



Moving Picture, Audio and Data Coding
by Artificial Intelligence
www.mpai.community

MPAI Technical Specification

Object and Scene Description (MPAI-OSD)

V1.4

WARNING

Use of the technologies described in this Technical Specification may infringe patents, copyrights, or intellectual property rights of MPAI Members or non-members.

MPAI and its Members accept no responsibility whatsoever for damages or liability, direct or consequential, which may result from the use of this Technical Specification.

Readers are invited to review **Error! Reference source not found.**[Notices and Disclaimers](#).

Technical Specification

Object and Scene Description (MPAI-OSD)

V1.4

1	Foreword	5
2	Introduction (informative).....	7
3	Scope	7
4	Definitions	8
5	References	10
5.1	Normative Reference.....	10
5.2	Informative References	11
6	AI Workflows	11
6.1	Technical Specifications	11
6.1.1	Television Media Analysis	11
7	AI Modules.....	20
7.1	Technical Specifications	20
7.1.1	Basic Scene Description	21
7.1.2	Scene Description.....	23
7.1.3	Event Description	24
7.1.4	Basic 3D Model Scene Description	26
7.1.5	3D Model Scene Description.....	27
7.1.6	3D Model Event Description.....	29
7.1.7	Basic Audio Scene Description.....	30
7.1.8	Audio Scene Descriptor.....	32
7.1.9	Audio Event Description	34
7.1.10	Basic Audio-Visual Scene Description.....	35
7.1.11	Audio-Visual Scene Description.....	38
7.1.12	Audio-Visual Event Description	41
7.1.13	Basic LiDAR Scene Description.....	42
7.1.14	LiDAR Scene Description	44
7.1.15	LiDAR Event Description	46
7.1.16	Basic Offline Map Scene Description.....	47
7.1.17	Offline Map Event Description	50
7.1.18	Basic RADAR Scene Description.....	51
7.1.19	RADAR Event Description	55
7.1.20	Basic Speech Scene Description	56
7.1.21	Speech Scene Description.....	58
7.1.22	Speech Event Description.....	59
7.1.23	Basic Ultrasound Scene Description.....	61
7.1.24	Ultrasound Scene Description	62
7.1.25	Ultrasound Event Description	64
7.1.26	Basic Visual Scene Description	65
7.1.27	Audio-Visual Alignment.....	70
7.1.28	Audio-Visual Scene Demultiplexing	73
7.1.29	Audio-Visual Scene Multiplexing.....	75
7.1.30	Direct Visual Identification	77
7.1.31	Television Splitting	79
7.1.32	Visual Change Detection	81

7.1.33	Visual Direction Identification	84
7.1.34	Visual Instance Identification	85
7.1.35	Visual Object Extraction	87
7.1.36	Visual Object Identification by Body	90
7.2	Reference Software	93
7.3	Performance Assessment	93
8	Data Types	94
8.1	Technical Specifications	94
8.1.1	Basic Object.....	96
8.1.2	Object.....	98
8.1.3	Basic Scene Descriptors.....	100
8.1.4	Scene Descriptors.....	101
8.1.5	Basic Scene Geometry	103
8.1.6	Scene Geometry	104
8.1.7	Event Descriptors	105
8.1.8	Basic 3D Model Object.....	106
8.1.9	3D Model Object.....	108
8.1.10	Basic 3D Model Scene Descriptors.....	109
8.1.11	3D Model Scene Descriptors	110
8.1.12	Basic 3D Model Scene Geometry	111
8.1.13	3D Model Scene Geometry	112
8.1.14	3D Model Event Descriptors	113
8.1.15	Basic Audio Object	114
8.1.16	Audio Object.....	116
8.1.17	Basic Audio Scene Descriptors	117
8.1.18	Audio Scene Descriptors	118
8.1.19	Basic Audio Scene Geometry.....	119
8.1.20	Audio Scene Geometry	120
8.1.21	Audio Event Descriptors.....	121
8.1.22	Basic Audio-Visual Object	122
8.1.23	Audio-Visual Object	123
8.1.24	Basic Audio-Visual Scene Descriptors	125
8.1.25	Audio-Visual Scene Descriptors	126
8.1.26	Basic Audio-Visual Scene Geometry	127
8.1.27	Audio-Visual Scene Geometry.....	128
8.1.28	Audio-Visual Event Descriptors	129
8.1.29	Basic LiDAR Object	130
8.1.30	LiDAR Object	131
8.1.31	Basic LiDAR Scene Descriptors	133
8.1.32	LiDAR Scene Descriptors	134
8.1.33	Basic LiDAR Scene Geometry.....	135
8.1.34	LiDAR Scene Geometry	136
8.1.35	LiDAR Event Descriptors.....	137
8.1.36	Basic Location Object.....	138
8.1.37	Location Object	139
8.1.38	Basic Offline Map Object.....	139
8.1.39	Offline Map Object	141
8.1.40	Basic Offline Map Scene Descriptors	142
8.1.41	Offline Map Scene Descriptors	143
8.1.42	Basic Offline Map Scene Geometry	144

8.1.43	Offline Map Scene Geometry.....	145
8.1.44	Offline Map Event Descriptors	146
8.1.45	Basic RADAR Object	147
8.1.46	RADAR Object	148
8.1.47	Basic RADAR Scene Descriptors	150
8.1.48	RADAR Scene Descriptors	151
8.1.49	RADAR Scene Geometry	153
8.1.50	RADAR Event Descriptors	154
8.1.51	Basic Speech Object.....	155
8.1.52	Speech Object.....	156
8.1.53	Basic Speech Scene Descriptors.....	157
8.1.54	Speech Scene Descriptors.....	158
8.1.55	Basic Speech Scene Geometry	160
8.1.56	Speech Scene Geometry	161
8.1.57	Speech Event Descriptors	162
8.1.58	Basic Text Object.....	162
8.1.59	Text Object	163
8.1.60	Simple Time	165
8.1.61	Time	166
8.1.62	Basic Ultrasound Object	166
8.1.63	Ultrasound Object	168
8.1.64	Basic Ultrasound Scene Descriptors	169
8.1.65	Ultrasound Scene Descriptors	170
8.1.66	Basic Ultrasound Scene Geometry.....	171
8.1.67	Ultrasound Scene Geometry	172
8.1.68	Ultrasound Event Descriptors.....	173
8.1.69	Basic Visual Object.....	174
8.1.70	Visual Object.....	176
8.1.71	Basic Visual Scene Descriptors.....	178
8.1.72	Visual Scene Descriptors	179
8.1.73	Basic Visual Scene Geometry	180
8.1.74	Visual Scene Geometry	181
8.1.75	Visual Event Descriptors	182
8.1.76	Annotation	183
8.1.77	Perceptible Entity	184
8.1.78	Audio Source	184
8.1.79	Point of View.....	186
8.1.80	Bounding Box.....	187
8.1.81	Position	188
8.1.82	Colour Object	189
8.1.83	Right Parallelepiped.....	190
8.1.84	Coordinates.....	191
8.1.85	Selector	191
8.1.86	Device Scene Geometry.....	192
8.1.87	Simple Location	193
8.1.88	Instance Identifier.....	194
8.1.89	Space-Time	195
8.1.90	Light Source	196
8.1.91	Spatial Attitude.....	197
8.1.92	Object Audio Characteristics.....	198

8.1.93	Texture Map	200
8.1.94	Orientation	201
8.1.95	Trajectory.....	202
8.1.96	Path.....	203
8.2	Conformance testing.....	204
8.3	Performance Assessment	204

1 Foreword

The international, unaffiliated, non-profit *Moving Picture, Audio, and Data Coding by Artificial Intelligence (MPAI)* organisation was established in September 2020 in the context of:

1. **Increasing** use of Artificial Intelligence (AI) technologies applied to a broad range of domains affecting millions of people
2. **Marginal** reliance on standards in the development of those AI applications
3. **Unprecedented** impact exerted by standards on the digital media industry affecting billions of people

believing that AI-based data coding standards will have a similar positive impact on the Information and Communication Technology industry.

The design principles of the MPAI organisation as established by the MPAI Statutes are the development of AI-based Data Coding standards in pursuit of the following policies:

1. Publish upfront clear Intellectual Property Rights licensing frameworks.
2. Adhere to a rigorous standard development process.
3. Be friendly to the AI context but, to the extent possible, remain agnostic to the technology thus allowing developers freedom in the selection of the more appropriate – AI or Data Processing – technologies for their needs.
4. Be attractive to different industries, end users, and regulators.
5. Address five standardisation areas:
 1. *Data Type*, a particular type of Data, e.g., Audio, Visual, Object, Scenes, and Descriptors with as clear semantics as possible.
 2. *Qualifier*, specialised Metadata conveying information on Sub-Types, Formats, and Attributes of a Data Type.
 3. *AI Module* (AIM), processing elements with identified functions and input/output Data Types.
 4. *AI Workflow* (AIW), MPAI-specified configurations of AIMs with identified functions and input/output Data Types.
 5. *AI Framework* (AIF), an environment enabling dynamic configuration, initialisation, execution, and control of AIWs.
6. Provide appropriate Governance of the ecosystem created by MPAI Technical Specifications enabling users to:
 1. *Operate* Reference Software Implementations of MPAI Technical Specifications provided together with Reference Software Specifications
 2. *Test* the conformance of an implementation with a Technical Specification using the Conformance Testing Specification.
 3. *Assess* the performance of an implementation of a Technical Specification using the Performance Assessment Specification.
 4. *Obtain* conforming implementations possibly with a performance assessment report from a trusted source through the MPAI Store.

MPAI operates on four solid pillars:

1. The [MPAI Patent Policy](#) specifies the MPAI standard development process and the Framework Licence development guidelines.

2. [Technical Specification: Artificial Intelligence Framework \(MPAI-AIF\) V2.1](#) specifies an environment enabling initialisation, dynamic configuration, and control of AI applications in the standard AI Framework environment depicted in Figure 1. An AI Framework can execute AI applications called AI Workflows (AIW) typically including interconnected AI Modules (AIM). MPAI-AIF supports small- and large-scale high-performance components and promotes solutions with improved explainability.

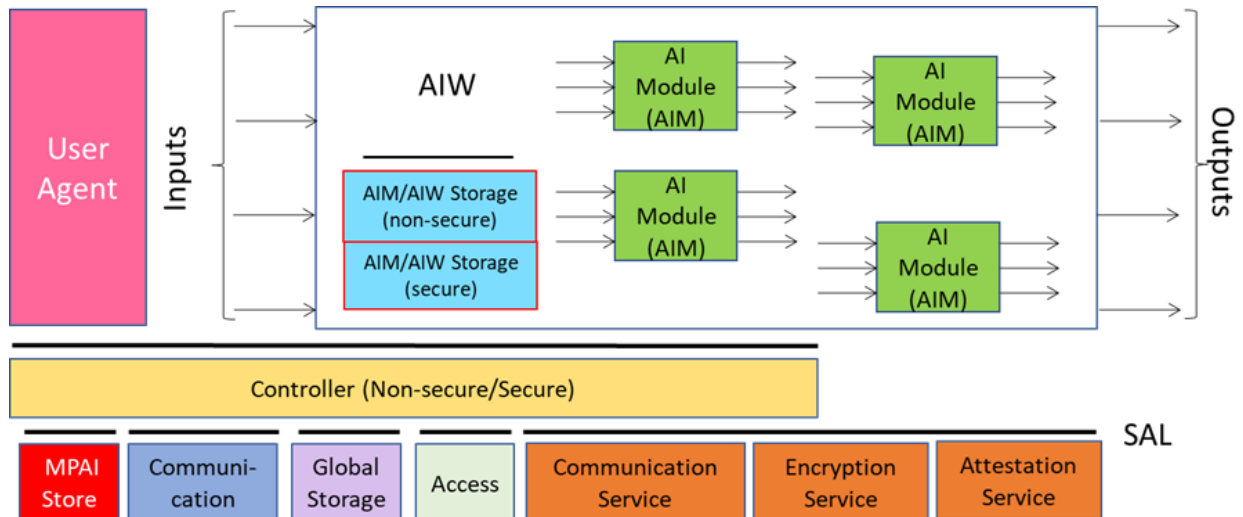


Figure 1 – The AI Framework (MPAI-AIF) V2 Reference Model

3. [Technical Specification: Data Types, Formats, and Attributes \(MPAI-TFA\) V1.4](#) specifies Qualifiers, a type of metadata supporting the operation of AIMs receiving data from other AIMs or from input data. Qualifiers convey information on Sub-Types (e.g., the type of colour), Formats (e.g., the type of compression and transport), and Attributes (e.g., semantic information in the Content). Although Qualifiers are human-readable, they are only intended to be used by AIMs. Therefore, Text, Speech, Audio, Visual, and other Data received by or exchanged between AIWs and AIMs should be interpreted as being composed of Content (Text, Speech, Audio, and Visual as appropriate) and associated Qualifiers. For instance, a Text Object is composed of Text Data and Text Qualifier. The specification of most MPAI Data Types reflects this point.
4. [Technical Specification: Governance of the MPAI Ecosystem \(MPAI-GME\) V2.0](#) defines the following elements:
 1. Standards, i.e., the ensemble of Technical Specifications, Reference Software, Conformance Testing, and Performance Assessment.
 2. Developers of MPAI-specified AIMs and Integrators of MPAI-specified AIWS (Implementers).
 3. MPAI Store in charge of making AIMs and AIWs submitted by Implementers available to Integrators and End Users.
 4. Performance Assessors, independent entities assessing the performance of implementations in terms of Reliability, Replicability, Robustness, and Fairness.
 5. End Users.

The interaction between and among actors of the MPAI Ecosystem are depicted in Figure 2.

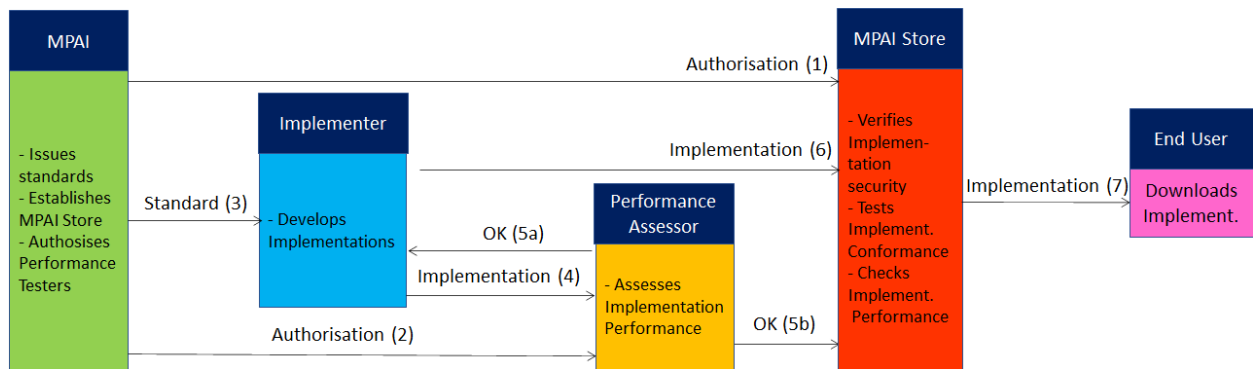


Figure 2 – The MPAI Ecosystem

2 Introduction (informative)

Several MPAI Technical Specifications need data types that refer to Objects and Scenes that are uni- and multimodal, refer to locations that may be in virtual spaces and include Space and Time information.

Technical Specification: Object and Scene Description (MPAI-OSD) V1.4 – in the following also called MPAI-OSD V1.34 or MPAI-OSD – has been developed to specify Time, Space, Object and Scene information for unified use across MPAI Technical Specifications and the specification to the Television Media Analysis (OSD-TMA) AI Workflow (AIW).

MPAI-OSD includes the complete OSD-TMA Reference Software Specification including the Open Source Software Implementation of the AIW and AIMs (released with BSD-3-clause licence). The Reference Software Implementation offers users the opportunity to access Implementations that are Conforming with the Technical Specification. The AIMs used by OSD-TMA are sourced by the MPAI-CAE, MPAI-MMC, MPAI-OSD, and MPAI-PAF Technical Specifications.

In all Chapters and Sections, Terms beginning with a capital letter are defined in [Table 1](#) if they are specific to MPAI-OSD. All MPAI-defined Terms are accessible [online](#). All Chapters, and Sections are Normative unless they are labelled as Informative.

3 Scope

Technical Specification: Object and Scenes Description (MPAI-OSD) V1.4 specifies:

1. Data Types and associated Qualifiers that are used to specify uni- and multi-modal Objects and Scenes in Virtual Environments with attributes of Space and Time.
2. The MPAI-OSD AI Modules and the Data Types.
3. The Television Media Analysis (OSD-TMA) AI Workflow using AI Modules and Data Types from MPAI-CAE, MPAI-MMC, and MPAI-PAF.

MPAI-OSD has been developed for uniform use across MPAI Technical Specifications with the cooperation of

1. The *AI Framework* (AIF-DC), *Context-based Audio Enhancement* (CAE-DC), *Multimodal Conversation* (MMC-DC), and *Portable Avatar Format* (PAF-DC).
2. The *Connected Autonomous Vehicle* (CAV) and *MPAI Metaverse Model* (MMM) groups of the Requirements Standing Committee.

In the future, MPAI may publish new Technical Specification in the MPAI-OSD scope that modify or extend the scope of this Technical Specification.

4 Definitions

Capitalised Terms have the meaning defined in [Table 1](#). All MPAI-defined Terms are accessible [online](#). Non-capitalised terms have the meaning commonly defined for the context in which they are used or represent an entity in the real world. For instance, Table 1 defines *Object*, *Scene*, and *User* but does not define *object*, *scene*, and *human*.

A dash “-” preceding a Term in [Table 1](#) means the following:

1. If the font is normal, the Term in [Table 1](#) without a dash and preceding the one with a dash should be placed before that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Data followed by one of the words Format and Type.
2. If the font is *italic*, the Term in the table without a dash and preceding the one with a dash should be placed after that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Descriptor preceded by one of the words Face and Body.

Table 1 – Terms and Definitions

Term	Definition
Attitude	
- <i>Spatial</i>	Position and Orientation and their velocities and accelerations of a Human and Visual Object in a Virtual Environment.
Audio	A Data Type an instance of which represents analogue signals – or is rendered to be perceived – in the human-audible range (16 Hz - 20 kHz).
Avatar	An Data Type including the 3D Model of an Avatar and the Face and Body Descriptors.
- Model	An inanimate Avatar exposing animation interfaces.
- <i>Portable</i>	A Data Type including Avatar ID, Time, Avatar, Language, Speech, Text, Speech Model, Personal Status, Audio-Visual Scene Descriptors, and potentially an input Portable Avatar.
Centre Point	The point of an Object selected to have coordinates (0,0,0).
Coordinate System	A system where the position of a point is specified by three numbers.
- <i>Cartesian</i>	A coordinate system where the three numbers are the signed distances from the point to three mutually perpendicular planes.
- <i>Spherical</i>	A coordinate system where the three numbers are: - the radial distance of that point from a fixed origin. - the polar angle measured from a fixed zenith direction. - the azimuthal angle of its orthogonal projection on a reference plane.
Data	Information in digital form.
- Format	A specific digital representation of Data.
- <i>Media</i>	Data representing Text, Speech, Audio, Visual, 3D Model, LiDAR, RADAR, Ultrasound information.
- Object	A Data Type including Data of a given Data Type and the Qualifier of that Data Type.
- Type	A recognised instance of Data.
Descriptor	The Digital Representation of a feature of an Object.

- <i>Audio-Visual</i>	A Data Type including the digital representation of the features of an audio-visual instance.
- <i>Body</i>	A Data Type including the digital representation of the features of the body of a real or digital human.
- <i>Face</i>	A Data Type including the digital representation of a feature of the face of a real or digital human.
- <i>Visual</i>	A Data Type including the digital representation of the features of a visual instance.
Digital Representation	Data corresponding to and representing a physical entity.
Environment	A Virtual Space that may be null or may include an Audio-Visual Scene.
Human	A human being in a real space.
- <i>Digital</i>	A Digitised or a Virtual Human.
- <i>Digitised</i>	An Object that has the appearance of a specific human when rendered.
- <i>Virtual</i>	An Object created by a computer that has a human appearance when rendered but is not a Digitised Human.
Identifier	The label uniquely associated with a human or an Object.
Instance	An element of a set of entities – Scenes, Digital Humans etc. – belonging to some levels in a hierarchical classification (taxonomy).
Object	A Data Type including Media Data and an optional Qualifier.
- <i>3D Model</i>	A Data Type including 3D Model Data and Qualifier.
- <i>Audio</i>	A Data Type including Audio Data and Qualifier.
- <i>Audio-Visual</i>	A Data Type including Audio-Visual Data and Qualifier.
- <i>Digital</i>	A Digitised or a Virtual Object.
- <i>Digitised</i>	Data representing a real object.
- <i>Speech</i>	A Data Type including Speech Data and Qualifier.
- <i>Text</i>	A Data Type including Text Data and Qualifier.
- <i>Visual</i>	A Data Type including Visual Data and Qualifier.
Orientation	The 3 Euler angles of an Object in a Virtual Space.
Position	The coordinates of a representative point for an object in a Virtual Space with respect to a set of coordinate axes.
Rendering	The process of instantiating Data or a Virtual Space as a human-perceptible entity.
Scene	A composition of Objects located according to a Scene Geometry.
- <i>3D Model</i>	A Scene composed of 3D Model Objects.
- <i>Audio</i>	A Scene composed of Audio Objects.
- <i>Audio-Visual</i>	A Scene composed of Speech and Audio Objects, Visual and 3D Model Objects, and co-located Audio-Visual Objects.
- <i>Speech</i>	A Scene composed of Speech Objects.
- <i>Visual</i>	A Scene composed of Visual Objects.

Scene Descriptors	A Data Type including the Media Objects and their spatial arrangement in a Scene.
- 3D Model	A Data Type including a Scene's 3D Model Objects and Sub-Scenes, and their spatial arrangement.
- <i>Audio</i>	A Data Type including an Audio Scene's Audio Objects and Sub-Scenes, and their spatial arrangement.
- <i>Audio-Visual</i>	A Data Type including an Audio Scene's Speech, Audio, Visual, 3D Model, and Audio-Visual Objects, and their spatial arrangement.
- <i>Visual</i>	A Data Type including a Visual Scene's Visual Objects and Sub-Scenes and their spatial arrangement.
Scene Geometry	A Data Type including the spatial arrangement of the Media Objects in a Scene.
<i>3D Model</i>	A Data Type describing the spatial arrangement of the 3D Model Objects and Sub-Scenes in a Scene.
- <i>Audio</i>	A Data Type describing the spatial arrangement of the Visual Objects and Sub-Scenes of a Scene.
- <i>Audio-Visual</i>	A Data Type describing the spatial arrangement of the Speech, Audio, Visual, 3D Model, and Audio-Visual Objects and Sub-Scenes of a Scene.
- <i>Speech</i>	A Data Type describing the Spatial arrangement of the Speech Objects and Sub-Scenes of a Scene.
- <i>Visual</i>	A Data Type describing the Spatial arrangement of the Visual Objects and Sub-Scenes of a Scene.
Selector	Input Data having the goal to set a parameter (e.g., use of Text vs Speech or Language Preference) or an operating mode of a Machine.
Speech	A Data Type an instance of which represents or is rendered to be perceived as human utterances.
Text	A series of characters drawn from a finite alphabet of a character set.
- <i>Recognised</i>	The Text at the output of an Automatic Speech Recognition AIM.
- <i>Refined Text</i>	The Text at the output of a Natural Language Understanding AIM.
Trajectory	The sequence of start and end Spatial Attitudes and corresponding Times expected and actual of a series of segments.
Virtual Space	A space generated and maintained by a computing platform that can be rendered.

5 References

5.1 Normative Reference

1. MPAI; Technical Specification: [Governance of the MPAI ecosystem](#) (MPAI-GME) V2.0.
2. MPAI; Technical Specification: [AI Framework](#) (MPAI-AIF) V2.2.
3. MPAI; Technical Specification: [Context-based Audio Enhancement](#) (MPAI-CAE) - [Use Cases](#) (CAE-USC) V2.4.
4. MPAI; Technical Specification: [Multimodal Conversation](#) (MPAI-MMC) V2.4.

5. MPAI; Technical Specification: [Portable Avatar Format](#) (MPAI-PAF) V1.5.
6. MPAI; Technical Specification: [Profiles](#) (MPAI-PRF) - [AI Modules](#) (PRG-AIM) V1.1.
7. MPAI; Technical Specification: [Data Types, Formats, and Attributes](#) (MPAI-TFA) V1.4.
8. IETF; Date and Time in the Internet: Timestamps; RFC 3339; July 2002.

5.2 Informative References

9. MPAI; [The MPAI Statutes](#).
10. MPAI; [Patent Policy](#)
11. MPAI; Framework Licence: [Object and Scene Description](#)
12. MPAI; Technical Specification: [Connected Autonomous Vehicles](#) (MPAI-CAV) – [Technologies](#) (CAV-TEC) V1.1.
13. MPAI; Technical Specification: [MPAI Metaverse Model](#) (MPAI-MMM) – [Technologies](#) (MMM-TEC) V2.1.

6 AI Workflows

6.1 Technical Specifications

Technical Specification: Object and Scene Descriptors (MPAI-OSD) V1.4 assumes that implementations will be based on *Technical Specification: AI Framework (MPAI-AIF) V2.1* enabling dynamic configuration, initialisation, and control of AI Workflows (AIW) composed of interconnected AI Modules (AIM) in a standard AI Framework (AIF).

Table 1 displays the AIW specified by MPAI-OSD V1.4. Click the AIW to access its dedicated page, which includes a its functions, reference model, I/O Data, Functions of AIMs, I/O Data of AIMs, and a table providing links to the AIW-related AIW, AIMs, and JSON metadata.

Table 1 - AIWs of MPAI-OSD V1.4

Acronym	Name and Specification	JSON
OSD-TMA	Television Media Analysis	X

6.1.1 Television Media Analysis

6.1.1.1 Functions

The Television Media Analysis (OSD-TMA) AI Workflow produces Audio-Visual Event Descriptors (OSD-AVE) composed of a set of Audio-Visual Scene Descriptors (AVS) that include

1. Relevant Audio, Visual, or Audio-Visual Object
2. IDs of speakers and ID of faces with their spatial positions
3. Text from utterances

of a television program provided as input together with textual information that OSD-TMA may use to improve its performance.

OSD-TMA assumes that there is only one speaker at a time.

6.1.1.2 Reference Model

Figure 1 depicts the Reference Model of TV Media Analysis (OSD-TMA).

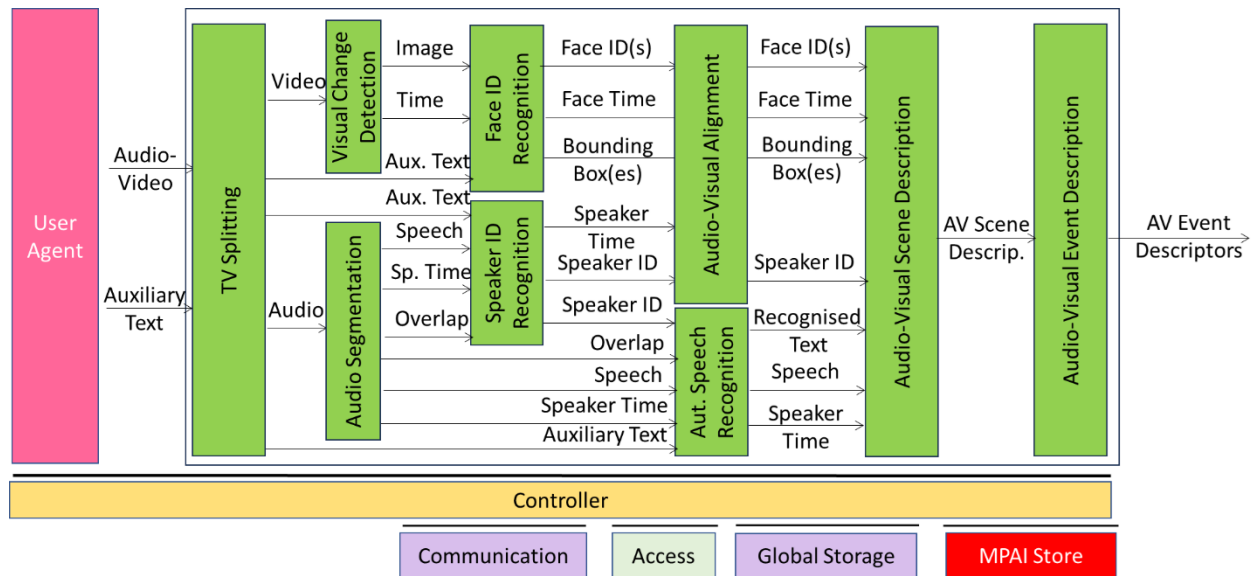


Figure 1 – Reference Model of OSD-TMA

6.1.1.3 Input/Output Data

Table 1 provides the input and output data of the TV Media Analysis Use Case:

Table 1 – I/O Data of TV Media Analysis (OSD-TMA)

Input	Descriptions
Audio-Video	Audio-Video to be analysed.
Text Object	Text helping OSD-TMA AIMS to improve their performance.
Output	Descriptions
Audio-Visual Event Descriptors	Resulting analysis of Input Audio-Video-Text.

6.1.1.4 Functions of AI Modules

Table 2 provides the functions of the TV Media Analysis Use Case. Note that processing proceeds asynchronously, e.g., TV Splitter separates audio and video for the entire duration of the file and passes the entire audio and video files.

