OSVP Clip Documentation

Introduction

The OSVP Clip (clip) is a collection of metadata parameters sampled over a specified duration. Each parameter is either:

- static: the parameter has at constant value over the duration of the clip
- dynamic: the parameter is sampled at regular intervals over the duration of the clip

Each parameter is identified by a unique name. It also has a general description as well as a specific set of constraints.

The OSVP Frame (frame) is a collection of metadata parameters that is dynamic and has a synchronous relationship with a video frame. In an OSVP environment this describes live camera position ('tracking') and lens data.

Clip Parameters

`activeSensorPhysicalDimensions`

Description

Height and width of the active area of the camera sensor in microns

Units

millimeter

Sampling

Static

Constraints

The height and width shall be each be real non-negative numbers.

`activeSensorResolution`

Description

Photosite resolution of the active area of the camera sensor in pixels

Units

pixel

Sampling

Static

Constraints

The height and width shall be each be an integer in the range [0..2,147,483,647].

`anamorphicSqueeze`

Description

Nominal ratio of height to width of the image of an axis-aligned square captured by the camera sensor. It can be used to de-squeeze images but is not however an exact number over the entire captured area due to a lens' intrinsic analog nature.

Units

n/a

Sampling

Static

Constraints

The parameter shall be a rational number whose numerator is in the range [0..2,147,483,647] and denominator in the range (0..4,294,967,295].

`firmwareVersion`

Description

Non-blank string identifying camera firmware version

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`label`

Description

Non-blank string containing user-determined camera identifier

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`make`

Description

Non-blank string naming camera manufacturer

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`model`

Description

Non-blank string identifying camera model

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`serialNumber`

Description

Non-blank string uniquely identifying the camera

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`captureFrameRate`

Description

Capture frame rate of the camera

Units

hertz

Sampling

Static

Constraints

The parameter shall be a rational number whose numerator is in the range [0..2,147,483,647] and denominator in the range (0..4,294,967,295].

`duration`

Description

Duration of the clip

Units

second

Sampling

Static

Constraints

The parameter shall be a rational number whose numerator is in the range [0..2,147,483,647] and denominator in the range (0..4,294,967,295].

`fdlLink`

Description

URN identifying the ASC Framing Decision List used by the camera.

Units

n/a

Sampling

Static

Constraints

The parameter shall be a UUID URN as specified in IETF RFC 4122. Only lowercase characters shall be used. Example: `f81d4fae-7dec-11d0-a765-00a0c91e6bf6`

`globalStage`

Description

Position of stage origin in global ENU and geodetic coordinates (E, N, U, lat0, lon0, h0). Note this may be dynamic if the stage is inside a moving vehicle.

Units

meter

Sampling

Regular

Constraints

```
Each field in the GlobalPosition shall be a real number
```

`isoSpeed`

Description

Arithmetic ISO scale as defined in ISO 12232

Units

n/a

Sampling

Static

Constraints

The parameter shall be a integer in the range (1..4,294,967,295].

`custom`

Description

This list provides optional additonal custom coefficients that can extend the existing lens model. The meaning of and how these characeristics are to be applied to a virtual camera would require negotiation between a particular producer and consumer.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a tuple of items of the class itemClass. The tuple can be empty

`distortionIsProjection`

Description

Indicator that the OpenLensIO distortion model is the Projection Characterization, not the Field-Of-View Characterization. This is primarily relevant when storing overscan values, not in transmission as the overscan should be calculated by the consumer.

Units

n/a

Sampling

Static

Constraints

The parameter shall be a boolean.

`distortionOffset`

Description

Offset in x and y of the centre of distortion of the virtual camera

Units

millimeter

Sampling

Regular

Constraints

X and Y centre shift shall each be real numbers.

`distortionOverscan`

Description

Overscan factor on lens distortion. This is primarily relevant when storing overscan values, not in transmission as the overscan should be calculated by the consumer.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a real number >= 1.

