the no-charge 3D JT viewer from Siemens. JT2Go puts 3D data at your fingertips by allowing anyone to download the no-charge viewer. JT2Go also allows anyone to embed 3D JT data directly into Microsoft Office documents. JT2Go offers full 3D interactivity on parts, assemblies, and even 2D drawings (CGM & TIF).
- [3] *Siemens: PLM Components: Parasolid: XT Pipeline* (<http://www.ugs.com/products/open/parasolid/pipeline.shtml>) --- This web page provides information on the Parasolid precise boundary representation format (XT) and how this XT format fits within the Siemens vision of seamless exchange of digital product models across enterprises, between different disciplines, using their PLM applications of choice.
- [4] *OpenGL Programming Guide : the official guide to learning OpenGL Version 2*, Fifth Edition, by OpenGL Architecture Review Board, Dave Shreiner, Mason Woo, Jackie Neider, and Tom Davis (Addison-Wesley 2005) --- This book gives in-depth explanation of the OpenGL Specification and will provide further insight into the significance of some of the data (e.g. Materials, Textures) that can exist in a JT file. Information in this book may also serve as a guide for how one could process the data contained in a JT file to produce/render an image on the screen.
- [5] Michael Deering, *Geometry Compression*, Computer Graphics, Proceedings SIGGRAPH '95, August 1995, pp. 13-20.
- [6] Michael Deering, Craig Gotsman, Stefan Gumhold, Jarek Rossignac, and Gabriel Taubin, *3D Geometry Compression*, Course Notes for SIGGRAPH 2000, July 25, 2000.
- [7] *OpenGL Shading Language Specification* (<http://www.opengl.org/documentation/glsl/>) --- OpenGL Shading Language (GLSL) as defined by the OpenGL Architectural Review Board, the governing body of OpenGL.
- [8] *Cg Toolkit Users Manual* (http://developer.nvidia.com/object/cg_users_manual.html) --- Explains everything you need to learn and use the Cg language as well as the Cg runtime library.
- [9] *The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics*, Randima Fernando and Mark J. Kilgard, nVIDIA Corporation, Addison Wesley Publishing Company, April 2003
- [10] K. Weiler. *Topological Structures for Geometric Modeling*, PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, 1986.
- [11] C. M. Hoffmann. *Geometric and Solid Modeling: An Introduction*. Morgan Kaufmann Publishers, Inc., San Mateo, California, 1989.
- [12] *Planetmath.org - Huffman Coding* (<http://planetmath.org/encyclopedia/HuffmanCoding.html>) --- This web page provides a technical overview of Huffman coding which is one form of data encoding used within the JT format.

- [13] Michael Schindler, *Practical Huffman Coding* (<http://www.compressconsult.com/huffman/#encoding>) --- This web page provides some coding hints for implementing Huffman coding which is one form of data encoding used within the JT format.
- [14] Glen G. Langdon Jr., *An Introduction to Arithmetic Coding*, IBM Journal of Research and Development, Volume 28, Number 2, March 1984, pp. 135-149.
- [15] Paul G. Howard and Jeffrey Scott Vitter, *Practical Implementation of Arithmetic Coding. Image and Text Compression*, ed. J. A. Storer, Kluwer Academic Publishers, April 1992, pp. 85-112.
- [16] zlib.net (<http://www.zlib.net/>) --- This web page provides (either directly or through links) complete detailed information on ZLIB compression including frequently asked questions, technical documentation, source code downloads, etc.

3 Definitions

3.1 Terms

It is assumed that readers of this document are familiar with concepts in the area of computer graphics and solid modeling. The intention of this section is not to provide comprehensive definitions, but is to provide a short introduction and clarification of the usage of terms within this document.

Assembly	– A related collection of <i>model</i> parts, represented in a JT format logical scene graph as a logical graph branch
Attribute	– Objects associated with nodes in a <i>logical scene graph</i> and specifying one of several appearances, positioning, or rendering characteristics of a <i>shape</i>
Boundary Representation	– A solid model representation where the solid volume is specified by its surface boundary (both its geometric and topological boundaries).
CodeText	– A collection of data in encoded form.
Directed Acyclic Graph	– A <i>graph</i> is a set of nodes, and a set of edges connecting the nodes in a tree like structure. A <i>directed graph</i> is one in which every edge has a direction such that edge (u,v), connecting node-u with node-v, is different from edge (v,u). A <i>Directed Acyclic Graph</i> is a directed graph with no cycles; where a cycle is a path (sequence of edges) from a node to itself. So with a <i>Directed Acyclic Graph</i> there is no path that can be followed within the graph such that the first node in the path is the same as the last node in the path.
JT Enabled Application	– Application which supports reading and/or writing reference compliant JT Format files.
Level of Detail	– One alternative graphical representation for some <i>model</i> component (e.g. part).
Logical Scene Graph	– A <i>scene graph</i> representing the logical organization of a <i>model</i> . Contains <i>shapes</i> and <i>attributes</i> representing the <i>model's</i> physical components, <i>properties</i> identifying arbitrary metadata (e.g. names, semantic roles) of those components, and a hierarchical structure expressing the component relationships.