

---

---

**Information technology — Genomic  
information representation —**

**Part 2:  
Coding of genomic information**

*Technologies de l'information — Représentation des informations  
génomiques —*

*Partie 2: Codage des informations génomiques*

ITeH Standards  
(<https://standards.iteh.ai>)  
Document Preview

ISO/IEC 23092-2:2020

<https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020>



**iTeh Standards**  
**(<https://standards.iteh.ai>)**  
**Document Preview**

[ISO/IEC 23092-2:2020](https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020)

<https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020>



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2020

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>
<b>4 Abbreviated terms</b>	<b>6</b>
<b>5 Conventions</b>	<b>6</b>
5.1 General	6
5.2 Arithmetic operators	7
5.3 Logical operators	7
5.4 Relational operators	7
5.5 Bit-wise operators	8
5.6 Assignment operators	8
5.7 Range notation	8
5.8 Mathematical functions	9
5.9 Order of operation precedence	9
5.10 Variables, syntax elements and tables	10
5.11 Text description of logical operators	11
5.12 Processes	12
<b>6 Syntax and semantics</b>	<b>12</b>
6.1 Method of specifying syntax in tabular form	12
6.2 Bit ordering	13
6.3 Specification of syntax functions and data types	13
6.4 Semantics	15
<b>7 Data structures</b>	<b>15</b>
7.1 General	15
7.2 Data unit	15
7.3 Raw reference	16
7.3.1 General	16
7.3.2 Syntax and semantics	16
7.4 Parameter set	17
7.4.1 Syntax and semantics	17
7.4.2 Encoding parameters	17
7.5 Access unit	23
7.5.1 Syntax and semantics	24
7.5.2 Access unit types	27
<b>8 Descriptors</b>	<b>28</b>
<b>9 Sequencing reads</b>	<b>31</b>
9.1 General	31
9.2 Supported symbols	31
9.3 Paired-end reads	33
9.4 Reverse-complement reads	33
9.5 Data classes	34
9.6 Aligned data	34
9.7 Unaligned data	35
<b>10 Decoding process</b>	<b>36</b>
10.1 General	36
10.2 dataset_type = 0 or 1	36
10.2.1 General	36
10.2.2 References padding	37

10.2.3	Type 1 AU (Class P).....	37
10.2.4	Type 2 AU (Class N).....	38
10.2.5	Type 3 AU (Class M).....	39
10.2.6	Type 4 AU (Class I).....	39
10.2.7	Type 5 AU (Class HM).....	41
10.2.8	Type 6 AU (Class U).....	41
10.3	dataset_type = 2.....	42
10.3.1	General.....	42
10.3.2	Type 1 AU.....	42
10.3.3	Type 2 AU.....	43
10.3.4	Type 3 AU.....	43
10.3.5	Type 4 AU.....	44
10.3.6	Type 6 AU.....	44
10.4	Genomic descriptors.....	44
10.4.1	General.....	44
10.4.2	pos.....	45
10.4.3	rcomp.....	45
10.4.4	flags.....	46
10.4.5	mmpos.....	47
10.4.6	mmtypes.....	49
10.4.7	clips.....	52
10.4.8	ureads.....	55
10.4.9	rln.....	55
10.4.10	pair.....	57
10.4.11	mscore.....	64
10.4.12	mmap.....	65
10.4.13	msar.....	67
10.4.14	rtype.....	68
10.4.15	rgroup.....	70
10.4.16	qv.....	70
10.4.17	rname.....	74
10.4.18	rftp.....	74
10.4.19	rftt.....	75
10.4.20	tokentype descriptors.....	76
10.5	sequence.....	85
10.5.1	General.....	85
10.5.2	Aligned reads (Classes P, N, M, I, HM).....	85
10.5.3	Unmapped reads (Class HM, U).....	86
10.6	e-cigar.....	86
10.6.1	Syntax.....	86
10.6.2	Decoding process for the first alignment.....	88
10.6.3	Decoding process for other alignments.....	95
10.6.4	Reference transformation.....	95
11	<b>Representation of reference sequences.....</b>	<b>97</b>
11.1	External reference.....	97
11.2	Embedded reference.....	97
11.3	Computed reference.....	97
11.3.1	General.....	97
11.3.2	Supported Algorithms.....	98
11.3.3	Reference transformation.....	98
11.3.4	PushIn.....	99
11.3.5	Local assembly.....	100
11.3.6	Global assembly.....	102
12	<b>Block payload parsing process.....</b>	<b>102</b>
12.1	General.....	102
12.2	Inverse binarizations.....	103
12.2.1	General.....	103

