

---

---

**Information technology — Coded  
representation of immersive media —  
Part 9:  
Geometry-based point cloud  
compression**

iTeh STANDARD PREVIEW  
(standards.iteh.ai)

*Technologies de l'information — Représentation codée de média  
immersifs —*

*Partie 9: Compression des nuages de points basée sur la géométrie*

ISO/IEC DIS 23090-9

<https://standards.iteh.ai/catalog/standards/sist/ffea08f2-71b7-4e84-80d8-ee6f45f4df14/iso-iec-dis-23090-9>



iTeh STANDARD PREVIEW  
(standards.iteh.ai)

ISO/IEC DIS 23090-9

<https://standards.iteh.ai/catalog/standards/sist/ffea08f2-71b7-4e84-80d8-ee6f45f4df14/iso-iec-dis-23090-9>



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2023

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
CP 401 • Ch. de Blandonnet 8  
CH-1214 Vernier, Geneva  
Phone: +41 22 749 01 11  
Email: [copyright@iso.org](mailto:copyright@iso.org)  
Website: [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

	Page
<b>Foreword</b> .....	<b>vii</b>
<b>Introduction</b> .....	<b>viii</b>
<b>1 Scope</b> .....	<b>1</b>
<b>2 Normative references</b> .....	<b>1</b>
<b>3 Terms and definitions</b> .....	<b>1</b>
3.1 General terms.....	1
3.2 Terms related to high-level syntax and entropy coding.....	3
3.3 Terms related to tree structure.....	6
3.4 Terms related to geometry coding.....	7
3.5 Terms related to attribute coding.....	7
<b>4 Abbreviated terms</b> .....	<b>8</b>
<b>5 Conventions</b> .....	<b>9</b>
5.1 General.....	9
5.2 Symbolic names.....	9
5.3 Numerical representation.....	10
5.4 Arithmetic operators.....	10
5.5 Logical operators.....	10
5.6 Relational operators.....	10
5.7 Bit-wise operators.....	11
5.8 Assignment operators.....	11
5.9 Range notation.....	11
5.10 Mathematical functions.....	12
5.10.1 General.....	12
5.10.2 IntAtan2.....	13
5.10.3 IntCos and IntSin.....	13
5.10.4 IntSqrt.....	14
5.10.5 IntRecipSqrt.....	14
5.10.6 Div.....	15
5.10.7 Morton.....	15
5.10.8 FromMorton.....	16
5.11 Order of operation precedence.....	16
5.12 Named expressions.....	17
5.12.1 General.....	17
5.12.2 Scope of a named expression.....	18
5.12.3 Arguments of named expressions.....	18
5.12.4 Sub-expressions.....	19
5.12.5 Definitions with multiple statements.....	19
5.12.6 Textual definitions.....	19
5.13 Variables, syntax elements and tables.....	19
<b>6 Point cloud format and relationship to coded and output representations</b> .....	<b>20</b>
6.1 General format.....	20
6.2 Attributes.....	20
6.2.1 General.....	20
6.2.2 Colour.....	20
6.2.3 Opacity.....	21
6.2.4 Reflectance.....	21
6.2.5 Normal vector.....	21
6.2.6 Material identifier.....	21
6.2.7 Frame number/index.....	21
6.2.8 User defined attributes.....	22
6.3 Codec-derived attributes.....	22
6.3.1 General.....	22