`distortionOverscanMax`

Description

Static maximum overscan factor on lens distortion. This is primarily relevant when storing overscan values, not in transmission as the overscan should be calculated by the consumer.

Units

n/a

Sampling

Static

Constraints

The parameter shall be a real number >= 1.

`distortion`

Description

A list of Distortion objects that each define the coefficients for calculating the distortion characteristics of a lens comprising radial distortion coefficients of the spherical distortion (k1-N) and the tangential distortion (p1-

N). An optional key 'model' can be used that describes the distortion model. The default is Brown-Conrady D-U (that maps Distorted to Undistorted coordinates).

Units

n/a

Sampling

Regular

Constraints

The list shall contain at least one Distortion object, and in each object the radial and tangential coefficients shall each be real numbers.

`encoders`

Description

Normalised real numbers (0-1) for focus, iris and zoom. Encoders are represented in this way (as opposed to raw integer values) to ensure values remain independent of encoder resolution, mininum and maximum (at an acceptable loss of precision). These values are only relevant in lenses with end-stops that demarcate the 0 and 1 range. Value should be provided in the following directions (if known): Focus: 0=infinite 1=closest Iris: 0=open 1=closed Zoom: 0=wide angle 1=telephoto

Units

n/a

Sampling

Regular

Constraints

```
The parameter shall contain at least one normalised values (0..1) for the FIZ encoders.
```

`entrancePupilOffset`

Description

Offset of the entrance pupil relative to the nominal imaging plane (positive if the entrance pupil is located on the side of the nominal imaging plane that is towards the object, and negative otherwise). Measured in meters as in a render engine it is often applied in the virtual camera's transform chain.

Units

meter

Sampling

Regular

Constraints

The parameter shall be a real number.

`exposureFalloff`

Description

Coefficients for calculating the exposure fall-off (vignetting) of a lens

Units

n/a

Sampling

Regular

Constraints

The coefficients shall each be real numbers.

`fStop`

Description

The linear f-number of the lens, equal to the focal length divided by the diameter of the entrance pupil.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a non-negative real number.

`firmwareVersion`

Description

Non-blank string identifying lens firmware version

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`focalLength`

Description

Focal length of the lens.

Units

millimeter

Sampling

Regular

Constraints

The parameter shall be a non-negative real number.

`focusDistance`

Description

Focus distance/position of the lens

Units

meter

Sampling

Regular

Constraints

The parameter shall be a non-negative real number.

`make`

Description

Non-blank string naming lens manufacturer

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`model`

Description

Non-blank string identifying lens model

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`nominalFocalLength`

Description

Nominal focal length of the lens. The number printed on the side of a prime lens, e.g. 50 mm, and undefined in the case of a zoom lens.

Units

millimeter

Sampling

Static

Constraints

The parameter shall be a non-negative real number.

`projectionOffset`

Description

Offset in x and y of the centre of perspective projection of the virtual camera

Units

millimeter

Sampling

Regular

Constraints

X and Y projection offset shall each be real numbers.

`rawEncoders`

Description

Raw encoder values for focus, iris and zoom. These values are dependent on encoder resolution and before any homing / ranging has taken place.

Units

n/a

Sampling

Regular

Constraints

```
The parameter shall contain at least one integer value for the FIZ encoders.
```

`serialNumber`

Description

Non-blank string uniquely identifying the lens

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`tStop`

Description

Linear t-number of the lens, equal to the F-number of the lens divided by the square root of the transmittance of the lens.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a non-negative real number.

`undistortionOverscan`

Description

Overscan factor on lens undistortion. This is primarily relevant when storing overscan values, not in transmission as the overscan should be calculated by the consumer.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a real number >= 1.

`undistortionOverscanMax`

Description

Static maximum overscan factor on lens undistortion. This is primarily relevant when storing overscan values, not in transmission as the overscan should be calculated by the consumer.

Units

n/a

Sampling

Static

Constraints

The parameter shall be a real number >= 1.