12.2.2	Binary (BI).....	104
12.2.3	Truncated unary (TU).....	104
12.2.4	Exponential golomb (EG) .....	104
12.2.5	If the output of step 2 is 1, symVal= -1*symValTruncated exponential golomb (TEG).....	105
12.2.6	Signed truncated exponential golomb (STEG) .....	105
12.2.7	Split unit-wise truncated unary (SUTU) .....	106
12.2.8	Signed split unit-wise truncated unary (SSUTU) .....	106
12.2.9	Double truncated unary (DTU) .....	107
12.2.10	Signed double truncated unary (SDTU) .....	107
12.3	Decoder configuration.....	107
12.3.1	Sequences and quality values .....	107
12.3.2	Support values .....	109
12.3.3	CABAC binarizations.....	109
12.3.4	Transformation parameters.....	112
12.3.5	Msar descriptor and read identifiers.....	113
12.3.6	State variables.....	114
12.4	Initialization process for context variables.....	117
12.5	Arithmetic decoding engine .....	118
12.5.1	Initialization.....	118
12.5.2	Arithmetic decoding process.....	118
12.6	Decoding process for sequence descriptors.....	125
12.6.1	General.....	125
12.6.2	Block payload decoding process .....	126
13	<b>Output format</b> .....	<b>141</b>
13.1	General.....	141
13.2	MPEG-G record.....	141
13.2.1	General.....	141
13.2.2	number_of_template_segments.....	143
13.2.3	number_of_record_segments.....	144
13.2.4	number_of_alignments .....	144
13.2.5	class_ID.....	144
13.2.6	read_group_len.....	144
13.2.7	reserved.....	144
13.2.8	read_1_first.....	144
13.2.9	seq_ID.....	144
13.2.10	as_depth.....	144
13.2.11	read_len.....	144
13.2.12	qv_depth.....	145
13.2.13	read_name_len.....	145
13.2.14	read_name .....	145
13.2.15	read_group .....	145
13.2.16	sequence .....	145
13.2.17	quality_values.....	145
13.2.18	mapping_pos.....	145
13.2.19	ecigar_len.....	145
13.2.20	ecigar_string.....	145
13.2.21	reverse_comp.....	145
13.2.22	mapping_score.....	145
13.2.23	split_alignment.....	146
13.2.24	delta .....	146
13.2.25	split_pos.....	146
13.2.26	split_seq_ID .....	146
13.2.27	flags .....	146
13.2.28	more_alignments .....	146
13.2.29	next_pos.....	146
13.2.30	next_seq_ID .....	146
13.3	Initialization process .....	146

<b>Annex A</b> (informative) <b>Tokenization of reads identifiers</b> .....	<b>150</b>
<b>Annex B</b> (informative) <b>Mapping quality</b> .....	<b>152</b>
<b>Annex C</b> (informative) <b>Inverse binarization examples</b> .....	<b>153</b>

**iTeh Standards**  
**(<https://standards.iteh.ai>)**  
**Document Preview**

[ISO/IEC 23092-2:2020](https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020)

<https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020>

## 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 specification 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 specification 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)).

Attention is drawn to the possibility that some of the elements of this specification 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 <http://patents.iec.ch>).

Any trade name used in this specification 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).

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

This second edition cancels and replaces the first edition (ISO/IEC 23092-1:2019), which has been technically revised.

The main changes compared to the previous edition are as follows:

- The sequence decoding process for mismatches in classes I and HM has been clarified.
- In [subclause 10.4](#) and its subclauses variable *numberOfAlignedRecordSegments* has been renamed to *numberOfMappedRecordSegments*.
- In [subclause 10.4.2](#) the decoding process of pos and rtype descriptors with computed reference has been clarified.
- In [subclause 11.3.4](#) the decoding process of pushin has been revised.
- The decoding of the *reverseComp* values has been revised.
- The determination of the offset of mismatches within spliced segments has been revised.
- The decoding process for signatures has been revised.
- The signalling of computed references has been clarified.
- In [Clause 12](#) some decoding processes and some transformations have been clarified.