6.3.2	Slice identifier.....	22
6.3.3	Slice tag.....	22
6.3.4	Canonical point order.....	22
6.3.5	Point Morton order.....	23
6.4	Coded point cloud format.....	23
6.4.1	Sequence coordinate system.....	23
6.4.2	Coding coordinate system.....	24
6.4.3	Coded point cloud sequence.....	25
6.4.4	Coded point cloud frame.....	25
6.4.5	Slice of a coded point cloud frame.....	25
6.4.6	Repetition of slices.....	26
6.4.7	Relationship between tiles and slices.....	26
6.4.8	Parameter sets.....	27
6.5	Output point cloud format.....	28
6.5.1	General.....	28
6.5.2	Coordinate system.....	28
6.5.3	Fixed-point conformance output.....	28
6.5.4	Attributes.....	28
6.5.5	Output point cloud sequence.....	28
6.5.6	Output point cloud frame.....	28
<b>7</b>	<b>Syntax and semantics.....</b>	<b>29</b>
7.1	Method of specifying syntax in tabular form.....	29
7.2	Specification of syntax functions and descriptors.....	30
7.3	Syntax in tabular form.....	30
7.3.1	General.....	30
7.3.2	Parameter sets, ancillary data and byte alignment.....	31
7.3.3	Geometry data unit.....	37
7.3.4	Attribute data unit.....	42
7.3.5	Defaulted attribute data unit syntax.....	44
7.4	Semantics.....	44
7.4.1	General.....	44
7.4.2	Parameter sets, ancillary data and byte alignment.....	45
7.4.3	Geometry data unit.....	58
7.4.4	Attribute data unit.....	59
7.4.5	Defaulted attribute data unit semantics.....	59
<b>8</b>	<b>Decoding process.....</b>	<b>60</b>
8.1	General decoding process.....	60
8.2	Frame decoding processes.....	60
8.2.1	General.....	60
8.2.2	Frame counter.....	60
8.3	Slice decoding processes.....	60
8.3.1	General.....	60
8.3.2	State variables.....	61
8.3.3	Geometry decoding process.....	61
8.3.4	Default attribute values.....	61
8.3.5	Attribute decoding process.....	61
8.3.6	At the end of a slice.....	61
<b>9</b>	<b>Slice geometry.....</b>	<b>62</b>
9.1	General.....	62
9.2	Occupancy tree.....	62
9.2.1	General.....	62
9.2.2	Coded occupancy tree.....	62
9.2.3	Occupancy tree syntax element semantics.....	64
9.2.4	Node dimensions per tree level.....	65
9.2.5	State representation.....	65
9.2.6	Occupancy tree node coding.....	66
9.2.7	Occupied neighbourhood patterns.....	70

9.2.8	Neighbourhood-permuted node occupancy bitmap	72
9.2.9	Dictionary coding of occupancy_byte	73
9.2.10	Bitwise occupancy coding	78
9.2.11	Planar occupancy coding	84
9.2.12	Direct nodes	90
9.2.13	Angular coding	95
9.2.14	Subtree scaling	104
9.3	Predictive tree	109
9.3.1	General	109
9.3.2	Syntax element semantics	109
9.3.3	Tree traversal for reconstruction of point positions	110
9.3.4	Reconstruction of point coordinates	111
<b>10</b>	<b>Slice attributes</b>	<b>113</b>
10.1	General	113
10.2	Point coordinates	113
10.2.1	General	113
10.2.2	Conversion to scaled angular coordinates	114
10.3	Syntax element semantics	114
10.3.1	Attribute data unit coefficients	114
10.3.2	Attribute coefficient tuples	114
10.3.3	Raw attribute values	115
10.4	Raw attribute decoding	115
10.5	Attribute decoding using the region-adaptive hierarchical transform	115
10.5.1	General	115
10.5.2	Transform tree	115
10.5.3	Coefficient order	118
10.5.4	Coefficient scaling	119
10.5.5	Transform domain prediction	121
10.5.6	Inverse transform	125
10.5.7	Reconstructed attribute values	127
10.6	Attribute decoding using levels of detail	127
10.6.1	General	127
10.6.2	Syntax element semantics	128
10.6.3	Reconstruction process	128
10.6.4	State variables	128
10.6.5	Levels of detail	129
10.6.6	Predictor search	135
10.6.7	Reconstruction of attribute values	141
10.6.8	Prediction mode coding	142
10.6.9	Scaling	144
10.6.10	Coefficient prediction	144
10.6.11	Transform coefficient weights	145
10.6.12	Transform	146
10.7	Attribute quantization parameters	147
10.7.1	Syntax element semantics	147
10.7.2	Per-point regional QP offset	148
10.7.3	Attribute coefficient QP	148
10.7.4	Definition of <i>AttrQstep</i>	148
<b>11</b>	<b>Parsing process</b>	<b>149</b>
11.1	General	149
11.2	Data unit buffer	151
11.2.1	General	151
11.2.2	State	151
11.2.3	Initialization at the start of parsing a data unit	151
11.2.4	Initialization at the start of parsing a geometry data unit footer	151
11.2.5	Definition of <i>DuNextBit</i>	152
11.3	Chunked bytestream parsing	152