`protocol`

Description

Name of the protocol in which the sample is being employed, and version of that protocol

Units

n/a

Sampling

Regular

Constraints

Protocol name is nonblank string; protocol version is basic x.y.z semantic versioning string

`relatedSampleIds`

Description

List of sampleId properties of samples related to this sample. The existence of a sample with a given sampleId is not guaranteed.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a tuple of items of the class itemClass. The tuple can be empty

`sampleId`

Description

URN serving as unique identifier of the sample in which data is being transported.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a UUID URN as specified in IETF RFC 4122. Only lowercase characters shall be used. Example: `f81d4fae-7dec-11d0-a765-00a0c91e6bf6`

`shutterAngle`

Description

Shutter speed as a fraction of the capture frame rate. The shutter speed (in units of 1/s) is equal to the value of the parameter divided by 360 times the capture frame rate.

Units

degree

Sampling

Static

Constraints

The parameter shall be a real number in the range (0..360].

`sourceId`

Description

URN serving as unique identifier of the source from which data is being transported.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a UUID URN as specified in IETF RFC 4122. Only lowercase characters shall be used. Example: `f81d4fae-7dec-11d0-a765-00a0c91e6bf6`

`sourceNumber`

Description

Number that identifies the index of the stream from a source from which data is being transported. This is most important in the case where a source is producing multiple streams of samples.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a integer in the range (0..4,294,967,295].

`mode`

Description

Enumerated value indicating whether the sample transport mechanism provides inherent ('external') timing, or whether the transport mechanism lacks inherent timing and so the sample must contain a PTP timestamp itself ('internal') to carry timing information.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be one of the allowed values.

`recordedTimestamp`

Description

PTP timestamp of the data recording instant, provided for convenience during playback of e.g. pre-recorded tracking data. The timestamp comprises a 48-bit unsigned integer (seconds), a 32-bit unsigned integer (nanoseconds)

Units

second

Sampling

Regular

Constraints

The parameter shall contain valid number of seconds, nanoseconds elapsed since the start of the epoch.

`sampleRate`

Description

Sample frame rate as a rational number. Drop frame rates such as 29.97 should be represented as e.g. 30000/1001. In a variable rate system this should is estimated from the last sample delta time.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a rational number whose numerator is in the range [0..2,147,483,647] and denominator in the range (0..4,294,967,295].

`sampleTimestamp`

Description

PTP timestamp of the data capture instant. Note this may differ from the packet's transmission PTP timestamp. The timestamp comprises a 48-bit unsigned integer (seconds), a 32-bit unsigned integer (nanoseconds)

Units

second

Sampling

Regular

Constraints

The parameter shall contain valid number of seconds, nanoseconds elapsed since the start of the epoch.

`sequenceNumber`

Description

Integer incrementing with each sample.

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a integer in the range (0..4,294,967,295].

`synchronization`

Description

Object describing how the tracking device is synchronized for this sample.

frequency: The frequency of a synchronization signal.This may differ from the sample frame rate for example in a genlocked tracking device. This is not required if the synchronization source is PTP or NTP. locked: Is the tracking device locked to the synchronization source offsets: Offsets in seconds between sync and sample. Critical for e.g. frame remapping, or when using different data sources for position/rotation and lens encoding present: Is the synchronization source present (a synchronization source can be present but not locked if frame rates differ for example) ptp: If the synchronization source is a PTP master, then this object contains:

- "master": The MAC address of the PTP master
- "offset": The timing offset in seconds from the sample timestamp to the PTP timestamp
- "domain": The PTP domain number source: The source of synchronization must be defined as one of the following:
- "genlock": The tracking device has an external black/burst or tri-level analog sync signal that is triggering the capture of tracking samples
- "videoIn": The tracking device has an external video signal that is triggering the capture of tracking samples
- "ptp": The tracking device is locked to a PTP master
- "ntp": The tracking device is locked to an NTP server

Units

n/a

Sampling

Regular

Constraints

```
The parameter shall contain the required valid fields.
```

`timecode`

Description

SMPTE timecode of the sample. Timecode is a standard for labeling individual frames of data in media systems and is useful for inter-frame synchronization.