A list of all parts in the ISO/IEC 23092 series can be found on the ISO website.

Any feedback or questions on this specification 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).

## Introduction

The advent of high-throughput sequencing (HTS) technologies has the potential to boost the adoption of genomic information in everyday practice, ranging from biological research to personalized genomic medicine in clinics. As a consequence, the volume of generated data has increased dramatically during the last few years, and an even more pronounced growth is expected in the near future.

At the moment genomic information is mostly exchanged through a variety of data formats, such as FASTA/FASTQ for unaligned sequencing reads and SAM/BAM/CRAM for aligned reads. With respect to such formats, the ISO/IEC 23092 series provides a new solution for the representation and compression of genome sequencing information by:

- Specifying an abstract representation of the sequencing data rather than a specific format with its direct implementation.
- Being designed at a time point when technologies and use cases are more mature. This permits addressing one limitation of the textual SAM format, for which the incremental ad-hoc addition of features followed along the years, resulting in an overall redundant and suboptimal format which was unnecessarily complicated.
- Separating free-field user-defined information with no clear semantics from the genomic data representation. This allows a fully interoperable and automatic exchange of information between different data producers.
- Allowing multiplexing of relevant metadata information with the data since data and metadata are partitioned at different conceptual levels.
- Following a strict and supervised development process which has proven successful in the last 30 years in the domain of digital media for the transport format, the file format, the compressed representation and the application program interfaces.

The ISO/IEC 23092 series provides the enabling technology that will allow the community to create an ecosystem of novel, interoperable, solutions in the field of genomic information processing. In particular it offers:

- Consistent, general and properly designed format definitions and data structures to store sequencing and alignment information. A robust framework which can be used as a foundation to implement different compression algorithms.
- Speed and flexibility in the selective access to coded data, by means of newly designed data clustering and optimized storage methodologies.
- Low latency in data transmission and consequent fast availability at remote locations, based on transmission protocols inspired by real-time application domains.
- Built-in privacy and protection of sensitive information, thanks to a flexible framework which allows customizable secured access at all layers of the data hierarchy.
- Reliability of the technology and interoperability among tools and systems, owing to the provision of a procedure to assess conformance to this document on an exhaustive dataset.
- Support to the implementation of a complete ecosystem of compliant devices and applications, through the availability of a normative reference implementation covering the totality of the ISO/IEC 23092 series.

The fundamental structure of the ISO/IEC 23092 series data representation is the *genomic record*. The genomic record is a data structure consisting of either a single sequencing read, or a paired sequencing read, and its associated sequencing and alignment information; it may contain detailed mapping and alignment data, a single or paired read identifier (read name) and quality values.



Without breaking traditional approaches, the genomic record introduced in the ISO/IEC 23092 series provides a more compact, simpler and manageable data structure grouping all the information related to a single DNA template, from simple sequencing data to sophisticated alignment information.