11.3.1	General	152
11.3.2	Chunk syntax	152
11.3.3	Chunk semantics	153
11.3.4	State	153
11.3.5	Span of chunked bytestream data within a data unit	153
11.3.6	The chunk buffer	153
11.3.7	State update at the start of every CBS	154
11.3.8	Unpacking a single chunk	154
11.3.9	Definition of <i>ChunkNextAeBit</i>	154
11.3.10	Definition of <i>ChunkNextBpBit</i>	154
11.3.11	Boundary between spliced chunked bytestreams	155
11.3.12	Location of chunked bytestream boundaries	156
11.4	General inverse binarization processes	156
11.4.1	Parsing unsigned fixed-length codes (FL)	156
11.4.2	Parsing signed fixed-length codes (FL+S)	156
11.4.3	Parsing <i>k</i> -th order exp-Golomb codes (EGk)	156
11.4.4	Parsing concatenated truncated unary and <i>k</i> -th order exp-Golomb codes (TU+EGk)	157
11.4.5	Parsing truncated unary codes (TU)	157
11.4.6	Mapping process for signed codes	157
11.4.7	Parsing ASN.1 object identifiers	158
11.5	CABAC parsing processes	158
11.5.1	Initialization	158
11.5.2	Definition of <i>AeReadBin</i>	158
11.5.3	Contextual probability models	159
11.5.4	Arithmetic decoding engine	162
11.6	Parsing state memorization and restoration	164
11.6.1	General	164
11.6.2	Geometry data units	164
11.6.3	Attribute data units	165
11.6.4	Defaulted attribute data units	165
<b>Annex A (normative) Profiles and levels</b>		<b>166</b>
<b>Annex B (normative) Type-length-value encapsulated bytestream format</b>		<b>172</b>
<b>Annex C (informative) Arithmetic encoding engine</b>		<b>174</b>
<b>Annex D (normative) Partial decoding and spatial scalability</b>		<b>177</b>
<b>Annex E (informative) Index of named expressions and variables</b>		<b>179</b>
<b>Bibliography</b>		<b>185</b>

## Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://www.iso.org/directives) or [www.iec.ch/members\\_experts/refdocs](http://www.iec.ch/members_experts/refdocs)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](http://www.iso.org/patents)) or the IEC list of patent declarations received (see <https://patents.iec.ch>).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html). In the IEC, see [www.iec.ch/understanding-standards](http://www.iec.ch/understanding-standards).

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23090 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](http://www.iso.org/members.html) and [www.iec.ch/national-committees](http://www.iec.ch/national-committees).

## Introduction

Advancements in 3D capturing and rendering technologies are enabling new applications and services in the fields of assisted and autonomous driving, cartography, cultural heritage, industrial processes, immersive real-time communication, and virtual/augmented/mixed reality (VR/AR/MR) content creation, transmission and communication. Point clouds have arisen as one of the main representations for such applications. A point cloud frame consists of a set of 3D points. Every point, in addition to having a 3D position, may also be associated with numerous other attributes such as colour, transparency, reflectance, timestamp, surface normal and classification. Such representations require a large amount of data, which can be costly in terms of storage and transmission. This document provides the method for efficiently compressing point cloud representations.

The International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of a patent.

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

The holder of this patent right has assured ISO and IEC that he/she 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 and IEC. Information may be obtained from the patent database available at [www.iso.org/patents](http://www.iso.org/patents) or <https://patents.iec.ch>.

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

(standards.iteh.ai)

ISO/IEC DIS 23090-9

<https://standards.iteh.ai/catalog/standards/sist/ffea08f2-71b7-4e84-80d8-ee6f45f4df14/iso-iec-dis-23090-9>



# Information technology — Coded representation of immersive media —

## Part 9: Geometry-based point cloud compression

### 1 Scope

This document specifies geometry-based point cloud compression.