- format.frameRate: The frame rate as a rational number. Drop frame rates such as 29.97 should be represented as e.g. 30000/1001. The timecode frame rate may differ from the sample frequency.

Units

n/a

Sampling

Regular

Constraints

The parameter shall contain a valid format and hours, minutes, seconds and frames with appropriate min/max values.

`firmwareVersion`

Description

Non-blank string identifying tracking device firmware version

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`make`

Description

Non-blank string naming tracking device manufacturer

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`model`

Description

Non-blank string identifying tracking device model

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`notes`

Description

Non-blank string containing notes about tracking system

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`recording`

Description

Boolean indicating whether tracking system is recording data

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a boolean.

`serialNumber`

Description

Non-blank string uniquely identifying the tracking device

Units

n/a

Sampling

Static

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`slate`

Description

Non-blank string describing the recording slate

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`status`

Description

Non-blank string describing status of tracking system

Units

n/a

Sampling

Regular

Constraints

The parameter shall be a Unicode string betwee 0 and 1023 codepoints.

`transforms`

Description

A list of transforms. Transforms are composed in order with the last in the list representing the X,Y,Z in meters of camera sensor relative to stage origin. The Z axis points upwards and the coordinate system is right-handed. Y points in the forward camera direction (when pan, tilt and roll are zero). For example in an LED volume Y would point towards the centre of the LED wall and so X would point to camera-right. Rotation expressed as euler angles in degrees of the camera sensor relative to stage origin Rotations are intrinsic and are measured around the axes ZXY, commonly referred to as [pan, tilt, roll] Notes on Euler angles: Euler angles are human readable and unlike quarternions, provide the ability for cycles (with angles >360 or <0 degrees). Where a tracking system is providing the pose of a virtual camera, gimbal lock does not present the physical challenges of a robotic system. Conversion to and from quarternions is trivial with an acceptable loss of precision.

Units

meter / degree

Sampling

Regular

Constraints

Each component of each transform shall contain Real numbers.

Reader coverage

The following table indicates the camera parameters supported by each of the readers.

Reader	activeSensorPhysicalDimensions	activeSensorResolution	anamorphicSqueeze	
RED	+		+	
ARRI	+		+	
Venice	+		+	
Canon			+	