Table 2 – Functions of AI Modules of TV Media Analysis (OSD-TMA)

AIM	Function
TelevisionSplitting	Receives Audio-Visual File composed of: - An Audio-Video component. - A Text component. Produces - Video file - Audio file - Text file When all the files of the full duration of the video are ready, AV Splitter informs the following AIMS.
VisualChangeDetection	Receives Video file. Iteratively:

	- Looks for a video frame that conveys a scene changed from the preceding scene (depends on threshold). - Assigns a video clip identifier to the video clip. - Produces a set of images with StartTime and EndTime.
AudioSegmentation	Receives Audio file. Iteratively: - For each audio segment (from one change to the next): - Becomes aware that there is speech. - Assigns a speech segment ID and anonymous speaker ID (i.e., the identity is unknown) to the segment. - Decides whether: - The existing speaker has stopped. - A new speaker has started a speech segment. - If a speaker has started a speech: - Assigns a new speech segment ID. - Check whether the speaker is new or old in the session. - If old retain old anonymous speaker ID. - If new assign a new anonymous speaker ID. - Produces a series of audio sequences each of which contains: - A speech segment. - Start and end time. - Anonymous Speaker ID. - Overlap information
Face Identity Recognition	- Receives a Text file. - Extracts semantic information from the Text file. - Receives a set of images per video clip. - For each image identifies the bounding boxes. - Extracts faces from the bounding boxes. - Extracts the embeddings that represent a face. - Compares the embeddings with those stored in the face recognition data base. - Associates the embeddings with a face ID taking into account the semantic information from the Text.
Speaker Identity Recognition	1. Receives a Text file. 2. Extracts semantic information from the Text file. 3. Receives a Speech Object and Speech Overlap information. 4. Extracts the embeddings that represent the speech segment. 5. Compares the embeddings with those stored in the speaker recognition data base. 6. Associates the embeddings with a Speaker ID taking into account the semantic information from the Text.
Audio-Visual Alignment	Receives: - Face ID - Bounding Box - Face Time - Speaker ID - Speaker Time Associates Speaker ID and Face ID

Automatic Speech Recognition	- Receives a Text file. - Extracts semantic information from the Text file. - Receives a Speech Object. - Produces the transcription of the speech payload taking into account the semantic information from the Text. - Attaches time stamps to specific portions of the transcription.
Audio-Visual Scene Description	Receives - Bounding box coordinates, Face ID, and time stamps - Speaker ID and time stamps. - Reconciles Face ID and Speaker ID. - Text and time stamps Produces Audio-Visual Scene Descriptors
Audio-Visual Event Description	Receives Audio-Visual Scene Descriptors Produces Audio-Visual Event Descriptors

6.1.1.5 I/O Data of AI Modules

Table 3 provides the I/O Data of the AI Modules of the TV Media Analysis Use Case.

Table 3 – I/O Data of AI Modules of TV Media Analysis (OSD-TMA)

AIM	Receives	Produces
TelevisionSplitting	Audio - Video - Auxiliary Text	- Audio - Video - Auxiliary Text
VisualChangeDetection	- Video	- Image
AudioSegmentation	- Speech	- SpeechObjects - SpeechOverlap
Face Identity Recognition	- Image - Time - Auxiliary Text	VisualSceneDescriptors with: - FaceID - FaceTime - BoundingBox
Speaker Identity Recognition	- SpeechObject - SpeakerTime	- SpeechSceneDescriptors : with: - Speaker ID - SpeakerTime
Audio-Visual Alignment	- SpeechOverlap - SpeechObject - SpeakerTime - Auxiliary Text	- Recognised Text - SpeechObject - SpeakerTime
Automatic Speech Recognition	- SpeechSceneDescriptors - VisualSceneDescriptors	- AVSceneDescriptor with: - AlignedFaceID - BoundingBox
Audio-Visual Scene Description	- BoundingBox - AlignedFaceID - SceneTime	- AVSceneDescriptors

	- SpeakerID - Recognised Text	
Audio-Visual Event Description	- AVSceneDescriptors :	- AVEventDescriptors

6.1.1.6 AIW, AIMS, and JSON Metadata

Table 4 – AIW, AIMS, and JSON Metadata of TV Media Analysis (OSD-TMA)

AIW	AIM	Name	JSON
OSD-TMA		Television Media Analysis	X
	MMC-AUS	Audio Segmentation	X
	OSD-AVA	Audio-Visual Alignment	X
	OSD-AVE	Audio-Visual Event Description	X
	OSD-AVS	Audio-Visual Scene Description	X
	MMC-ASR	Automatic Speech Recognition	X
	PAF-FIR	Face Identity Recognition	X
	MMC-SIR	Speaker Identity Recognition	X
	OSD-TVS	Television Splitting	X
	OSD-VCD	Visual Change Detection	X

6.1.1.6.1 Disclaimers

1. This OSD-TMA Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to provide a working Implementation of OSD-TMA, not to provide a ready-to-use product.
3. MPAI disclaims the suitability of the Software for any other purposes and does not guarantee that it is secure.
4. Use of this Reference Software may require acceptance of licences from the respective repositories. Users shall verify that they have the right to use any third-party software required by this Reference Software.

6.1.1.6.2 Guide to the OSD-TMA code

The OSD-TMA Reference Software:

1. Receives any input Audio-Visual file that can be demultiplexed by FFMPEG.
2. Uses FFMPEG to extract:
 1. A WAV file (uncompressed audio).
 2. A video file.
3. Produces descriptors of the input Audio-Visual file represented by [Audio-Visual Event Descriptors](#).

The current OSD-TMA implementation does not support Auxiliary Text.

The OSD-TMA uses the following software components:

1. [RabbitMQ](#) is an MPAI-AIF function but is adapted for use in OSD-TMA. This service starts, stops and removes all OSD-TMA AIMS automatically using Docker in Docker (DinD): the controller lets only one dockerized OSD-TMA AIM run at any time, which saves computing resources. Docker is used to automate the deployment of all OSD-TMA AIMS so that they can run in different environments.

2. [Portainer](#) is a service that helps manage Docker containers, images, volumes, networks and stacks using a Graphical User Interface (GUI). Here it is mainly used to manage two containerized services: RabbitMQ and Controller
3. Docker Compose has a yml file to build and run all OSD-TMA AIMs and the Module for the Controller
4. [Controller](#) builds the Docker images of all OSD-TMA AIMs and of the Controller by reading a YAML file called compose.yml. Docker Compose helps run RabbitMQ and Controller as Docker containers.
5. Python code manages the MPAI-AIF operation but is adapted for use in OSD-TMA.

The OSD-TMA software includes compose.yml and controller. They can be found at https://experts.mpai.community/software/mpai-aif/osd_tma/orchestrator. compose.yml starts RabbitMQ, not Portainer. <https://docs.portainer.io/start/install-ce> describes how to install Portainer.

The OSD-TMA code includes [compose.yml and the code for the Controller](#).

compose.yml starts RabbitMQ, not Portainer. Portainer installation is [described](#).

The OSD-TMA code is released with a [demo](#). Recorded [demo](#): The [saved JSON](#) of OSD-AVE.

The OSD-TMA Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC), in particular, Francesco Gallo (EURIX) and Mattia Bergagio (EURIX) for developing this software.

The implementation includes the AI Modules of the following table. AIMs in red achieve high performance if a GPU to used. However, the software can also operate without GPU.

AIM	Name	Library	Dataset
MMC-ASR	Speech recognition	whisper-timestamped	Labeled Faces in the Wild
MMC-SIR	Speaker identification	SpeechBrain	VoxCeleb1
MMC-AUS	Audio segmentation	pyannote.audio	
OSD-AVE	Event descriptors		
OSD-AVS	Scene descriptors		
OSD-TVS	TV splitting	ffmpeg	
OSD-VCD	Visual change	PySceneDetect	
PAF-FIR	Face recognition	DeepFace	

6.1.1.6.3 RabbitMQ

RabbitMQ is an open-source message broker that implements AMQP (Advanced Message Queuing Protocol). It enables asynchronous communication among the modules, allows them to be decoupled and improves the scalability and resilience of the system.

RabbitMQ provides a number of key features:

- Asynchronous messaging: RabbitMQ supports asynchronous messaging, which decouples senders from receivers and allows applications to communicate in a non-blocking manner.
- Queue management: RabbitMQ implements message queuing, which helps distribute messages among consumers.
- Reliability: RabbitMQ provides mechanisms to ensure message reliability, such as delivery acknowledgement and message persistence.
- Flexible routing: RabbitMQ supports different types of exchanges for message routing, so messages can be routed according to various criteria.
- Clustering: RabbitMQ supports clustering, which allows multiple RabbitMQ servers to be grouped into a single logical broker, thereby improving reliability and availability.

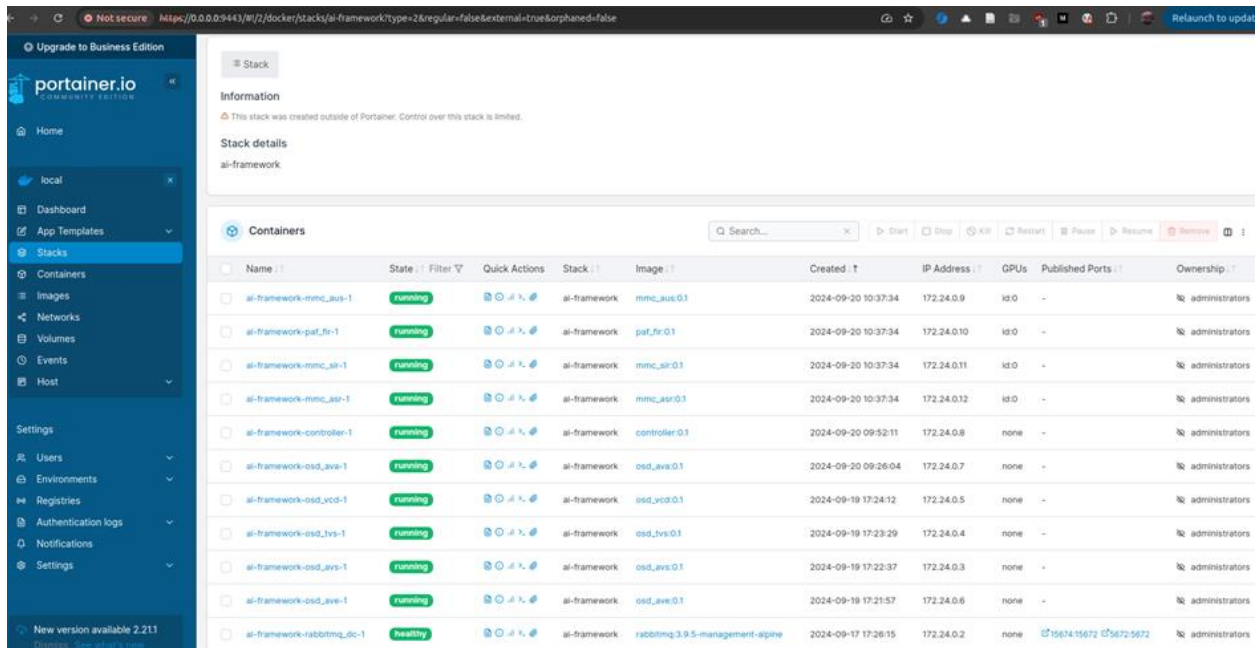
RabbitMQ is usable in a wide range of applications, from simple data transfer among processes to more complex messaging patterns such as publish/subscribe and work queues. Example of queues:

Overview				Messages			Message rates			+/-
Name	Type	Features	State	Ready	Unacked	Total	incoming	deliver / get	ack	
queue_module_controller	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_controller_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_mmc_asr	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_mmc_asr_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_mmc_ava	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_mmc_ava_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_mmc_sir	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_mmc_sir_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_osd_ava	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_osd_ava_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_osd_ave	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_osd_ave_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_osd_avs	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_osd_avs_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_osd_tvs	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_osd_tvs_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_osd_vcd	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_osd_vcd_DLQ	classic	D TTL DLX DLK	idle	0	0	0				
queue_module_paf_fir	classic	D DLX DLK	idle	0	0	0	0.00/s	0.00/s	0.00/s	
queue_module_paf_fir_DLQ	classic	D TTL DLX DLK	idle	0	0	0				

6.1.1.6.4 Portainer

Portainer is an open-source user interface that streamlines the management of Docker containers. It allows you to manage Docker containers, images, stacks, networks and volumes from a web interface; no need for a command-line interface. You can use Portainer to perform tasks such as building images, adding containers or stacks, and monitoring resource usage.

Example of containers in one of our stacks:



You can build and start all modules by running
`docker compose --env-file .env -f compose.yml up`
 compose.yml is in [repo](#).

.env lists the environment variables below:

`TAG=0.1`

`LOG_LEVEL`: log level. Can be *CRITICAL*, *ERROR*, *WARNING*, *INFO*, *DEBUG*, or *NOTSET*

`PATH_SHARED`: path of storage on disk

`AI_FW_DIR`: path of storage in container

`HUGGINGFACE_TOKEN`: HuggingFace token. Used by MMC-AUS

`GIT_NAME`: Git username

`GIT_TOKEN`: Git token

`MIDDLEWARE_USER`: RabbitMQ username

`MIDDLEWARE_PASSWORD`: RabbitMQ password

`MIDDLEWARE_PORT`: RabbitMQ port

`MIDDLEWARE_EXTERNAL_PORT`: outer RabbitMQ port

`MIDDLEWARE_VIRTUALHOST`: RabbitMQ virtual host. Default:

`MIDDLEWARE_VIRTUALHOST=/`

`EXCHANGE_NAME`: RabbitMQ exchange name

To start module x run

`docker compose --env-file .env -f compose.yml up controller rabbitmq_dc x`

Send a suitable payload to the queue of module x, as shown for the controller [here](#):

6.1.1.6.5 Controller

The controller is the module that orchestrates the workflow. It sends a message to the right module for processing.

The controller receives the input message from queue `queue_module_controller`. Then, the controller sends a tweaked message to OSD-TVS. This message contains the input message. OSD-TVS sends its output message to the controller, which sends a tweaked output message to MMC-AUS.

Modules are called in this order:

1. OSD-TVS
2. MMC-AUS

3. OSD-VCD
4. MMC-SIR
5. MMC-ASR
6. PAF-FIR
7. OSD-AVA
8. OSD-AVS
9. OSD-AVE.

6.1.1.6.6 Examples/README.md

examples/README.md details how to run examples are [here](#).

6.1.1.6.7 Acknowledgements

This version of the OSD-TMA Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

6.1.1.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-TMA AIW. Conformance Testing of the individual AIMS of the OSD-TMA Composite AIM are given by the individual AIM Specification.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-TMA AIW

Receives	Audio-Video	Shall validate against Audio-Visual Object schema. Also, Audio-Visual Data shall conform with Audio-Visual Qualifier.
	Text Object	Shall validate against Text Object schema. Also, Text Data shall conform with Text Qualifier.
Produces	Audio-Visual Event Descriptors	Shall validate against Audio-Visual Event Descriptors schema.

6.1.1.7.1 Reference Software

As a rule, MPAI provides Reference Software implementing the AI Modules released with the BSD-3-Clause licence and the following disclaimers:

1. The purpose of the Reference Software is to demonstrate a working Implementation of an AIW, not to provide a ready-to-use product.
2. MPAI disclaims the suitability of the Software for any other purposes than those of the MPAI-OSD Standard, and does not guarantee that it offers the best performance and that it is secure.
3. Users shall verify that they have the right to use any third-party software required by this Reference Software, e.g., by accepting the licences from third-party repositories.

6.1.1.7.2 Conformance Testing

An implementation of an AI Workflow conforms with MPAI-OSD if it accepts as input and produces as output Data and/or Data Objects (Data of a Data Type and its Qualifier) conforming with those specified by MPAI-HMC.

The Conformance of an instance of a Data is to be expressed by a sentence like “Data validates against the Data Type Schema”. This means that:

- Any Data Sub-Type is as indicated in the Qualifier.
- The Data Format is indicated by the Qualifier.
- Any File and/or Stream have the Formats indicated by the Qualifier.
- Any Attribute of the Data is of the type or validates against the Schema specified in the Qualifier.

The method to Test the Conformance of a Data or Data Object instance is specified in the *Data Types* chapter.

6.1.1.7.3 Performance Assessment

Performance is a multidimensional entity because it can have various connotations. Therefore, the Performance Assessment Specification should provide methods to measure how well an AIW performs its function, using a metric that depends on the nature of the function, such as:

1. Quality: the Performance of a [Television Media Analysis](#) AIW can measure how well an AIM identifies the humans in a scene.
2. Bias: Performance of a [Television Media Analysis](#) AIW can measure the quality of face identification in dependence of the types of humans involved.
3. Legal compliance: the Performance of an AIW can measure the compliance of the AIW to a regulation, e.g., the European AI Act.
4. Ethical compliance: the Performance Assessment of an AIW can measure the compliance of an AIW to a target ethical standard.

The current MPAI-OSD V1.4 Standard does not provide AIW Performance Assessment methods.

7 AI Modules

7.1 Technical Specifications

Table 1 provides the links to the specifications and the JSON syntax of all AIMs specified by ***Technical Specification: Object and Scene Description (MPAI-OSD) V1.4***. All previously specified MPAI-OSD AI-Modules are superseded by those specified by V1.4 but may still be used by explicitly signaling their version. AI Modules in bold are Composite.

Table 1 - Specifications and JSON syntax of AIMs used by MPAI-OSD V1.3

Acron.	Name	J	Acron.	Name	J	Acron.	Name	J
OSD-BSC	Basic Scene Description	X	OSD-SCD	Scene Description	X	OSD-EVD	Event Description	X
OSD-B3D	Basic 3D Model Scene Description	X	OSD-3SD	3D Model Scene Description	X	OSD-3ED	3D Model Event Description	X
OSD-BAS	Basic Audio Scene Description	X	OSD-ASD	Audio Scene Description	X	OSD-AED	Audio Event Description	X
OSD-BMD	Basic Audio-Visual Scene Description	X	OSD-MSD	Audio-Visual Scene Description	X	OSD-MED	Audio-Visual Event Description	X
OSD-BLD	Basic LiDAR Scene Description	X	OSD-LSD	LiDAR Scene Description	X	OSD-LED	LiDAR Event Description	X

OSD-BOD	Basic Offline Map Scene Description	X	OSD-OSD	Offline Map Scene Description	X	OSD-OED	Offline Map Event Description	X
OSD-BRD	Basic RADAR Scene Description	X	OSD-RSD	RADAR Scene Description	X	OSD-RED	RADAR Event Description	X
OSD-BSD	Basic Speech Scene Description	X	OSD-SSD	Speech Scene Description	X	OSD-SED	Speech Event Description	X
OSD-BUD	Basic Ultrasound Scene Description	X	OSD-USD	Ultrasound Scene Description	X	OSD-UED	Ultrasound Event Description	X
OSD-BVD	Basic Visual Scene Description	X	OSD-VSD	Visual Scene Description	X	OSD-VED	Visual Event Description	X
OSD-AVA	Audio-Visual Alignment	X	OSD-SDX	Audio-Visual Scene Demultiplexing	X	OSD-SMX	Audio-Visual Scene Multiplexing	X
OSD-DVI	Direct Visual Identification	X	OSD-TVS	Television Splitting	X	OSD-VCD	Visual Change Detection	X
OSD-VDI	Visual Direction Identification	X	OSD-VII	Visual Instance Identification	X	OSD-VOE	Visual Object Extraction	X
OSD-VOI	Visual Object Identification	X	OSD-VOB	Visual Object Identification by Body	X			

Note that:

1. MPAI-OSD Data Types that are specified by V1.3 are superseded by those specified by V1.4. Use of earlier versions of Data Types is permitted if their version is explicitly signaled.
2. MPAI-OSD AIWs and AIMs utilise Data Types specified by other MPAI Technical Specifications.
3. MPAI-OSD V1.3 specifies Basic Scene Geometry and Scene Geometry; Basic Scene Descriptors and Scene Descriptors; and Basic Scene Geometry and Scene Geometry, but not their media specific versions because they can be obtained by referencing their Qualifiers.

7.1.1 Basic Scene Description

7.1.1.1 Functions

The Basic Description (OSD-BSC) AIM receives Objects with their Space-Time information and produces the Descriptors of a Scene composed of Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	Objects	Individual Objects.
Produces	Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.1.2 Reference Model

The Reference Architecture is depicted in Figure 1.

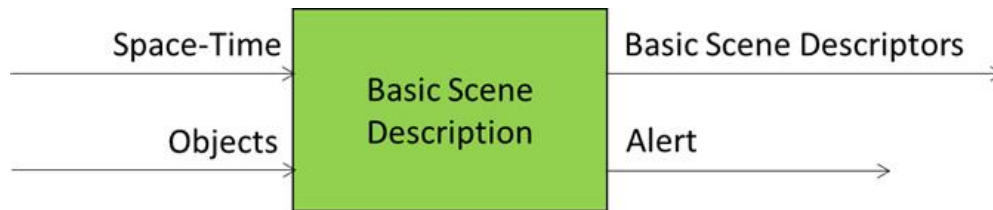


Figure 1 – The Basic Scene Description (OSD-BSC) AIM

7.1.1.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Basic Scene Description (OSD-BSC) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Scene Description (OSD-BSC) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Objects	Input Objects.
Output	Description
Basic Scene Descriptors	The output Basic Scene Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.1.4 SubAIMs

No SubAIMs.

7.1.1.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicSceneDescription.json>

7.1.1.6 Profiles

No profiles.

7.1.1.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BSC AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BSC AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.2 Scene Description

7.1.2.1 Functions

The Scene Description (OSD-SCD) AIM receives Objects their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of Objects and Scenes as output. The OSD-SCD AIM may also receive an Alert conveying information on potential anomalies in the input Objects.:

Receives	Space-Time	of the input Objects having the same time base.
	Objects.	individual Objects.
	Scene Descriptors	Scene the input Objects belong to.
Integrates	Space-Time and Object	with Scene Descriptors.
Produces	Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.2.2 Reference Model

The Reference Architecture is depicted in Figure 1.

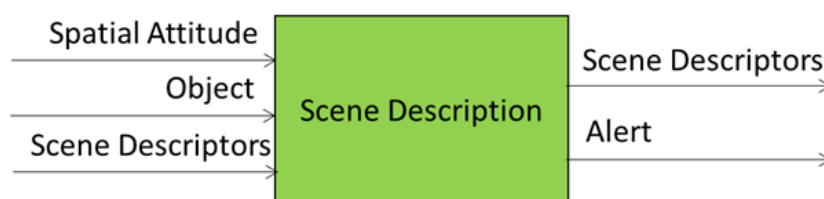


Figure 1 – The Scene Description (OSD-SCD) AIM

7.1.2.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Scene Description (OSD-VSD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Scene Description (OSD-SCD) AIM

Input	Description
Space-Time	Space-Time of input Objects.

Objects	Input Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.2.4 SubAIMs

No SubAIMs.

7.1.2.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/SceneDescription.json>

7.1.2.6 Profiles

No profiles.

7.1.2.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-SCD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-SCD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.3 Event Description

7.1.3.1 Functions

The Event Description (OSD-EVD) AIM produces the Descriptors of an Event from a sequence of Scene Descriptors:

Receives	<i>Scene Descriptors.</i>	A sequence of Scenes.
Produces	<i>Event Descriptors</i>	The Event.

7.1.3.2 Reference Model

The Event Description (OSD-EVD) AIM Reference Model is depicted in Figure 1.

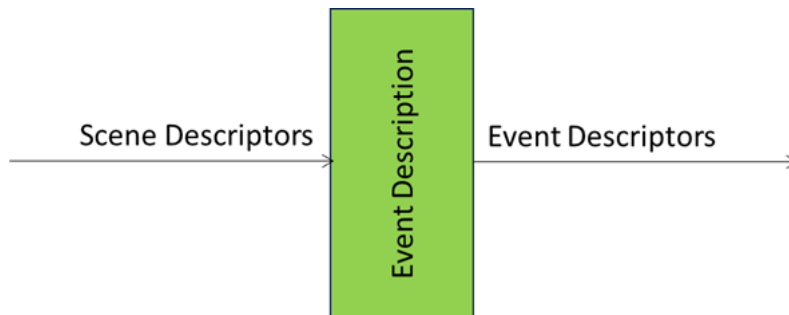


Figure 1 – The Event Description (OSD-EVD) AIM Reference Model

7.1.3.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Event Description (OSD-EVD) AIM

Input	Description
Scene Descriptors	Sequence of Scene Descriptors.
Output	Description
Event Descriptors	The Event Descriptors of the Scene.

7.1.3.4 SubAIMs

No SubAIMs

7.1.3.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/EventDescription.json>

7.1.3.6 Profiles

No Profiles

7.1.3.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-EVD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-EVD AIM

Receives	Scene Descriptors	Shall validate against Scene Descriptors schema
Produces	Event Descriptors	Shall validate against Event Descriptors schema

7.1.4 Basic 3D Model Scene Description

7.1.4.1 Functions

The Basic 3D Model Scene Description (OSD-B3D) AIM receives 3D Model Objects and their Space-Time information and produces the Descriptors of a Scene composed of 3D Model Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	3D Model Objects	Individual Objects.
Produces	Basic 3D Model Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.4.2 Reference Model

The Reference Architecture is depicted in Figure 1.

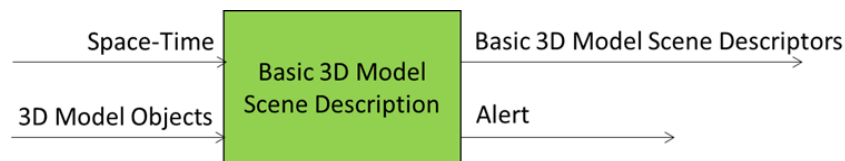


Figure 1 – The Basic 3D Model Scene Description (OSD-B3D) AIM

7.1.4.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Basic 3D Model Scene Description (OSD-B3D) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic Scene Description (OSD-B3D) AIM

Input	Description
Space-Time	Space-Time of input Objects.
3D Model Objects	Input Objects.
Output	Description
Basic 3D Model Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.4.4 SubAIMs

No SubAIMs.

7.1.4.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/Basic3DModelSceneDescription.json>

7.1.4.6 Profiles

No profiles.

7.1.4.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-B3D AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-B3D AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	3D Model Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic 3D Model Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.5 3D Model Scene Description

7.1.5.1 Functions

The 3D Model Scene Description (OSD-3SD) AIM receives 3D Model Objects, the Descriptors of the Scene the Objects belong to, and their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of 3D Model Objects and Scenes as output. The OSD-3SD AIM may also receive an Alert conveying information on potential anomalies in the input 3D Model Objects.

Receives	Space-Time	of the input Objects having the same time base.
	3D Model Objects	individual Objects.
	Scene Descriptors	Additional Descriptors of a Scene the Objects belong to.
Integrates	Space-Time and 3D Model Object	with Scene Descriptors.
Produces	3D Model Scene Descriptors	Output#1 of AIM

	Alert	Output#2 of AIM signaling potential anomalies in Objects.
--	-------	---

7.1.5.2 Reference Model

The Reference Architecture is depicted in Figure 1.

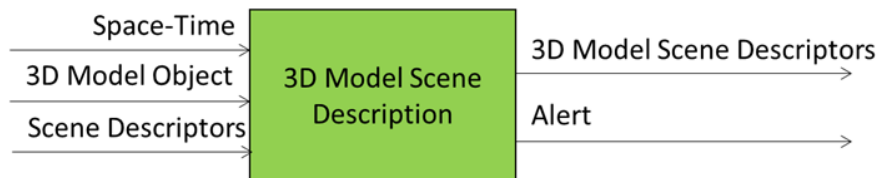


Figure 1 – The 3D Model Scene Description (OSD-3SD) AIM

7.1.5.3 Input/Output Data

Table 1 specifies the Input and Output Data of the 3D Model Scene Description (OSD-3SD) AIM.

Table 1 – I/O Data of the 3D Model Scene Description (OSD-3SD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
3D Model Objects	Input 3D Model Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
3D Model Scene Descriptors	The output 3D Model Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.5.4 SubAIMs

No SubAIMs.

7.1.5.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/3DModelSceneDescription.json>

7.1.5.6 Profiles

No profiles.

7.1.5.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-3SD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-3SD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	3D Model Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	3D Model Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.6 3D Model Event Description

7.1.6.1 Functions

The 3D Model Event Description (OSD-3ED) AIM produces the Descriptors of a 3D Model Event from a sequence of 3D Model Scene Descriptors:

Receives	<i>3D Model Scene Descriptors</i>	A sequence.
Produces	<i>3D Model Event Descriptors</i>	A file

7.1.6.2 Reference Model

The 3D Model Event Description (OSD-3ED) AIM Reference Model is depicted in Figure 1.

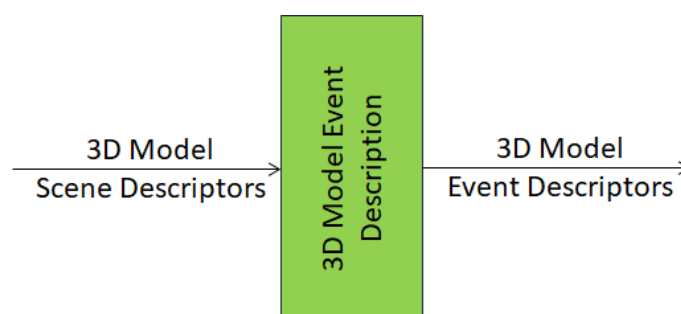


Figure 1 – The 3D Model Event Description (OSD-3ED) AIM Reference Model

7.1.6.3 Input/Output Data

Table 1 specifies the Input and Output Data of the 3D Model Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the 3D Model Event Description (OSD-3ED) AIM

Input	Description
3D Model Scene Descriptors	Sequence of 3D Model Scene Descriptors.
Output	Description
3D Model Event Descriptors	The 3D Model Event Descriptors of the 3D Model Scene.

7.1.6.4 SubAIMs

No SubAIMs

7.1.6.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/3DModelEventDescription.json>

7.1.6.6 Profiles

No Profiles

7.1.6.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-3ED AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-3ED AIM

Receives	3D Model Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	3D Model Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.7 Basic Audio Scene Description

7.1.7.1 Functions

The Basic Audio Scene Description (OSD-BAD) AIM receives Audio Objects and their Space-Time information and produces the Descriptors of a Scene composed of Audio Objects and optionally an Alert:

Receives	Space-Time Audio Objects	of the input Objects having the same time base. Individual Objects.
Produces	Basic Audio Scene Descriptors Alert	Scene Descriptors produced by AIM. Data concerning potential anomalies in input Object.

7.1.7.2 Reference Model

The Reference Architecture is depicted in Figure 1.

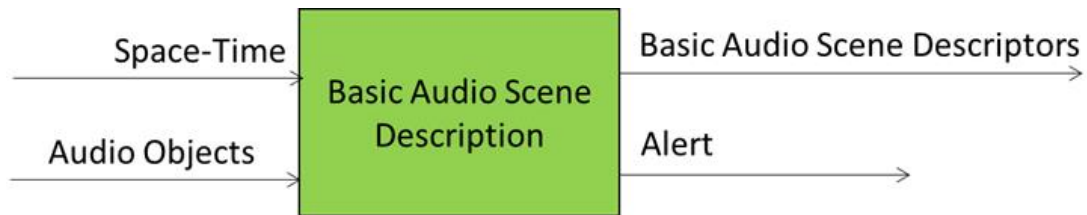


Figure 1 – The Basic Audio Scene Description (OSD-BAD) AIM

7.1.7.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Scene Description (OSD-BAD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Scene Description (OSD-BAD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Audio Objects	Input Objects.
Output	Description
Basic Audio Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.7.4 SubAIMs

Basic Audio Scene Description (CAE-BAS) is a Composite AIM with the structure depicted in Figure 2.

