INTERNATIONAL STANDARD

ISO/IEC 23092-5

First edition
2020-11

Information technology — Genomic information representation —

Part 5:
Conformance

Technologie de l'information — Représentation des informations génomiques —

Partie 5: Conformité

Reference number
ISO/IEC 23092-5:2020(E)

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

Published in Switzerland

Contents

Page

Foreword iv
Introduction v
1 Scope 1
2 Normative references 1
3 Terms and definitions 1
4 ISO/IEC 23092-1 conformance 1
4.1 Definition of ISO/IEC 23092-1 conformance 1
4.1.1 Assumptions 1
4.1.2 Definition of ISO/IEC 23092-1 file conformity 2
4.1.3 Definition of ISO/IEC 23092-1 decoder conformity 2
4.2 Requirements and functionality under test 2
4.3 Procedure to test file conformity 3
4.4 Procedure to test ISO/IEC 23092-1 decoder conformity 3
4.5 Test items for ISO/IEC 23092-1 conformance 3
4.5.1 Test items 3
4.5.2 Specification of tests 5
4.5.3 Support tool for reference verification 11
5 ISO/IEC 23092-2 conformance 12
5.1 Definition of ISO/IEC 23092-2 conformance 12
5.1.1 Assumptions 12
5.1.2 Definition of ISO/IEC 23092-2 bitstream conformity 12
5.1.3 Definition of ISO/IEC 23092-2 decoder conformity 12
5.2 Requirements and functionality under test 13
5.3 Procedure to test bitstream conformity 13
5.4 Procedure to test decoder conformity 13
5.5 Test items for ISO/IEC 23092-2 conformance 14
5.5.1 Set I: Genome sequencing data with single alignment 14
5.5.2 Set II: Quality values 16
5.5.3 Set III: Compressed references 18
5.5.4 Set IV: Genome sequencing data with multiple alignments 18
6 Conformance repository 19

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).

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) or the IEC list of patent declarations received (see http://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.

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 23092 series can be found on the ISO website.

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.

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 at the same time results not general and 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 sequence read, or a paired sequence 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 is a too small unit 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 the ISO/IEC 23092 series. 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 sequence 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.