### 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

Rec. ITU-T X.690 | ISO/IEC 8825-1, *Information technology — ASN.1 encoding rules — Part 1: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

Rec. ITU-T X.660 | ISO/IEC 9834-1, *Information technology — Procedures for the operation of object identifier registration authorities: General procedures and top arcs of the international object identifier tree — Part 1:*

Rec. ITU-T X.667 | ISO/IEC 9834-8, *Information technology — Procedures for the operation of object identifier registration authorities — Part 8: Generation of universally unique identifiers (UUIDs) and their use in object identifiers*

ISO/IEC 23091-2, *Information technology — Coding-independent code points — Part 2: Video*

Rec. ITU-T T.35, *Procedure for the allocation of ITU T defined codes for non-standard facilities*

### 3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

#### 3.1 General terms

##### 3.1.1 point

fundamental element of a *point cloud* (3.1.2) comprising a position specified as *Cartesian coordinates* (3.1.8) and zero or more *attributes* (3.1.19)

##### 3.1.2

##### point cloud

unordered list of *points* (3.1.1)

### 3.1.3

#### **point cloud sequence**

sequence of one or more *point clouds* ([3.1.2](#))

### 3.1.4

#### **point cloud frame**

*point cloud* ([3.1.2](#)) in a *point cloud sequence* ([3.1.3](#))

### 3.1.5

#### **coded point cloud frame**

coded representation of a *point cloud frame* ([3.1.4](#))

### 3.1.6

#### **canonical point order**

order of *points* ([3.1.1](#)) decoded from a *slice* ([3.1.21](#)) according to the decoding processes

Note 1 to entry: The decoding processes are specified in [Clause 9](#).

### 3.1.7

#### **bounding box**

axis-aligned cuboid defining a spatial region that bounds a set of *points* ([3.1.1](#))

### 3.1.8

#### **Cartesian coordinates**

three scalar multiples of respective orthogonal *XYZ* ([3.1.11](#)) unit vectors with finite precision and bounds that specify a position relative to a fixed reference

### 3.1.9

#### **angular coordinates**

position specified as the radial distance  $\rho$  from the V axis, an azimuth angle  $\varphi$  in the S-T plane and an indexed elevation

### 3.1.10

#### **attribute coordinates**

*STV* ([3.1.12](#)) or scaled *RPI* ([3.1.13](#)) point coordinates used to code an *attribute* ([3.1.19](#))

### 3.1.11

#### **XYZ**

#### **XYZ axes**

X, Y and Z axes, in that order, used to represent *Cartesian coordinates* ([3.1.8](#))

### 3.1.12

#### **STV**

#### **STV axes**

S, T and V axes, in that order, that are a sequence-dependent permutation of the *XYZ axes* ([3.1.11](#)); used to represent the coded *geometry* ([3.1.18](#))

### 3.1.13

#### **RPI**

#### **RPI axes**

R, P and I axes, in that order, used to represent *angular coordinates* ([3.1.9](#))

### 3.1.14

#### **sequence coordinate system**

scaled and translated application-specific coordinate system that applies to an entire coded *point cloud sequence* ([3.1.3](#)) and in which all *points* ([3.1.1](#)) have non-negative fixed-point coordinates

### 3.1.15

#### **coding coordinate system**

scaled *sequence coordinate system* ([3.1.14](#)) that applies to an entire coded *point cloud sequence* ([3.1.3](#)) and in which all *points* ([3.1.1](#)) have non-negative integer coordinates

**3.1.16****slice coordinate system**

translated *coding coordinate system* (3.1.15) that applies to a single *slice* (3.1.21) and in which all *points* (3.1.1) in the slice have non-negative integer coordinates

**3.1.17****beam**

sampler of point positions using *angular coordinates* (3.1.9) by rays cast with a fixed elevation and from a point on and rotating around the V axis at the angular origin

**3.1.18****geometry**

*point positions* (3.4.1) associated with a set of *points* (3.1.1)

**3.1.19****attribute**

scalar or vector property associated with each *point* (3.1.1) in a *point cloud* (3.1.2)

EXAMPLE Colour, reflectance, frame index.

**3.1.20****position**

<bit> bit in a binary string or value, representing the factor  $2^{\text{position}}$

EXAMPLE The LSB has bit position 0.

**3.1.21****slice**

*geometry* (3.1.18) and *attributes* (3.1.19) for part of, or an entire, *coded point cloud frame* (3.1.5)

Note 1 to entry: the *bounding boxes* (3.1.7) of any two slices can intersect.