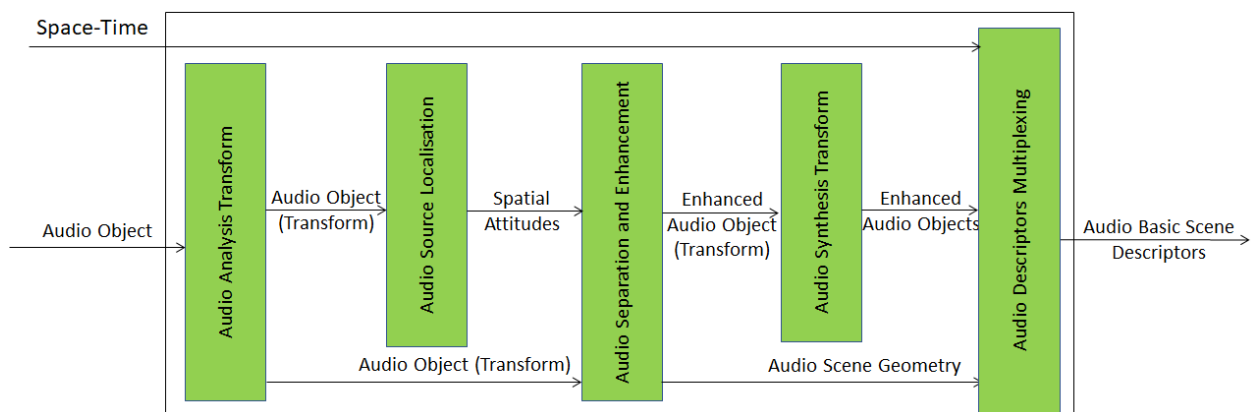


Figure 2 - The Basic Audio Scene Description Composite (OSD-ABS) AIM

The specifications of the OSD-BAS Basic AIMs are provided at the links of Table 2.

Table 2 – BASIC AIMs of Basic Audio Scene Description

AIM	AIMs	Names	JSON
OSD-BAS		Basic Audio Scene Description	File
	CAE-AAT	Audio Analysis Transform	File
	CAE-ASL	Audio Source Localisation	File
	CAE-ASE	Audio Separation and Enhancement	File
	CAE-AST	Audio Synthesis Transform	File
	CAE-AMX	Audio Descriptors Multiplexing	File

7.1.7.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicAudioSceneDescription.json>

7.1.7.6 Profiles

No profiles.

7.1.7.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BAD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BAD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Audio Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Audio Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.8 Audio Scene Descriptor

7.1.8.1 Functions

The Audio Scene Description (OSD-ASD) AIM receives Audio Objects, the Descriptors of the Scene the Objects belong to, and their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of Audio Objects and Scenes as output. The OSD-ASD

AIM may also receive an Alert conveying information on potential anomalies in the input Audio Objects:

Receives	Space-Time	of the input Objects having the same time base.
	Audio Objects	individual Audio Objects.
	Scene Descriptors	Scene to Objects belong to.
Integrates	Space-Time and 3D Model Object	with Scene Descriptors.
Produces	Audio Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.8.2 Reference Model

The Reference Architecture is depicted in Figure 1.

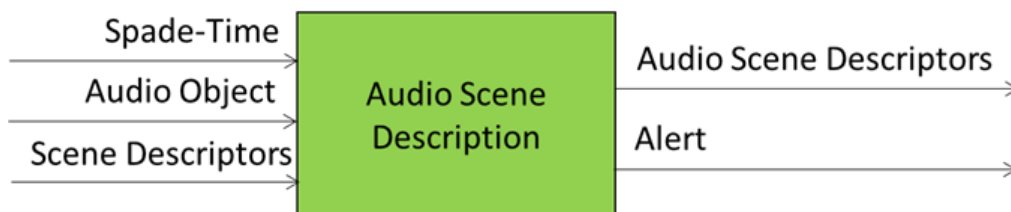


Figure 1 – The Audio Scene Description (OSD-ASD) AIM

7.1.8.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Scene Description (OSD-ASD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Audio Scene Description (OSD-ASD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Audio Objects	Input Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Audio Scene Descriptors	The output Audio Scene Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.8.4 SubAIMs

No SubAIMs.

7.1.8.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/SceneDescription.json>

7.1.8.6 Profiles

No profiles.

7.1.8.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-3SD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-3SD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Audio Objects	Shall validate against Audio Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Audio Scene Descriptors	Shall validate against Audio Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.9 Audio Event Description

7.1.9.1 Functions

The Audio Event Description (OSD-AED) AIM produces the Descriptors of a Audio Event from a sequence of Audio Scene Descriptors:

Receives	<i>Audio Scene Descriptors.</i>	A sequence.
Produces	<i>Audio Event Descriptors</i>	A file

7.1.9.2 Reference Model

The Audio Event Description (OSD-AED) AIM Reference Model is depicted in Figure 1.

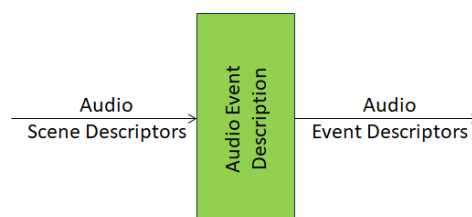


Figure 1 – The Audio Event Description (OSD-AED) AIM Reference Model

7.1.9.3 *Input/Output Data*

Table 1 specifies the Input and Output Data of the Audio Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Audio Event Description (OSD-AED) AIM

Input	Description
Audio Scene Descriptors	Sequence of Audio Scene Descriptors.
Output	Description
Audio Event Descriptors	The Audio Event Descriptors of the Audio Scene.

7.1.9.4 *SubAIMs*

No SubAIMs

7.1.9.5 *JSON Metadata*

<https://schemas.mpai.community/OSD/V1.4/AIMs/AudioEventDescription.json>

7.1.9.6 *Profiles*

No Profiles

7.1.9.7 *Conformance Testing*

Table 2 provides the Conformance Testing Method for OSD-AED AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-AED AIM

Receives	Audio Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Audio Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.10 Basic Audio-Visual Scene Description

7.1.10.1 *Functions*

The Basic Audio-Visual Scene Description (OSD-BMS) AIM receives Audio Objects, Speech Objects, Visual Objects, and Audio-Visual Objects with their Space-Time information and produces the Descriptors of a Scene composed of Audio Objects, Speech Objects, Visual Objects, and Audio-Visual Objects and optionally an Alert:

Receives	Space-Time	of the Audio Basic Visual Scene.
	Audio Objects	In input.
	Speech Objects	In input.
	Visual Objects	In input.
	Audio-Visual Objects	In input.
Processes	All Objects	
Creates	Audio, Speech, Visual, and Audio-Visual Scene Descriptors	From the Objects, if possible
Combines	Scene Descriptors	Of all Objects.
Produces	Audio-Visual Scene Descriptors	

7.1.10.2 Reference Model

Figure 1 specifies the Reference Model of the Audio-Visual Basic Scene Description (OSD-BMS) AIM.

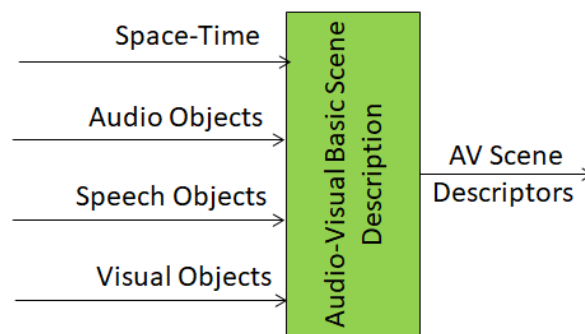


Figure 1 – The Basic Audio-Visual Scene Description (OSD-BMS) AIM

7.1.10.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio-Visual Basic Scene Description (OSD-BMS).

Table 1 – I/O Data of the Basic Audio-Visual Scene Description (OSD-BMS) AIM

Input	Description
SpaceTime	Space-Time information of Objects.
Audio Objects	Input Audio Objects.
Speech Objects	Input Speech Objects.
Visual Objects	Input Visual Objects.
Output	Description
Audio-Visual Scene Descriptors	The Audio-Visual Descriptors of the Scene.

7.1.10.4 SubAIMs

Audio-Visual Basic Scene Description (OSD-AVB) is a Composite AIM whose reference Model is depicted in Figure 2.

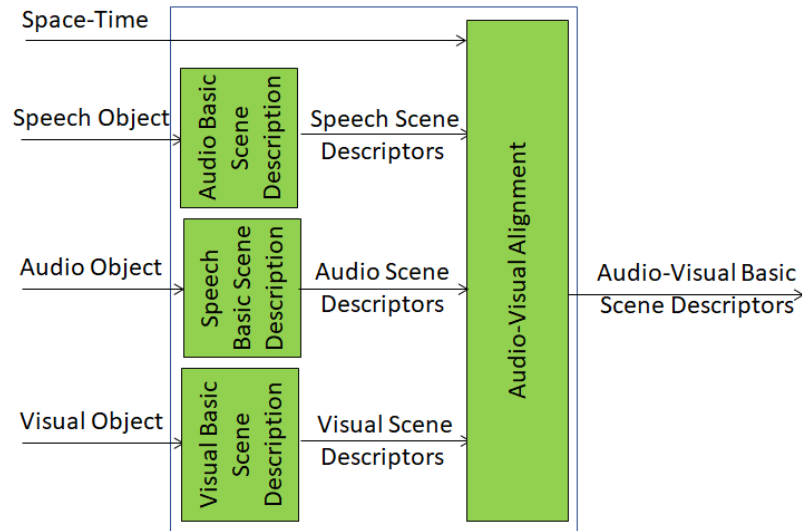


Figure 2 - Reference Model of Basic Audio-Visual Scene Description Composite (OSD-BMS) AIM

Table 2 provides the AI Modules composing the AIM.

Table 2 – AI Modules of the Audio-Visual Basic Scene Description (OSD-AVB) AIM

AIM	Acronym	AIMs	JSON
OSD-AVS		Basic Audio-Visual Basic Scene Description	X
	OSD-BBS	Basic Audio Scene Description	X
	OSD-BSS	Basic Speech Scene Description	X
	OSD-BVS	Basic Visual Scene Description	X
	OSD-AVA	Audio-Visual Alignment	X

7.1.10.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicAudioVisualSceneDescription.json>

7.1.10.6 Profiles

No Profiles.

7.1.10.7 Conformance Testing

Table 3 provides the Conformance Testing Method for OSD-BMS AIM. Conformance Testing of the individual AIMs of the OSD-BMS Composite AIM are given by the individual AIM Specification.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 3 – Conformance Testing Method for OSD-BMS AIM

Receives	Space-Time	Shall validate against Space-Time schema
	Speech Objects	Shall validate against Speech Objects schema Speech Data shall conform with Qualifier
	Audio Objects	Shall validate against Audio Objects schema Audio Data shall conform with Qualifier
	Visual Objects	Shall validate against Visual Objects schema Visual Data shall conform with Qualifier
	Audio-Visual Objects	Shall validate against Audio-Visual Objects schema Audio-Visual Data shall conform with Qualifier
Produces	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema

7.1.11 Audio-Visual Scene Description

7.1.11.1 Functions

The Audio-Visual Scene Description (OSD-MSD) AIM receives Audio-Visual Objects, the Descriptors of the Scene the Objects belong to, and their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of Audio-Visual Objects and Scenes as output. The OSD-MSD AIM may also receive an Alert conveying information on potential anomalies in the input Audio-Visual Objects:

Receives	<i>Space-Time</i>	Of output Audio-Visual Scene Descriptors,
	<i>Speech Objects</i>	
	<i>Audio Objects</i>	
	<i>Visual Objects</i>	
	<i>Audio-Visual Scene Descriptors</i>	Of Scene to be augmented.
Augments	<i>Audio-Visual Scene Descriptors</i>	
Produces	<i>Audio-Visual Scene Descriptors</i>	

7.1.11.2 Reference Model

Figure 1 specified the Reference Model of Audio-Visual Scene Description (OSD-AVS) aim.

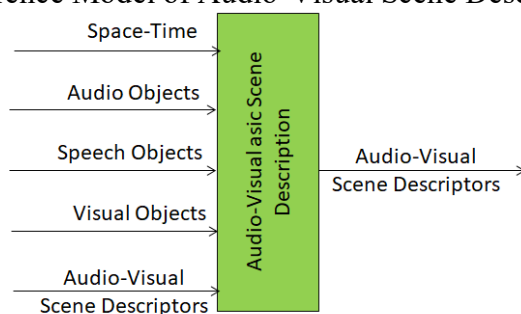


Figure 1 – The Audio-Visual Scene Description (OSD-AVS) AIM

7.1.11.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio-Visual Scene Description (OSD-AVS) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Audio-Visual Scene Description (OSD-AVS) AIM

Input	Description
Space-Time	Space-Time information of output Audio-Visual Scene Descriptors
Speech Object	Speech Object
Audio Objects	Audio Objects.
Visual Objects	Visual Objects.
Audio-Visual Scene Descriptors	The Audio-Visual Descriptors of the Scene part of the target Audio-Visual Scene.
Output	Description
Audio-Visual Scene Descriptors	The Audio-Visual Descriptors of the Scene.

7.1.11.4 SubAIMs

Figure 2 specified the Reference Model of Audio-Visual Scene Description (CAE-ASD) Composite AIM.

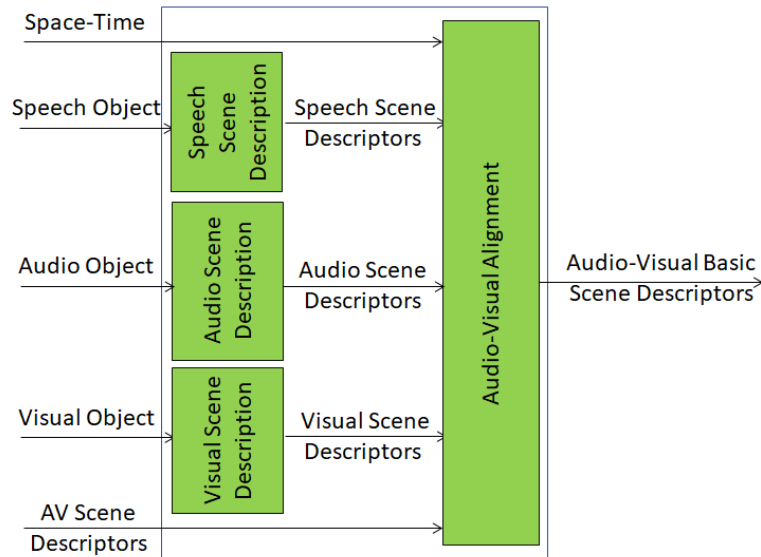


Figure 2 – The Audio-Visual Scene Description (OSD-AVS) Composite AIM

Table 2 provides the links to the specifications of the OSD-AVS AIMs.

Table 2 – AIMs of the Audio-Visual Scene Description (OSD-AVS) Composite AIM

AIMs	Names	JSON
OSD-SSD	Speech Scene Description	X
OSD-ASD	Audio Scene Description	X
OSD-VSD	Visual Scene Description	X

7.1.11.5 JSON Metadata

<http://schemas.mpai.community/OSD/V1.4/AIMs/AudioVisualSceneDescription.json>

7.1.11.6 Profiles

No Profiles.

7.1.11.6.1 Disclaimers

1. This OSD-AVS Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this OSD-AVS Reference Software is to show a working Implementation of OSD-AVS, not to provide a ready-to-use product.
3. MPAI disclaims the suitability of the Software for any other purposes and does not guarantee that it is secure.
4. Use of this Reference Software may require acceptance of licences from the respective repositories. Users shall verify that they have the right to use any third-party software required by this Reference Software.

7.1.11.6.2 Guide to the OSD-AVS code

OSD-AVS arranges the aligned visual and speech objects into [Audio-Visual Scene Descriptors](#). Use of this Reference Software for the OSD-AVS AI Module is for developers who are familiar with Python, Docker, and RabbitMQ.

The OSD-AVS Reference Software is found at the MPAI [gitlab](#) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image.

7.1.11.6.3 Acknowledgements

This OSD-AVS Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

7.1.11.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-AVS AIM. AIM. Conformance Testing of the individual AIMs of the OSD-AVS Composite AIM are given by the individual AIM Specification.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-AVS AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Speech Objects	Shall validate against Speech Objects schema. Speech Data shall conform with Qualifier.
	Audio Objects	Shall validate against Audio Objects schema. Audio Data shall conform with Qualifier.

	Visual Objects	Shall validate against Visual Objects schema. Visual Data shall conform with Qualifier.
Produces	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema.

7.1.12 Audio-Visual Event Description

7.1.12.1 Functions

The Audio-Visual Event Description (OSD-MED) AIM produces the Descriptors of an Audio-Visual Event from a sequence of Audio-Visual Scene Descriptors:

Receives	<i>Audio-Visual Scene Descriptors</i>	A sequence.
Produces	<i>Audio-Visual Event Descriptors</i>	

7.1.12.2 Reference Model

The Audio-Visual Event Description (OSD-MED) AIM Reference Model is depicted in Figure 1.

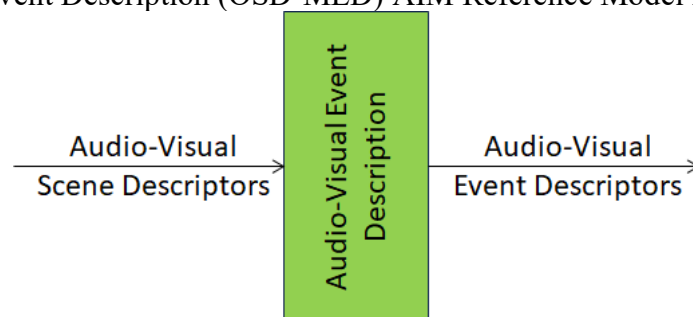


Figure 1 – The Audio-Visual Event Description (OSD-MED) AIM Reference Model

7.1.12.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio-Visual Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Audio-Visual Event Description (OSD-MED) AIM

Input	Description
Audio-Visual Scene Descriptors	Sequence of Audio-Visual Scene Descriptors.
Output	Description
Audio-Visual Event Descriptors	The Audio-Visual Event Descriptors of the Audio-Visual Scene.

7.1.12.4 SubAIMs

No SubAIMs

7.1.12.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/AudioVisualEventDescription.json>

7.1.12.6 Profiles

No Profiles

7.1.12.7 Reference Software

1. This OSD-MED Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to show a working Implementation of OSD-MED, not to provide a ready-to-use product.
3. MPAI disclaims the suitability of the Software for any other purposes and does not guarantee that it is secure.
4. Use of this Reference Software may require acceptance of licences from the respective repositories. Users shall verify that they have the right to use any third-party software required by this Reference Software.

7.1.12.7.1 Guide to the OSD-AVE code

OSD-MED arranges the audio-visual scene descriptors from OSD-AVS into [Audio-Visual Event Descriptors](#).

Use of this Reference Software for the OSD-MED AI Module is for developers who are familiar with Python, Docker, and RabbitMQ.

The OSD-MED Reference Software is found at the MPAI [gitlab](#) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image.

7.1.12.7.2 Acknowledgements

This version of the OSD-MED Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

7.1.12.8 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-MED AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-MED AIM

Receives	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Audio-Visual Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.13 Basic LiDAR Scene Description

7.1.13.1 Functions

Basic LiDAR Scene Description (OSD-BLD) V1.3 produces the Descriptors of a Scene composed of LiDAR Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	LiDAR Objects	Individual Objects.
Produces	Basic LiDAR Scene Descriptors	Scene Descriptors produced by AIM.

	Alert	Data concerning potential anomalies in input Object.
--	-------	--

7.1.13.2 Reference Model

The Reference Architecture is depicted in Figure 1.

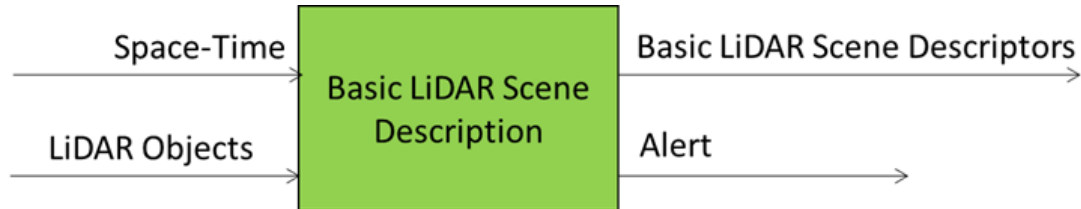


Figure 1 – The Basic LiDAR Scene Description (OSD-BLD) AIM

7.1.13.3 Input/Output Data

Table 1 specifies the Input and Output Data of the LiDAR Scene Description (OSD-BLD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic LiDAR Scene Description (OSD-BLD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
LiDAR Objects	Input Objects.
Output	Description
Basic LiDAR Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.13.4 SubAIMs

No SubAIMs.

7.1.13.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicLiDARSceneDescription.json>

7.1.13.6 Profiles

No profiles.

7.1.13.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BLD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BLD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	LiDAR Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic LiDAR Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.13.8

7.1.14 LiDAR Scene Description

7.1.14.1 *Functions*

LiDAR Scene Description (OSD-LSD) V1.3 produces the Descriptors of a Scene composed by LiDAR Objects and Scenes:

Receives	Space-Time	of the input Objects having the same time base.
	LiDAR Objects	Individual LiDAR Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and LiDAR Object	with Scene Descriptors.
Produces	LiDAR Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.14.2 *Reference Model*

The Reference Architecture is depicted in Figure 1.

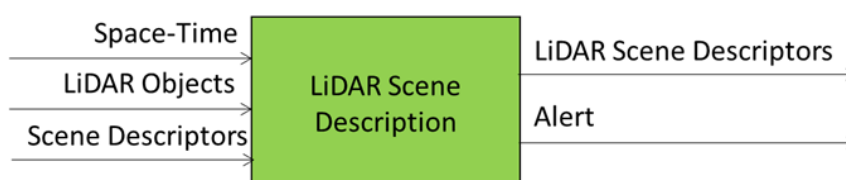


Figure 1 – The LiDAR Scene Description (OSD-LSD) AIM

7.1.14.3 *Input/Output Data*

Table 1 specifies the Input and Output Data of the LiDAR Scene Description (OSD-LSD) AIM. .

Table 1 – I/O Data of the LiDAR Scene Description (OSD-LSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
LiDAR Objects	Input LiDAR Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
LiDAR Scene Descriptors	The output LiDAR Descriptors.
Alert	Data signaling potential anomalies in Object.

7.1.14.4 SubAIMs

No SubAIMs.

7.1.14.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/LiDARSceneDescription.json>

7.1.14.6 Profiles

No profiles.

7.1.14.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-LSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-3SD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	LiDAR Objects	Shall validate against LiDAR Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	LiDAR Scene Descriptors	Shall validate against LiDAR Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.15 LiDAR Event Description

7.1.15.1 Functions

LiDAR Event Description (OSD-LIE) V1.3 produces the Descriptors of a LiDAR Event from a sequence of Visual Scene Descriptors:

Receives	<i>LiDAR Scene Descriptors</i> .	A sequence.
Produces	<i>LiDAR Event Descriptors</i>	a file.

7.1.15.2 Reference Model

The LiDAR Event Description (OSD-LIE) AIM Reference Model is depicted in Figure 1.

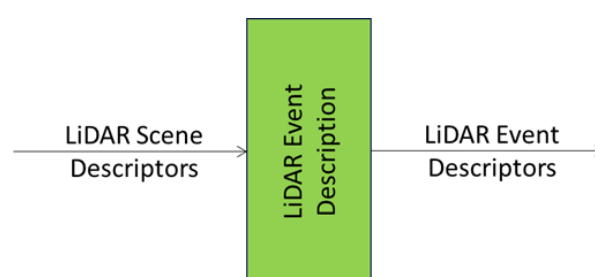


Figure 1 – The LiDAR Event Description (OSD-LIE) AIM Reference Model

7.1.15.3 Input/Output Data

Table 1 specifies the Input and Output Data of the LiDAR Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the LiDAR Event Description (OSD-LIE) AIM

Input	Description
LiDAR Scene Descriptors	Sequence of LiDAR Scene Descriptors.
Output	Description
LiDAR Event Descriptors	The LiDAR Event Descriptors of the LiDAR Scene.

7.1.15.4 SubAIMs

No SubAIMs

7.1.15.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/LiDAREventDescription.json>

7.1.15.6 Profiles

No Profiles

7.1.15.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-LIE AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-LIE AIM

Receives	LiDAR Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	LiDAR Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.16 Basic Offline Map Scene Description

7.1.16.1 Functions

Basic Offline Map Scene Description (OSD-BOD) V1.4 produces the Descriptors of a Scene composed of LiDAR Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	Offline Map Objects	Individual Objects.
Produces	Basic Offline Map Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.16.2 Reference Model

The Reference Architecture is depicted in Figure 1.

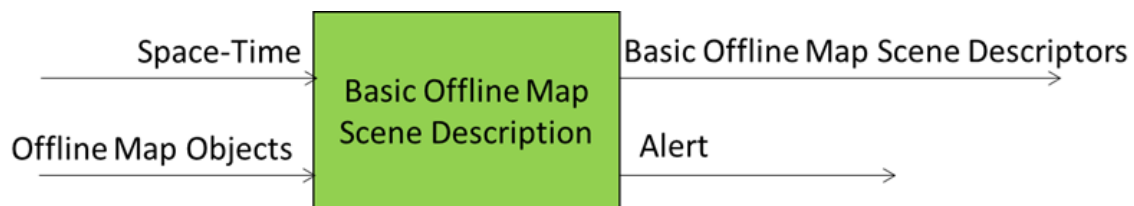


Figure 1 – The Basic Offline Map Scene Description (OSD-BOD) AIM

7.1.16.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Offline Map Scene Description (OSD-BOD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic Offline Map Scene Description (OSD-BOD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Offline Map Objects	Input Objects.
Output	Description
Basic Offline Map Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.16.4 SubAIMs

No SubAIMs.

7.1.16.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicOfflineMapSceneDescription.json>

7.1.16.6 Profiles

No profiles.

7.1.16.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BOD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BOD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Offline Map Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Offline Map Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.16.8 Offline Map Scene Description

7.1.16.9 Functions

Offline Map Scene Description (OSD-OSD) produces the Descriptors of a Scene composed by Offline Map Objects and Scenes:

Receives	Space-Time	of the input Objects having the same time base.
	Offline Map Objects	Individual Offline Map Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and Offline Map Object	with Scene Descriptors.
Produces	Offline Map Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.16.10 Reference Model

The Reference Architecture is depicted in Figure 1.

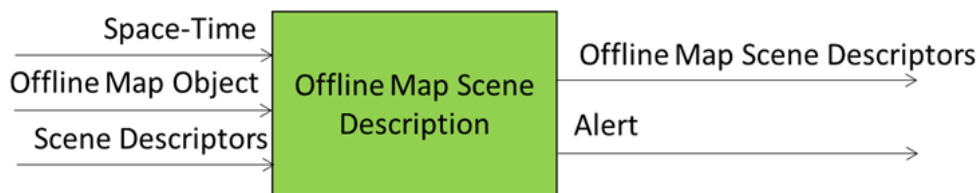


Figure 1 – The Offline Map Scene Description (OSD-OSD) AIM

7.1.16.11 Input/Output Data

Table 1 specifies the Input and Output Data of the Offline Map Scene Description (OSD-OSD) AIM. .

Table 1 – I/O Data of the Offline Map Scene Description (OSD-OSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Offline Map Objects	Input Offline Map Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Offline Map Scene Descriptors	The output Offline Map Scene Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.16.12 SubAIMs

No SubAIMs.

7.1.16.13 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/OfflineMapSceneDescription.json>

7.1.16.14 Profiles

No profiles.

7.1.16.15 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-OSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-OSD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Offline Map Objects	Shall validate against Offline Map Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Offline Map Scene Descriptors	Shall validate against Offline Map Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.17 Offline Map Event Description

7.1.17.1 Functions

Offline Map Event Description (OSD-OME) produces the Descriptors of of a Offline Map Event from a sequence of Offline Map Scene Descriptors:

Receives	<i>Offline Map Scene Descriptors.</i>	A sequence.
Produces	<i>Offline Map Event Descriptors</i>	A File.

7.1.17.2 Reference Model

The Offline Map Event Description (OSD-OME) AIM Reference Model is depicted in Figure 1.

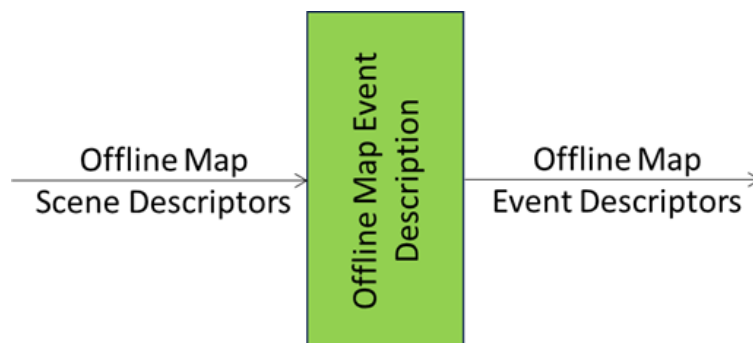


Figure 1 – The Offline Map Event Description (OSD-OME) AIM Reference Model

7.1.17.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Offline Map Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Offline Map Event Description (OSD-OME) AIM

Input	Description
Offline Map Scene Descriptors	Sequence of Offline Map Scene Descriptors.
Output	Description
Offline Map Event Descriptors	The Offline Map Event Descriptors of the Offline Map Scene.

7.1.17.4 SubAIMs

No SubAIMs

7.1.17.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/OfflineMapEventDescription.json>

7.1.17.6 Profiles

No Profiles

7.1.17.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-OME AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-OME AIM

Receives	Offline Map Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Offline Map Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.18 Basic RADAR Scene Description

7.1.18.1 Functions

Basic RADAR Scene Description (OSD-BRD) produces the Descriptors of a Scene composed of RADAR Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
----------	------------	---

	RADAR Objects	Individual Objects.
Produces	Basic RADAR Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.18.2 Reference Model

The Reference Architecture is depicted in Figure 1.