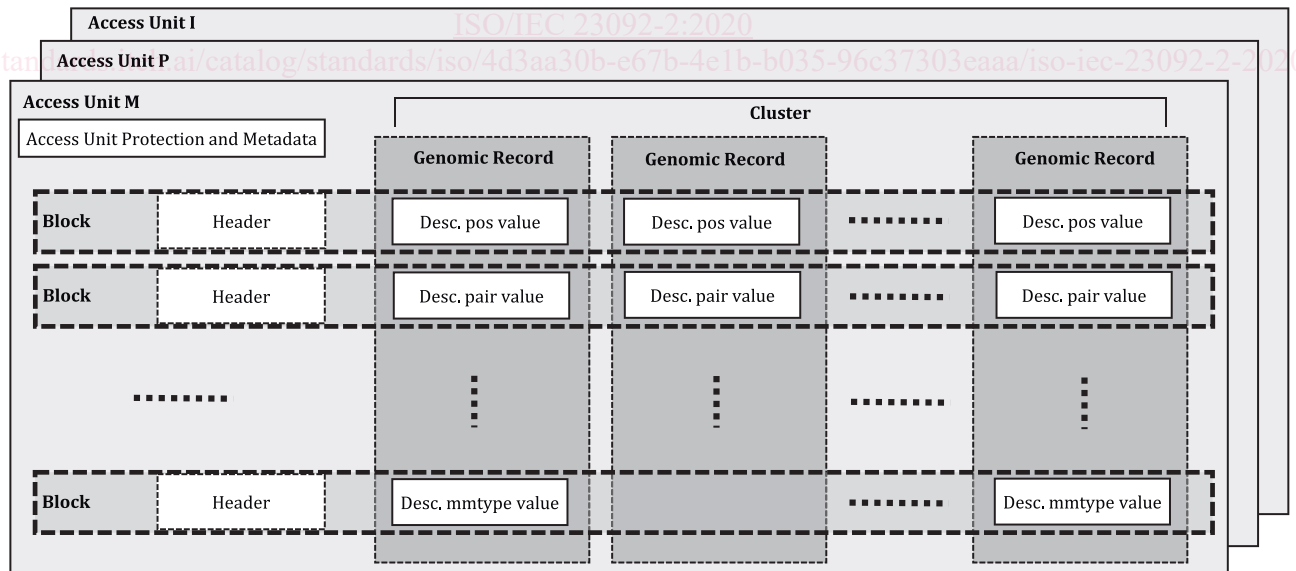
The genomic record, although it is an appropriate logic data structure for interaction and manipulation of coded information, is not a suitable atomic data structure for compression. To achieve high compression ratios, it is necessary to group genomic records into clusters and to transform the information of the same type into sets of descriptors structured into homogeneous blocks. Furthermore, when dealing with selective data access, the genomic record unit is too small to allow effective and fast information retrieval.

For these reasons, this document introduces the concept of access unit, which is the fundamental structure for coding and access to information in the compressed domain.

The access unit is the smallest data structure that can be decoded by a decoder compliant with ISO/IEC 23092-2. An access unit is composed of one block for each descriptor used to represent the information of its genomic records; therefore, a block payload is the coded representation of all the data of the same type (i.e. a descriptor) in a cluster.

In addition to clusters of genomic records compressed into access units, reads are further classified in six data classes: five classes are defined according to the result of their alignment against one or more reference sequences; the sixth class contains either reads that could not be mapped or raw sequencing data. The classification of sequencing reads into classes enables the development of powerful selective data access. In fact access units inherit a specific data characterization (e.g. perfect matches in class P, substitutions in class M, indels in class I, half-mapped reads in class HM) from the genomic records composing them, and thus constitute a data structure capable of providing powerful filtering capability for the efficient support of many different use cases.

Access units are the fundamental, finest grain data structure in terms of content protection and in terms of metadata association. In other words each access unit can be protected individually and independently. [Figure 1](#) shows how access units, blocks and genomic records relate to each other in the ISO/IEC 23092 series data structure.