**3.1.22****tile**

set of *slices* (3.1.21) identified by a common *slice\_tag syntax element* (3.2.16) value whose *geometry* (3.1.18) should be contained within a *bounding box* (3.1.7) specified in a *tile inventory data unit* (3.2.13)

**3.1.23****Morton code**

non-negative integer obtained by interleaving the bits of three integers

**3.1.24****Morton order**

order of elements according to their *Morton code* (3.1.23)

**3.1.25****sparse array**

array with fewer set elements than total addressable elements

Note 1 to entry: Unset elements can have an inferred value when accessed.

**3.2 Terms related to high-level syntax and entropy coding****3.2.1****ASN.1**

abstract syntax notation one

notation specified by Rec. ITU-T X.680 | ISO/IEC 8824-1 that is used for the definition of data types, values and constraints on data types

3.2.2

**bin**

binary symbol (bit) of the *binarized* (3.2.3) representation of a *syntax element's* (3.2.16) value

3.2.3

**binarization**

specification of a *syntax element's* (3.2.16) value as a sequence of *bins* (3.2.2)

3.2.4

**bypass**

<contextual probability model> static, equiprobable probability model

3.2.5

**bypass symbol**

*bypass-contextualized* (3.2.4) *bin* (3.2.2)

3.2.6

**bypass stream**

sequence of *bypass symbols* (3.2.5) that are not encoded in an arithmetic-coded *bitstream* (3.2.7)

3.2.7

**bitstream**

<data> sequence of bits

3.2.8

**bitstream**

<coded sequence> sequence of bits, in the form of encapsulated *data units* (3.2.13), that represents a coded *point cloud sequence* (3.1.3)

3.2.9

**set bit**

bit with the value 1

3.2.10

**unset bit**

bit with the value 0

3.2.11

**byte**

sequence of 8 bits, typeset with the most significant bit on the left and the least significant bit on the right.

Note 1 to entry: When represented in a bitstream, the most significant bit of a byte is first.

3.2.12

**byte aligned**

*bitstream* (3.2.7) position that is an integer multiple of eight bits from the position of the first bit in the bitstream

3.2.13

**data unit**

**DU**

sequence of *bytes* (3.2.11) conveying a single *syntax structure* (3.2.17) of known length

3.2.14

**data unit header**

parameters, located from the start of a *data unit* (3.2.13)

3.2.15

**data unit footer**

parameters, located from the end of a *data unit* (3.2.13)

**3.2.16****syntax element**

element of data represented in the *bitstream* (3.2.7)

**3.2.17****syntax structure**

zero or more *syntax elements* (3.2.16) present together in the *bitstream* (3.2.7) in a specified order

**3.2.18****parameter set**

collection of parameters that apply when activated

**3.2.19****sequence parameter set****SPS**

parameters for an entire coded *point cloud sequence* (3.1.3), conveyed by an SPS *data unit* (3.2.13) and activated when referenced by a *geometry* (3.1.18) data unit

**3.2.20****geometry parameter set****GPS**

parameters for the coding of *slice* (3.1.21) *geometry* (3.1.18), conveyed by a GPS *data unit* (3.2.13) and activated when referenced by a geometry data unit

**3.2.21****attribute parameter set****APS**

parameters for the coding of a *slice* (3.1.21) *attribute* (3.1.19), conveyed by an APS *data unit* (3.2.13) and activated when referenced by an attribute data unit

**3.2.22****object identifier****OID**

<ASN.1> ordered list of primary integer values from the root of the *international object identifier tree* (3.2.23) to a node, which unambiguously identifies that node

[SOURCE: Rec. ITU-T X.660 | ISO/IEC 9834-1:2012, 3.5.11, modified — The abbreviated term "OID" and the domain "<ASN.1>" have been added.]

**3.2.23****international object identifier tree**

tree whose root corresponds to Rec. ITU-T X.660 | ISO/IEC 9834-1 and whose nodes correspond to *registration authorities* (3.2.25) responsible for allocating arcs from a *parent node* (3.3.10)

[SOURCE: Rec. ITU-T X.660 | ISO/IEC 9834-1:2012, 3.5.5]

**3.2.24****registration**

<object identifier> assignment of an unambiguous name to an object in a way which makes the assignment available to interested parties

[SOURCE: Rec. ITU-T X.660 | ISO/IEC 9834-1:2012, 3.5.16, modified — The domain "<object identifier>" has been added.]