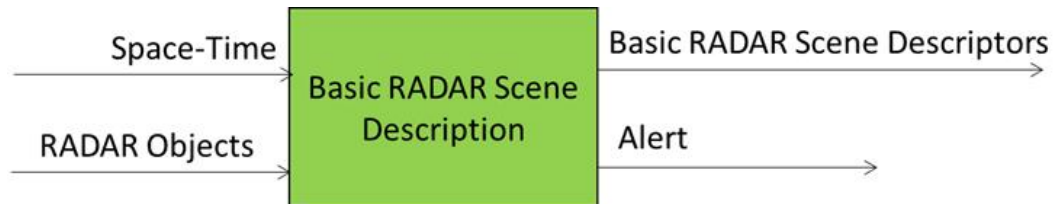


Figure 1 – The Basic RADAR Scene Description (OSD-BRD) AIM

7.1.18.3 Input/Output Data

Table 1 specifies the Input and Output Data of the RADAR Scene Description (OSD-BRD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic RADAR Scene Description (OSD-BRD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
RADAR Objects	Input Objects.
Output	Description
Basic RADAR Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.18.4 SubAIMs

No SubAIMs.

7.1.18.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicRADARSceneDescription.json>

7.1.18.6 Profiles

No profiles.

7.1.18.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BRD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BRD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	RADAR Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic RADAR Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.18.8 RADAR Scene Description

7.1.18.9 Functions

RADAR Scene Description (OSD-RSD) produces the Descriptors of a Scene composed by RADAR Objects and Scenes:

Receives	Space-Time	of the input Objects having the same time base.
	RADAR Objects	Individual RADAR Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and RADAR Object	with Scene Descriptors.
Produces	RADAR Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.18.10 Reference Model

The Reference Architecture is depicted in Figure 1.

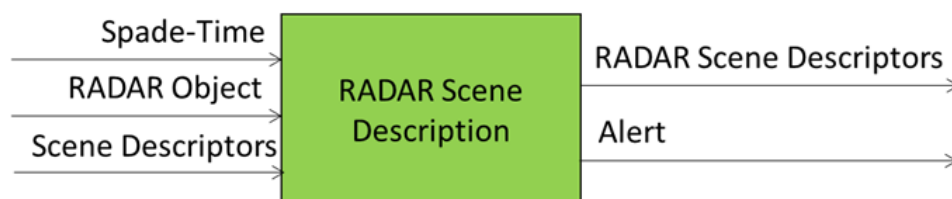


Figure 1 – The RADAR Scene Description (OSD-RSD) AIM

7.1.18.11 *Input/Output Data*

Table 1 specifies the Input and Output Data of the RADAR Scene Description (OSD-RSD) AIM. .

Table 1 – I/O Data of the RADAR Scene Description (OSD-RSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
RADAR Objects	Input RADAR Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
RADAR Scene Descriptors	The output RADAR Scene Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.18.12 *SubAIMs*

No SubAIMs.

7.1.18.13 *JSON Metadata*

<https://schemas.mpai.community/OSD/V1.4/AIMs/RADARSceneDescription.json>

7.1.18.14 *Profiles*

No profiles.

7.1.18.15 *Conformance Testing*

Table 2 provides the Conformance Testing Method for OSD-RSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-RSD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	RADAR Objects	Shall validate against RADAR Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	RADAR Scene Descriptors	Shall validate against RADAR Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.19 RADAR Event Description

7.1.19.1 Functions

RADAR Event Description (OSD-RAE) produces the Descriptors of a RADAR Event from a sequence of Visual Scene Descriptors:

Receives	<i>RADAR Scene Descriptors</i> .	A sequence.
Produces	<i>RADAR Event Descriptors</i>	a fille.

7.1.19.2 Reference Model

The RADAR Event Description (OSD-RAE) AIM Reference Model is depicted in Figure 1.

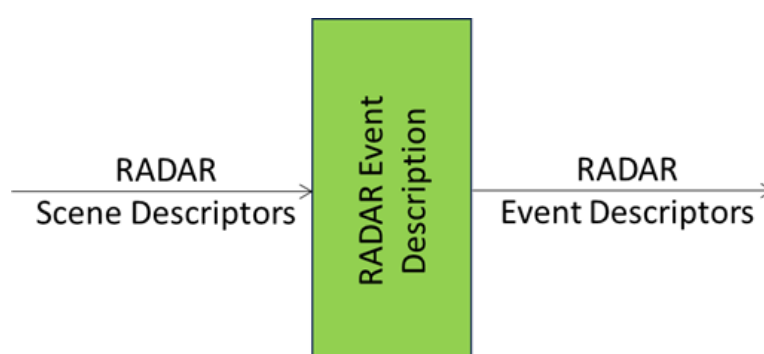


Figure 1 – The RADAR Event Description (OSD-RAE) AIM Reference Model

7.1.19.3 Input/Output Data

Table 1 specifies the Input and Output Data of the RADAR Event Description AIM. Ranks are to the Data Type specifications.

Table 1 – I/O Data of the RADAR Event Description (OSD-RAE) AIM

Input	Description
RADAR Scene Descriptors	Sequence of RADAR Scene Descriptors.
Output	Description
RADAR Event Descriptors	The RADAR Event Descriptors of the RADAR Scene.

7.1.19.4 SubAIMs

No SubAIMs

7.1.19.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/RADAREventDescription.json>

7.1.19.6 Profiles

No Profiles

7.1.19.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-RAE AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-RAE AIM

Receives	RADAR Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	RADAR Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.20 Basic Speech Scene Description

7.1.20.1 Functions

The Basic Speech Scene Description (OSD-BSD) receives Speech Objects with their Space-Time information and produces the Descriptors of a Scene composed of Speech Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	Speech Objects	Individual Objects.
Produces	Basic Speech Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.20.2 Reference Model

The Reference Architecture is depicted in Figure 1.

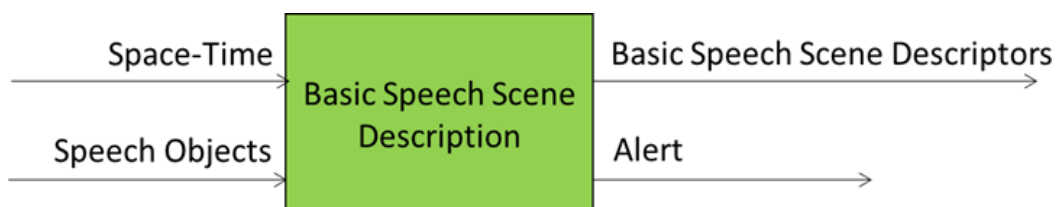


Figure 1 – The Basic Speech Scene Description (OSD-BSD) AIM

7.1.20.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Speech Scene Description (OSD-BSD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic Speech Scene Description (OSD-BSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Speech Objects	Input Objects.
Output	Description
Basic Speech Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.20.4 SubAIMs

No SubAIMs.

7.1.20.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicSpeechSceneDescription.json>

7.1.20.6 Profiles

No profiles.

7.1.20.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BSD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Speech Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Speech Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.21 Speech Scene Description

7.1.21.1 Functions

The Speech Scene Description (OSD-SSD) AIM receives Speech Objects their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of Speech Objects and Speech Scenes as output. The OSD-SSD AIM may also receive an Alert conveying information on potential anomalies in the input Speech Objects:

Receives	Space-Time	of the input Objects having the same time base.
	Speech Objects	Individual Speech Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and Speech Object	with Scene Descriptors.
Produces	Speech Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signaling potential anomalies in Object.

7.1.21.2 Reference Model

The Reference Architecture is depicted in Figure 1.

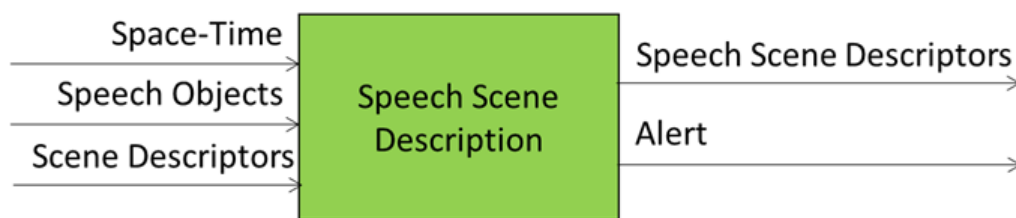


Figure 1 – The Speech Scene Description (OSD-SSD) AIM

7.1.21.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Speech Scene Description (OSD-SSD) AIM. .

Table 1 – I/O Data of the Speech Scene Description (OSD-SSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Speech Objects	Input Speech Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Speech Scene Descriptors	The output Speech Scene Descriptors.

Alert	Data signalling potential anomalies in Object.
-----------------------	--

7.1.21.4 SubAIMs

No SubAIMs.

7.1.21.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/SpeechSceneDescription.json>

7.1.21.6 Profiles

No profiles.

7.1.21.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-SSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-SSD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Speech Objects	Shall validate against Speech Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Speech Scene Descriptors	Shall validate against Speech Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.22 Speech Event Description

7.1.22.1 Functions

The Speech Event Description (OSD-SED) AIM produces the Descriptors of a Speech Event from a sequence of Speech Scene Descriptors:

Receives	<i>Speech Scene Descriptors.</i>	A sequence.
Produces	<i>Speech Event Descriptors</i>	A file

7.1.22.2 Reference Model

The Speech Event Description (OSD-SED) AIM Reference Model is depicted in Figure 1.

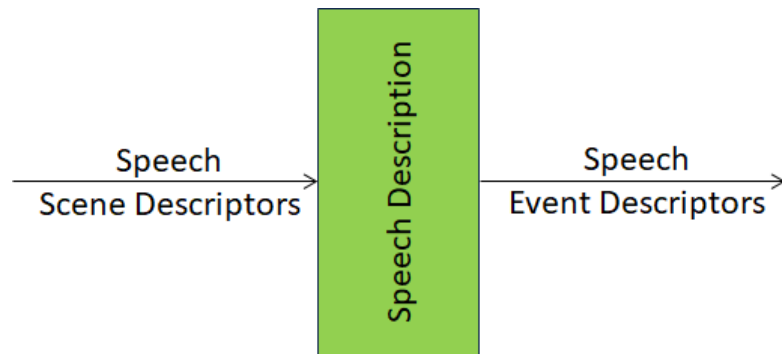


Figure 1 – The Speech Event Description (OSD-SED) AIM Reference Model

7.1.22.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Speech Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Speech Event Description (OSD-SED) AIM

Input	Description
Speech Scene Descriptors	Sequence of Speech Scene Descriptors.
Output	Description
Speech Event Descriptors	The Speech Event Descriptors of the Speech Scene.

7.1.22.4 SubAIMs

No SubAIMs

7.1.22.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/SpeechEventDescription.json>

7.1.22.6 Profiles

No Profiles

Under development.

7.1.22.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-SED AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-SED AIM

Receives	Speech Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Speech Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.23 Basic Ultrasound Scene Description

7.1.23.1 Functions

Basic Ultrasound Scene Description (OSD-BUD) produces the Descriptors of a Scene composed of Ultrasound Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	Ultrasound Objects	Individual Objects.
Produces	Basic Ultrasound Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.23.2 Reference Model

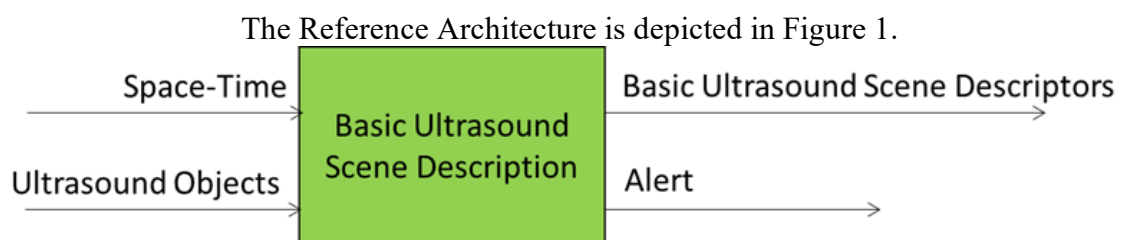


Figure 1 – The Basic Ultrasound Scene Description (OSD-BUD) AIM

7.1.23.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Ultrasound Scene Description (OSD-BUD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic Ultrasound Scene Description (OSD-BUD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Ultrasound Objects	Input Objects.
Output	Description
Basic Ultrasound Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.23.4 SubAIMs

No SubAIMs.

7.1.23.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicUltrasoundSceneDescription.json>

7.1.23.6 Profiles

No profiles.

7.1.23.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BUD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BUD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Ultrasound Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Ultrasound Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.24 Ultrasound Scene Description

7.1.24.1 Functions

Ultrasound Scene Description (OSD-USD produces the Descriptors of a Scene composed by Ultrasound Objects and Scenes:

Receives	Space-Time	of the input Objects having the same time base.
	Ultrasound Objects	Individual Ultrasound Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and Ultrasound Object	with Scene Descriptors.
Produces	Ultrasound Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.24.2 Reference Model

The Reference Architecture is depicted in Figure 1.

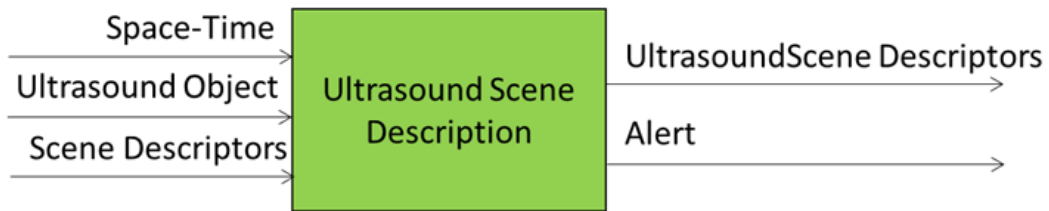


Figure 1 – The Ultrasound Scene Description (OSD-USD) AIM

7.1.24.3 3 Input/Output Data

Table 1 specifies the Input and Output Data of the Ultrasound Scene Description (OSD-USD) AIM. .

Table 1 – I/O Data of the Ultrasound Scene Description (OSD-USD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Ultrasound Objects	Input Ultrasound Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Ultrasound Scene Descriptors	The output Ultrasound Scene Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.24.4 SubAIMs

No SubAIMs.

7.1.24.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/UltraSoundSceneDescription.json>

7.1.24.6 Profiles

No profiles.

7.1.24.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-USD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-USD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Ultrasound Objects	Shall validate against Ultrasound Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Ultrasound Scene Descriptors	Shall validate against Ultrasound Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.25 Ultrasound Event Description

7.1.25.1 Functions

Ultrasound Event Description (OSD-USE) V1.3 produces the Descriptors of an Ultrasound Event from a sequence of its Ultrasound Scene Descriptors:

Receives	<i>Ultrasound Scene Descriptors.</i>	A sequence of Ultrasound Scene Descriptors.
Produces	<i>Ultrasound -Visual Event Descriptors</i>	The output Ultrasound Event Descriptors

7.1.25.2 Reference Model

The Ultrasound Event Description (OSD-USE) AIM Reference Model is depicted in Figure 1.

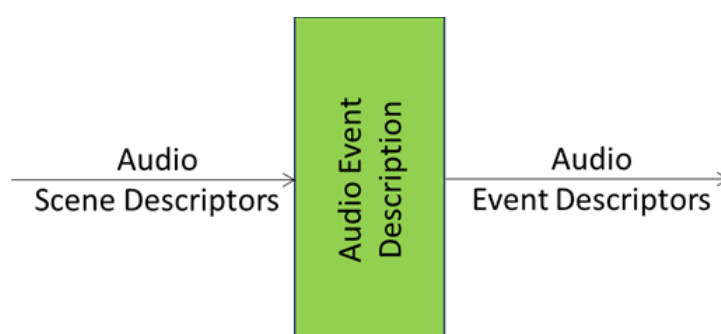


Figure 1 – The Ultrasound Event Description (OSD-USE) AIM Reference Model

7.1.25.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Ultrasound Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Ultrasound Event Description (OSD-USE) AIM

Input	Description
Ultrasound Scene Descriptors	Sequence of Ultrasound Scene Descriptors.
Output	Description
Ultrasound Event Descriptors	The Ultrasound Event Descriptors of the Ultrasound Scene.

7.1.25.4 SubAIMs

No SubAIMs

7.1.25.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/UltrasoundEventDescription.json>

7.1.25.6 Profiles

No Profiles

7.1.25.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-USE AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-USE AIM

Receives	Ultrasound Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Ultrasound Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.26 Basic Visual Scene Description

7.1.26.1 Functions

The Basic Visual Scene Description (OSD-BVD) receives Visual Objects with their Space-Time information and produces the Descriptors of a Scene composed of Visual Objects and optionally an Alert:

Receives	Space-Time	of the input Objects having the same time base.
	Visual Objects	Individual Objects.
Produces	Basic Visual Scene Descriptors	Scene Descriptors produced by AIM.
	Alert	Data concerning potential anomalies in input Object.

7.1.26.2 Reference Model

The Reference Architecture is depicted in Figure 1.

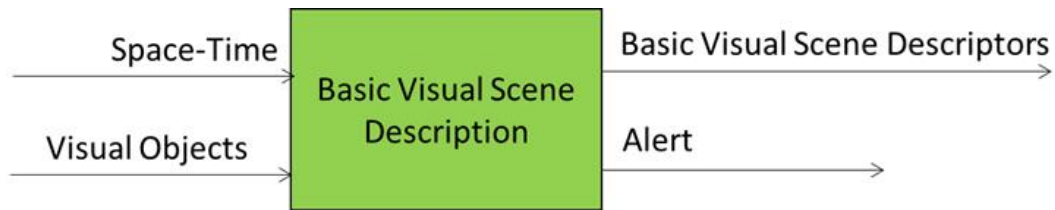


Figure 1 – The Basic Visual Scene Description (OSD-BVD) AIM

7.1.26.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Scene Description (OSD-BVD) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Basic Visual Scene Description (OSD-BVD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Visual Objects	Input Objects.
Output	Description
Basic Visual Scene Descriptors	The output Descriptors.
Alert	Data concerning potential anomalies in input Object.

7.1.26.4 SubAIMs

No SubAIMs.

7.1.26.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/BasicVisualSceneDescription.json>

7.1.26.6 Profiles

No profiles.

7.1.26.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-BVD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-BVD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Visual Objects	Shall validate against Object schema. Media-specific Data shall conform with their Qualifiers.
Produces	Basic Visual Scene Descriptors	Shall validate against Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.26.8 *Visual Scene Description*

7.1.26.9 *Functions*

The Visual Scene Description (OSD-VSD) AIM receives Visual Objects, the Descriptors of the Scene the Objects belong to, and their Space-Time information as inputs and produces the Descriptors of a Scene that is composed of Visual Objects and Scenes as output. The OSD-VSD AIM may also receive an Alert conveying information on potential anomalies in the input Visual Objects:

Receives	Space-Time	of the input Objects having the same time base.
	Visual Objects	Individual Visual Objects.
	Scene Descriptors	Scene the Objects belong to.
Integrates	Space-Time and Visual Object	with Scene Descriptors.
Produces	Visual Scene Descriptors	Output#1 of AIM
	Alert	Output#2 of AIM signalling potential anomalies in Object.

7.1.26.10 *Reference Model*

The Reference Architecture is depicted in Figure 1.

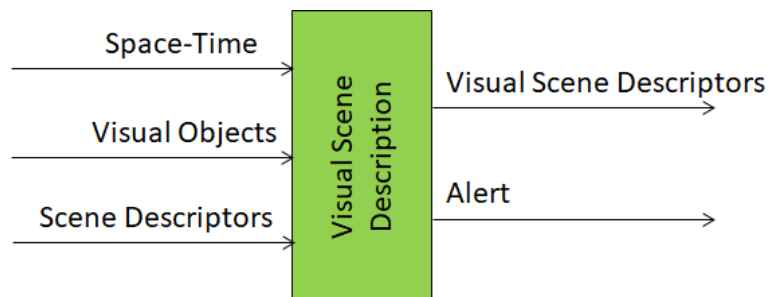


Figure 1 – The Visual Scene Description (OSD-VSD) AIM

7.1.26.11 *Input/Output Data*

Table 1 specifies the Input and Output Data of the Visual Scene Description (OSD-VSD) AIM. .

Table 1 – I/O Data of the Visual Scene Description (OSD-VSD) AIM

Input	Description
Space-Time	Space-Time of input Objects.
Visual Objects	Input Visual Objects.
Scene Descriptors	Input Scene Descriptors.
Output	Description
Visual Scene Descriptors	The output Visual Scene Descriptors.
Alert	Data signalling potential anomalies in Object.

7.1.26.12 *SubAIMs*

No SubAIMs.

7.1.26.13 *JSON Metadata*

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualSceneDescription.json>

7.1.26.14 *Profiles*

No profiles.

7.1.26.15 *Conformance Testing*

Table 2 provides the Conformance Testing Method for OSD-VSD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-VSD AIM

Receives	Space-Time	Shall validate against Space-Time schema.
	Visual Objects	Shall validate against Visual Object schema. Media-specific Data shall conform with their Qualifiers.
	Scene Descriptors	Shall validate against Scene Descriptors schema.
Produces	Visual Scene Descriptors	Shall validate against Visual Scene Descriptors schema.
	Alert	Shall validate against Alert schema.

7.1.26.16 *Visual Event Description*

7.1.26.17 *Functions*

The Visual Event Description (OSD-VED) AIM produces the Descriptors of a Visual Event from a sequence of Visual Scene Descriptors:

Receives	<i>Visual Scene Descriptors.</i>	A sequence.
Produces	<i>Visual Event Descriptors</i>	A file

7.1.26.18 *Reference Model*

The Visual Event Description (OSD-VED) AIM Reference Model is depicted in Figure 1.

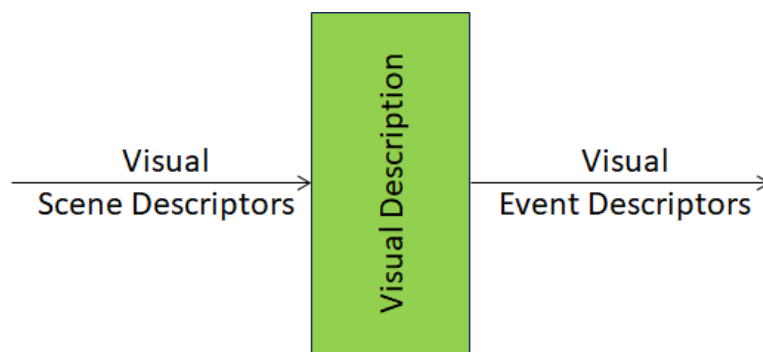


Figure 1 – The Visual Event Description (OSD-VED) AIM Reference Model

7.1.26.19 *Input/Output Data*

Table 1 specifies the Input and Output Data of the Visual Event Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Visual Event Description (OSD-VED) AIM

Input	Description
Visual Scene Descriptors	Sequence of Visual Scene Descriptors.
Output	Description
Visual Event Descriptors	The Visual Event Descriptors of the 3D Model Scene.

7.1.26.20 *SubAIMs*

No SubAIMs

7.1.26.21 *JSON Metadata*

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualEventDescription.json>

7.1.26.22 Profiles

No Profiles

7.1.26.23 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-VED AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-VED AIM

Receives	Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema
Produces	Visual Event Descriptors	Shall validate against AV Event Descriptors schema

7.1.27 Audio-Visual Alignment

7.1.27.1 Functions

The Audio-Visual Alignment (OSD-AVA) AIM provides the Descriptors of an Audio-Visual Scene whose Audio Objects, Speech Objects, 3D Model Objects, and Visual Objects have compatible Identifiers if they have the same Position.

Receives	<i>Speech Scene Descriptors</i>	Descriptors of potentially present Speech Scene.
	<i>Audio Scene Descriptors</i>	Descriptors of potentially present Audio Scene.
	<i>Visual Scene Descriptors</i>	Descriptors of potentially present Visual Scene.
	<i>3D Model Scene Descriptors</i>	Descriptors of potentially present 3D Model Scene.
Aligns	Speech, Audio, and Visual Objects	Sharing the same Spatial Attitude
Produces	<i>Audio-Visual Scene Descriptors</i>	Where Speech Objects, Audio Objects, 3D Model Objects, and Visual Objects have compatible Identifiers if they have the same Spatial Attitude.

7.1.27.2 Reference Model

Figure 1 specifies the Reference Model of the Audio-Visual Alignment (OSD-AVA) AIM.

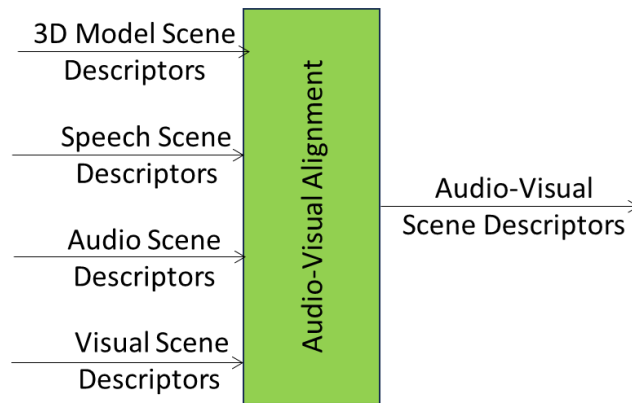


Figure 1 - Reference Model of the Audio-Visual Alignment (OSD-AVA) AIM

7.1.27.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio-Visual Alignment (OSD-AVA) AIM.

Table 1 – I/O Data of the Audio-Visual Alignment AIM

Input	Description
Speech Scene Descriptors	The IDs and the geometry of the Speech Objects of the Scene.
Audio Scene Descriptors	The IDs and the geometry of the Audio Objects of the Scene.
Visual Scene Descriptors	The IDs and the geometry of the Audio Objects of the Scene.
3D Model Scene Descriptors	
Output	Description
Audio-Visual Scene Descriptors	The IDs and the geometry of the Audio, Speech, 3D Model, Visual and Audio-Visual Objects of the Scene.

7.1.27.4 SubAIMs

No SubAIMs.

7.1.27.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/AudioVisualAlignment.json>

7.1.27.6 Profiles

No profiles.

7.1.27.7 Reference Software

7.1.27.7.1 Disclaimers

1. This OSD-AVA Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to show a working Implementation of OSD-AVA, not to provide a ready-to-use product.
3. MPai disclaims the suitability of this Reference Software for any other purposes and does not guarantee that it is secure.

4. Use of this Reference Software may require acceptance of licences from the respective repositories. Users shall verify that they have the right to use any third-party software required by this Reference Software.

7.1.27.7.2 Guide to OSD-AVA code

OSD-AVA arranges the output [Visual Objects](#) and [Speech Objects](#) with the corresponding Time information: scene cuts/transitions and speakers' turns. Each Object is bounded by two adjacent times from a list of unique times that are either 1) scene cuts/transitions or 2) starts and ends of speakers' turns.

Use of this Reference Software for the OSD-AVA AI Module is for developers who are familiar with Python, Docker, and RabbitMQ.

OSD-AVA computes segments as unique intervals from scene bounds and from speech segments. Moreover, OSD-AVA outputs visual objects and speech objects.

The OSD-AVA Reference Software is found at the MPAI [gitlab](#) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image.

7.1.27.7.3 Acknowledgements

This version of the OSD-AVA Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

7.1.27.8 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-AVA AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-AVA AIM

Receives	Speech Scene Descriptors	Shall validate against Speech Scene Descriptors schema
	Audio Scene Descriptors	Shall validate against Audio Scene Descriptors schema
	Visual Scene Descriptors	Shall validate against Visual Scene Descriptors schema
	3D Model Scene Descriptors	Shall validate against 3D Model Scene Descriptors schema
Produces	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema

7.1.27.9 Performance Assessment

Performance Assessment of an OSD-AVA AIM Implementation shall be performed using a dataset of scenes containing Audio and/or Speech and Visual objects.

The Performance Assessment Report of an OSD-AVA AIM Implementation shall include:

1. The Identifier of the OSD-AVA AIM whose Performance is being Assessed.
2. The Identifier of the scene dataset used which include the identifiers of the aligned objects.
3. The data type of the scenes: analogue, digital, without or with separated objects.

4. The Performance of the OSD-AVA AIM expressed as the number of times the OSD-AVA AIM being Assessed for Performance:
 - Correctly identifies as aligned the objects that the data set declares as aligned divided by the total number of aligned objects (Truly aligned objects).
 - Incorrectly identifies as aligned the object that the dataset declares aligned in the dataset divided by the total number of aligned objects (Falsely aligned objects).
 - Incorrectly identifies as non-aligned object that are declared aligned in the dataset referenced in 2 divided by the total number of aligned objects (Missed aligned objects).

7.1.28 Audio-Visual Scene Demultiplexing

7.1.28.1 Functions

The Audio-Visual Scene Demultiplexing (OSD-SDX) receives Audio-Visual Scene Descriptors and extracts the component elements - Audio Scene Geometry, Speech Scene Geometry, Visual Scene Geometry, Audio Objects, Speech Objects, and Visual Objects - from Audio-Visual Scene Descriptors:

Receives	<i>Audio-Visual Scene Descriptors</i>
Demultiplexes	<i>Audio-Visual Scene Descriptors</i>
Produces	<i>Speech Scene Geometry</i>
	<i>Audio Scene Geometry</i>
	<i>Visual Scene Geometry</i>
	<i>Speech Objects</i>
	<i>Audio Objects</i>
	<i>Visual Objects</i>

7.1.28.2 Reference Model

Figure 1 depicts the Reference Model of the Audio-Visual Scene Demultiplexing (OSD-SDX) AIM.

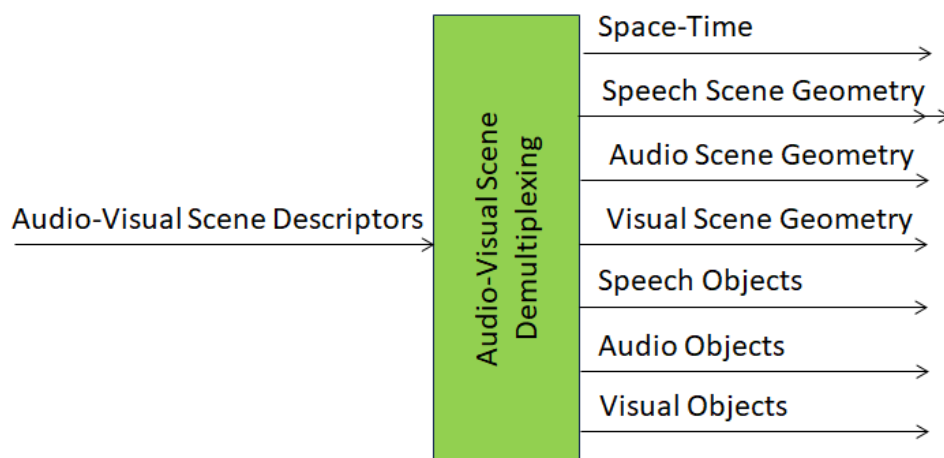


Figure 1 – Audio-Visual Scene Demultiplexing (OSD-SDX) AIM Reference Model

7.1.28.3 Input/Output Data

Table 1 specifies the Input and Output Data of the of the Audio-Visual Scene Demultiplexing (OSD-SDX) AIM.