Clip JSON Schema

```
"$id": "https://opentrackio.org/schema.json",
"$schema": "https://json-schema.org/draft/2020-12/schema",
"type": "object",
"properties": {
  "static": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "camera": {
        "type": "object",
        "additionalProperties": false,
        "properties": {
```

```
            "activeSensorPhysicalDimensions": {
              "type": "object",
              "additionalProperties": false,
              "required": [
                "height",
                "width"
              ],
              "properties": {
                "height": {
                  "type": "number",
                  "minimum": 0.0
                },
                "width": {
                  "type": "number",
                  "minimum": 0.0
                }
              },
              "description": "Height and width of the active area of the camera sensor in micro
              "units": "millimeter"
            },
            "activeSensorResolution": {
              "type": "object",
              "additionalProperties": false,
              "required": [
                "height",
                "width"
              ],
              "properties": {
                "height": {
                  "type": "integer",
                  "minimum": 0,
                  "maximum": 2147483647
                },
                "width": {
                  "type": "integer",
                  "minimum": 0,
                  "maximum": 2147483647
                }
              },
              "description": "Photosite resolution of the active area of the camera sensor in p
              "units": "pixel"
            },
            "anamorphicSqueeze": {
              "type": "object",
              "properties": {
                "num": {
                  "type": "integer",
                  "minimum": 1,
                  "maximum": 2147483647
                },
                "denom": {
                  "type": "integer",
                  "minimum": 1,
                  "maximum": 4294967295
                }
              },
              "required": [
```

```
        "num",
        "denom"
      ],
      "additionalProperties": false,
      "description": "Nominal ratio of height to width of the image of an axis-aligned
    },
    "firmwareVersion": {
      "type": "string",
      "minLength": 1,
      "maxLength": 1023,
      "description": "Non-blank string identifying camera firmware version"
    },
    "label": {
      "type": "string",
      "minLength": 1,
      "maxLength": 1023,
      "description": "Non-blank string containing user-determined camera identifier"
    },
    "make": {
      "type": "string",
      "minLength": 1,
      "maxLength": 1023,
      "description": "Non-blank string naming camera manufacturer"
    },
    "model": {
      "type": "string",
      "minLength": 1,
      "maxLength": 1023,
      "description": "Non-blank string identifying camera model"
    },
    "serialNumber": {
      "type": "string",
      "minLength": 1,
      "maxLength": 1023,
      "description": "Non-blank string uniquely identifying the camera"
    },
    "captureFrameRate": {
      "type": "object",
      "properties": {
        "num": {
          "type": "integer",
          "minimum": 1,
          "maximum": 2147483647
        },
        "denom": {
          "type": "integer",
          "minimum": 1,
          "maximum": 4294967295
        }
      },
      "required": [
        "num",
        "denom"
      ],
      "additionalProperties": false,
      "description": "Capture frame rate of the camera",
      "units": "hertz"
```

```
        },
        "fdlLink": {
          "type": "string",
          "pattern": "^urn:uuid:[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12
          "description": "URN identifying the ASC Framing Decision List used by the camera.
        },
        "isoSpeed": {
          "type": "integer",
          "minimum": 1,
          "maximum": 4294967295,
          "description": "Arithmetic ISO scale as defined in ISO 12232"
        },
        "shutterAngle": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 360.0,
          "description": "Shutter speed as a fraction of the capture frame rate. The shutte
          "units": "degree"
        }
      }
    },
    "duration": {
      "type": "object",
      "properties": {
        "num": {
          "type": "integer",
          "minimum": 1,
          "maximum": 2147483647
        },
        "denom": {
          "type": "integer",
          "minimum": 1,
          "maximum": 4294967295
        }
      },
      "required": [
        "num",
        "denom"
      ],
      "additionalProperties": false,
      "description": "Duration of the clip",
      "units": "second"
    },
    "lens": {
      "type": "object",
      "additionalProperties": false,
      "properties": {
        "distortionIsProjection": {
          "type": "boolean",
          "description": "Indicator that the OpenLensIO distortion model is the Projection
        },
        "distortionOverscanMax": {
          "type": "number",
          "minimum": 1.0,
          "description": "Static maximum overscan factor on lens distortion. This is primar
        },
        "firmwareVersion": {
```

```
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string identifying lens firmware version"
          },
          "make": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string naming lens manufacturer"
          },
          "model": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string identifying lens model"