**3.2.25****registration authority**

<international object identifier tree> entity such as an organization, a standard or an automated facility that performs *registration* (3.2.24) of one or more types of objects

[SOURCE: Rec. ITU-T X.660 | ISO/IEC 9834-1, 3.5.17, modified — The domain "<international object identifier tree>" has been added.]

### 3.2.26

#### **application specific**

defined by an application or an application standard

### 3.2.27

#### **unspecified**

having no specified meaning in this document and will not have a specified meaning in future editions of this document

Note 1 to entry: It is used to specify values of a particular *syntax element* (3.2.16).

## 3.3 Terms related to tree structure

### 3.3.1

#### **tree**

recursive structure of *nodes* (3.3.7) without loops and containing a single *root node* (3.3.5)

### 3.3.2

#### **top**

<tree> *tree level* (3.3.4) with a *depth* (3.3.8) of 0, consisting of the *root node* (3.3.5)

### 3.3.3

#### **bottom**

<tree> *tree level* (3.3.4) with the greatest *depth* (3.3.8)

### 3.3.4

#### **tree level**

set of *nodes* (3.3.7) at the same *depth* (3.3.8) in a *tree* (3.3.1)

### 3.3.5

#### **root node**

*node* (3.3.7) without a *parent node* (3.3.10)

### 3.3.6

#### **leaf node**

terminal *node* (3.3.7) without any *child nodes* (3.3.9)

### 3.3.7

#### **node**

element of a *tree* (3.3.1)

### 3.3.8

#### **depth**

<node> number of descendent hops from the *root node* (3.3.5) to a *node* (3.3.7)

### 3.3.9

#### **child node**

direct descendent of a *node* (3.3.7)

### 3.3.10

#### **parent node**

direct ancestor of a *node* (3.3.7)

### 3.3.11

#### **grandparent node**

direct ancestor of a *node's* (3.3.7) *parent node* (3.3.10)

### 3.3.12

#### **great-grandparent node**

direct ancestor of a *node's* (3.3.7) *grandparent node* (3.3.11)

**3.3.13****sibling nodes**

*nodes* (3.3.7) that are *child nodes* (3.3.9) of the same *parent node* (3.3.10)

**3.3.14****subtree**

part of a *tree* (3.3.1) comprising a *subtree root node* (3.3.15) and all its descendants over all subsequent *tree levels* (3.3.4)

**3.3.15****subtree root node**

single *node* (3.3.7) of a *subtree* (3.3.14) from which all other nodes in the same subtree are descendants

**3.4 Terms related to geometry coding****3.4.1****position**

<point> three-dimensional coordinates of a *point* (3.1.1)

**3.4.2****occupancy tree**

eight-ary *tree* (3.3.1) of *occupancy tree nodes* (3.4.4) representing the *geometry* (3.1.18) of a *slice* (3.1.21)

**3.4.3****predictive tree**

*tree* (3.3.1) of *predictive tree nodes* (3.4.5) representing the *geometry* (3.1.18) of a *slice* (3.1.21)

**3.4.4****occupancy tree node**

*node* (3.3.7) of an *occupancy tree* (3.4.2) representing a sub-volume of the 3D space (or volume) containing the *point cloud* (3.1.2)

**3.4.5****predictive tree node**

*node* (3.3.7) of a *predictive tree* (3.4.3) representing a single *position* (3.4.1) for one or more *points* (3.1.1)

**3.4.6****direct node**

terminal *occupancy tree node* (3.4.4) that codes one or more *point positions* (3.4.1)

**3.4.7****occupancy bitmap**

8-bit bitmap for an *occupancy tree node* (3.4.4) whose bits indicate the existence of *child nodes* (3.3.9) at particular locations in the next *tree level* (3.3.4)

**3.4.8****occupied neighbourhood pattern**

pattern that indicates the existence and arrangement of the six possible *occupancy tree nodes* (3.4.4) that share faces with a central node

**3.5 Terms related to attribute coding****3.5.1****primary attribute component**

first, or only, *attribute* (3.1.19) component, identified by the index 0

**3.5.2****secondary attribute component**

*attribute* (3.1.19) component other than the first component, identified by an index greater than 0