Table 1 – I/O Data of the Audio-Visual Scene Demultiplexing (OSD-SDX) AIM

Input	Description
Audio-Visual Scene Descriptors	The Descriptors of the Audio-Visual Scene.
Output	Description
Space-Time	Space-Time information of the Audio-Visual Scene
Speech Scene Geometry	The Descriptors of the Speech Scene.
Audio Scene Geometry	The Descriptors of the Audio Scene.
Visual Scene Geometry	The Descriptors of the Visual Scene.
Audio Object	The Audio Objects in the Scene.
Speech Object	The Speech Objects in the Scene.
Visual Object	The Visual Objects in the Scene.

7.1.28.4 SubAIMs

No SubAIMs.

7.1.28.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/AudioVisualSceneDemultiplexing.json>

7.1.28.6 Profiles

No Profiles.

7.1.28.7 Conformance Testing

Table 2 provides the Conformance Testing Method for the OSD-SDX AIM as a Basic AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-SDX AIM

Receives	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema
----------	--	--

Produces	Speech Scene Geometry	Shall validate against Speech Scene Geometry schema
	Audio Scene Geometry	Shall validate against Audio Scene Geometry schema
	Visual Scene Geometry	Shall validate against Visual Scene Geometry schema
	Speech Objects	Shall validate against Speech Objects schema Speech Data shall conform with Qualifier
	Audio Objects	Shall validate against Audio Objects schema Audio Data shall conform with Qualifier
	Visual Objects	Shall validate against Visual Objects schema Visual Data shall conform with Qualifier

7.1.29 Audio-Visual Scene Multiplexing

7.1.29.1 Functions

The Audio-Visual Scene Multiplexing (OSD-SMX) produces Audio-Visual Scene Descriptors by combining the component elements – Audio Scene Geometry, Speech Scene Geometry, Visual Scene Geometry, Audio Objects, Speech Objects, and Visual Objects:

Receives	<i>Space-Time</i>
	<i>Speech Objects</i>
	<i>Audio Objects</i>
	<i>Visual Objects</i>
	<i>Speech Scene Geometry</i>
	<i>Audio Scene Geometry</i>
	<i>Visual Scene Geometry</i>
Produces	<i>Audio-Visual Scene Descriptors</i>

7.1.29.2 Reference Model

Figure 1 specifies the Reference Model of the Audio-Visual Scene Multiplexing (OSD-SMX) AIM.

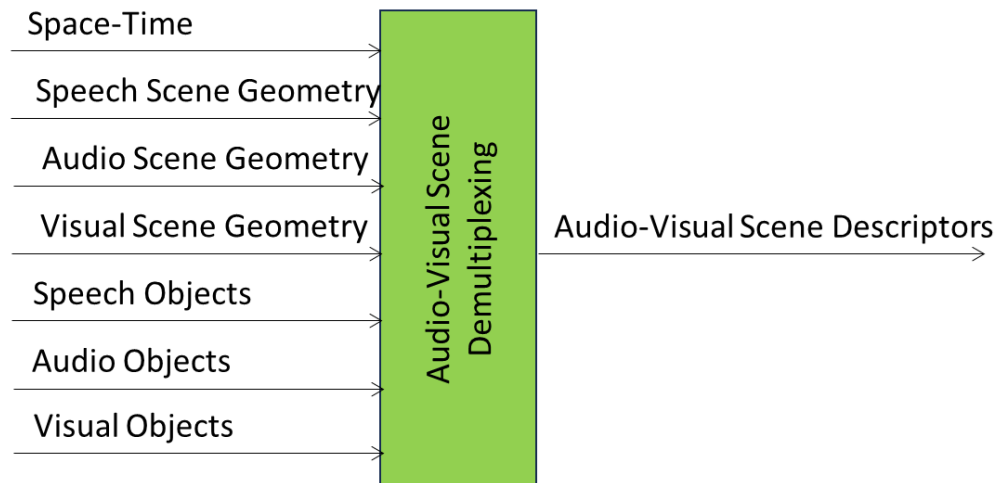


Figure 1 – The Audio-Visual Scene Multiplexing (OSD-SMX) AIM Reference Model

7.1.29.3 Input/Output Data

Table 13 specifies the Input and Output Data of the Audio-Visual Scene Multiplexing (OSD-SMX).

Table 13 – I/O Data of the Audio-Visual Scene Multiplexing (OSD-SMX) AIM

Input	Description
Space-Time	The Space-Time information.
Speech Object	The Speech Objects of the Scene.
Audio Object	The Audio Objects of the Scene.
Visual Object	The Visual Objects of the Scene.
Speech Scene Geometry	The Geometry of the Audio, Visual, Audio-Visual Objects of the Scene.
Audio Scene Geometry	The Geometry of the Audio, Visual, Audio-Visual Objects of the Scene.
Visual Scene Geometry	The Geometry of the Audio, Visual, Audio-Visual Objects of the Scene.
Output	Description
Audio-Visual Scene Descriptors	The combination of the Audio, Visual, and Audio-Visual Objects, and the Geometry of the Objects of the Scene.

7.1.29.4 SubAIMs

No SubAIMs.

7.1.29.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/AudioVisualSceneMultiplexing.json>

7.1.29.6 Profiles

No Profiles.

7.1.29.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-SMX AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-SMX AIM

Receives	SpaceTime	Shall validate against Space-Time schema.
	Speech Objects	Shall validate against Speech Objects schema. Speech Data shall conform with Qualifier.
	Audio Objects	Shall validate against Audio Objects schema. Audio Data shall conform with Qualifier.
	Visual Objects	Shall validate against Visual Objects schema. Visual Data shall conform with Qualifier.
	Speech Scene Geometry	Shall validate against Speech Scene Geometry schema.
	Audio Scene Geometry	Shall validate against Audio Scene Geometry schema.
	Visual Scene Geometry	Shall validate against Visual Scene Geometry schema.
Produces	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors schema.

7.1.30 Direct Visual Identification

7.1.30.1 Functions

The Direct Visual Identification (OSD-DVI) AIM receives a Visual Object and produces a Taxonomy-based identifier of an input Visual Object using the Point of View of a human body pointing to the Visual Object:

Receives	<i>Visual Object</i>	An individual Visual Object or a Visual Object containing individual Visual Objects.
Produces	<i>Visual Visual ID</i>	That identifies the Visual Object or a Visual Object contained in the input Visual Object.

7.1.30.2 Reference Model

Figure 1 depicts the Direct Visual Identification (OSD-DVI) AIM Reference Model.

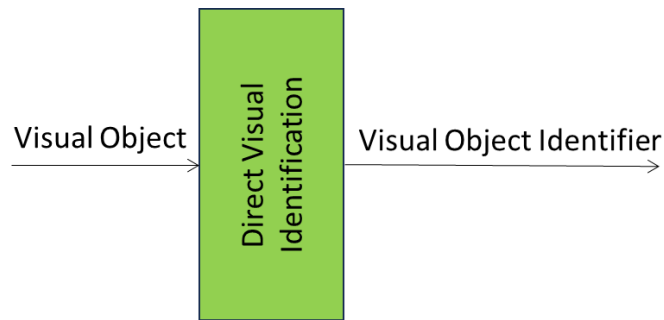


Figure - The Direct Visual Identification (OSD-DVI) AIM Reference Model

7.1.30.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Direct Visual Identification (OSD-DVI)AIM.

Table 1 – I/O Data of the Direct Visual Identification (OSD-DVI) AIM

Input	Description
Visual Object	The Visual Objects in the Visual Scene that are not Entities.
Output	Description
Visual Object Instance Identifier	The Identifier of a specific Visual Object.

7.1.30.4 SubAIMs

Direct Visual Identification (OSD-DVI) is a Composite AIM depicted in Figure 2.

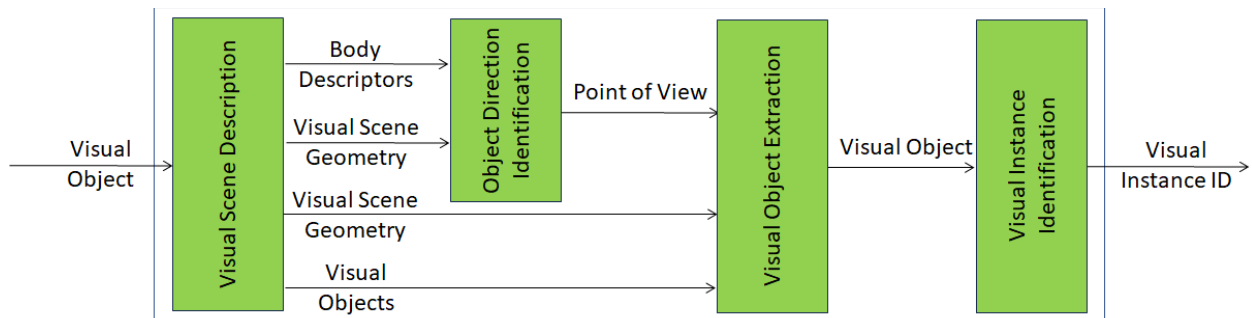


Figure 2 - The Direct Visual Identification (OSD-DVI) Composite AIM

Note that the Direct Visual Direction Identification AIM can parse either an AV Scene Geometry or its Visual Scene Geometry subset.

The AIMs composing the Direct Visual Identification (OSD-DVI) Composite AIM are:

AIM1	AIM2	AIM3	Names	JSON
OSD-DVI			Direct Visual Identification	Link
	OSD-VSD		Visual Scene Description	Link

	OSD-VOI		Visual Object Identification	Link
		OSD-VDI	Visual Direction Identification	Link
		OSD-VOE	Visual Object Extraction	Link
		OSD-VII	Visual Instance Identification	Link

7.1.30.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/DirectVisualIdentification.json>

7.1.30.6 Profiles

No Profiles.

7.1.30.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-DVI AIM. AIM. Conformance Testing of the individual AIMs of the OSD-DVI Composite AIM are given by the individual AIM Specification.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-DVI AIM

Receives	Visual Object	Shall validate against Visual Objects schema. Visual Data shall conform with Qualifier.
Produces	Visual Object Instance Identifier	Shall validate against Instance ID schema.

7.1.31 Television Splitting

7.1.31.1 Functions

The Television Splitting (OSD-TVS) AIM receives a video file with associated auxiliary text and produces the audio and video file and the text:

Receives	Audio-Visual Object	As a file.
	Auxiliary Text	A Text Object.
Demultiplexes	Audio-Visual Object	
Produces	Text Object	
	Audio Object	
	Video Object	

7.1.31.2 Reference Model

The Reference Model is specified in Figure 1.

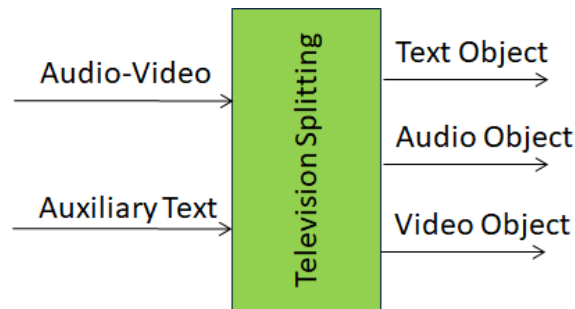


Figure 1 – The Television Splitting (OSD-TVS) AIM Reference Model

7.1.31.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Television Splitting (OSD-TVS) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Television Splitting (OSD-TVS) AIM

Input	Description
Audio-Video Object	Input audio-video file.
Auxiliary Text Object	Text provided to facilitate operation.
Output	Description
Text Object	The Text component of the input Audio-Video-Text file.
Audio Object	The Audio component of the input Audio-Video-Text file.
Video Object	The Video component of the input Audio-Video-Text file.

7.1.31.4 SubAIMs

No SubAIMs

7.1.31.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/TelevisionSplitting.json>

7.1.31.6 Profiles

No Profiles.

7.1.31.7 Reference Software

7.1.31.7.1 Disclaimers

1. This OSD-TVS Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to show a working Implementation of OSD-TVS, not to provide a ready-to-use product.
3. MPai disclaims the suitability of the Software for any other purposes and does not guarantee that it is secure.
4. Users shall verify that they have the right to use any third-party software required by this Reference Software.

7.1.31.7.2 Guide to the code

Use of this AI Modules is for developers who are familiar with Python, Docker, RabbitMQ, and downloading models from HuggingFace.

OSD-TVS extracts a WAV (uncompressed audio) from an input video using FFMPEG. Therefore, any input Audio-Visual file that can be demultiplexed by FFMPEG can be used as input. The current OSD-TVS implementation does not support Auxiliary Text.

The OSD-TVS Reference Software is found at the MPAI [gitlab](#) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image

Library: <https://www.ffmpeg.org/>

7.1.31.7.3 Acknowledgements

This version of the OSD-TVS Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

7.1.31.8 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-TVS AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-TVS AIM

Receives	Audio-Video Object	Shall validate against Audio-Visual Object schema. Audio-Visual Data shall conform with Qualifier.
	Auxiliary Text Object	Shall validate against Text Object schema. Text Data shall conform with Qualifier.
Produces	Text Object	Shall validate against Text Object schema. Text Data shall conform with Qualifier.
	Audio Object	Shall validate against Audio Object schema. Audio Data shall conform with Qualifier.
	Video Object	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.

7.1.32 Visual Change Detection

7.1.32.1 Functions

Visual Change Detection (OSD-VCD) receives a Visual Object, detects scene cuts/transitions, and produces as output, for each scene change, the first changed image and the time of change:

Receives	Visual Object	Input Video Visual Object.
Detects	Scene cuts/transitions	In the input video at a time a video frame is sufficiently different from preceding ones to signal a change.
Produces	Video frames	Per scene set by external parameters.

	Time	Including: 1. Time of changed video frame 2. Time of last video frame before a new change is detected.
--	------	--

7.1.32.2 Reference Model

The Reference Model of the Visual Change Detection (OSD-BCD) AIM is specified in Figure 1.

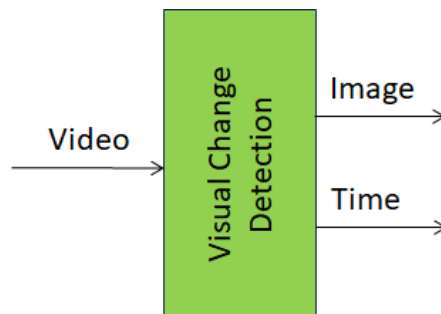


Figure 1 – The Visual Change Detection AIM Reference Model

7.1.32.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Scene Description AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Visual Scene Description AIM

Input	Description
Video	Input video file.
Output	Description
Image	The Image that has been detected as changed.
Time	The time of scene change detection (and the time before a new change).

7.1.32.4 SubAIMs

No SubAIMs

7.1.32.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualChangeDetection.json>

7.1.32.6 Profiles

No Profiles

7.1.32.7 Reference Software

7.1.32.7.1 Disclaimers

1. This OSD-VCD Reference Software Implementation is released with the BSD-3-Clause licence.
2. The purpose of this OSD-VCD Reference Software is to show a working Implementation of OSD-VCD, not to provide a ready-to-use product.
3. MPai disclaims the suitability of the Software for any other purposes and does not guarantee that it is secure.

4. Use of this Reference Software may require acceptance of licences from the respective repositories. Users shall verify that they have the right to use any third-party software required by this Reference Software.

7.1.32.7.2 *Guide to the OSD-VCD code*

OSD-VCD detects scene cuts/transitions in the input video and saves a given number of frames per scene, according to parameter "images_per_scene" in the input message. If parameter "scenedet_check" in the input message is set to "true", a helper video will be saved to the output folder. This video helps to visually check whether scene cuts/transitions are detected correctly.

The OSD-VCD Reference Software is found at the MPAI [gitlab](https://gitlab.com/Breakthrough/PySceneDetect) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image Library:
<https://github.com/Breakthrough/PySceneDetect>

7.1.32.7.3 *Acknowledgements*

This version of the OSD-VCD Reference Software has been developed by the MPAI *AI Framework* Development Committee (AIF-DC).

7.1.32.8 *Conformance Testing*

Table 2 provides the Conformance Testing Method for OSD-VCD AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-VCD AIM

Receives	Visual Object	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.
Produces	Video frames	Shall validate against Visual Object (images) schema. Visual Data shall conform with Qualifier.
	Time	Shall validate against Time schema.

7.1.32.9 *Performance Assessment*

Performance Assessment of an OSD-VCD AIM Implementation shall be performed using a dataset of video sequences for which scene changes have been identified. The video sequences may include none, one, or more than one scene change.

The Performance Assessment Report of an OSD-VCD AIM Implementation shall include:

1. The Identifier of the OSD-VCD AIM.
2. The Identifier of the video sequence dataset.
3. The classification of the video sequence according to an identified Taxonomy,
4. The Performance of the OSD-VCD AIM expressed by the following three numbers:
 1. The number of times a scene change was correctly identified divided by the total number of scene changes in the dataset (true positives).
 2. The number of times a scene change was incorrectly identified divided by the total number of scene changes (false positives).
 3. The number of times a scene change was incorrectly not identified when there was a scene change (false negatives).

7.1.33 Visual Direction Identification

7.1.33.1 Functions

The Visual Direction Identification (OSD-VDI) AIM identifies the Point of View signaled by the index finger or a human body in a space described by a Visual Scene Geometry:

Receives	Visual Scene Geometry	The Geometry of the Visual Scene
	Body Descriptors	The Descriptors of a Body.
Produces	Point of View	The direction of a line traversing a point of the forefinger of the Entity.

7.1.33.2 Reference Model

Figure 1 depicts the Reference Model of the Visual Direction Identification (OSD-VOI) AIM.

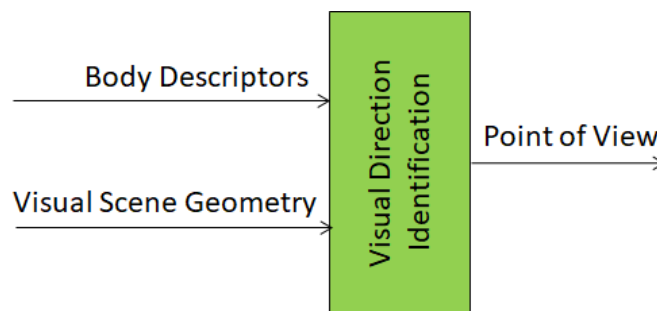


Figure 1 – The Visual Direction Identification (OSD-VOI) AIM Reference Model

7.1.33.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Direction Identification (OSD-VOI) AIM.

Table 1 – I/O Data of the Visual Direction Identification (OSD-VOI) AIM

Input	Description
Body Descriptors Object	The Descriptors of the Body Objects in the Visual Scene.
Visual Scene Geometry	The digital representation of the spatial arrangement of the Visual Objects of the Scene.
Output	Description
Point of View	The direction of the line traversing the forefinger of the target Entity.

7.1.33.4 SubAIMs

No SubAIMs.

7.1.33.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualDirectionIdentification.json>

7.1.33.6 Profiles

No Profiles.

7.1.33.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-VDI AIM.

If a schema contains references to other schemas, conformance of data for the primary schema implies that any data referencing a secondary schema shall also validate against the relevant schema, if present and conform with the Qualifier, if present.

Table 2 – Conformance Testing Method for OSD-VDI AIM

Receives	Visual Scene Geometry	Shall validate against Visual Scene Geometry schema
	Body Descriptors Object	Shall validate against Body Descriptors XML schema
Produces	Point of View	Shall validate against Point of View schema

7.1.34 Visual Instance Identification

7.1.34.1 Functions

The Visual Instance Identification (OSD-VII) AIM receives a Visual Object provides the identifier of the Visual Object based on a Taxonomy:

Receives	<i>Visual Object</i>	To be identified.
Produces	<i>An Instance ID</i>	Identifying an element of a set of Visual Objects belonging to a level in a taxonomy.

7.1.34.2 Reference Model

Figure 1 specifies the Reference Model of the Visual Instance Identification (OSD-VII) AIM.

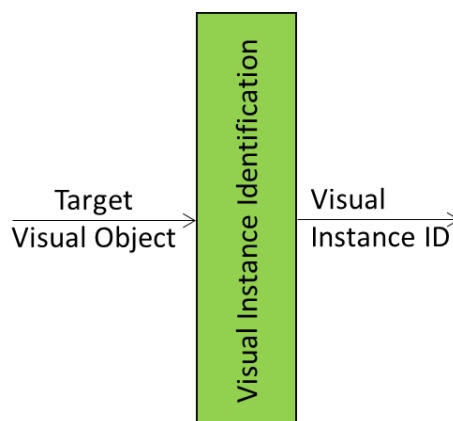


Figure 1 – The Visual Instance Identification (OSD-VII) AIM Reference Model

7.1.34.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Instance Identification (OSD-VII) AIM.

Table 1 – I/O Data of Visual Instance Identification (OSD-VII) AIM

Input	Description
Target Visual Object	The Visual Object crossed by the line traversing the forefinger of the Entity.
Output	Description
Visual Instance Identifier	The Identifier of the specific Visual Object belonging to a level in the taxonomy.

7.1.34.4 SubAIMs

No SubAIMs.

7.1.34.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualInstanceIdentification.json>

7.1.34.6 Profiles

No Profiles

7.1.34.7 Conformance Testing

Table 2 provides the Conformance Testing Method for OSD-VII AIM.

Note that a schema may contain references to other schemas. In this case, validation of data for the primary schema implies that any data that refers to a secondary schema shall also validate.

Table 2 – Conformance Testing Method for OSD-VII AIM

Receives	Visual Object	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.
Produces	Instance ID	Shall validate against Instance ID schema.

Table 3 provides an example of MMC-AQM AIM conformance testing.

Table 3 – An example MMC-AQM AIM conformance testing

Input Data	Data Type	Input Conformance Testing Data
Input Image	JPEG	All input Text files to be drawn from Image files .
Output Data	Data Type	Data Format
Object Instance ID	Identifier	All Identifiers of Visual Objects shall be represented according to Instance Identifier

7.1.34.8 Performance Assessment

Performance Assessment of an OSD-VII AIM Implementation shall be performed using a dataset of object of a category of objects of an identified Taxonomy.

The Performance Assessment Report of an OSD-VII AIM Implementation shall include:

1. The Identifier of the OSD-VII AIM.
2. The Identifier of the object dataset.

3. The data type of object: analogue, digital, 2D, 3D etc.
4. The Performance of the OSD-VII AIM expressed as the Accuracy of the Identifiers provided by the OSD-VII AIM computed on all objects of the dataset referenced in 2.

7.1.35 Visual Object Extraction

7.1.35.1 Functions

The Visual Object Extraction (OSD-VOE) AIMs receives a Visual Scene Geometry with its Visual Objects and a Point of View and provides the Visual Object that is crossed by the Point of View:

Receives	<i>Visual Scene Geometry</i>	Spatial description of object arrangement.
	<i>Visual Objects</i>	To be extracted for identification.
	<i>Point of View</i>	Crossed by line.
Extracts	<i>Visual Object</i>	Crossed by line from Point of View.

7.1.35.2 Reference Model

Figure 1 depicts the Reference Model of the Visual Object Extraction (OSD-VOE) AIM.

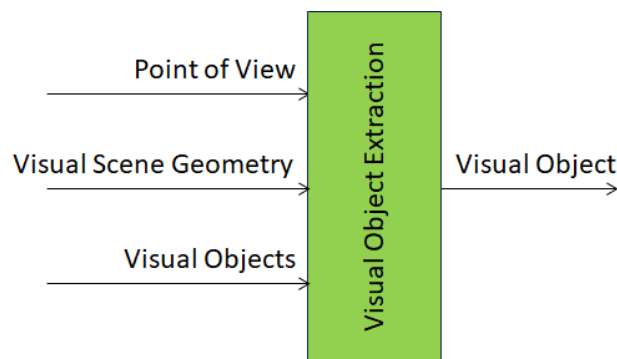


Figure 1 – The Visual Object Extraction (OSD-VOE) AIM Reference Model

7.1.35.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Object Extraction (OSD-VOE) AIM.

Table 1 – I/O Data of the Visual Object Extraction (OSD-VOE) AIM

Input	Description
Point of View	The direction of the line traversing the forefinger of the Entity.
Visual Scene Geometry	The digital representation of the spatial arrangement of the Visual Objects of the Scene.
Visual Objects	The Visual Objects of the Visual Scene Geometry.
Output	Description

Target Visual Object Identifier	The Visual Object crossed by the line traversing the forefinger of the Entity.
--	--

7.1.35.4 SubAIMs

No SubAIMs.

7.1.35.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualObjectExtraction.json>

7.1.35.6 Profiles

No Profiles.

7.1.35.7 Conformance Testing

Table 2 provides the Conformance Testing Method for MMC-VOE AIM.

Note that a schema may contain references to other schemas. In this case, validation of data for the primary schema implies that any data that refers to a secondary schema shall also validate.

Table 2 – Conformance Testing Method for MMC-VOE AIM

Receives	Visual Scene Geometry	Shall validate against Visual Scene Geometry schema.
	Visual Objects	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.
	Point of View	Shall validate against Point of View schema.
Extracts	Visual Object	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.

7.1.35.8 Visual Object Identification

7.1.35.9 Functions

The Visual Object Identification (OSD-VOI) AIM produces the Identifier based on a Taxonomy of a Visual Object included in a Visual Scene Geometry that is crossed by a Point of View:

Receives	<i>Visual Scene Geometry</i>	The arrangement of the objects in the Scene, a subset of Visual Scene Descriptors.
	<i>Visual Objects</i>	The Objects in the Scene.
	<i>Body Descriptors</i>	Descriptors of the Body indicating the object.
Produces	<i>Visual Instance ID</i>	Identifying a Visual Object in the Scene that belongs to some level in a taxonomy.

7.1.35.10 Reference Model

Figure 1 specifies the Reference Model of Visual Object Identification (OSD-VOI) AIM.

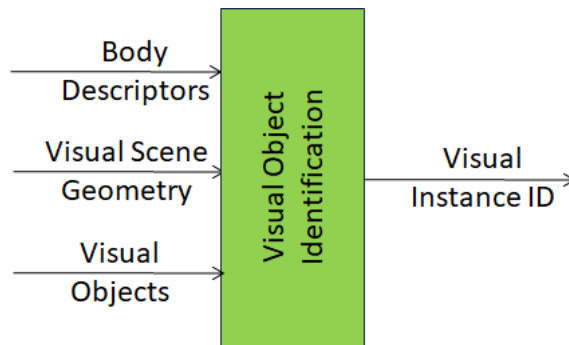


Figure - The Visual Object Identification (OSD-VOI) AIM Reference Model

7.1.35.11 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Object Identification (OSD-VOI) AIM.

Table 1 – I/O Data of the Visual Object Identification (OSD-VOI) AIM

Input	Description
Body Descriptors Object	The Descriptors of the Body Objects of Entities in the Visual Scene.
Visual Scene Geometry	The digital representation of the spatial arrangement of the Visual Objects of the Scene.
Visual Object	The Visual Objects in the Visual Scene that are not Entities.
Output	Description
Visual Instance Identifier	The Identifier of the specific Visual Object belonging to a level in the taxonomy.

7.1.35.12 SubAIMs

Visual Object Identification (OSD-VOI) is a Composite AIM specified by Figure 2.

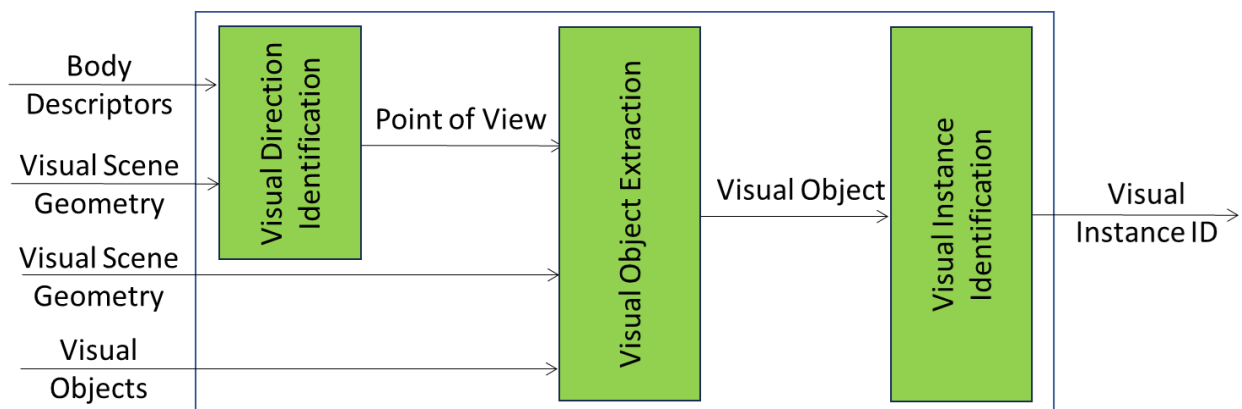


Figure 2 - The Visual Object Identification (OSD-VOI) Composite AIM

Note that the Visual Direction Identification AIM can parse either an AV Scene Geometry or its Visual Scene Geometry subset.

The AIMs composing the Visual Object Identification (OSD-VOI) Composite AIM are:

AIM	AIMs	Names	JSON
OSD-VOI		Visual Object Identification	Link
	OSD-VDI	Visual Direction Identification	Link
	OSD-VOE	Visual Object Extraction	Link
	OSD-VII	Visual Instance Identification	Link

7.1.35.13 *JSON Metadata*

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualObjectIdentification.json>

7.1.35.14 *Profiles*

No Profiles.

7.1.35.15 *Conformance Testing*

Table 2 provides the Conformance Testing Method for OSD-VOI AIM. Conformance Testing of the individual AIMs of the OSD-VOI Composite AIM are given by the individual AIM Specification.

Note that a schema may contain references to other schemas. In this case, validation of data for the primary schema implies that any data that refers to a secondary schema shall also validate.

Table 2 – Conformance Testing Method for OSD-VOI AIM

Receives	Visual Scene Geometry	Shall validate against Visual Scene Geometry schema.
	Visual Objects	Shall validate against Visual Objects schema. Visual Data shall conform with Qualifier.
	Body Descriptors Object	Shall validate against Body Descriptors XML schema.
Produces	Visual Instance ID	Shall validate against Instance ID schema.