**Figure 1 — Access units, blocks and genomic records**

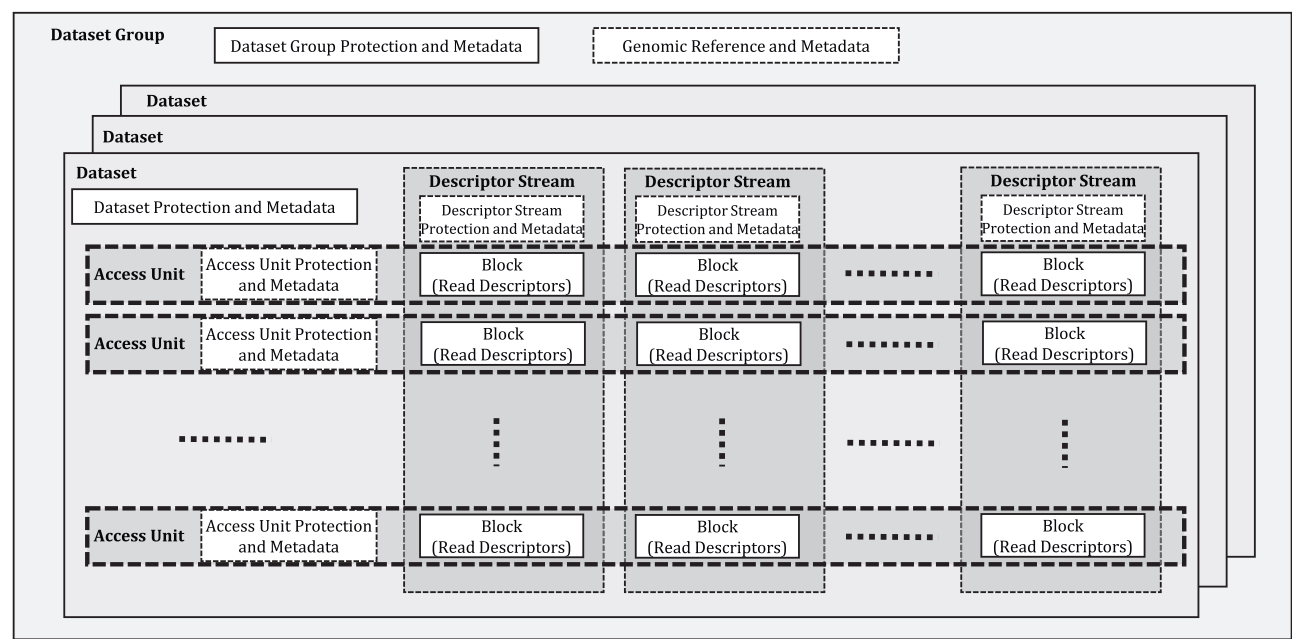
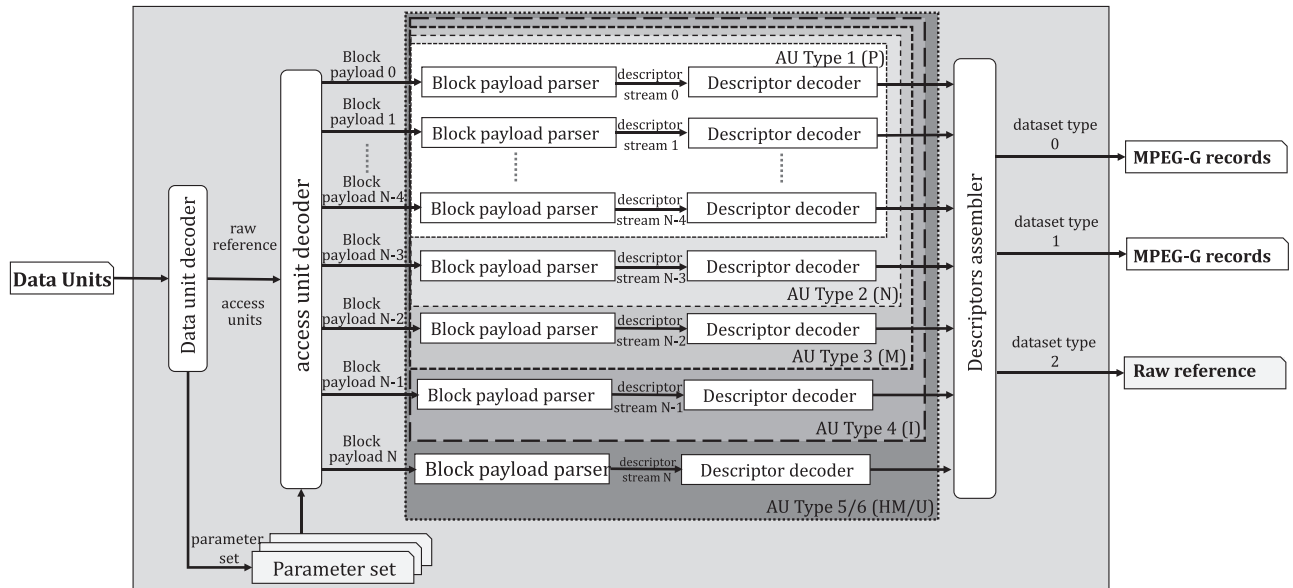


Figure 2 — High-level data structure: datasets and dataset group

A dataset is a coded data structure containing headers and one or more access units. Typical datasets could, for example, contain the complete sequencing of an individual, or a portion of it. Other datasets could contain for example a reference genome or a subset of its chromosomes. Datasets are grouped in dataset groups, as shown in [Figure 2](#).