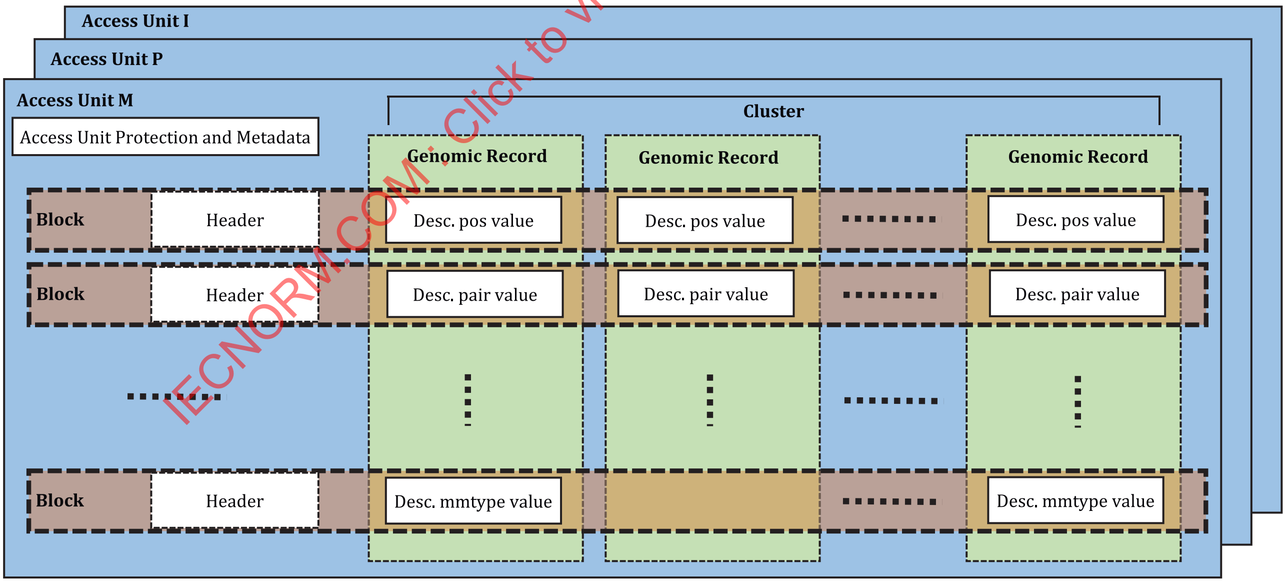

Figure 1 — Access units, blocks and genomic records

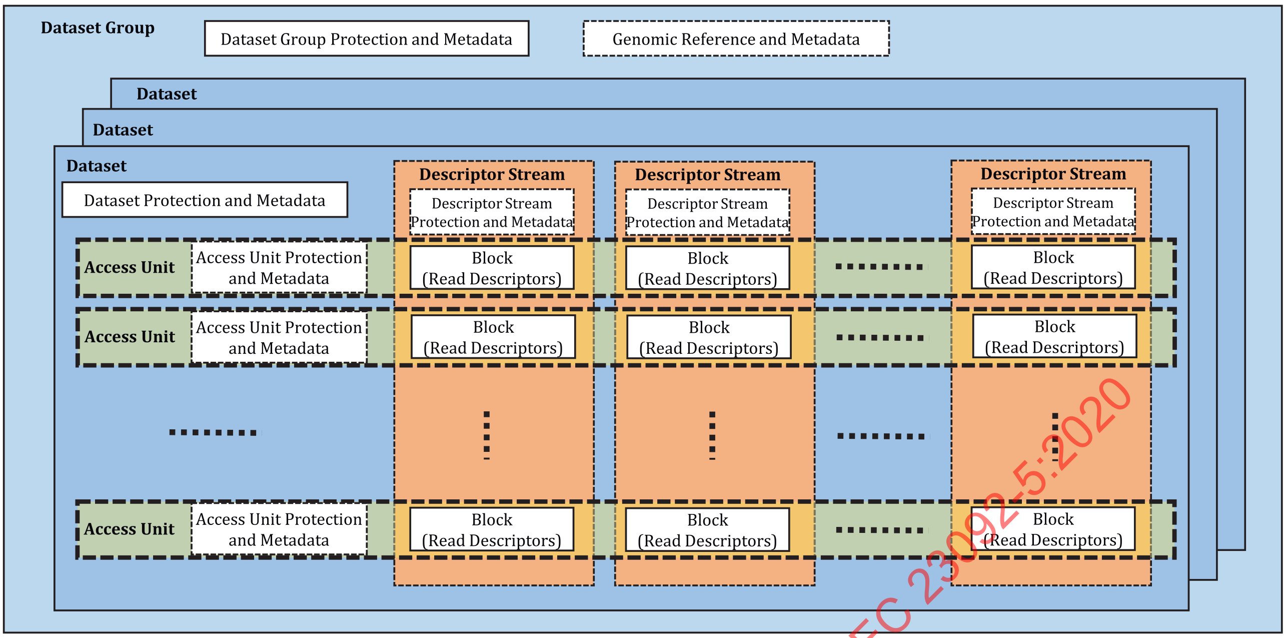
Dataset Group
Dataset Group Protection and Metadata
Genomic Reference and Metadata
Dataset
Dataset
Dataset
Dataset Protection and Metadata
Descriptor Stream
Descriptor Stream Protection and Metadata
Access Unit
Access Unit Protection and Metadata
Block (Read Descriptors)
....

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.

A simplified diagram of the dataset decoding process is shown in Figure 3.

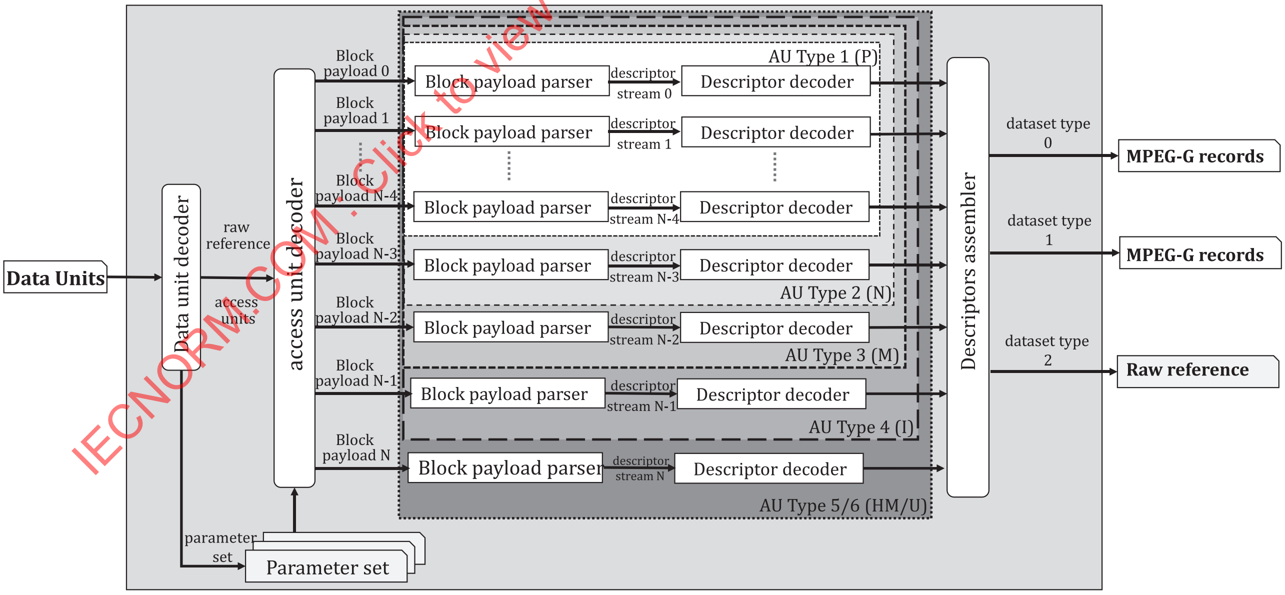

Figure 3 — Decoding process

This document defines a set of test procedures designed to verify whether bitstreams and decoders meet requirements specified in ISO/IEC 23092-1 and ISO/IEC 23092-2. In this document encoders are not addressed.