7.1.36 Visual Object Identification by Body

7.1.36.1 *Functions*

The Visual Object Identification by Body (OSD-VOB) AIM receives a Visual Scene described by its Visual Scene Geometry and two Visual Objects - a Body Object and a generic Visual Object. The OSD-VOB AIM finds the Point of View indicated by the index finger of the Body Object and produces the Identifier - based on a Taxonomy - of the generic Visual Object that is crossed by the Point of View:

Receives	Visual Object	Known to be a Body Object.
----------	---------------	----------------------------

	Visual Object	A generic Visual Object to be identified.
	Visual Scene Geometry	Geometry of the Visual Scene that includes the two Visual Objects.
Processes	Body Object	To identify the Visual Body's index finger.
Finds	Point of View	Defined by the direction of the index finger.
Produces	Identifier	Of the Visual Object crossed by the line corresponding to the Point of View.

7.1.36.2 Reference Model

The Reference Model of the Visual Object Identification by Body (OSD-VOB) AIM is specified in Figure 1.

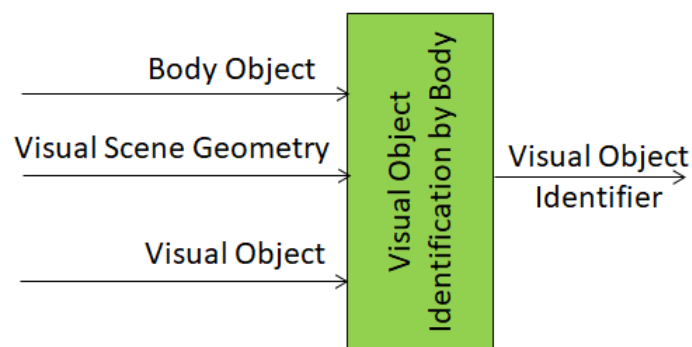


Figure 1 - The Visual Object Identification by Body (OSD-VOB) AIM Reference Model

7.1.36.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Object Identification by Body (OSD-VOB) AIM. Links are to the Data Type specifications.

Table 1 – I/O Data of the Visual Scene Description AIM

Input	Description
Visual Object	A Body Object
Visual Object	Generic Visual Object to be identified
Visual Scene Geometry	Common Scene Geometry of the two Visual Objects
Output	Description
Instance Identifier	Identifier of generic visual Objects

7.1.36.4 SubAIMs

Visual Object Identification by Body (OSD-VOB) is a Composite AIM specified by Figure 2.

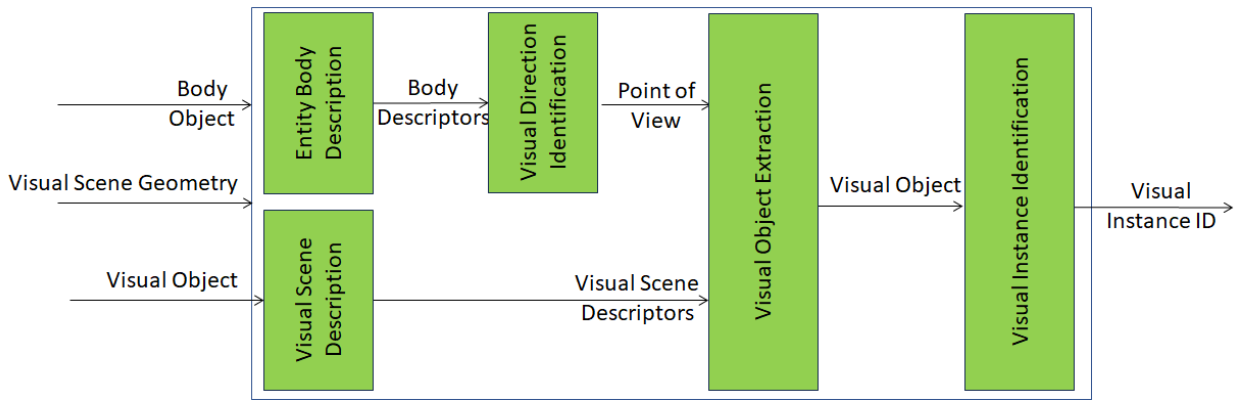


Figure 1 - The Visual Object Identification by Body (OSD-VOB) Composite AIM Reference Model

The AIMs composing the Visual Object Identification by Body (OSD-VOB) Composite AIM are:

Table2 – AIMs of the Visual Object Identification by Body (OSD-VOB) Composite AIM

AIM	AIMs	Names	JSON
OSD-VOB		Visual Object Identification by Body	Link
	PAF-EBD	Entity Body Description	Link
	OSD-VOD	Visual Scene Description	Link
	OSD-VDI	Visual Direction Identification	Link
	OSD-VOE	Visual Object Extraction	Link
	OSD-VII	Visual Instance Identification	Link

7.1.36.5 JSON Metadata

<https://schemas.mpai.community/OSD/V1.4/AIMs/VisualObjectIdentificationByBody.json>

7.1.36.6 Profiles

No Profiles.

7.1.36.7 Conformance Testing

Table 3 provides the Conformance Testing Method for OSD-VOB AIM. Conformance Testing of the individual AIMs of the OSD-VOB Composite AIM are given by the individual AIM Specification.

Note that a schema may contain references to other schemas. In this case, validation of data for the primary schema implies that any data that refers to a secondary schema shall also validate.

Table 3 – Conformance Testing Method for OSD-VOI AIM

Receives	Visual Object (Body)	Shall validate against Visual Object schema. Visual Data shall conform with Qualifier.
----------	--------------------------------------	--

	Visual Object	Shall validate against Visual Objects schema. Visual Data shall conform with Qualifier.
	Visual Scene Geometry	Shall validate against Visual Scene Descriptors schema.
Produces	Visual Instance ID	Shall validate against Instance ID schema.

7.1.36.8 Conformance testing

A Data instance of a Data Type Conforms with MPAI-OSD V1.4 if the JSON Data validate against the relevant MPAI-OSD V1.4 JSON Schema and if the Data Conforms with the relevant Data Qualifier, if present. MPAI-OSD V1.4 does not provide method for testing the Conformance of the Semantics of the Data instance to the MPAI-OSD V1.5 specification.

Conformance testing can be performed by a human using a JSON Validator to verify the Conformance of the syntax of JSON Data to the relevant JSON Schema; and, if the Data has a Qualifier, to verify that the syntax of the Data conforms with the relevant values in the Data Qualifier. Alternatively, Conformance testing can be performed by software implementing the steps above.

7.2 Reference Software

As a rule, MPAI provides Reference Software implementing the AI Modules released with the BSD-3-Clause licence and the following disclaimers:

1. The purpose of the Reference Software is to provide a working Implementation of an AIM, not a ready-to-use product.
2. MPAI disclaims the suitability of the Software for any other purposes than those of the MPAI-OSD Standard, and does not guarantee that it offers the best performance and that it is secure.
3. Users shall verify that they have the right to use any third-party software required by this Reference Software, e.g., by accepting the licences from third-party repositories.

MPAI-OSD V1.3 provides Reference Software Implementation for some AIMs.

7.3 Performance Assessment

Performance is a multidimensional entity because it can have various connotations. Therefore, the Performance Assessment Specification should provide methods to measure how well an AIW performs its function, using a metric that depends on the nature of the function, such as:

1. Quality: Performance Assessment measures how well an AIM performs its function, using a metric that depends on the nature of the function, e.g., how well a Visual Change Detection (VCD) AIM can detect the change of a visual scene.
2. Bias: Performance Assessment measures the preference given by an AIM to certain elements, using a metric that depends on a bias related to certain attributes of the AIM. For instance, a Visual Instance Identification (VII) AIM tends to have a higher correct identification of visual objects that have a certain shape.
3. Legal compliance: Performance Assessment measures how well an AIM performs its function, using a metric that assesses its accordance with a certain legal standard.
4. Ethical compliance: the Performance Assessment of an AIM can measure the compliance of an AIM to a target ethical standard.

MPAI-OSD V1.3 provides Performance Assessment methods for some AIMs.

8 Data Types

8.1 Technical Specifications

Table 1 provides the Data Types specified by *Technical Specification: Object Scene Description (MPAI-OSD) V1.4*. The Table is organised to give first the Data Types for which a "Basic" and a "Full" versions are specified (Generic, 3D Model, Audio, Audio-Visual, LiDAR, Location, Offline Map, RADAR. Speech, Text, Time, Ultrasound, Visual) followed by Data Types that do not have this characteristic.

Table 1 - Data Types specified by MPAI-OSD V1.4

Type	Acronym	Name	J	Acronym	Name	J
Generic	OSD-BOB	Basic Object	X	OSD-OBJ	Object	X
	OSD-BSD	Basic Scene Descriptors	X	OSD-SCD	Scene Descriptors	X
	OSD-BSG	Basic Scene Geometry	X	OSD-SCG	Scene Geometry	X
				OSD-EVT	Event Descriptors	X
3D Model	OSD-B3O	Basic 3D Model Object	X	OSD-3DO	3D Model Object	X
	OSD-B3D	Basic 3D Model Scene Descriptors	X	OSD-3SD	3D Model Scene Descriptors	X
	OSD-B3G	Basic 3D Model Scene Geometry	X	OSD-3SG	3D Model Scene Geometry	X
				OSD-3EV	3D Model Event Descriptors	X
Audio	OSD-BAO	Basic Audio Object	X	OSD-AUO	Audio Object	X
	OSD-BAS	Basic Audio Scene Descriptors	X	OSD-ASD	Audio Scene Descriptors	X
	OSD-BAG	Basic Audio Scene Geometry	X	OSD-ASG	Audio Scene Geometry	X
				OSD-AEV	Audio Event Descriptors	X
Audio-Visual	OSD-BMO	Basic Audio-Visual Object	X	OSD-AVO	Audio-Visual Object	X
	OSD-BMS	Basic Audio-Visual Scene Descriptors	X	OSD-MSD	Audio-Visual Scene Descriptors	X
	OSD-BMG	Basic Audio-Visual Scene Geometry	X	OSD-MSG	Audio-Visual Scene Geometry	X
				OSD-MEV	Audio-Visual Event Descriptors	X

LiDAR	OSD-BLO	Basic LiDAR Object	X	OSD-LIO	LiDAR Object	X
	OSD-BLS	Basic LiDAR Scene Descriptors	X	OSD-LSD	LiDAR Scene Descriptors	X
	OSD-BLG	Basic LiDAR Scene Geometry	X	OSD-LSG	LiDAR Scene Geometry	X
				OSD-LEV	LiDAR Event Descriptors	X
Location	OSD-BLC	Basic Location Object	X	OSD-LOC	Location Object	X
Offline Map	OSD-BOO	Basic Offline Map Object	X	OSD-OMO	Offline Map Object	X
	OSD-BOS	Basic Offline Map Scene Descriptors	X	OSD-OSD	Offline Map Scene Descriptors	X
	OSD-BOG	Basic Offline Map Scene Geometry	X	OSD-OSG	Offline Map Scene Geometry	X
				OSD-OEV	Offline Map Event Descriptors	X
RADAR	OSD-BRO	Basic RADAR Object	X	OSD-RAO	RADAR Object	X
	OSD-BRS	Basic RADAR Scene Descriptors	X	OSD-RSD	RADAR Scene Descriptors	X
	OSD-BRG	Basic RADAR Scene Geometry	X	OSD-RSG	RADAR Scene Geometry	X
				OSD-REV	RADAR Event Descriptors	X
Speech	OSD-BSO	Basic Speech Object	X	OSD-SPO	Speech Object	X
	OSD-BSS	Basic Speech Scene Descriptors	X	OSD-SSD	Speech Scene Descriptors	X
	OSD-BSG	Basic Speech Scene Geometry	X	OSD-SSG	Speech Scene Geometry	X
				OSD-SEV	Speech Event Descriptors	X
Text	OSD-BTO	Basic Text Object	X	OSD-TXO	Text Object	X
Time	OSD-BTM	Simple Time	X	OSD-TIM	Time	X
Ultrasound	OSD-BUO	Basic Ultrasound Object	X	OSD-USO	Ultrasound Object	X
	OSD-BUS	Basic Ultrasound Scene Descriptors	X	OSD-USD	Ultrasound Scene Descriptors	X
	OSD-BUG	Basic Ultrasound Scene Geometry	X	OSD-USG	Ultrasound Scene Geometry	X

				OSD- UED	Ultrasound Event Descriptors	X
Visual	OSD- BVO	Basic Visual Object	X	OSD- VIO	Visual Object	X
	OSD- BVD	Basic Visual Scene Descriptors	X	OSD- VSD	Visual Scene Descriptors	X
	OSD- BVG	Basic Visual Scene Geometry	X	OSD- VSG	Visual Scene Geometry	X
				OSD- VSG	Visual Event Descriptors	X
Other	OSD- ANN	Annotation	X	OSD- PCE	Perceptible Entity	X
	OSD- AUS	Audio Source	X	OSD- OPV	Point of View	X
	OSD- BBX	Bounding Box	X	OSD- OPS	Position	X
	OSD- CLO	Colour Object	X	OSD- RPP	Right Parallelepiped	X
	OSD- CRD	Coordinates	X	OSD- SEL	Selector	X
	OSD- DSG	Device Scene Geometry	X	OSD- SLC	Simple Location	X
	OSD- IID	Instance Identifier	X	OSD- SPT	Space-Time	X
	OSD- LTS	Light Source	X	OSD- OSA	Spatial Attitude	X
	OSD- OAC	Object Audio Characteristics	X	OSD- TEM	Texture Map	X
	OSD- OOR	Orientation	X	OSD- TRJ	Trajectory	X
	OSD- PAT	Path	X			

Note that:

1. All MPAI-OSD Data Types specified by previous Versions are superseded. Use of earlier versions of Data Types is permitted if their version is explicitly signaled.
2. MPAI-OSD AIWs and AIMs may utilise Data Types specified by other MPAI Technical Specifications.

8.1.1 Basic Object

8.1.1.1 Definition

A Data Type including:

1. Data of a specific media type perceptible by a specific device and/or a human.
2. Descriptive Data regarding Sub-Types, Formats and Attributes of the Data (optionally).

8.1.1.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

The following table gives the Media Type-specific Objects.

Acronym	Object name	JSON
OSD-B3O	Basic-3DModel Object	Link
OSD-BAO	Basic Audio Object	Link
OSD-BMO	Basic-Audio-Visual Object	Link
OSD-BLO	Basic LiDAR Object	Link
OSD-BOO	Basic Offline Map Object	Link
OSD-BRO	Basic RADAR Object	Link
OSD-BSO	Basic Speech Object	Link
OSD-BTO	Basic Text Object	Link
OSD-BUO	Basic Ultrasound Object	Link
OSD-BVO	Basic Visual Object	Link

8.1.1.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicObject.json>

8.1.1.4 Semantics

Label	Description
Header	Basic Object Header
– Standard-BasicObject	The characters “OSD-BOB-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
Basic ObjectID	Identifier of the Basic Object.

ParentObjects[]	Set of Parent Objects.
- ParentObject	A Parent Object.
- ParentObjectSpaceTime	Space-Time of a Parent Object.
ChildObjects[]	Set of Child Objects.
- ChildObject	Identifier of a Child Object.
- ChildObjectSpaceTime	Space-Time of a Child Object.
BasicObjectSpace-Time	Space-Time of Data.
BasicObjectData	the Data of the Basic Object.
BasicObjectQualifier	Media-specific Data Qualifier .
BasicObjectAnnotations[]	Set of Data Annotations.
– Annotation	An Annotation
– AnnotationSpaceTime	Where Annotation is attached - when it will be active.
– Rights	Actions that may be performed on the Annotation.
BasicObjectRights	Rights to perform Process Actions on the Object.
DescrMetadata	Descriptive Metadata

8.1.1.5 Conformance Testing

A Data instance Conforms with Basic Object (OSD-BOB) if:

1. The Data validates against the Basic Object's JSON Schema.
2. All Data in the Basic Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.2 Object

8.1.2.1 Definition

A Data Type including a collection of Basic Objects possibly of different Media Types.

An Object can have a hierarchical structure where Objects contain Basic Objects and Objects.

8.1.2.2 Functional Requirements

An Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Object.
3. Space-Time information of the Object.
4. Basic Object and Objects included in the Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Object.
6. The Rights that may be exercised on the Object.

Note that.

1. An Object that does not include Objects but only one Basic Object is a Basic Object.
 2. The Space-Time information of a Basic Object, Object, or Basic Scene included in an Object may be superseded by the Space-Time information of the Object containing it.
- The following table gives the Media Type-specific Objects.

Acronym	Object name	JSON
OSD-3DO	3DModel Object	Link
OSD-AUO	Audio Object	Link
OSD-AVO	Audio-Visual Object	Link
OSD-LIO	LiDAR Object	Link
OSD-OMO	Map Object	Link
OSD-RAO	RADAR Object	Link
OSD-SPO	Speech Object	Link
OSD-TXO	Text Object	Link
OSD-USO	Ultrasound Object	Link
OSD-VIO	Visual Object	Link

8.1.2.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Object.json>

8.1.2.4 Semantics

Label	Description
Header	Object Header
– Standard-Object	The characters “OSD-OBJ-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
ObjectID	Identifier of the Object.
ObjectSpaceTime	Space-Time of Object.
BasicObjectCount	Set of Parent Objects.
BasicObjects[]	Set of Basic Objects.
- BasicObjectSpaceTime	Space Time of a Basic Object in the Object.
- BasicObject	A Basic Object in the Object.
ObjectCount	Set of Child Objects.
Objects[]	Set of Objects.
- ObjectSpaceTime	Space Time of an Object in the Object.

- Object	An Object in the Object
ObjectAnnotations[]	Set of Object Annotation.
- Annotation	An Annotation.
- AnnotationSpaceTime	Where Annotation is attached and when it will be active.
- Rights	Actions that may be performed on the Annotation
ObjectRights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.2.5 Conformance Testing

A Data instance Conforms with Object (OSD-OBJ) if:

1. The Data validates against the Object's JSON Schema.
2. All Data in the Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.3 Basic Scene Descriptors

8.1.3.1 Definition

A Data Type representing the Objects of a scene and their arrangement in the scene.

8.1.3.2 Functional Requirements

Basic Scene Descriptors include:

1. ID of a Virtual Space where it is or will be located.
2. ID of the Scene Descriptors.
3. The number of Objects in the Scene.
4. Basic Scene Space-Time.
5. Objects that include, for each Object:
 1. The Object ID or the Object.
 2. Space-Time values potentially different from their intrinsic Space Times values.
6. Rights that may be exercised on the Basic Scene

The following Table gives the Media Type-specific Basic Scene Descriptors

Basic Scene Descriptors name	Acronym	JSON
Basic 3DModel Scene Descriptors	OSD-B3S	Link
Basic Audio Scene Descriptors	OSD-BAS	Link
Basic Audio-Visual Scene Descriptors	OSD-MBS	Link
Basic LiDAR Scene Descriptors	OSD-BLS	Link
Basic Offline Map Scene Descriptors	OSD-BOS	Link
Basic RADAR Scene Descriptors	OSD-BRS	Link
Basic Speech Scene Descriptors	OSD-BSS	Link
Basic Text Scene Descriptors	OSD-BTS	Link
Basic Ultrasound Scene Descriptors	OSD-BUS	Link

Basic Visual Scene Descriptors	OSD-BVS	Link
--------------------------------	---------	----------------------

8.1.3.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicSceneDescriptors.json>

8.1.3.4 Semantics

Label	Description
Header	Basic Scene Descriptors Header
- Standard-BasicSceneDescriptors	The characters “OSD-BSD-”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicSceneDescriptorsID	Identifier of the Basic Scene.
SpaceTime	Data about Space-Time of Basic Scene.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- ObjectID and/or Object	ObjectID or Object.
- ObjectSpaceTime	Space-Time of Object.
DescrMetadata	Descriptive Metadata.

8.1.3.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BSD) if:

1. The Data validates against the Basic Scene Descriptors’ JSON Schema.
2. All Data in the Basic Scene Descriptors' JSON Schema
 1. Have the specified type.
 2. Validate against their JSON Schemas.
 3. Conform with their Data Qualifiers.

8.1.4 Scene Descriptors

8.1.4.1 Definition

A Data Type including the Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.4.2 Functional Requirements

Scene Descriptors include

1. Objects
2. The Descriptors of the Scenes includes in the Scene called Sub-Scenes.
3. Rights that may be exercised on the Scene.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

Scene Descriptors name	Acronym	JSON
3DModel Scene Descriptors	OSD-3SD	Link
Audio Scene Descriptors	OSD-ASD	Link

Audio-Visual Scene Descriptors	OSD-MSD	Link
LiDAR Scene Descriptors	OSD-LSD	Link
Offline Map Scene Descriptors	OSD-OSD	Link
RADAR Scene Descriptors	OSD-RSD	Link
Speech Scene Descriptors	OSD-SSD	Link
Ultrasound Scene Descriptors	OSD-USD	Link
Visual Basic Descriptors	OSD-VSD	Link

8.1.4.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SceneDescriptors.json>

8.1.4.4 Semantics

Label	Description
Header	Scene Descriptors Header
- Standard-SceneDescriptors	The characters “OSD-SCD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.4.5 Conformance Testing

A Data instance Conforms with Scene Descriptors (OSD-SCD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas

3. Conform with their Data Qualifiers.

8.1.5 Basic Scene Geometry

8.1.5.1 Definition

A Data Type representing the spatial arrangement of the Objects in a scene.

8.1.5.2 Functional Requirements

Basic Scene Geometry includes:

1. The ID of a Virtual Space where it is/will be located.
2. The ID of the Basic Scene Geometry.
3. The Basic Scene Space-Time info.
4. The number of Objects in the Scene.
5. The Object ID and Space-Time of each Object:
6. The Rights that may be exercised on the Object.

The following Table gives the Media Type-dependent Scene Geometries.

Basic Scene Geometry name	Acronym	JSON
3DModel Basic Scene Geometry	OSD-B3G	Link
Audio Basic Scene Geometry	OSD-BAG	Link
Audio-Visual Basic Scene Geometry	OSD-BMG	Link
LiDAR Basic Scene Geometry	OSD-BLG	Link
Offline Map Basic Scene Geometry	OSD-BOG	Link
RADAR Basic Scene Geometry	OSD-BRG	Link
Speech Basic Scene Geometry	OSD-BSG	Link
Ultrasound Basic Scene Geometry	OSD-BUG	Link
Visual Basic Scene Geometry	OSD-BVG	Link

8.1.5.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicSceneGeometry.json>

8.1.5.4 Semantics

Label	Description
Header	Basic Scene Geometry Header
- Standard-BasicSceneGeometry	The characters “OSD-BSG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicSceneGeometryID	Identifier of the Basic Scene Geometry.
ObjectCount	Number of Objects in Basic Scene

BasicSceneSpaceTime	Space and Time info of Basic Scene Geometry
BasicSceneGeometry[]	Set of Basic Scene Geometry Data
- ObjectID	Object ID.
- ObjectSpaceTime	Object Space-Time - may supersede Object's own Space-Time
DescrMetadata	Descriptive Metadata

8.1.5.5 Conformance Testing

A Data instance Conforms with Basic Scene Geometry (OSD-BSG) if:

1. The Data validates against the Scene Geometry's JSON Schema.
2. All Data in the Basic Scene Geometry's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.6 Scene Geometry

8.1.6.1 Definition

A Data Type representing the arrangement of the Objects of a Scene that may include Objects and sub-scenes in a hierarchical fashion.

8.1.6.2 Functional Requirements

Scene Geometry may reference Objects and Scenes in a hierarchical fashion.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

The following Table gives the Media Type-dependent Scene Geometries.

Scene Geometry name	Acronym	JSON
3DModel Scene Geometry	OSD-3SG	Link
Audio Scene Geometry	OSD-ASG	Link
Audio-Visual Scene Geometry	OSD-MSG	Link
LiDAR Scene Geometry	OSD-LSG	Link
Offline Map Scene Geometry	OSD-OSG	Link
RADAR Scene Geometry	OSD-RSG	Link
Speech Scene Geometry	OSD-SSG	Link
Ultrasound Scene Geometry	OSD-USG	Link
Visual Basic Geometry	OSD-VSG	Link

8.1.6.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SceneGeometry.json>

8.1.6.4 Semantics

Label	Description
Header	Scene Geometry Header
- Standard-SceneGeometry	The characters "OSD-SCG-V"
- Version	Major version – 1 or 2 characters

- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- ObjectID	Qualifiers of Object.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Scenes.
- SubSceneID	ID of Sub-Scene.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.6.5 Conformance Testing

A Data instance Conforms with Scene Geometry (OSD-SCG) if:

1. The Data validates against the Visual Scene Geometry’s JSON Schema.
2. All Data in the Visual Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.7 Event Descriptors

8.1.7.1 Definition

An Item represented by a series of Scene Descriptors for a certain duration (from a start Time to an end Time).

8.1.7.2 Functional Requirements

Event Descriptors contains Scene Descriptors between two Times.

8.1.7.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/EventDescriptors.json>

8.1.7.4 Semantics

Label	Description
Header	Event Descriptors Header
- Standard - EventDescriptors	The characters “OSD-EVD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters

MInstanceID	Identifier of M-Instance.
EventDescriptorsID	Identifier of Event Descriptors.
EventDescriptorsSpaceTime	Data about start and end Space-Time.
EventDescriptors[]	Collection of Scene Descriptors between Start Time and End Time
- SceneDescriptors	Set of Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.7.5 Conformance Testing

A Data instance Conforms with Event Descriptors (OSD-EVD) if:

1. The Data validates against the Event Descriptors' JSON Schema.
2. All Data in the Event Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.8 Basic 3D Model Object

8.1.8.1 Definition

A Data Type including

1. Data representing:
 1. The surface and relevant features of physical objects from the real world, or
 2. Synthetic Data intended for visual rendering, or
 3. A mixture of the two types, and
2. Optionally, descriptive Data regarding Sub-Types, Formats and Attributes of the 3D Model Data.

8.1.8.2 Functional Requirements

A Basic 3D Model Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic 3D Model Object.
3. The ID(s) of 3D Model Objects at a preceding time (Parent Objects) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent 3D Model Objects in an M-Instance.
5. The ID(s) of 3D Model Objects at a following time (Child Objects).
6. The Space-Time information of Child 3D Model Objects in an M-Instance.
7. The Space-Time information of the Basic 3D Model Object in an M-Instance.
8. The Data of the 3D Model Object.
9. The Qualifier of the specific 3D Model Data Type.
10. Additional Basic 3D Model Object information:
 1. - Mass of the Object/
 2. - Material Property - is material or immaterial.
 3. - Gravity Property - is subject to gravity or not.
 4. - Texture Map - the Objects texture map.
 5. - Object Audio Characteristics - the audio characteristics of the Object.
 6. The Euler angles of the Object.

11. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.
12. The Rights that can be exercised on the Basic 3D Model Object.

8.1.8.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Basic3DModelObject.json>

8.1.8.4 Semantics

Label	Description
Header	Basic 3D Model Object Header
– Standard-Basic3DModelObject	The characters “OSD-B3O-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
Basic3DModelObjectID	Identifier of the Basic 3D Model Object.
3DModelData	The Data of the 3D Model Object.
3DModelDataQualifier	Qualifier of Basic 3D Model Data.
BasicVisualObjectAttributes	Additional information on Object.
- Mass	Mass of the Object in kg.
- Material	Object has material consistence (0=No, 1=Yes).
- Gravity	Object is subject to gravity (0=No, 1=Yes).
- TextureMapID	ID of Texture Map.
- ObjectAudioCharacteristics	Basic 3D Model Object's Audio Characteristics
- EulerAngles	The intrinsic three axes of the Basic 3D Model Object with respect to the coordinate system. The Orientation of the Object is with respect to the intrinsic axes of the Object.
Basic3DModelObjectSpaceTime	Space-Time info of the Basic 3D Model Object.
Basic3DModelObjectAnnotations[]	Annotations of Basic 3D Model Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions on the Basic 3D Model Object.
DescrMetadata	Descriptive Metadata

8.1.8.5 Conformance Testing

A Data instance Conforms with Basic 3D Model Object if:

1. The Data validates against the Basic 3D Model Object's JSON Schema.
2. All Data in the Basic 3D Model Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.9 3D Model Object

8.1.9.1 Definition

A Data Type including a collection of Basic 3D Model Objects.

A 3D Model Object can have a hierarchical structure where 3D Model Objects contain Basic 3D Model Objects and 3D Model Objects.

8.1.9.2 Functional Requirements

A 3D Model Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the 3D Model Object.
3. Space-Time information of the 3D Model Object.
4. Basic 3D Model Objects and other 3D Model Objects included in the 3D Model Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the 3D Model Object Annotation.
6. The Rights that may be exercised on the 3D Model Object.

Note that.

1. A 3D Model Object that does not include 3D Model Objects and only one Basic 3D Model Object is a Basic 3D Model Object.
2. The Space-Time information of a Basic 3D Model Object and a 3D Model Object included in an 3D Model Object may be superseded by the Space-Time information of the 3D Model Object containing them.

8.1.9.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/3DModelObject.json>

8.1.9.4 Semantics

Label	Description
Header	3D Model Object Header
– Standard-3DModelObject	The characters “OSD-3DO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
3DModelObjectID	Identifier of the 3D Model Object.
3DModelObjectSpaceTime	Space-Time of 3D Model Object.
Basic3DModelObjectCount	Set of Parent 3D Model Objects.

Basic3DModelObjects[]	Set of Basic 3D Model Objects.
- SpaceTime	Space Time of a Basic 3D Model Object in the 3D Model Object.
- Basic3DModelObject	A Basic 3D Model Object in the 3D Model Object.
3DModelObjectCount	Number of 3D Model Objects.
3DModelObjects[]	Set of 3D Model Objects.
- SpaceTime	Space Time of an 3D Model Object in the 3D Model Object.
- 3DModelObject	A 3D Model Object in the 3D Model Object
Annotations[]	Set of 3D Model Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Process Actions that may be performed on the Annotation
Rights	Process Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.9.5 Conformance Testing

A Data instance Conforms with 3D Model Object (OSD-3DO) if:

1. The Data validates against the 3D Model Object's JSON Schema.
2. All Data in the 3D Model Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.10 Basic 3D Model Scene Descriptors

8.1.10.1 Definition

A Data Type including the 3D Model Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as 3D Model Object and 3D Model Scene, respectively.