According to the ISO/IEC 23092 series, the compressed sequencing data can be multiplexed into a bitstream suitable for packetization for real-time transport over typical network protocols. In storage use cases, coded data can be encapsulated into a file format with the possibility to organize blocks per descriptor stream or per access unit, to further optimize the selective access performance to the type of data access required by the different application scenarios. The ISO/IEC 23092 series further provides a reference process to convert a transport stream into a file format and vice versa.

The ISO/IEC 23092 series defines the syntax and semantics of the compressed genome sequencing data representation and the deterministic decoding process that reconstructs the contents of datasets. The decoding process is fully specified such that all decoders that conform to this document will produce identical decoded output. A simplified diagram of the decoding process is shown in [Figure 3](#).



**Figure 3 — The decoding process**

The International Organization for Standardization (ISO) and 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).

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.



# Information technology — Genomic information representation —

## Part 2: Coding of genomic information

### 1 Scope

This document provides specifications for the representation of the following types of genomic information:

- unaligned sequencing reads including read identifiers and quality values;
- aligned sequencing reads including read identifiers and quality values;
- reference sequences.

### 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.

ISO/IEC 10646, *Information technology — Universal Coded Character Set (UCS)*

ISO/IEC 23092-1:2020, *Information technology — Genomic information representation — Part 1: Transport and storage of genomic information* 23092-2:2020

<https://standards.iteh.ai/catalog/standards/iso/4d3aa30b-e67b-4e1b-b035-96c37303eaaa/iso-iec-23092-2-2020>

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23092-1 and the following apply.

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

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

#### 3.1 alignment

information describing the similarity between a sequence [typically a *sequencing read* (3.28)] and a reference sequence (for instance, a reference genome)

Note 1 to entry: An alignment is described in terms of a position within the reference, the strand of the reference, and a set of edit operations (matches, mismatches, insertions and deletions, clipping of the sequence ends and splicing information) needed to turn the first sequence into the second.

### 3.2

#### **CIGAR string**

CIGAR

textual way of representing an *alignment* ([3.1](#))

Note 1 to entry: Several definitions have been used by different programs; the one referred to here is the one used in the SAM format. It encodes a set of edit operations (matches, mismatches, insertions and deletions, clipping of the sequence ends and splicing information) needed to turn the sequencing read into the reference.

### 3.3

#### **dataset**

compression unit containing one or more of: reference sequences; *sequencing reads* ([3.28](#)); and *alignment* ([3.1](#)) information

Note 1 to entry: Datasets shall be as specified in ISO/IEC 23092-1.

### 3.4

#### **deletion**

contiguous removal of one or more bases from a genomic sequence

### 3.5

#### **E-CIGAR**

extended CIGAR syntax specified as a superset of the CIGAR syntax

Note 1 to entry: Among other things, E-CIGAR enables the unambiguous representation of substitutions, spliced reads and splice strandedness.

### 3.6

#### **edit operation**

modification of a sequence of *nucleotides* ([3.20](#)) by means of a substitution, *deletion* ([3.4](#)), *insertion* ([3.18](#)) or clip

### 3.7

#### **FASTA**

GIR that includes a name and a *nucleotide* ([3.20](#)) sequence for each *sequencing read* ([3.28](#))

Note 1 to entry: Additional information is usually encoded in the read identifier by bioinformatics tools (such as database information, and base calling information).

### 3.8

#### **FASTQ**

GIR that includes *FASTA* ([3.7](#)) and *quality values* ([3.22](#))

### 3.9

#### **first end**

end 1

read 1

first segment of a paired-end *template* ([3.33](#))

Note 1 to entry: Illumina platforms usually store first and second ends in two separate files and in the same order — i.e. the n-th read of the first FASTQ file and the n-th read of the second FASTQ file belong to the same template.

### 3.10

#### **genomic descriptor**

descriptor

element of the syntax used to represent a feature of a genomic *sequencing read* ([3.28](#)) or associated information such as *alignment* ([3.1](#)) information or *quality values* ([3.22](#))