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.

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 5:
Conformance

1 Scope

This document specifies a set of test procedures designed to verify whether bitstreams and decoders meet requirements specified in ISO/IEC 23092-1 and ISO/IEC 23092-2.

Procedures are described for testing conformity of bitstreams and decoders to the requirements that are fully determined in ISO/IEC 23092-1 and ISO/IEC 23092-2. This document identifies those requirements, associates them to functionality under test and defines how conformity with them can be tested. Test bitstreams implemented according to those functionalities are provided in electronic form.

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

ISO/IEC 23092-2:2020, *Information technology — Genomic information representation — Part 2: Coding of genomic information*

3 Terms and definitions

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

4 ISO/IEC 23092-1 conformance

4.1 Definition of ISO/IEC 23092-1 conformance

4.1.1 Assumptions

In this document, the following assumptions are made in reference to ISO/IEC 23092-1:

The term ‘file’ means ISO/IEC 23092-1 file; the term ‘transport’ means ISO/IEC 23092-1 transport.

The term ‘decapsulator’ means ISO/IEC 23092-1 decapsulator, i.e. an implementation of the parsing and demultiplexing processes specified by ISO/IEC 23092-1. A decapsulator operates on data structures that are specified in ISO/IEC 23092-1:2020, Clause 6.

If any statement made in this document accidentally contradicts a statement or requirement in ISO/IEC 23092-1, the text of ISO/IEC 23092-1 prevails.

The following subclauses specify the tests to verify the conformity of files and decapsulators. Those tests make use of test data (test files and reference outputs), made available as specified in Clause 6, and make use of the reference software specified in ISO/IEC 23092-4, with source code available as described in ISO/IEC 23092-4.

This document does not specify tests to verify the conformity of transport.

4.1.2 Definition of ISO/IEC 23092-1 file conformity

An ISO/IEC 23092-1 file is a file that conforms to the specification defined by the requirements of ISO/IEC 23092-1.

A conformant file shall meet all the requirements and implement all the restrictions in the syntax specified in ISO/IEC 23092-1.

Subclause 4.3 defines the test that a file shall pass successfully in order to be claimed in conformity with ISO/IEC 23092-1.

4.1.3 Definition of ISO/IEC 23092-1 decoder conformity

An ISO/IEC 23092-1 decoder, or decapsulator, is an implementation of the processes necessary to parse and demultiplex the data structures of ISO/IEC 23092-1 and to perform operations associated to these data structures.

A conformant ISO/IEC 23092-1 decoder shall meet all the requirements and implement all the restrictions in the syntax defined by ISO/IEC 23092-1.

Subclause 4.4 defines the tests that a decoder shall pass successfully in order to be claimed in conformity with this document.

A conformant decoder shall implement parsing and decapsulation procedures that are equivalent to the ones specified in ISO/IEC 23092-1 and meet all the general requirements defined in ISO/IEC 23092-1.

Fundamental requirement areas for ISO/IEC 23092-1 decoders and their mapping to functionality under test are listed in subclause 4.2.

4.2 Requirements and functionality under test

Table 1 — Requirement areas for ISO/IEC 23092-1

Requirement area	Functionality
Dataset group	Dataset extraction from dataset group
Reference	Get reference with checksum calculation
Indexing by positions	Selective access by position ranges
Indexing by signatures	Selective access by signatures for non-aligned content (signature decoding)
Labels	Selective access by labels (single dataset)
Non-indexed content	Content extraction without indexing table
DSC and AUC storage mode	Access in AUC and DSC mode
Ordered blocks	Content extraction with and without ordered blocks

4.3 Procedure to test file conformity

ISO/IEC 23092-4 contains the source code of a software decoder that checks that a file implements properly the specification in ISO/IEC 23092-1.

A file that claims conformity with ISO/IEC 23092-1 shall pass the following test:

When processed by the reference software, the file shall not cause errors or non-conformity messages.

To verify the correctness of a file, it is necessary to parse it entirely, i.e. to parse all the syntactic elements and values derived from those syntactic elements used by the decoding procedures specified in ISO/IEC 23092-1.

4.4 Procedure to test ISO/IEC 23092-1 decoder conformity

This document provides test bitstreams in digital form; it also contains the reference output of each test bitstream as generated by the reference software (ISO/IEC 23092-4).