8.1.10.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Value of Gravity
4. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.10.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Basic3DModelSceneDescriptors.json>

8.1.10.4 Semantics

Label	Description
Header	Basic 3D Model Scene Descriptors Header

- Standard-Basic3DModelSceneDescriptors	The characters “OSD-B3S-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
GravityValue	The value of Gravity in the Basic Scene measured in m/s ² .
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.10.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-B3S) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.11 3D Model Scene Descriptors

8.1.11.1 Definition

A Data Type including the 3D Model Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.11.2 Functional Requirements

3D Model Scene Descriptors include

1. 3D Model Objects
2. The Descriptors of the Scenes includes in the Scene called Sub-Scenes.
3. Rights that may be exercised on the Scene.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

8.1.11.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/3DModelSceneDescriptors.json>

8.1.11.4 Semantics

Label	Description
-------	-------------

Header	3D Model Scene Descriptors Header
- Standard-3DModelSceneDescriptors	The characters “OSD-3SD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.11.5 Conformance Testing

A Data instance Conforms with 3D Model Scene Descriptors (OSD-3SD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.12 Basic 3D Model Scene Geometry

8.1.12.1 Definition

A Data Type including the arrangement of the 3D Model Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as 3D Model Data, 3D Model Objects, 3D Model Qualifiers, and 3D Model Scenes

8.1.12.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.12.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Basic3DModelSceneGeometry.json>

8.1.12.4 Semantics

Label	Description
Header	Basic 3D Model Scene Geometry Header
- Standard-Basic3DModelSceneGeometry	The characters “OSD-B3G-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.12.5 Conformance Testing

A Data instance Conforms with Basic 3D Model Scene Geometry (OSD-B3G) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.13 3D Model Scene Geometry

8.1.13.1 Definition

An Data Type including the arrangement of the 3D Model Objects in a scene with their 3D Model Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as 3D Model Data, 3D Model Objects, 3D Model Qualifiers, and 3D Model (Sub-)Scenes

8.1.13.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.13.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/3DModelSceneGeometry.json>

8.1.13.4 Semantics

Label	Description
Header	3D Model Scene Geometry Header
- Standard-3DModelSceneGeometry	The characters “OSD-3SG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectID	ID of Object.
- SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
- SceneSubSceneID	ID of Sub-Scene.
- SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.13.5 Conformance Testing

A Data instance Conforms with 3D Model Scene Geometry (OSD-3SG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.14 3D Model Event Descriptors

8.1.14.1 Definition

An Item including a series of 3D Model Scene Descriptors for a certain duration.

8.1.14.2 Functional Requirements

3D Model Event Descriptors contains 3D Model Scene Descriptors for a Time.

8.1.14.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/3DModelEventDescriptors.json>

8.1.14.4 Semantics

Label	Description
Header	3D Model Event Descriptors Header
· Standard-3DModelEventDescriptors	The characters “OSD-3DE-V”
· Version	Major version – 1 or 2 characters
· Dot-separator	The character “.”
· Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.14.5 Conformance Testing

A Data instance Conforms with MPAI-OSD 3D Model Event Descriptors (OSD-3DE) if:

1. The Data validates against the 3D Model Event Descriptors’ JSON Schema.
2. All Data in the 3D Model Event Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.14.6

8.1.15 Basic Audio Object

8.1.15.1 Definition

A Data Type including

1. Data representing:
 1. A sound of the real world, or
 2. Synthetic Data intended for rendering as sound, or
 3. A mixture of the two types, and
2. Optionally, descriptive Data regarding Audio Sub-Types, Formats and Attributes.

In the following, Data, Qualifier, and Object should be read as Audio Data, Audio Qualifiers, and Audio Object, respectively.

8.1.15.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.

2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

8.1.15.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioObject.json>

8.1.15.4 Semantics

Label	Description
Header	Basic Audio Object Header
– Standard-BasicAudioObject	The characters “OSD-BAO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.15.5 Conformance Testing

A Data instance Conforms with Basic Audio Object (OSD-BAO) if:

1. The Data validates against the Basic Object’s JSON Schema.
2. All Data in the Basic Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.16 Audio Object

8.1.16.1 Definition

A Data Type including Basic Audio Objects and Audio Objects.

Audio Objects have a hierarchical structure where Audio Objects contain Basic Audio Objects and Audio Objects.

8.1.16.2 Functional Requirements

An Audio Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Audio Object.
3. Space-Time information of the Audio Object.
4. Basic Audio Object and Audio Objects included in the Audio Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Audio Object.
6. The Rights that may be exercised on the Audio Object.

Note that.

1. An Audio Object that does not include Sub-Scenes and only one Basic Audio Object is a Basic Audio Object.
2. The Space-Time information of a Basic Audio Object, Audio Object included in an Audio Object may be superseded by the Space-Time information of the Audio Object containing it.

8.1.16.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioObject.json>

8.1.16.4 Semantics

Label	Description
Header	Audio Object Header
– Standard-AudioObject	The characters “OSD-AUO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
AudioObjectID	Identifier of the Audio Object.
AudioObjectSpaceTime	Space-Time of Audio Object.
BasicAudioObjectCount	Set of Parent Audio Objects.
BasicAudioObjects[]	Set of Basic Audio Objects.
- SpaceTime	Space Time of a Basic Audio Object in the Audio Object.
- BasicAudioObject	A Basic Audio Object in the Audio Object.

AudioObjectCount	Number of Audio Objects.
AudioObjects[]	Set of Audio Objects.
- SpaceTime	Space Time of an Audio Object in the Audio Object.
- AudioObject	An Audio Object in the Audio Object
Annotations[]	Set of Audio Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.16.5 Conformance Testing

A Data instance Conforms with Audio Object (OSD-AUO) if:

1. The Data validates against the Audio Object's JSON Schema.
2. All Data in the Audio Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.17 Basic Audio Scene Descriptors

8.1.17.1 Definition

A Data Type including the Audio Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as Audio Object and Audio Scene, respectively.

8.1.17.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.17.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioSceneDescriptors.json>

8.1.17.4 Semantics

Label	Description
Header	Basic Audio Scene Descriptors Header
- Standard-BasicAudioSceneDescriptors	The characters "OSD-BAS-V"
- Version	Major version – 1 or 2 characters

- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.17.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BAS) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.17.6

8.1.18 Audio Scene Descriptors

8.1.18.1 Definition

Audio Scene Descriptors are defined as a Data Type including the Audio Objects of a scene, their sub-scenes, and their arrangement in the scene. Audio Scene Descriptors may be hierarchical, i.e., they may contain Objects and Audio Scene Descriptors.

8.1.18.2 Functional Requirements

Audio Scene Descriptors include

1. Audio Objects
2. The Descriptors of the Scenes includes in the Scene called Sub-Scenes.
3. Rights that may be exercised on the Scene.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

8.1.18.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioSceneDescriptors.json>

8.1.18.4 Semantics

Label	Description
Header	Audio Scene Descriptors Header

- Standard-AudioSceneDescriptors	The characters “OSD-ASD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.18.5 Conformance Testing

A Data instance Conforms with Audio Scene Descriptors (OSD-ASD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.19 Basic Audio Scene Geometry

8.1.19.1 Definition

A Data Type including the arrangement of the Audio Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Audio Data, Audio Objects, Audio Qualifiers, and Audio Scenes

8.1.19.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.19.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioSceneGeometry.json>

8.1.19.4 Semantics

Label	Description
-------	-------------

Header	Basic Audio Scene Geometry Header
- Standard-BasicAudioSceneGeometry	The characters “OSD-BAG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.19.5 Conformance Testing

A Data instance Conforms with Basic Audio Scene Geometry (OSD-BAG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.20 Audio Scene Geometry

8.1.20.1 Definition

A Data Type including the space-time arrangement of the Audio Objects in a scene.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as Audio Data, Audio Objects, Audio Qualifiers, and Audio (Sub-)Scenes

8.1.20.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.20.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioSceneGeometry.json>

8.1.20.4 Semantics

Label	Description
Header	Audio Scene Geometry Header
- Standard-AudioSceneGeometry	The characters “OSD-ASG-V”

- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectID	ID of Object.
- SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
- SceneSubSceneID	ID of Sub-Scene.
- SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.20.5 Conformance Testing

A Data instance Conforms with Audio Scene Geometry (OSD-ASG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.21 Audio Event Descriptors

8.1.21.1 Definition

An Item including a series of Audio Scene Descriptors for a certain duration.

8.1.21.2 Functional Requirements

Audio Event Descriptors contains Audio Scene Descriptors for a Time.

8.1.21.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioEventDescriptors.json>

8.1.21.4 Semantics

Label	Description
Header	Audio Event Descriptors Header
· Standard	The characters “OSD-AUE-V”

· Version	Major version – 1 or 2 characters
· Dot-separator	The character “.”
· Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.21.5 Conformance Testing

A Data instance Conforms with MPAA-OSD Audio Event Descriptors (OSD-AUE) if:

1. The Data validates against the Audio Event Descriptors’ JSON Schema.
2. All Data in the Audio Event Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.22 Basic Audio-Visual Object

8.1.22.1 Definition

A Data Type including

1. Data representing:
 1. A sound and visual of the real world, or
 2. Synthetic Data intended for rendering as sound and visual, or
 3. A mixture of the two types, and
2. Optionally, descriptive Data regarding Audio-Visual Sub-Types, Formats and Attributes.

In the following, Data, Qualifier, and Object should be read as Audio-Visual Data, Audio-Visual Qualifiers, and Audio-Visual Object, respectively.

8.1.22.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.

10. The set of Annotations including, for each Annotation:

1. Space-Time information of the Annotation.
2. Rights to perform Actions on the Annotation.

8.1.22.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioVisualObject.json>

8.1.22.4 Semantics

Label	Description
Header	Basic Audio-Visual Object Header
– Standard-BasicAudioVisualObject	The characters “OSD-BMO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.22.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object’s JSON Schema.
2. All Data in the Basic Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.23 Audio-Visual Object

8.1.23.1 Definition

Data whose rendering has both Audio and Visual perceptibility attributes.

8.1.23.2 Functional Requirements

Audio-Visual Object includes:

1. The ID of a Virtual Space (M-Instance) where it is or will be located.

2. The 3DModel-Speech-Audio-Visual Objects' Space-Time location.
3. The IDs of the 3DModel, Speech, Audio, and Visual Objects' and their Space-Time information.

8.1.23.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioVisualObject.json>

8.1.23.4 Semantics

Label	Description
Header	Audio-Visual Object Header
- Standard-AudioVisualObject	The characters “OSD-AVO-V”
- Version	Major version – 1 or 2 Bytes
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 Bytes
MInstanceID	Identifier of M-Instance.
AudioVisualObjectID	Identifier of Audio-Visual Object.
AudioVisualObjectSpaceTime	Space-Time of Audio-Visual Object
AudioVisualQualifier	Qualifier of the Audio-Visual Object
3DModelObjectData	3D Model Object Data
- 3DModelObjectID and/or 3DModelObject	3D Model Object ID and/or Object
- 3DModelObjectSpaceTime	Space-Time of Speech Object
SpeechObjectData	Speech Object Data
- SpeechObjectID and/or Speech Object	Speech Object ID and/or Object
- SpeechObjectSpaceTime	Space-Time of Speech Object
AudioObjectData	Audio Object Data
- AudioObjectID and/or Audio Object	Audio Object ID and/or Object
- AudioObjectSpaceTime	Space-Time of Audio Object
VisualObjectData	Visual Object Data
- VisualObjectID and/or Visual Object	Visual Object ID and/or Object
- VisualObjectSpaceTime	Space-Time of Visual Object
Annotations[]	Set of Audio Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation

Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.23.5 Conformance Testing

A Data instance Conforms with Audio-Visual Object (OSD-AVO) if:

1. The Data validates against the Audio-Visual Object's JSON Schema.
2. All Data in the Audio-Visual Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.24 Basic Audio-Visual Scene Descriptors

8.1.24.1 Definition

A Data Type including the Audio-Visual Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as Audio-Visual Object and Audio-Visual Scene, respectively.

8.1.24.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.24.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioVisualSceneDescriptors.json>

8.1.24.4 Semantics

Label	Description
Header	Basic Audio-Visual Scene Descriptors Header
- Standard-BasicAudioVisualSceneDescriptors	The characters "OSD-BMS-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.

- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.24.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BMS) if:

1. The Data validates against the Scene Descriptors' JSON Schema.
2. All Data in the Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.25 Audio-Visual Scene Descriptors

8.1.25.1 Definition

A Data Type including the Audio-Visual Scene's Objects and Sub-Scenes and their arrangement in the Scene.

8.1.25.2 Functional Requirements

Audio-Visual Scene Descriptors includes Scenes in addition to Objects.

8.1.25.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioVisualSceneDescriptors.json>

8.1.25.4 Semantics

Label	Description
Header	Audio-Visual Scene Descriptors Header
- Standard-AVSceneDescriptors	The characters "OSD-AVS-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
AVBasicSceneDescriptorsID	Identifier of the AV Object.
ObjectCount	Number of Objects in Scene
AVSceneSpaceTime	Data about Space and Time
SpeechObjects[]	Set of Speech Objects
- SpeechObject	Speech Object
- SpeechObjectSpaceTime	Space-Time of Speech Object
AudioObjects[]	Set of Audio Objects
- AudioObject	ID of Audio Object
- AudioObjectSpaceTime	Space-Time of Audio Object
VisualObjects[]	Set of Visual Objects

- VisualObjectID	ID of Visual Object
- VisualObjectSpaceTime	Space-Time of Visual Object
AudioVisualObjects[]	Set of Audio-Visual Objects
- AudioVisualObjectID	ID of Audio-Visual Object
- AudioObjectSpaceTime	Space-Time of Audio-Visual Object
SubSceneCount	Number of Sub-Scenes in Scene
SpeechSubScenes[]	Set of Speech Objects
- SpeechSubScene	Speech SubScene
- SpeechSubSceneSpaceTime	Space-Time of Speech SubScene
AudioSubScenes[]	Set of Audio SubScenes
- AudioSubScene	ID of Audio SubScene
- AudioSubSceneSpaceTime	Space-Time of Audio SubScene
VisualSubScenes[]	Set of Visual SubScenes
- VisualSubSceneID	ID of Visual SubScene
- VisualSubSceneSpaceTime	Space-Time of Visual SubScene
AudioVisualSubScenes[]	Set of Audio-Visual SubScenes
- AudioVisualSubSceneID	ID of Audio-Visual SubScene
- AudioSubSceneSpaceTime	Space-Time of Audio-Visual SubScene
DescrMetadata	Descriptive Metadata

8.1.25.5 Conformance Testing

A Data instance Conforms with Audio-Visual Scene Descriptors (OSD-AVS) V1.3 if:

1. The Data validates against the Audio-Visual Scene Descriptors' JSON Schema.
2. All Data in the Audio-Visual Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.26 Basic Audio-Visual Scene Geometry

8.1.26.1 Definition

A Data Type including the arrangement of the 3D Model Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Audio-Visual Data, Audio-Visual Objects, Audio-Visual Qualifiers, and Audio-Visual Scenes.

8.1.26.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.26.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioVisualSceneGeometry.json>

8.1.26.4 Semantics

Label	Description
-------	-------------

Header	Basic Audio-Visual Scene Geometry Header
- Standard-BasicAudioVisualSceneGeometry	The characters “OSD-BMG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.26.5 Conformance Testing

A Data instance Conforms with Basic Audio-Visual Scene Geometry (OSD-BMG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.27 Audio-Visual Scene Geometry

8.1.27.1 Definition

A Data Type including the arrangement of the 3D Model Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Audio-Visual Data, Audio-Visual Objects, Audio-Visual Qualifiers, and Audio-Visual Scenes.

8.1.27.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.27.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicAudioVisualSceneGeometry.json>

8.1.27.4 Semantics

Label	Description
-------	-------------

Header	Basic Audio-Visual Scene Geometry Header
- Standard-BasicAudioVisualSceneGeometry	The characters “OSD-BMG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.27.5 Conformance Testing

A Data instance Conforms with Basic Audio-Visual Scene Geometry (OSD-BMG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.27.6

8.1.28 Audio-Visual Event Descriptors

8.1.28.1 Definition

An Item including a series of Audio-Visual Scene Descriptors for a certain duration.

8.1.28.2 Functional Requirements

Audio-Visual Event Descriptors contains Audio-Visual Scene Descriptors for a Time.

8.1.28.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioVisualEventDescriptors.json>

8.1.28.4 Semantics

Label	Description
Header	Audio-Visual Event Descriptors Header

- Standard-AudioVisualEventDescriptors	The characters “OSD-AVE-V”
· Version	Major version – 1 or 2 characters
· Dot-separator	The character “.”
· Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.28.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Audio-Visual Event Descriptors (OSD-AVE) if:

1. The Data validates against the Annotation’s JSON Schema.
2. All Data in the Annotation’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.29 Basic LiDAR Object

8.1.29.1 Definition

Basic LiDAR Object refers to

1. LiDAR Data representing captured by a single LiDAR Device.
2. LiDAR Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of LiDAR Data.

In the following, Data, Qualifier, and Object should be read as LiDAR Data, LiDAR Qualifiers, and LiDAR Object, respectively.

8.1.29.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:

1. Space-Time information of the Annotation.
2. Rights to perform Actions on the Annotation.

8.1.29.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicLiDARObject.json>

8.1.29.4 Semantics

Label	Description
Header	Basic LiDAR Object Header
– Standard-BasicLiDARObject	The characters “OSD-BLO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.29.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object’s JSON Schema.
2. All Data in the Basic Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.30 LiDAR Object

8.1.30.1 Definition

A Data Type including a collection of Basic LiDAR Objects.

A LiDAR Object can have a hierarchical structure where LiDAR Objects contain Basic LiDAR Objects and LiDAR Objects.

8.1.30.2 Functional Requirements

A LiDAR Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the LiDAR Object.
3. Space-Time information of the LiDAR Object.
4. Basic LiDAR Object and LiDAR Objects included in the LiDAR Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the LiDAR Object.
6. The Rights that may be exercised on the LiDAR Object.

Note that.

1. A LiDAR Object that does not include Sub-Scenes and only one Basic LiDAR Object is a Basic LiDAR Object.
2. The Space-Time information of a Basic LiDAR Object and LiDAR Object included in a LiDAR Object may be superseded by the Space-Time information of the LiDAR Object containing them.

8.1.30.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LiDARObject.json>

8.1.30.4 Semantics

Label	Description
Header	LiDAR Object Header
– Standard-LiDARObject	The characters “OSD-LIO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
LiDARObjectID	Identifier of the LiDAR Object.
LiDARObjectSpaceTime	Space-Time of LiDAR Object.
BasicLiDARObjectCount	Set of Parent LiDAR Objects.
BasicLiDARObjects[]	Set of Basic LiDAR Objects.
- SpaceTime	Space Time of a Basic LiDAR Object in the LiDAR Object.
- BasicLiDARObject	A Basic LiDAR Object in the LiDAR Object.
LiDARObjectCount	Number of LiDAR Objects.
LiDARObjects[]	Set of LiDAR Objects.
- SpaceTime	Space Time of a LiDAR Object in the LiDAR Object.
- 3DModelObject	A LiDAR Object in the LiDAR Object
Annotations[]	Set of LiDAR Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.30.5 Conformance Testing

A Data instance Conforms with LiDAR Object (OSD-LIO) if:

1. The Data validates against the LiDAR Object's JSON Schema.
2. All Data in the LiDAR Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.31 Basic LiDAR Scene Descriptors

8.1.31.1 Definition

A Data Type including the LiDAR Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as LiDAR Object and LiDAR Scene, respectively.

8.1.31.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.31.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicLiDARSceneDescriptors.json>

8.1.31.4 Semantics

Label	Description
Header	Basic LiDAR Scene Descriptors Header
- Standard-BasicLiDARSceneDescriptors	The characters “OSD-BRS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.

DescrMetadata	Descriptive Metadata
----------------------	----------------------

8.1.31.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BLS) if:

1. The Data validates against the Scene Descriptors' JSON Schema.
2. All Data in the Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.32 LiDAR Scene Descriptors

8.1.32.1 Definition

A Data Type including the LiDAR Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.32.2 Functional Requirements

LiDAR Scene Descriptors include

1. LiDAR Objects
2. The Descriptors of the LiDAR Scenes includes in the LiDAR Scene called LiDAR Sub-Scenes.
3. Rights that may be exercised on the LiDAR Scene.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

8.1.32.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LiDARSceneDescriptors.json>

8.1.32.4 Semantics

Label	Description
Header	LiDAR Scene Descriptors Header
- Standard-LiDARSceneDescriptors	The characters "OSD-LSD-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.

- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.32.5 Conformance Testing

A Data instance Conforms with LiDAR Scene Descriptors (OSD-LSD) if:

1. The Data validates against the Scene Descriptors' JSON Schema.
2. All Data in the Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.33 Basic LiDAR Scene Geometry

8.1.33.1 Definition

A Data Type including the arrangement of the LiDAR Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as LiDAR Data, LiDAR Objects, LiDAR Qualifiers, and LiDAR Scenes.

8.1.33.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.33.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicLiDARSceneGeometry.json>

8.1.33.4 Semantics

Label	Description
Header	Basic LiDAR Scene Geometry Header
- Standard-BasicLiDARSceneGeometry	The characters "OSD-BLG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.33.5 Conformance Testing

A Data instance Conforms with Basic Audio Scene Geometry (OSD-BLG) if:

1. The Data validates against the Scene Geometry's JSON Schema.
2. All Data in the Scene Geometry's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.34 LiDAR Scene Geometry

8.1.34.1 Definition

An Data Type including the arrangement of the LiDAR Objects in a scene with their LiDAR Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as LiDAR Data, LiDAR Objects, LiDAR Qualifiers, and LiDAR (Sub-)Scenes

8.1.34.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.34.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LiDARSceneGeometry.json>

8.1.34.4 Semantics

Label	Description
Header	LiDAR Scene Geometry Header
- Standard-LiDARSceneGeometry	The characters "OSD-LSG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.
– SceneObjectSpaceTime	Space Time of Object.

SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.34.5 Conformance Testing

A Data instance Conforms with LiDAR Scene Geometry (OSD-LSG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.35 LiDAR Event Descriptors

8.1.35.1 Definition

An Item including a series of LiDAR Scene Descriptors for a certain duration.

8.1.35.2 Functional Requirements

LiDAR Event Descriptors contains LiDAR Scene Descriptors for a Time.

8.1.35.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LiDAREventDescriptors.json>

8.1.35.4 Semantics

Label	Description
Header	LiDAR Event Descriptors Header
- Standard-LiDAREventDescriptors	The characters “OSD-LIE-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.35.5 Conformance Testing

A Data instance Conforms with MPAI-OSD LiDAR Event Descriptors (OSD-LIE) if:

1. The Data validates against the LiDAR Event Descriptors' JSON Schema.
2. All Data in the LiDAR Event Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.36 Basic Location Object

8.1.36.1 Definition

ML Model Object refers to

1. A region with Space-Time attributes that is not exposed as further subdivided.
2. Location Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes.

8.1.36.2 Functional Requirements

A Basic Location Object is composed of spatial units having the same Qualifier.

8.1.36.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicLocationObject.json>

8.1.36.4 Semantics

Label	Description
Header	Basic Location Object Header
– Standard-BasicLocationObject	The characters “OSD-BLO-V”
– Version	Major version
– Dot-separator	The character “.”
– Subversion	Minor version
MInstanceID	Identifier of M-Instance.
BasicLocationObjectID	Identifier of Basic Location.
BasicLocationSpaceTime	Space-Time of the Basic Location.
BasicLocationData	The Data of a Basic Location.
BasicLocationQualifier	The Qualifier of the Basic Location.
DescrMetadata	Descriptive Metadata.

8.1.36.5 Conformance Testing

A Data instance Conforms with Basic Location (OSD-BLC) if the Data:

1. The Data validates against the Basic Location's JSON Schema.
2. All Data in the Basic Location's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas

8.1.37 Location Object

8.1.37.1 Definition

A region with Space-Time attributes that is further subdivided in Basic Locations.

8.1.37.2 Functional Requirements

A Location

1. Has an extension limited in Space and Time.
2. Is composed of Basic Locations, e.g.:
 1. A room can be a Basic Location of the Location defined as an apartment.
 2. An apartment is a Location that is contained in the Location called building.

8.1.37.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LocationObject.json>

8.1.37.4 Semantics

Label	Description
Header	Location Header
– Standard	The characters “MMM-LOC-V”
– Version	Major version
– Dot-separator	The character “.”
– Subversion	Minor version
MInstanceID	Identifier of M-Instance.
BasicLocationCount	Number of Basic Locations in Location.
BasicLocations[]	Set of Basic Locations in Location.
– BasicLocationID or BasicLocation	Basic Location as an ID or as Basic Location
– BasicLocationSpaceTime	Basic Location Space-Time
LocationCount	Number of Locations in Location.
Locations[]	Set of Locations in Location.
– LocationID or Location	ObjectID or Object.
– LocationSpaceTime	Space-Time of Object.
DescrMetadata	Descriptive Metadata.

8.1.37.5 Conformance Testing

A Data instance Conforms with Location (OSD-LOC) if:

1. The Data validates against the Location’s JSON Schema.
2. All Data in the Location’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.38 Basic Offline Map Object

8.1.38.1 Definition

Basic Offline Map Object refers to

1. Offline Map Data representing from a single Offline Map.
2. Offline Map Data Qualifier specified by MPai-TFA providing information on Sub-Types, Formats and Attributes of LiDAR Data.

In the following, Data, Qualifier, and Object should be read as Basic Offline Map Data, Basic Offline Map Qualifiers, and Basic Offline Map Object, respectively.

8.1.38.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

8.1.38.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicOfflineMapObject.json>

8.1.38.4 Semantics

Label	Description
Header	Basic Basic Offline Map Object Header
– Standard-BasicBasicOfflineMapObject	The characters “OSD-BOO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.38.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object's JSON Schema.
2. All Data in the Basic Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.39 Offline Map Object

8.1.39.1 Definition

A Data Type including a collection of Basic Offline Map Objects.

An Offline Map Object can have a hierarchical structure where Offline Map Objects contain Basic Offline Map Objects and Offline Map Objects.

8.1.39.2 Functional Requirements

An Offline Map Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Offline Map Object.
3. Space-Time information of the Offline Map Object.
4. Basic Offline Map Object and Offline Map Objects included in the Offline Map Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Offline Map Object.
6. The Rights that may be exercised on the Offline Map Object.

Note that.

1. An Offline Map Object that does not include Sub-Scenes and only one Basic Offline Map Object is a Basic Offline Map Object.
2. The Space-Time information of a Basic Offline Map Object and Offline Map Object included in an Offline Map Object may be superseded by the Space-Time information of the Offline Map Object containing them.

8.1.39.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/OfflineMapObject.json.json>

8.1.39.4 Semantics

Label	Description
Header	Offline Map Object Header
– Standard-AudioObject	The characters “OSD-OMO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
OfflineMapObjectID	Identifier of the Offline Map Object.
OfflineMapObjectSpaceTime	Space-Time of Offline Map Object.
BasicOfflineMapObjectCount	Set of Parent Offline Map Objects.

BasicOfflineMapObjects[]	Set of Basic Offline Map Objects.
- SpaceTime	Space Time of a Basic Offline Map Object in the Offline Map Object.
- Basic OfflineMapObject	A Basic Offline Map Object in the Offline Map Object.
OfflineMapObjectCount	Number of Offline Map Objects.
OfflineMapObjects[]	Set of Offline Map Objects.
- SpaceTime	Space Time of an Offline Map Object in the Offline Map Object.
- AudioObject	An Offline Map Object in the Offline Map Object
Annotations[]	Set of Offline Map Object Annotation.
- Annotation	An Annotation.
- AnnotationSpaceTime	Where Annotation is attached and when it will be active.
- Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.39.5 Conformance Testing

A Data instance Conforms with Offline Map Object (OSD-AUO) if:

1. The Data validates against the Offline Map Object's JSON Schema.
2. All Data in the Offline Map Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.40 Basic Offline Map Scene Descriptors

8.1.40.1 Definition

A Data Type including the Offline Map Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as Offline Map Object and Offline Map Scene, respectively.

8.1.40.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.40.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicOfflineMapSceneDescriptors.json>

8.1.40.4 Semantics

Label	Description
Header	Basic Offline Map Scene Descriptors Header

- Standard-BasicOfflineMapSceneDescriptors	The characters “OSD-BOS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.40.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BOS) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.41 Offline Map Scene Descriptors

8.1.41.1 Definition

A Data Type including the Offline Map Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.41.2 Functional Requirements

Offline Map Scene Descriptors include

1. Offline Map Objects
2. The Descriptors of the Offline Map Scenes includes in the Offline Map Scene called Offline Map Sub-Scenes.
3. Rights that may be exercised on the Offline Map Scene.

Scenes may be hierarchical, i.e., they may contain Offline Map Objects and Offline Map Scenes.

8.1.41.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/OfflineMapSceneDescriptors.json>

8.1.41.4 Semantics

Label	Description
-------	-------------

Header	Offline Map Scene Descriptors Header
- Standard-Offline MapSceneDescriptors	The characters “OSD-OSD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.41.5 Conformance Testing

A Data instance Conforms with Offline Map Scene Descriptors (OSD-OSD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.42 Basic Offline Map Scene Geometry

8.1.42.1 Definition

A Data Type including the arrangement of the Offline Map Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Offline Map Data, Offline Map Objects, Offline Map Qualifiers, and Offline Map Scenes.

8.1.42.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.42.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicOfflineMapSceneGeometry.json>

8.1.42.4 Semantics

Label	Description
Header	Basic Offline Scene Geometry Header
- Standard-BasicOfflineMapSceneGeometry	The characters “OSD-BLG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.42.5 Conformance Testing

A Data instance Conforms with Basic Offline Scene Geometry (OSD-BOG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.43 Offline Map Scene Geometry

8.1.43.1 Definition

An Data Type including the arrangement of the Offline Map Objects in a scene with their Offline Map Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as Offline Map Data, Offline Map Objects, Offline Map Qualifiers, and Offline Map (Sub-)Scenes

8.1.43.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.43.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/OfflineMapSceneGeometry.json>

8.1.43.4 Semantics

Label	Description
Header	Offline Map Scene Geometry Header
- Standard-OfflineMapSceneGeometry	The characters “OSD-OSG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.
– SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.43.5 Conformance Testing

A Data instance Conforms with Offline Map Scene Geometry (OSD-OSG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.44 Offline Map Event Descriptors

8.1.44.1 Definition

An Item including a series of Offline Map Event Descriptors Scene Descriptors for a certain duration.

8.1.44.2 Functional Requirements

Offline Map Event Descriptors contains Offline Map Scene Descriptors for a Time.

8.1.44.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/OfflineMapEventDescriptors.json>

8.1.44.4 Semantics

Label	Description
Header	Offline Map Event Descriptors Header
- Standard-OfflineMapEventDescriptors	The characters “OSD-OME-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.44.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Offline Map Event Descriptors (OSD-OME) if:

1. The Data validates against the Offline Map Event Descriptors’ JSON Schema.
2. All Data in the Offline Map Event Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.45 Basic RADAR Object

8.1.45.1 Definition

Basic RADAR Object refers to:

1. RADAR Data representing captured by a single RADAR Device.
2. RADAR Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of RADAR Data.