          },
          "nominalFocalLength": {
            "type": "number",
            "minimum": 0.0,
            "description": "Nominal focal length of the lens. The number printed on the side
            "units": "millimeter"
          },
          "serialNumber": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string uniquely identifying the lens"
          },
          "undistortionOverscanMax": {
            "type": "number",
            "minimum": 1.0,
            "description": "Static maximum overscan factor on lens undistortion. This is prim
          }
        }
      },
      "tracker": {
        "type": "object",
        "additionalProperties": false,
        "properties": {
          "firmwareVersion": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string identifying tracking device firmware version"
          },
          "make": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string naming tracking device manufacturer"
          },
          "model": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string identifying tracking device model"
```

```
          },
          "serialNumber": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023,
            "description": "Non-blank string uniquely identifying the tracking device"
          }
        }
      }
    }
  },
  "globalStage": {
    "type": "object",
    "additionalProperties": false,
    "required": [
      "E",
      "N",
      "U",
      "lat0",
      "lon0",
      "h0"
    ],
    "properties": {
      "E": {
        "type": "number"
      },
      "N": {
        "type": "number"
      },
      "U": {
        "type": "number"
      },
      "lat0": {
        "type": "number"
      },
      "lon0": {
        "type": "number"
      },
      "h0": {
        "type": "number"
      }
    },
    "description": "Position of stage origin in global ENU and geodetic coordinates (E, N, U,
    "units": "meter"
  },
  "lens": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "custom": {
        "type": "array",
        "items": {
          "type": "number"
        },
        "description": "This list provides optional additonal custom coefficients that can  e
      },
      "distortionOffset": {
```

```
      "type": "object",
      "additionalProperties": false,
      "required": [
        "x",
        "y"
      ],
      "properties": {
        "x": {
          "type": "number"
        },
        "y": {
          "type": "number"
        }
      },
      "description": "Offset in x and y of the centre of distortion of the virtual camera "
      "units": "millimeter"
    },
    "distortionOverscan": {
      "type": "number",
      "minimum": 1.0,
      "description": "Overscan factor on lens distortion. This is primarily relevant when s
    },
    "distortion": {
      "type": "array",
      "minItems": 1,
      "items": {
        "type": "object",
        "additionalProperties": false,
        "required": [
          "radial"
        ],
        "properties": {
          "model": {
            "type": "string"
          },
          "radial": {
            "type": "array",
            "items": {
              "type": "number"
            },
            "minLength": 1
          },
          "tangential": {
            "type": "array",
            "items": {
              "type": "number"
            },
            "minLength": 1
          }
        }
      },
      "description": "A list of Distortion objects that each define the coefficients for ca
    },
    "encoders": {
      "type": "object",
      "additionalProperties": false,
      "properties": {
```

```
        "focus": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 1.0
        },
        "iris": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 1.0
        },
        "zoom": {
          "type": "number",
          "minimum": 0.0,
          "maximum": 1.0
        }
      },
      "anyOf": [
        {
          "required": [
            "focus"
          ]
        },
        {
          "required": [
            "iris"
          ]
        },
        {
          "required": [
            "zoom"
          ]
        }
      ],
      "description": " Normalised real numbers (0-1) for focus, iris and zoom. Encoders are
    },
    "entrancePupilOffset": {
      "type": "number",
      "description": "Offset of the entrance pupil relative to the nominal imaging plane (p
      "units": "meter"
    },
    "exposureFalloff": {
      "type": "object",
      "additionalProperties": false,
      "required": [
        "a1"
      ],
      "properties": {
        "a1": {
          "type": "number"
        },
        "a2": {
          "type": "number"
        },
        "a3": {
          "type": "number"
        }
      },
```

```
      "description": "Coefficients for calculating the exposure fall-off (vignetting) of a
    },
    "fStop": {
      "type": "number",
      "minimum": 0.0,
      "description": "The linear f-number of the lens, equal to the focal length divided by
    },
    "focalLength": {
      "type": "number",
      "minimum": 0.0,
      "description": "Focal length of the lens.",
      "units": "millimeter"
    },
    "focusDistance": {
      "type": "number",
      "minimum": 0.0,
      "description": "Focus distance/position of the lens",
      "units": "meter"
    },
    "projectionOffset": {
      "type": "object",
      "additionalProperties": false,
      "required": [
        "x",
        "y"
      ],
      "properties": {
        "x": {
          "type": "number"