A decoder that claims conformity with ISO/IEC 23092-1 shall pass the following tests:

When processed by the decoder under test, each standard test file contained in this document and associated to ISO/IEC 23092-1 shall generate a sequence of output data units byte-per-byte identical to the corresponding reference output.

To verify the conformity of the decoder, it is necessary to decode all the standard test items associated to ISO/IEC 23092-1 and to check the identity of all the resulting data units. Data units are specified in ISO/IEC 23092-2:2020, 7.1.

It may not be possible to perform this type of test with a production decoder; in that case, the conformity must be assessed by the implementer during the design and development phase.

This document provides, in electronic form, a shell script, running on Linux OS or compatible terminals, to automate the whole test and verification process for the decoder conformity of the reference software (ISO/IEC 23092-4).

4.5 Test items for ISO/IEC 23092-1 conformance

4.5.1 Test items

Table 2 describes the test items for ISO/IEC 23092-1 conformance. Coverage is limited to ISO/IEC 23092-1:2020, subclause 5.5 and Clause 6, which specify the requirements for the decoder of ISO/IEC 23092-1.

All test items until, and including, AbL-016 are coded with AUC mode enabled.

Table 2 — Test items for the abstraction layer

Test Item	Description	ISO/IEC 23092-1 content coverage	Functionality under test
AbL-001	Extract a dataset from dataset group. Include extraction of raw reference (from FASTA) associated to the dataset.	Subclause 6.4.2 Subclause 6.4.1.2	Dataset extraction from dataset group
AbL-002	Extract a dataset from dataset group. Include extraction of AUs of reference (ISO/IEC 23092 compressed) associated to the dataset.	Subclause 6.4.2 Subclause 6.4.1.2	Dataset extraction from dataset group
AbL-003	Get raw reference from FASTA + MD5 checksum.	Subclause 6.4.1.2.4 Subclause 6.4.1.2.5	Get reference with checksum

Table 2 *(continued)*

Test Item	Description	ISO/IEC 23092-1 content coverage	Functionality under test
AbL-004	Get raw reference from FASTA + SHA-256 checksum.	Subclause 6.4.1.2.4 Subclause 6.4.1.2.5	Get reference with checksum
AbL-005	Get ISO/IEC 23092 compressed reference + SHA-256 checksum.	Subclause 6.4.1.2.5	Get reference checksum
AbL-006	Selective access by position range on a single reference sequence. Include at least the necessary part of reference.	Subclause 5.5 Subclause 6.5.2.1	Selective access by position ranges
AbL-007	Selective access by position range on several reference sequences. Include at least the necessary part of reference.	Subclause 5.5 Subclause 6.5.2.1	Selective access by position ranges
AbL-008	Selective access by position range, partially covered range on several reference sequences. Include at least the necessary parts of reference.	Subclause 5.5 Subclause 6.5.2.1	Selective access by position ranges
AbL-009	Selective access by position range on a single reference sequence. No data coverage in the range (no output).	Subclause 6.5.2.1	Selective access by position ranges
AbL-010	Selective access by signature with non-IUPAC alphabet; file with single signature.	Subclause 6.5.2.1 Subclause 6.5.2.2	Selective access for non-aligned content
AbL-011	Selective access by signatures with non-IUPAC alphabet; file with 2 signatures.	Subclause 6.5.2.1 Subclause 6.5.2.2	Selective access for non-aligned content
AbL-012	Selective access by signatures with IUPAC alphabet (reference sequence); dataset with single signature.	Subclause 6.5.2.1 Subclause 6.5.2.2	Selective access for non-aligned content
AbL-013	Selective access by Labels, single file with different Labels across multiple datasets, multiple regions. Tests with different queries.	Subclause 6.5.2.1 Subclause 6.4.1.4	Selective access by labels
AbL-014	File without MIT. Extract a complete dataset. Include the extracted reference.	Subclause 6.4.3	Content extraction without indexing table
AbL-015	File without MIT. Extract content with selective access without relying on MIT. Include the extracted range on reference.	Subclause 6.4.3	Content extraction without indexing table
AbL-016	File with 2 datasets using 2 different references. Selective access covering the two at the same time.	Subclause 6.5.2.1	Selective access by position ranges
AbL-017	The same as AbL-001 with file in DSC mode. Ordered block flag set to 1.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.4.1.2	Access in DSC mode
Abl-018	The same as AbL-001 with file in DSC mode. Ordered block flag set to 0.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.4.1.2	Content extraction without ordered blocks
Abl-019	The same as AbL-006 with file in DSC mode. Ordered block flag set to 1.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.5.2.1	Access in DSC mode

