INTERNATIONAL STANDARD ISO/IEC 23090-10:2022
TECHNICAL CORRIGENDUM 1

Published 2023-05

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Coded representation of immersive media — Part 10: Carriage of visual volumetric video-based coding data

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Représentation codée de média immersifs — Partie 10: Transport de données de codage basé sur la vidéo volumétrique

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 23090-10:2022 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

ICS 35.040 **Ref. No. ISO/IEC 23090-10:2022/Cor.1:2023(E)**

Published in Switzerland

1 v3c_unit and nal_unit types

In the syntax of `V3CDecoderConfigurationRecord` the `v3c_parameter_set` is using the `v3c_unit` type which is defined in ISO/IEC 23090-5 but not in ISO/IEC 23090-10.

```
v3c_unit v3c_parameter_set(v3c_parameter_set_length);
```

Even worse is the fact that while the syntax defines the parameter set to be a `v3c_unit`, the semantics section only refers to a V3C Unit payload:

> `v3c_parameter_set` is a V3C unit payload for V3C unit of type `V3C_VPS`, as defined in ISO/IEC 23090-5.

This is clearly a mismatch between semantics and the syntax, because ISO/IEC 23090-5 defines the `v3c_unit`, as a `v3c_unit_header + v3c_unit_payload`.

Similarly, `nal_unit` is also defined in ISO/IEC 23090-5 and should be replaced by generic `bit(8)` type. It is also proposed to remove unnecessary inline comments.

1.1 Proposed corrigendum

In 7.2.1.2, replace:

```
aligned(8) class V3CDecoderConfigurationRecord {
   // version 0
   unsigned int(3) unit_size_precision_bytes_minus1;
   unsigned int(5) num_of_v3c_parameter_sets;
   for (int i=0; i < num_of_v3c_parameter_sets; i++) {
      unsigned int(16) v3c_parameter_set_length;
      // v3c_unit() as defined in ISO/IEC 23090-5
      v3c_unit v3c_parameter_set(v3c_parameter_set_length);
   }
   unsigned int(8) num_of_setup_unit_arrays;
   for (int j=0; j < num_of_setup_unit_arrays; j++) {
      unsigned int(1) array_completeness;
      bit(1) reserved = 0;
      unsigned int(6) nal_unit_type;
      unsigned int(8) num_nal_units;
      for (int i=0; i < num_nal_units; i++) {
         unsigned int(16) setup_unit_length;
         // nal_unit(size) as defined in ISO/IEC 23090-5
      nal_unit setup_unit(setup_unit_length);

    }

  }

  // additional fields

}
```

with:

```
aligned(8) class V3CDecoderConfigurationRecord(int version) {
  if(version == 0){
    unsigned int(3) unit_size_precision_bytes_minus1;
    unsigned int(5) num_of_v3c_parameter_sets;
    for (int i=0; i < num_of_v3c_parameter_sets; i++) {
      unsigned int(16) v3c_parameter_set_length;
      bit(8) v3c_parameter_set[v3c_parameter_set_length];
    }
    unsigned int(8) num_of_setup_unit_arrays;
    for (int j=0; j < num_of_setup_unit_arrays; j++) {
      unsigned int(1) array_completeness;
      bit(1) reserved = 0;
      unsigned int(6) nal_unit_type;
      unsigned int(8) num_nal_units;
      for (int i=0; i < num_nal_units; i++) {
        unsigned int(16) setup_unit_length;
        bit(8) setup_unit[setup_unit_length];
      }
    }
  }
}
```

In 7.2.1.3, replace:

v3c_parameter_set_length indicates the size, in bytes, of the v3c_parameter_set field.

v3c_parameter_set is a V3C unit payload for V3C unit of type V3C_VPS, as defined in ISO/IEC 23090-5.

...

setup_unit_length indicates the size, in bytes, of the setup_unit field. The length field includes the size of both the NAL unit header and the NAL unit payload but does not include the length field itself.

setup_unit contains a NAL unit according to related nal_unit_type. When present in setup_unit, NAL_PREFIX_ESEI, NAL_PREFIX_NSEI, NAL_SUFFIX_ESEI, or NAL_SUFFIX_NSEI contain SEI messages of a 'declarative' nature, that is, those that provide information about the stream as a whole. An example of such an SEI could be a user-data SEI.

with:

v3c_parameter_set_length indicates the size, in bytes, of the v3c_parameter_set array. The signalled value shall not be equal to 0.

v3c_parameter_set is an array of data containing the entire v3c_unit of the type V3C_VPS, as defined in ISO/IEC 23090-5.

...

setup_unit_length indicates the size, in bytes, of the setup_unit array~~.~~ The signalled value shall not be equal to 0.

setup_unit is an array of data containing the entire `nal_unit` as defined in ISO/IEC 23090-5. The contained NAL unit shall be of the same type as specified by `nal_unit_type`. When present in `setup_unit`, `NAL_PREFIX_ESEI`, `NAL_PREFIX_NSEI`, `NAL_SUFFIX_ESEI`, or `NAL_SUFFIX_NSEI` contain SEI messages of a 'declarative' nature, that is, those that provide information about the stream as a whole. An example of such an SEI could be a user-data SEI.

In 7.2.2.2, replace:

```
class V3CConfigurationBox extends FullBox('v3cC', version = 0, 0) {

    V3CDecoderConfigurationRecord();

}
```

with:

```
class V3CConfigurationBox extends FullBox('v3cC', version = 0, 0) {
   V3CDecoderConfigurationRecord v3c_config(version);
}
```

In 7.2.2.3, replace:

`V3CDecoderConfigurationRecord` is defined in subclause 7.2.1

with:

`v3c_config` is an instance of `V3CDecoderConfigurationRecord` as defined in subclause 7.2.1.

2 sample_stream_nal_unit type definition

In the context of ISO/IEC 23090-10 the `sample_stream_nal_unit` type is undefined. The following syntax of `V3CAtlasSample` addresses the problem.

2.1 Proposed corrigendum

In 7.4.4.2, replace:

```
aligned(8) class V3CAtlasSample {

   // sample_size value is the size of the sample from the SampleSizeBox

   for (int i=0; i < sample_size; ) {

      sample_stream_nal_unit ss_nal_unit;   // as defined in ISO/IEC 23090-5

      i += ss_nal_unit.ssnu_nal_unit_size +

            V3CDecoderConfigurationRecord.unit_size_precision_bytes_minus1 + 1;

   }

}
```

with:

```
aligned(8) class V3CAtlasSample {
   // sample_size value is the size of the sample from the SampleSizeBox
   for (int i=0; i < sample_size; ) {
      unsigned int((v3c_config.unit_size_precision_bytes_minus1 + 1)*8) nal_size;
      bit(8) ss_nal_unit[nal_size];
      i += nal_size + v3c_config.unit_size_precision_bytes_minus1 + 1;
   }
}
```