        },
        "y": {
          "type": "number"
        }
      },
      "description": "Offset in x and y of the centre of perspective projection of the virt
      "units": "millimeter"
    },
    "rawEncoders": {
      "type": "object",
      "additionalProperties": false,
      "properties": {
        "focus": {
          "type": "integer",
          "minimum": 0
        },
        "iris": {
          "type": "integer",
          "minimum": 0
        },
        "zoom": {
          "type": "integer",
          "minimum": 0
        }
      },
      "anyOf": [
        {
          "required": [
```

```
              "focus"
            ]
          },
          {
            "required": [
              "iris"
            ]
          },
          {
            "required": [
              "zoom"
            ]
          }
        ],
        "description": " Raw encoder values for focus, iris and zoom. These values are depend
      },
      "tStop": {
        "type": "number",
        "minimum": 0.0,
        "description": "Linear t-number of the lens, equal to the F-number of the lens divide
      },
      "undistortionOverscan": {
        "type": "number",
        "minimum": 1.0,
        "description": "Overscan factor on lens undistortion. This is primarily relevant wher
      }
    }
  },
  "protocol": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "name": {
        "type": "string",
        "minLength": 1,
        "maxLength": 1023
      },
      "version": {
        "type": "array",
        "items": {
          "type": "integer",
          "minValue": 0,
          "maxValue": 9
        },
        "minItems": 3,
        "maxItems": 3
      }
    },
    "description": "Name of the protocol in which the sample is being employed, and version o
  },
  "relatedSampleIds": {
    "type": "array",
    "items": {
      "type": "string",
      "pattern": "^urn:uuid:[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$"
    },
    "description": "List of sampleId properties of samples related to this sample. The existe
```

```
    },
    "sampleId": {
      "type": "string",
      "pattern": "^urn:uuid:[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$",
      "description": "URN serving as unique identifier of the sample in which data is being tra
    },
    "sourceId": {
      "type": "string",
      "pattern": "^urn:uuid:[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}$",
      "description": "URN serving as unique identifier of the source from which data is being t
    },
    "sourceNumber": {
      "type": "integer",
      "minimum": 0,
      "maximum": 4294967295,
      "description": "Number that identifies the index of the stream from a source from which d
    },
    "timing": {
      "type": "object",
      "additionalProperties": false,
      "properties": {
        "mode": {
          "type": "string",
          "enum": [
            "internal",
            "external"
          ],
          "description": "Enumerated value indicating whether the sample transport mechanism
        },
        "recordedTimestamp": {
          "type": "object",
          "additionalProperties": false,
          "properties": {
            "seconds": {
              "type": "integer",
              "minimum": 0,
              "maximum": 281474976710655
            },
            "nanoseconds": {
              "type": "integer",
              "minimum": 0,
              "maximum": 4294967295
            }
          },
          "required": [
            "seconds",
            "nanoseconds"
          ],
          "description": " PTP timestamp of the data recording instant, provided for convenienc
          "units": "second"
        },
        "sampleRate": {
          "type": "object",
          "properties": {
            "num": {
              "type": "integer",
              "minimum": 1,
```

```
          "maximum": 2147483647
        },
        "denom": {
          "type": "integer",
          "minimum": 1,
          "maximum": 4294967295
        }
      },
      "required": [
        "num",
        "denom"
      ],
      "additionalProperties": false,
      "description": "Sample frame rate as a rational number. Drop frame rates such as 29.9
    },
    "sampleTimestamp": {
      "type": "object",
      "additionalProperties": false,
      "properties": {
        "seconds": {
          "type": "integer",
          "minimum": 0,
          "maximum": 281474976710655
        },
        "nanoseconds": {
          "type": "integer",
          "minimum": 0,
          "maximum": 4294967295
        }
      },
      "required": [
        "seconds",
        "nanoseconds"
      ],
      "description": "PTP timestamp of the data capture instant. Note this may differ   fro
      "units": "second"
    },
    "sequenceNumber": {
      "type": "integer",
      "minimum": 0,
      "maximum": 4294967295,
      "description": "Integer incrementing with each sample."
    },
    "synchronization": {
      "type": "object",
      "additionalProperties": false,
      "description": "Object describing how the tracking device is synchronized for this sa
      "properties": {
        "frequency": {
          "type": "object",
          "additionalProperties": false,
          "required": [
            "num",
            "denom"
          ],
          "properties": {
            "num": {
```

```
            "type": "integer",
            "minimum": 1,
            "maximum": 4294967295
          },
          "denom": {
            "type": "integer",
            "minimum": 1,