Table 2 *(continued)*

Test Item	Description	ISO/IEC 23092-1 content coverage	Functionality under test
AbL-020	The same as AbL-007 with file in DSC mode. Ordered block flag set to 1.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.5.2.1	Access in DSC mode
AbL-021	The same as AbL-013 with file in DSC mode. Ordered block flag set to 1.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.4.1.4	Access in DSC mode
AbL-022	The same as AbL-006 with file in DSC mode. Ordered block flag set to 0.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.5.2.1	Content extraction without ordered blocks
AbL-023	The same as AbL-007 with file in DSC mode (test extraction of the last block). Ordered block flag set to 0.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.5.2.1	Content extraction without ordered blocks
AbL-024	The same as AbL-013 with file in DSC mode. Ordered block flag set to 0.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.4.1.4	Content extraction without ordered blocks
AbL-025	The same as AbL-016 with file in DSC mode. Ordered block flag set to 1.	Subclause 6.5.3 Subclause 6.4.2.1.4 Subclause 6.4.1.2 Subclause 6.4.1.4	Selective access by position ranges in DSC mode
AbL-026	Get metadata using the offset mechanism.	Subclause 6.6.5	Offset

Subclauses which are covered by all tests are: ISO/IEC 23092-1:2020, subclauses 6.1, 6.2, 6.4.1, 6.4.2, 6.4.3, 6.5.1, 6.6.5.

4.5.2 Specification of tests

This subclause describes the steps to implement in order to run tests on test items specified in Table 2. For all items, the necessary parameter set(s) shall always be output first, prior to any other data unit.

Table 3 — Execution of tests

Test item	Test procedure
AbL-001	Input: AbL-001.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the dataset with ID equal to 2 contained in the file; identify the reference (FAST-A) related to the dataset and retrieve it completely Expected output: One file with the sequence of data units composing the dataset (in the same order as in the MIT) and a file with the related complete raw reference Criteria: The data units of the dataset and the raw reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-001

Table 3 *(continued)*

Test item	Test procedure
AbL-002	Input: AbL-002.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the dataset with ID equal to 2 contained in the file; identify the reference (compressed as another dataset in the dataset group) related to the dataset and retrieve it completely Expected output: One file with the sequence of data units composing the dataset and one file with data units composing the related compressed reference (in the same order as in the file) Criteria: The data units of the dataset and the compressed reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-002
AbL-003	Input: AbL-003.mgg (ISO/IEC 23092-1 file) Steps: Open the file; identify the reference (FASTA) related to the dataset with ID equal to 1 and retrieve it; calculate MD-5 checksum on each reference sequence Expected output: The checksums for each reference sequence Criteria: The checksums obtained as output shall by identical to the reference ones provided for test item AbL-003
AbL-004	Input: AbL-004.mgg (ISO/IEC 23092-1 file) Steps: Open the file; identify the reference (FASTA) related to the dataset with ID equal to 2 and retrieve it; calculate SHA-256 checksum on each reference sequence Expected output: The checksums for each reference sequence Criteria: The checksums obtained as output shall by identical to the reference ones provided for test item AbL-004
AbL-005	Input: AbL-005.mgg (ISO/IEC 23092-1 file) Steps: Open the file; identify the compressed reference related to the dataset with ID equal to 1 and retrieve the checksum for the reference Expected output: The checksum for the reference Criteria: The checksum obtained as output shall by identical to the reference one provided for test item AbL-005
AbL-006	Input: AbL-006.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 10000000 to 20000000 (included) of the sequence 1; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned region. Expected output: One file with the sequence of data units covering the region of the dataset (in the same order as in the file) and one file with the required data units of the related reference. Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-006

Table 3 *(continued)*

Test item	Test procedure
AbL-007	Input: AbL-007.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the regions a) 10’000’000 to 20’000’000 (included) of the sequence 1; b) 20’000’000 to 30’000’020 (included) of the sequence 3; c) 0 to 500’000 (included) of sequence 17; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned regions. Expected output: One file with the sequence of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-007
AbL-008	Input: AbL-008.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the regions a) 240’000’000 to 250’000’000 (included) of the sequence 1; b) 0 to 10’000’120 (included) of the sequence 3; c) 140’000’000 to 200’000’000 (included) of sequence 20; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned regions. Expected output: One file with the sequence of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-008
AbL-009	Input: AbL-009.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the regions a) 100’000’000 to 200’000’000 (included) of the sequence 18; b) 120’000’000 to 130’000’020 (included) of the sequence 21 Expected output: none Criteria: The test item shall generate no output
AbL-010	Input: AbL-010.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 0 contained in the file and with signature matching the string “AAAAAAAA”. Expected output: One file with the sequence of data units identified by the requested signature. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-010
AbL-011	Input: AbL-011.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 0 contained in the file and with signature matching either the string “AAAAAAAA” or the string “TTTTAAAA”. Expected output: One file with the sequence of data units identified by the requested signatures. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-011

Table 3 *(continued)*

Test item	Test procedure
AbL-012	Input: AbL-012.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the data units of the dataset with ID equal to 0 contained in the file and with signature matching the string “YYYYYYYY”. Expected output: One file with the sequence of data units identified by the requested signature. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-012
AbL-013	Input: AbL-013.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains 3 datasets in a dataset group, one is a reference); extract the data units of the dataset with Label matching the string “CHD5”, then matching the string “LCK”, then matching the string “NGF” on dataset with ID equal to 2 only; identify the reference (compressed as another dataset in the file) and retrieve at least the minimum amount of its data units necessary to decode the above mentioned labelled regions. Expected output: One file with the sequence of data units identified by the requested labels. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-013
AbL-014	Input: AbL-014.mgg (ISO/IEC 23092-1 file) Steps: Open the file; extract the dataset with ID equal to 1 contained in the file, which has no MIT; identify the reference related to the dataset and retrieve it completely Expected output: One file with the sequence of data units composing the dataset and one file with the related compressed reference (in the same order as in the file) Criteria: The data units of the dataset and the compressed reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-014
AbL-015	Input: AbL-015.mgg (ISO/IEC 23092-1 file) Steps: Open the file, which has no MIT; parse the file AU headers and extract the data units of the dataset with ID equal to 1 and with covered region matching, partially or completely, the region 5000000 to 10000000 (included) of the sequence 1; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned region. Expected output: One file with the sequence of data units covering the region of the dataset (in the same order as in the file) and one file with the required data units of the related reference. Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-015

Table 3 *(continued)*

Test item	Test procedure
AbL-016	Input: AbL-016.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains 4 datasets in a dataset group, 2 of them are references); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 10'000'000 to 20'000'000 (included) of the sequence 1; extract the class M-only data units of the dataset with ID equal to 2 contained in the file and with covered region matching, partially or completely, the region 20'000'000 to 30'000'020 (included) of the sequence 3; identify the references (compressed as other datasets in the file) related to the two datasets and retrieve at least the minimum amount of their data units necessary to decode the above mentioned regions. Expected output: For each dataset: one file for each of the sequences of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the datasets and of the related references, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-016
AbL-017	Input: AbL-017.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 1); extract the dataset with ID equal to 2 contained in the file; identify the reference (FASTA) related to the dataset and retrieve it completely Expected output: One file with the sequence of data units composing the dataset (in the same order as in the MIT) and one file with the related complete raw reference Criteria: The data units of the dataset and the raw reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-017
AbL-018	Input: AbL-018.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 0); extract the dataset with ID equal to 2 contained in the file; identify the reference (FASTA) related to the dataset and retrieve it completely Expected output: One file with the sequence of data units composing the dataset (in the same order as in the MIT) and one file with the related complete raw reference Criteria:. The data units of the dataset and the raw reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-018
AbL-019	Input: AbL-019.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 1); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 5000000 to 10000000 (included) of the sequence 1; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned region. Expected output: One file with the sequence of data units covering the region of the dataset (in the same order as in the file) and one file with the required data units of the related reference. Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-019

Table 3 *(continued)*

Test item	Test procedure
AbL-020	Input: AbL-020.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 1); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the regions a) 10’000’000 to 20’000’000 (included) of the sequence 1; b) 20’000’000 to 30’000’020 (included) of the sequence 3; c) 0 to 1’500’000 (included) of sequence 17; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned regions. Expected output: One file with the sequence of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-020
AbL-021	Input: AbL-021.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains 3 datasets in a dataset group, DSC mode, OBF set to 1); extract the data units of the dataset with label matching the string “CHD5”, then matching the string “LCK”, then matching the string “NGF” on dataset with ID equal to 3 only. Expected output: One file with the sequence of data units identified by the requested labels. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-021
AbL-022	Input: AbL-022.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 0); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 5000000 to 10000000 (included) of the sequence 1; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve the minimum amount of its data units necessary to decode the above mentioned region. Expected output: One file with the sequence of data units covering the region of the dataset (in the same order as in the file) and one file with the required data units of the related reference. Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-022
AbL-023	Input: AbL-023.mgg (ISO/IEC 23092-1 file) Steps: Open the file (DSC mode, OBF set to 0); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the regions a) 10’000’000 to 20’000’000 (included) of the sequence 1; b) 20’000’000 to 30’000’020 (included) of the sequence 3; c) 0 to 1’500’000 (included) of sequence 17; identify the reference (compressed as another dataset in the file) related to the dataset and retrieve at least the minimum amount of its data units necessary to decode the above mentioned regions. Expected output: One file with the sequence of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the dataset and of the related reference, obtained as output files, shall be byte-by-byte identical to the reference output provided for test item AbL-023