            "maximum": 4294967295
          }
        }
      },
      "locked": {
        "type": "boolean"
      },
      "offsets": {
        "type": "object",
        "additionalProperties": false,
        "properties": {
          "translation": {
            "type": "number"
          },
          "rotation": {
            "type": "number"
          },
          "lensEncoders": {
            "type": "number"
          }
        }
      },
      "present": {
        "type": "boolean"
      },
      "ptp": {
        "type": "object",
        "additionalProperties": false,
        "properties": {
          "master": {
            "type": "string",
            "pattern": "^([A-F0-9]{2}:){5}[A-F0-9]{2}$"
          },
          "offset": {
            "type": "number"
          },
          "domain": {
            "type": "integer",
            "minimum": 0,
            "maximum": 127
          }
        }
      },
      "source": {
        "type": "string",
        "enum": [
          "genlock",
          "videoIn",
          "ptp",
          "ntp"
```

```
            ]
          }
        },
        "required": [
          "locked",
          "source"
        ]
      },
      "timecode": {
        "type": "object",
        "additionalProperties": false,
        "required": [
          "hours",
          "minutes",
          "seconds",
          "frames",
          "format"
        ],
        "properties": {
          "hours": {
            "type": "integer",
            "minimum": 0,
            "maximum": 23
          },
          "minutes": {
            "type": "integer",
            "minimum": 0,
            "maximum": 59
          },
          "seconds": {
            "type": "integer",
            "minimum": 0,
            "maximum": 59
          },
          "frames": {
            "type": "integer",
            "minimum": 0,
            "maximum": 119
          },
          "format": {
            "type": "object",
            "description": "The timecode format is defined as a rational frame rate and - whe
            "required": [
              "frameRate"
            ],
            "additionalProperties": false,
            "properties": {
              "frameRate": {
                "type": "object",
                "additionalProperties": false,
                "required": [
                  "num",
                  "denom"
                ],
                "properties": {
                  "num": {
                    "type": "integer",
```

```
                    "minimum": 1,
                    "maximum": 4294967295
                  },
                  "denom": {
                    "type": "integer",
                    "minimum": 1,
                    "maximum": 4294967295
                  }
                }
              },
              "subFrame": {
                "type": "integer",
                "minimum": 0,
                "maximum": 4294967295
              }
            }
          }
        },
        "description": "SMPTE timecode of the sample. Timecode is a standard for labeling inc
      }
    }
  },
  "tracker": {
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "notes": {
        "type": "string",
        "minLength": 1,
        "maxLength": 1023,
        "description": "Non-blank string containing notes about tracking system"
      },
      "recording": {
        "type": "boolean",
        "description": "Boolean indicating whether tracking system is recording data"
      },
      "slate": {
        "type": "string",
        "minLength": 1,
        "maxLength": 1023,
        "description": "Non-blank string describing the recording slate"
      },
      "status": {
        "type": "string",
        "minLength": 1,
        "maxLength": 1023,
        "description": "Non-blank string describing status of tracking system"
      }
    }
  },
  "transforms": {
    "type": "array",
    "minItems": 1,
    "uniqueItems": false,
    "items": {
      "type": "object",
      "additionalProperties": false,
```

```
        "properties": {
          "translation": {
            "type": "object",
            "additionalProperties": false,
            "properties": {
              "x": {
                "type": "number"
              },
              "y": {
                "type": "number"
              },
              "z": {
                "type": "number"
              }
            },
            "units": "meter"
          },
          "rotation": {
            "type": "object",
            "additionalProperties": false,
            "properties": {
              "pan": {
                "type": "number"
              },
              "tilt": {
                "type": "number"
              },
              "roll": {
                "type": "number"
              }
            },
            "units": "degree"
          },
          "scale": {
            "type": "object",
            "additionalProperties": false,
            "properties": {
              "x": {
                "type": "number"
              },
              "y": {
                "type": "number"
              },
              "z": {
                "type": "number"
              }
            }
          },
          "id": {
            "type": "string",
            "minLength": 1,
            "maxLength": 1023
          }
        },
        "required": [
          "translation",
          "rotation"
```

```
        ]
      },
      "description": "A list of transforms. Transforms are composed in order with the last in t
      "units": "meter / degree"
    }
  }
}
```