Table 3 *(continued)*

Test item	Test procedure
AbL-024	Input: AbL-024.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains 3 Datasets in a Dataset Group, DSC mode, OBF set to 0); extract the data units of the dataset with Label matching the string “CHD5”, then matching the string “LCK”, then matching the string “NGF” on Dataset with ID equal to 3 only. Expected output: One file with the sequence of data units identified by the requested Labels. Criteria: The data units of the dataset obtained as output shall be byte-by-byte identical to the reference output provided for test item AbL-024
AbL-025	Input: AbL-025.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains 4 datasets in a dataset group, 2 of them are references, DSC mode, OBF set to 1); extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 10’000’000 to 20’000’000 (included) of the sequence 1; extract the data units of the dataset with ID equal to 2 contained in the file and with covered region matching, partially or completely, the region 20’000’000 to 30’000’020 (included) of the sequence 3; identify the references (compressed as other datasets in the file) related to the two datasets and retrieve at least the minimum amount of their data units necessary to decode the above mentioned regions. Expected output: For each dataset: one file for each of the sequences of data units covering the regions of the dataset (in the same order as in the file) and one file with the required data units of the related reference Criteria: The data units of the datasets and of the related references, obtained as outputs, shall be byte-by-byte identical to the reference output provided for test item AbL-025
AbL-026	Input: AbL-026.mgg (ISO/IEC 23092-1 file) Steps: Open the file (the file contains two datasets with ID 0 and 1); extract the dtmd box for dataset with ID 0; extract the data units of the dataset with ID equal to 1 contained in the file and with covered region matching, partially or completely, the region 10’000’000 to 20’000’000 (included) of the sequence “1”; extract the auin boxes of the selected region. Expected output: One file containing the value field of gen_info box with key dtmd for dataset with ID 0 and a file containing the concatenation of the value field of gen_info boxes with key auin contained in the region 10000000 to 20000000 (included) of the sequence “1” of dataset with ID 0. Criteria: The files obtained as output, shall be byte-by-byte identical to the reference output provided for test item AbL-026

4.5.3 Support tool for reference verification

A majority of the tests to assess the conformity of decoders to ISO/IEC 23092-1, which are described in subclause 4.5.2, require the retrieval of data units of the reference genome related to the data units of the aligned content and necessary to further decode it.

Different decoders may use different approaches to the retrieval of the reference related to a dataset. This document provides as reference output the full reference genome for every test requiring references genomes, with the specification of the minimum necessary range to pass the test. In order to simplify the verification of the retrieved reference genome, it further provides a support tool helping the assessment of the decoder under test. The tool, available in source code and as an executable for most common OS and distributions, can be invoked with the following command line:

```
diffReference <refOutput.rref> <output.rref> <startChr> <startPos> <endChr> <endPos>
```

where the first parameter is the file name of the reference output (complete reference), the second parameter is the reference file as retrieved by the decoder under test, and the following parameters are integer values to pass the range in terms of reference sequence and positions on which the match will be calculated. Reference files shall be in rawReference format as specified in ISO/IEC 23092-2.

5 ISO/IEC 23092-2 conformance

5.1 Definition of ISO/IEC 23092-2 conformance

5.1.1 Assumptions

In this document, the following assumptions are made in reference to ISO/IEC 23092-2:

The term 'bitstream' means ISO/IEC 23092-2 bitstream.

The term 'decoder' means ISO/IEC 23092-2 decoder, i.e. an implementation of the decoding process specified by ISO/IEC 23092-2. The output of a decoder is specified in ISO/IEC 23092-2:2020, Clause 16.

If any statement made in this clause accidentally contradicts a statement or requirement in ISO/IEC 23092-2, the text of ISO/IEC 23092-2 prevails.

The following subclauses specify the tests to verify conformity of bitstreams and decoders. Those tests make use of test data (bitstream test items and reference outputs) made available as specified in Clause 6 and of the reference software specified in ISO/IEC 23092-4, with source code available as described in ISO/IEC 23092-4.

5.1.2 Definition of ISO/IEC 23092-2 bitstream conformity

An ISO/IEC 23092-2 bitstream is a bitstream that conforms to the specification defined by ISO/IEC 23092-2.

A conformant bitstream shall meet all the requirements and implement all the restrictions in the syntax defined by ISO/IEC 23092-2.

Subclause 5.3 defines the test that a bitstream shall pass successfully in order to be claimed in conformity with this document.

5.1.3 Definition of ISO/IEC 23092-2 decoder conformity

An ISO/IEC 23092-2 decoder is an implementation of the process defined by the requirements of ISO/IEC 23092-2.

A conformant decoder shall meet all the requirements and implement all the restrictions in the syntax defined by ISO/IEC 23092-2.

Subclause 5.4 defines the test procedure and the tests that a decoder shall pass successfully in order to be claimed in conformity with ISO/IEC 23092-2.

A conformant decoder shall implement a decoding process that is equivalent to the one specified in ISO/IEC 23092-2 and meets all the general requirements defined in ISO/IEC 23092-2.

A decoder that fails one or a part of the tests defined by subclause 5.4 shall not claim conformity to ISO/IEC 23092-2.

In the following subclauses the term 'reference decoder' means the reference software decoder (ISO/IEC 23092-4), which is the decoder that implements the above-mentioned decoding process.

Fundamental requirement areas for decoders and their mapping to functionality under test are listed in Table 4.

5.2 Requirements and functionality under test

Table 4 — Requirement areas for ISO/IEC 23092-2

Requirement area	Functionality
Parameter set	Decoder configuration
Read names	Compression of read names
Unmapped sequences of nucleotides	Dataset type equal to 0
Advanced coding of unmapped sequences of nucleotides	Dataset type equal to 0 with computed references
Alphabets	IUPAC, non-IUPAC alphabets (incl. RNA)
Reference-less data coding of aligned content	Reference-less data coding of mapped and unmapped segments, order of segments, substitutions, indels, clips
Reference-based data coding of aligned content	Reference-based data coding of mapped and unmapped segments, single and paired-end reads, order of segments, substitutions, indels, clips (incl. reference padding)
Coding of alignment accessory information	Flags, mapping quality, read groups
Coding of variable length content	Variable read length for unmapped and mapped content
Advanced coding of aligned content	Reference transformation
Entropy coding and CABAC	Different binarizations, transform sub-sequences
Quality values	Unquantized and quantized compression, alphabets
References	Reference-less and reference-based compression of references
Multiple alignments	Multiple and secondary alignments
Coding of splicing information	Alignments with splices

5.3 Procedure to test bitstream conformity

The ISO/IEC 23092 series reference software (ISO/IEC 23092-4) contains the source code of a software decoder that checks that a bitstream implements properly the specification defined in ISO/IEC 23092-2.

A bitstream that claims conformity with ISO/IEC 23092-2 shall pass the following test:

When processed by the reference software, the bitstream shall not cause errors or non-conformity messages. This test shall be run only on bitstreams that are known to be free of errors introduced by transmission.

To verify the correctness of a bitstream, it is necessary to parse the entire bitstream and to decode all the syntactic elements and values derived from those syntactic elements used by the decoding process specified in ISO/IEC 23092-2. This verification does not necessarily perform all stages of the decoding process in ISO/IEC 23092-2 (e.g. generation of MPEG-G records as specified in ISO/IEC 23092-2:2020, 13.2).

5.4 Procedure to test decoder conformity

This document contains standard test bitstreams in digital form; it also contains the standard reference output of each test bitstream as generated by the reference software (ISO/IEC 23092-4).

A decoder that claims conformity with ISO/IEC 23092-2 shall pass the following tests:

When processed by the decoder under test, each standard test bitstream contained in this document and associated to ISO/IEC 23092-2 shall generate an output byte-per-byte identical to the corresponding reference output.

To verify the conformity of the decoder, it is necessary to decode all the standard test items associated to ISO/IEC 23092-2 and to check the identity of all the resulting records or raw references. It may not be possible to perform this type of test with a production decoder; in that case, the conformity must be assessed by the implementer during the design and development phase.

This document provides, in electronic form, a shell script, running on Linux OS or compatible terminals, to automate the whole test and verification process for the decoder conformity of the reference decoder (ISO/IEC 23092-4).

5.5 Test items for ISO/IEC 23092-2 conformance

For convenience, given the wide functionality, test items associated to ISO/IEC 23092-2 are subdivided in four sets.

5.5.1 Set I: Genome sequencing data with single alignment

5.5.1.1 Test descriptions

ISO/IEC 23092 supports both reference-less and reference-based compression of genome sequencing content. Reference-less compression applies to both mapped and unmapped content (in this case a computed reference may be used). Reference-based compression of a reference genome only applies to aligned content.

Table 5 describes the test items for Set I of ISO/IEC 23092-2 conformance. The requested reference genome is indicated below the test item identification (in parentheses) where needed.

Table 5 — Test items for single-alignment genome sequencing

Test Item	Description	ISO/IEC 23092-2 content coverage	Functionality under test
Gen-001	Simple unmapped content with multiple inherited parameter sets, every AU or small subsets of AUs referring to the different sets	Subclause 7.3 Subclause 7.3.2 Clause 8	Parent-child inheritance, several different parameter sets to maintain
Gen-002	Unmapped content, paired-end reads (constant length), no computed reference	Subclause 10.4.7	Sequence coding (ureads descriptor only)
Gen-003	Unmapped content, paired-end reads (constant length), no computed reference, paired coding	Subclause 10.4.7 Subclause 10.4.9	Sequence coding (ureads descriptor only), pairing for class U
Gen-004	Unmapped content, single reads (variable short length), no computed reference	Subclause 10.4.7 Subclause 10.4.8	Variable read length, sequence coding
Gen-005	Unmapped content, single reads (variable long length), no computed reference	Subclause 10.4.7 Subclause 10.4.8	Variable read length, sequence coding
Gen-006	Unmapped content, paired-end reads (constant length), computed reference algorithm 2	Subclause 10.2 Subclause 11.3 Subclause 11.3.3	Push-in computed reference with constant read length
Gen-007	Unmapped content, single reads, variable length, computed reference algorithm 2	Subclause 10.2 Subclause 10.4.8 Subclause 11.3 Subclause 11.3.3	Push-in computed reference with variable read length
Gen-009	Unmapped content, paired-end reads (constant length), computed reference algorithm 4	Subclause 10.2 Subclause 11.3 Subclause 11.3.5	Global assembly computed reference with pairing