In the following, Data, Qualifier, and Object should be read as RADAR Data, RADAR Qualifiers, and RADAR Object, respectively.

8.1.45.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.

8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

8.1.45.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicRADARObject.json>

8.1.45.4 Semantics

Label	Description
Header	Basic RADAR Object Header
– Standard-BasicRADARObject	The characters “OSD-BRO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.45.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object’s JSON Schema.
2. All Data in the Basic Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.46 RADAR Object

8.1.46.1 Definition

A Data Type including a collection of Basic RADAR Objects.

A RADAR Object can have a hierarchical structure where RADAR Objects contain Basic RADAR Objects and RADAR Objects.

8.1.46.2 Functional Requirements

A RADAR Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the RADAR Object.
3. Space-Time information of the RADAR Object.
4. Basic RADAR Object and RADAR Objects included in the RADAR Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the RADAR Object.
6. The Rights that may be exercised on the RADAR Object.

Note that.

1. A RADAR Object that does not include Sub-Scenes and only one Basic RADAR Object is a Basic RADAR Object.
2. The Space-Time information of a Basic RADAR Object and RADAR Object included in a RADAR Object may be superseded by the Space-Time information of the RADAR Object containing them.

8.1.46.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/RADARObject.json>

8.1.46.4 Semantics

Label	Description
Header	RADAR Object Header
– Standard-RADARObject	The characters “OSD-RAO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
RADARObjectID	Identifier of the RADAR Object.
RADARObjectSpaceTime	Space-Time of RADAR Object.
BasicRADARObjectCount	Set of Parent RADAR Objects.
BasicRADARObjects[]	Set of Basic RADAR Objects.
- SpaceTime	Space Time of a Basic RADAR Object in the RADAR Object.
- BasicRADARObject	A Basic RADAR Object in the RADAR Object.
RADARObjectCount	Number of RADAR Objects.
RADARObjects[]	Set of RADAR Objects.
- SpaceTime	Space Time of a RADAR Object in the RADAR Object.
- RADARObject	A RADAR Object in the RADAR Object
Annotations[]	Set of RADAR Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation

Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.46.5 Conformance Testing

A Data instance Conforms with RADAR Object (OSD-RAO) if:

1. The Data validates against the RADAR Object's JSON Schema.
2. All Data in the RADAR Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.47 Basic RADAR Scene Descriptors

8.1.47.1 Definition

A Data Type including the RADAR Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as RADAR Object and RADAR Scene, respectively.

8.1.47.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.47.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicRADARSceneDescriptors.json>

8.1.47.4 Semantics

Label	Description
Header	Basic RADAR Scene Descriptors Header
- Standard-BasicRADARSceneDescriptors	The characters "OSD-BRS-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.

- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.47.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BRS) if:

1. The Data validates against the Scene Descriptors' JSON Schema.
2. All Data in the Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.48 RADAR Scene Descriptors

8.1.48.1 Definition

A Data Type including the RADAR Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.48.2 Functional Requirements

RADAR Scene Descriptors include

1. RADAR Objects
2. The Descriptors of the RADAR Scenes includes in the RADAR Scene called RADAR Sub-Scenes.
3. Rights that may be exercised on the RADAR Scene.

Scenes may be hierarchical, i.e., they may contain RADAR Objects and RADAR Scenes.

8.1.48.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/RADARSceneDescriptors.json>

8.1.48.4 Semantics

Label	Description
Header	RADAR Scene Descriptors Header
- Standard-RADARSceneDescriptors	The characters "OSD-RSD-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.

- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.48.5 Conformance Testing

A Data instance Conforms with RADAR Scene Descriptors (OSD-RSD) if:

1. The Data validates against the Scene Descriptors' JSON Schema.
2. All Data in the Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.48.6 Definition

A Data Type including the arrangement of the RADAR Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as RADAR Data, RADAR Objects, RADAR Qualifiers, and RADAR Scenes.

8.1.48.7 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.48.8 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicRADARSceneGeometry.json>

8.1.48.9 Semantics

Label	Description
Header	Basic RADAR Scene Geometry Header
- Standard-BasicRADARSceneGeometry	The characters "OSD-BRG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.

- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.48.10 Conformance Testing

A Data instance Conforms with Basic RADAR Scene Geometry (OSD-BRG) if:

1. The Data validates against the Scene Geometry's JSON Schema.
2. All Data in the Scene Geometry's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.49 RADAR Scene Geometry

8.1.49.1 Definition

An Data Type including the arrangement of the RADAR Objects in a scene with their RADAR Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as RADAR Data, RADAR Objects, RADAR Qualifiers, and RADAR (Sub-)Scenes

8.1.49.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.49.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/RADARSceneGeometry.json>

8.1.49.4 Semantics

Label	Description
Header	RADAR Scene Geometry Header
- Standard-RADARSceneGeometry	The characters "OSD-RSG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.

– SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.49.5 Conformance Testing

A Data instance Conforms with RADAR Scene Geometry (OSD-RSG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.50 RADAR Event Descriptors

8.1.50.1 Definition

An Item including a RADAR Event Descriptors Scene Descriptors for a certain duration.

8.1.50.2 Functional Requirements

RADAR Event Descriptors contains RADAR Scene Descriptors for a Time.

8.1.50.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/RADAREventDescriptors.json>

8.1.50.4 Semantics

Label	Description
Header	RADAR Event Descriptors Header
- Standard-RADAREventDescriptors	The characters “OSD-RAE-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.50.5 Conformance Testing

A Data instance Conforms with MPAI-OSD RADAR Event Descriptors (OSD-RAE) if:

1. The Data validates against the RADAR Event Descriptors' JSON Schema.
2. All Data in the RADAR Event Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.51 Basic Speech Object

8.1.51.1 Definition

A Data Type including:

1. Data representing:
 1. Content represented as Data whose rendering has vocal attributes, or
 2. Synthetic Data intended for rendering as speech, or
 3. A mixture of the two types, and
2. Optionally, descriptive Data regarding Speech Sub-Types, Formats and Attributes.

In the following, Data, Qualifier, and Object should be read as Speech Data, Speech Qualifiers, and Speech Object, respectively.

8.1.51.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

8.1.51.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicSpeechObject.json>

8.1.51.4 Semantics

Label	Description
Header	Basic Speech Object Header
– Standard-BasicSpeechObject	The characters “OSD-BSO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.

BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Speech Data.
BasicObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.51.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object's JSON Schema.
2. All Data in the Basic Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.52 Speech Object

8.1.52.1 Definition

A Data Type including a collection of Basic Speech Objects.

A Speech Object can have a hierarchical structure where Speech Objects contain Basic Speech Objects and Speech Objects.

8.1.52.2 Functional Requirements

A Speech Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Speech Object.
3. Space-Time information of the Speech Object.
4. Basic Speech Object and Speech Objects included in the Speech Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Speech Object.
6. The Rights that may be exercised on the Speech Object.

Note that.

1. A Speech Object that does not include Sub-Scenes and only one Basic Speech Object is a Basic Speech Object.
2. The Space-Time information of a Basic Speech Object and Speech Object included in a Speech Object may be superseded by the Space-Time information of the Speech Object containing them.

8.1.52.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpeechObject.json>

8.1.52.4 Semantics

Label	Description
Header	Speech Object Header
– Standard-SpeechObject	The characters “OSD-SPO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SpeechObjectID	Identifier of the Speech Object.
SpeechObjectSpaceTime	Space-Time of Speech Object.
BasicSpeechObjectCount	Set of Parent Speech Objects.
BasicSpeechObjects[]	Set of Basic Speech Objects.
- SpaceTime	Space Time of a Basic Speech Object in the Speech Object.
- BasicSpeechObject	A Basic Speech Object in the Speech Object.
SpeechObjectCount	Number of Speech Objects.
SpeechObjects[]	Set of Speech Objects.
- SpaceTime	Space Time of a Speech Object in the Speech Object.
- SpeechObject	A Speech Object in the Speech Object
Annotations[]	Set of Speech Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.52.5 Conformance Testing

A Data instance Conforms with Speech Object (OSD-SPO) if:

1. The Data validates against the Speech Object’s JSON Schema.
2. All Data in the Speech Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.53 Basic Speech Scene Descriptors

8.1.53.1 Definition

A Data Type including the Speech Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as Speech Object and Speech Scene, respectively.

8.1.53.2 Functional Requirements

Basic Scene Descriptors include

1. Objects

2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.53.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicSpeechSceneDescriptors.json>

8.1.53.4 Semantics

Label	Description
Header	Basic Speech Scene Descriptors Header
- Standard-BasicSpeechSceneDescriptors	The characters “OSD-BSS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.53.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BSS) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.54 Speech Scene Descriptors

8.1.54.1 Definition

A Data Type including the Speech Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.54.2 Functional Requirements

Speech Scene Descriptors include

1. Speech Objects

2. The Descriptors of the Speech Scenes includes in the Speech Scene called Speech Sub-Scenes.
 3. Rights that may be exercised on the Speech Scene.
- Scenes may be hierarchical, i.e., they may contain Speech Objects and Speech Scenes.

8.1.54.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpeechSceneDescriptors.json>

8.1.54.4 Semantics

Label	Description
Header	Speech Scene Descriptors Header
- Standard-SpeechSceneDescriptors	The characters “OSD-SSD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.54.5 Conformance Testing

A Data instance Conforms with Speech Scene Descriptors (OSD-SSD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.55 Basic Speech Scene Geometry

8.1.55.1 Definition

A Data Type including the arrangement of the Speech Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Speech Data, Speech Objects, Speech Qualifiers, and Speech Scenes.

8.1.55.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.55.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicSpeechSceneGeometry.json>

8.1.55.4 Semantics

Label	Description
Header	Basic Speech Scene Geometry Header
- Standard-BasicSpeechSceneGeometry	The characters “OSD-BSG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.55.5 Conformance Testing

A Data instance Conforms with Basic Speech Scene Geometry (OSD-BSG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.56 Speech Scene Geometry

8.1.56.1 Definition

A Data Type including the arrangement of the Speech Objects in a scene with their Speech Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as Speech Data, Speech Objects, Speech Qualifiers, and Speech (Sub-)Scenes

8.1.56.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.56.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpeechSceneGeometry.json>

8.1.56.4 Semantics

Label	Description
Header	Speech Scene Geometry Header
- Standard-SpeechSceneGeometry	The characters “OSD-SSG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.
– SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.56.5 Conformance Testing

A Data instance Conforms with Speech Scene Geometry (OSD-SSG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas

3. Conform with their Data Qualifiers.

8.1.57 Speech Event Descriptors

8.1.57.1 Definition

An Item including a series of Speech Scene Descriptors for a certain duration.

8.1.57.2 Functional Requirements

Speech Event Descriptors contains Speech Scene Descriptors for a Time.

8.1.57.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpeechEventDescriptors.json>

8.1.57.4 Semantics

Label	Description
Header	Speech Event Descriptors Header
- Standard-SpeechEventDescriptors	The characters “OSD-SPE-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.57.5 Conformance Testing

A Data instance Conforms with MPai-OSD Speech Event Descriptors (OSD-SPE) if:

1. The Data validates against the Speech Event Descriptors' JSON Schema.
2. All Data in the Speech Event Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.58 Basic Text Object

8.1.58.1 Definition

A Data Type including characters and Character Qualifier.

8.1.58.2 Functional Requirements

A Basic Text Object may include:

1. The ID of a Virtual Space (M-Instance) to which it belongs.

2. The ID of the Basic Text Object.
3. The Space-Time information of the Basic Text Object in an M-Instance.
4. The Qualifier of the characters composing the Text Data.
5. The Rights that can be exercised on the Basic Text Object.
6. The set of Annotations including, for each Annotation, Rights to perform Actions on the Annotation.

8.1.58.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicTextObject.json>

8.1.58.4 Semantics

Label	Description
Header	Basic Text Object Header
– Standard-BasicTextObject	The characters “OSD-BTO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Text Object.
BasicTextObjectSpaceTime	Space-Time info of the Basic Text Object.
BasicTextData	The Text in the Basic Text Object.
TextDataQualifier	Qualifier of Text Data.
BasicTextObjectAnnotations[]	Annotations of Basic Object.
– Annotation	ID of Annotation
– AnnotationTime	The Time of the Annotation.
- AnnotationRight	Rights to Act on the Annotation.
BasicTextObjectRights	Rights to perform Actions of the Basic Text Object.
DescrMetadata	Descriptive Metadata

8.1.58.5 Conformance Testing

A Data instance Conforms with Basic Text Object if:

1. The Data validates against the Basic Text Object’s JSON Schema.
2. All Data in the Basic Text Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.59 Text Object

8.1.59.1 Definition

A Data Type including a collection of Basic Text Objects.

A Text Object can have a hierarchical structure where Text Objects contain Basic Text Objects and Text Objects.

8.1.59.2 Functional Requirements

A Text Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Text Object.
3. Space-Time information of the Text Object.
4. Basic Text Object and Text Objects included in the Text Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Text Object.
6. The Rights that may be exercised on the Text Object.

Note that.

1. A Text Object that does not include Sub-Scenes and only one Basic Text Object is a Basic Text Object.
2. The Space-Time information of a Basic Text Object and Text Object included in a Text Object may be superseded by the Space-Time information of the Text Object containing them.

8.1.59.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/TextObject.json>

8.1.59.4 Semantics

Label	Description
Header	Text Object Header
– Standard-TextObject	The characters “OSD-TXO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
TextObjectID	Identifier of the Text Object.
TextObjectSpaceTime	Space-Time of Text Object.
BasicTextObjectCount	Set of Parent Text Objects.
BasicTextObjects[]	Set of Basic Text Objects.
- SpaceTime	Space Time of a Basic Text Object in the Text Object.
- BasicTextObject	A Basic Text Object in the Text Object.
TextObjectCount	Number of Text Objects.
TextObjects[]	Set of Text Objects.
- SpaceTime	Space Time of a Text Object in the Text Object.
- TextObject	A Text Object in the Text Object
Annotations[]	Set of Text Object Annotation.
– Annotation	An Annotation.

– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.59.5 Conformance Testing

A Data instance Conforms with Text Object (OSD-TXO) if:

1. The Data validates against the Text Object's JSON Schema.
2. All Data in the Text Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.60 Simple Time

8.1.60.1 Definition

The digital representation of (duration of) time specified by MPAL.

8.1.60.2 Functional Requirements

Time includes start time and the end time of a duration. The origin of Time can be Absolute (from 1970/01/01) or relative to a user-selected value.

8.1.60.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SimpleTime.json>

8.1.60.4 Semantics

Label	Description
Header	Simple Time Header
- Standard-SimpleTime	The characters "OSD-STM-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
SimpleTimeID	Identifier of M-Instance.
SimpleTimeData	Data about Time
- TimeType	0=Relative: start at 0000/00/00T00:00 1=Absolute: start at 1970/01/01T00:00.
- Reserved	reserved
- TimeUnit	00=seconds, 01=milliseconds, 10=microseconds, 11=nanoseconds.
- StartTime	Start of Time.

- EndTime	End of Time.
DescrMetadata	Descriptive Metadata

8.1.60.5 Conformance Testing

A Data instance Conforms with OSD-STM if:

1. The Data validates against the Basic Times's JSON Schema.
2. All Data in the JSON Basic Times have the specified type.

8.1.61 Time

8.1.61.1 Definition

The digital representation of duration of time.

8.1.61.2 Functional Requirements

Time includes the digital representation of time specified by MPAI or other Standard Setting Organisations as indicated by Time Qualifier

8.1.61.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Time.json>

8.1.61.4 Semantics

Label	Description
Header	Time Header
- Standard-Object	The characters "OSD-TIM-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
TimeID	Identifier of M-Instance.
TimeData	Data about Time
TimeQualifier	End of Time.
DescrMetadata	Descriptive Metadata

8.1.61.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Time (OSD-TIM) if:

1. The Data validates against the Times's JSON Schema.
2. All Data in JSON Times has the specified type.

8.1.62 Basic Ultrasound Object

8.1.62.1 Definition

Basic Ultrasound Object refers to Data captured by a single Ultrasound Device representing

1. A sound of the real world, or
2. Synthetic Data intended for rendering as sound, or
3. A mixture of the two types

and Ultrasound Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of Ultrasound Data.

In the following, Data, Qualifier, and Object should be read as Ultrasound Data, UltrasoundR Qualifiers, and Ultrasound Object, respectively.

In the following, Data, Qualifier, and Object should be read as Ultrasound Data, Ultrasound Qualifiers, and Ultrasound Object, respectively.

8.1.62.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Parent Object(s) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Child Object(s).
6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the specific Data Type.
9. The Rights that can be exercised on the Basic Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.

8.1.62.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicUltrasoundObject.json>

8.1.62.4 Semantics

Label	Description
Header	Basic Ultrasound Object Header
– Standard-BasicUltrasoundObject	The characters “OSD-BUO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicObjectID	Identifier of the Basic Object.
BasicObjectSpaceTime	Space-Time info of the Basic Object.
BasicObjectData	Data of Basic Object.
Qualifier	Qualifier of Basic Data.
BasicObjectAnnotations[]	Annotations of Basic Object.

– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
Rights	Rights to perform Actions of the Basic Object.
DescrMetadata	Descriptive Metadata

8.1.62.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object's JSON Schema.
2. All Data in the Basic Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.63 Ultrasound Object

8.1.63.1 Definition

A Data Type including a collection of Basic Ultrasound Objects.

An Ultrasound Object can have a hierarchical structure where Ultrasound Objects contain Basic Ultrasound Objects and Ultrasound Objects.

8.1.63.2 Functional Requirements

A Ultrasound Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Ultrasound Object.
3. Space-Time information of the Ultrasound Object.
4. Basic Ultrasound Object and Ultrasound Objects included in the Ultrasound Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Ultrasound Object.
6. The Rights that may be exercised on the Ultrasound Object.

Note that.

1. An Ultrasound Object that does not include Sub-Scenes and only one Basic Ultrasound Object is a Basic Ultrasound Object.
2. The Space-Time information of a Basic Ultrasound Object and Ultrasound Object included in a Ultrasound Object may be superseded by the Space-Time information of the Ultrasound Object containing them.

8.1.63.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/UltrasoundObject.json>

8.1.63.4 Semantics

Label	Description
Header	Ultrasound Object Header
– Standard-UltrasoundObject	The characters “OSD-USO-V”
– Version	Major version – 1 or 2 characters

– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
UltrasoundObjectID	Identifier of the Ultrasound Object.
UltrasoundObjectSpaceTime	Space-Time of Ultrasound Object.
BasicUltrasoundObjectCount	Set of Parent Ultrasound Objects.
BasicUltrasoundObjects[]	Set of Basic Ultrasound Objects.
- SpaceTime	Space Time of a Basic Ultrasound Object in the Ultrasound Object.
- BasicUltrasoundObject	A Basic Ultrasound Object in the Ultrasound Object.
UltrasoundObjectCount	Number of Ultrasound Objects.
UltrasoundObjects[]	Set of Ultrasound Objects.
- SpaceTime	Space Time of a Ultrasound Object in the Ultrasound Object.
- UltrasoundObject	An Ultrasound Object in the Ultrasound Object
Annotations[]	Set of Ultrasound Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.63.5 Conformance Testing

A Data instance Conforms with Ultrasound Object (OSD-USO) if:

1. The Data validates against the Ultrasound Object’s JSON Schema.
2. All Data in the Ultrasound Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.64 Basic Ultrasound Scene Descriptors

8.1.64.1 Definition

A Data Type including the Ultrasound Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following Object and Scene are to be read as Ultrasound Object and Ultrasound Scene, respectively.

8.1.64.2 Functional Requirements

Basic Scene Descriptors include

1. Objects
2. Space-Time information.
3. Rights that may be exercised on the Scene.

The Space-Time of the Objects may be superseded by the Space-Time of the Scene.

8.1.64.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicUltrasoundSceneDescriptors.json>

8.1.64.4 Semantics

Label	Description
Header	Basic Ultrasound Scene Descriptors Header
- Standard-BasicUltrasoundSceneDescriptors	The characters “OSD-BUS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
SceneObjects[]	Set of Objects.
- SceneObject	An Object.
- SceneObjectSpaceTime	Space Time of Object.
Rights	Rights that may be exercised on the Scene.
DescrMetadata	Descriptive Metadata

8.1.64.5 Conformance Testing

A Data instance Conforms with Basic Scene Descriptors (OSD-BUS) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.65 Ultrasound Scene Descriptors

8.1.65.1 Definition

A Data Type including the Ultrasound Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.65.2 Functional Requirements

Ultrasound Scene Descriptors include

1. Ultrasound Scene Objects
2. The Descriptors of the Ultrasound Scenes includes in the Ultrasound Scene called Ultrasound Sub-Scenes.
3. Rights that may be exercised on the Ultrasound Scene.

Scenes may be hierarchical, i.e., they may contain Ultrasound Objects and Ultrasound Scenes.

8.1.65.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/UltrasoundSceneDescriptors.json>

8.1.65.4 Semantics

Label	Description
Header	Ultrasound Scene Descriptors Header
- Standard-UltrasoundSceneDescriptors	The characters “OSD-USD-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneDescriptorsID	Identifier of Scene Descriptors.
SceneDescriptorsSpaceTime	Space and Time of Scene Descriptors.
ObjectCount	Number of Objects in Scene.
Objects[]	Set of Objects.
- Object or ObjectID	Object in the Scene of its ID.
- ObjectSpaceTime	Space Time of Object.
SubSceneCount	Number of Sub-Scenes in Scene.
SubScenes[]	Set of Sub-Scenes in the Scene.
- SubScene or SubSceneID	Sub-Scene in the Scene or its ID.
- SubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.65.5 Conformance Testing

A Data instance Conforms with Ultrasound Scene Descriptors (OSD-USD) if:

1. The Data validates against the Scene Descriptors’ JSON Schema.
2. All Data in the Scene Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.66 Basic Ultrasound Scene Geometry

8.1.66.1 Definition

A Data Type including the arrangement of the Ultrasound Objects in a scene.

In the following, Data, Objects, Qualifiers, and Scenes should be read as Ultrasound Data, Ultrasound Objects, Ultrasound Qualifiers, and Ultrasound Scenes.

8.1.66.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.66.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicUltrasoundSceneGeometry.json>

8.1.66.4 Semantics

Label	Description
Header	Basic Ultrasound Scene Geometry Header
- Standard-BasicUltrasoundSceneGeometry	The characters “OSD-BUG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.66.5 Conformance Testing

A Data instance Conforms with Basic Ultrasound Scene Geometry (OSD-BUG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.67 Ultrasound Scene Geometry

8.1.67.1 Definition

An Data Type including the arrangement of the Ultrasound Objects in a scene with their Ultrasound Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as Ultrasound Data, Ultrasound Objects, Ultrasound Qualifiers, and Ultrasound (Sub-)Scenes

8.1.67.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.67.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/UltrasoundSceneGeometry.json>

8.1.67.4 Semantics

Label	Description
Header	Ultrasound Scene Geometry Header
- Standard-UltrasoundSceneGeometry	The characters “OSD-USG-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.
– SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.
DescrMetadata	Descriptive Metadata.

8.1.67.5 Conformance Testing

A Data instance Conforms with Ultrasound Scene Geometry (OSD-USG) if:

1. The Data validates against the Scene Geometry’s JSON Schema.
2. All Data in the Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.68 Ultrasound Event Descriptors

8.1.68.1 Definition

An Item including a series of Ultrasound Scene Descriptors for a certain duration.

8.1.68.2 Functional Requirements

Ultrasound Event Descriptors contains Ultrasound Scene Descriptors for a Time.

8.1.68.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/UltrasoundEventDescriptors.json>

8.1.68.4 Semantics

Label	Description
Header	Ultrasound Event Descriptors Header
- Standard-UltrasoundEventDescriptors	The characters “OSD-USE-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.68.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Ultrasound Event Descriptors (OSD-USE) if:

1. The Data validates against the Ultrasound Event Descriptors’ JSON Schema.
2. All Data in the Ultrasound Event Descriptors’ JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.68.6

8.1.69 Basic Visual Object

8.1.69.1 Definition

A Data Type including:

1. Visual Data perceptible by a visual device or visible to a human when rendered.
2. Descriptive Data regarding Sub-Types, Formats and Attributes of the Visual Data (optionally).

In the following, Data, Qualifier, and Object should be read as Visual Data, Visual Qualifiers, and Visual Object, respectively.

8.1.69.2 Functional Requirements

A Basic Object may include:

1. The ID of a Virtual Space (M-Instance) where it is or is intended to be located.
2. The ID of the Basic Object.
3. The ID(s) of Objects at a preceding time (Parent Objects) supporting two cases:
 1. The Parent Object has spawned two (or more) Objects.
 2. Two (or more) Parent Objects have merged into one.
4. The Space-Time information of Parent Objects in an M-Instance.
5. The ID(s) of Objects at a following time (Child Objects).

6. The Space-Time information of Child Objects in an M-Instance.
7. The Space-Time information of the Basic Object in an M-Instance.
8. The Qualifier of the Visual Data.
9. Additional Basic Visual Object information.
 - Mass of the Object
 - Material Property - is material or immaterial
 - Gravity Property - is subject to gravity or not
 - Texture Map - the Objects texture map
 - Object Audio Characteristics - the audio characteristics of the Object.
10. The set of Annotations including, for each Annotation:
 1. Space-Time information of the Annotation.
 2. Rights to perform Actions on the Annotation.
11. The Rights that can be exercised on the Basic Object.

8.1.69.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicVisualObject.json>

8.1.69.4 Semantics

Label	Description
Header	Visual Object Header
- Standard-VisualObject	The characters “OSD-BVO-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
VisualObjectID	Identifier of the Visual Object.
ParentVisualObjects[]	Identifier(s) of Parent Visual Objects.
- ParentVisualObjectID	Identifier of a Parent Visual Object.
- ParentVisualDataSpaceTime	SpaceTime of a Parent Visual Object.
ChildVisualObjects[]	Identifier(s) of Parent Visual Objects.
- ChildVisualObjectID	Identifier of a Child Visual Object.
- ChildVisualDataSpaceTime	SpaceTime of a Child Visual Object.
BasicObjectSpace-Time	Space-Time info of Visual Data.
BasicVisualObjectData	Data of Basic Object.
VisualDataQualifier	Qualifier of Visual Data.
BasicVisualObjectProperties	Additional information on Object
- Mass	Mass of the Object
- MaterialProperty	Object has material consistence (0=No, 1=Yes)

- GravityProperty	Object is subject to gravity (0=No, 1-Yes)
- TextureMap	Texture Map
- ObjectAudioCharacteristics	Basic Visual Object's Audio Characteristics
- EulerAngles	The three axes of the Basic 3D Model Object with respect to the coordinate system.
BasicVisualObjectsAnnotations[]	Annotations of Visual Data
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
– RightsID	What is possible to do with the Annotation
BasicVisualObjectRights	Rights to Act on the Basic Visual Object.
DescrMetadata	Descriptive Metadata

8.1.69.5 Conformance Testing

A Data instance Conforms with Basic Object if:

1. The Data validates against the Basic Object's JSON Schema.
2. All Data in the Basic Object's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.70 Visual Object

8.1.70.1 Definition

A Data Type including a collection of Basic Visual Objects.

A Visual Object can have a hierarchical structure where Visual Objects contain Basic Visual Objects and Visual Objects.

8.1.70.2 Functional Requirements

A Visual Object may include:

1. ID of a Virtual Space (M-Instance) where it is or intended to be located.
2. ID of the Visual Object.
3. Space-Time information of the Visual Object.
4. Basic Visual Object and Visual Objects included in the Visual Objects.
5. Annotation data set including:
 1. Annotations
 2. Space-Times of the Annotations.
 3. Rights to perform Actions on the Visual Object.
6. The Rights that may be exercised on the Visual Object.

Note that.

1. A Visual Object that does not include Sub-Scenes and only one Basic Visual Object is a Basic Visual Object.
2. The Space-Time information of a Basic Visual Object and Visual Object included in a Visual Object may be superseded by the Space-Time information of the Visual Object containing them.

8.1.70.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/VisualObject.json>

8.1.70.4 Semantics

Label	Description
Header	Visual Object Header
– Standard-VisualObject	The characters “OSD-VIO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
VisualObjectID	Identifier of the Visual Object.
VisualObjectProperty	Properties of Visual Object - If any is present overrides the properties of component objects.
- MaterialProperty	Object has material consistence (0=No, 1=Yes)
- GravityProperty	Object is subject to gravity (0=No, 1-Yes)
VisualObjectSpaceTime	Space-Time of Visual Object.
BasicVisualObjectCount	Set of Parent Visual Objects.
BasicVisualObjects[]	Set of Basic Visual Objects.
- SpaceTime	Space Time of a Basic Visual Object in the Visual Object.
- BasicVisualObject	A Basic Visual Object in the Visual Object.
VisualObjectCount	Number of Visual Objects.
VisualObjects[]	Set of Visual Objects.
- SpaceTime	Space Time of a Visual Object in the Visual Object.
- VisualObject	A Visual Object in the Visual Object
Annotations[]	Set of Visual Object Annotation.
– Annotation	An Annotation.
– AnnotationSpaceTime	Where Annotation is attached and when it will be active.
– Rights	Actions that may be performed on the Annotation
Rights	Actions that may be performed on the Object.
DescrMetadata	Descriptive Metadata

8.1.70.5 Conformance Testing

A Data instance Conforms with Visual Object (OSD-VIO) if:

1. The Data validates against the Visual Object’s JSON Schema.
2. All Data in the Visual Object’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.71 Basic Visual Scene Descriptors

8.1.71.1 Definition

A Data Type including the Visual Objects of a scene, their time and arrangement in the scene, and the Rights that may be exercised on the scene.

In the following, "Object" and "Scene" are to be read as "Visual Object" and "Visual Scene", respectively.

8.1.71.2 Functional Requirements

Basic Visual Scene Descriptors include

1. Visual Objects
2. Space-Time information.
3. Rights that may be exercised on the Basic Visual Scene.

8.1.71.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicVisualSceneDescriptors.json>

8.1.71.4 Semantics

Label	Description
Header	Basic Visual Scene Descriptors Header
- Standard-BasicVisualSceneDescriptors	The characters “OSD-BVS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
BasicVisualSceneDescriptorsID	Identifier of Basic Visual Scene Descriptors.
GravityValue	The value of Gravity in the Basic Scene measure in m/s^2 .
VisualObjectCount	Number of Visual Objects in Basic Visual Scene.
SceneDescriptorsSpaceTime	Space and Time of Basic Visual Scene Descriptors.
BasicVisualSceneObjects[]	Set of Visual Objects.
- BasicVisualSceneObject	A Visual Object.
- BasicVisualSceneObjectSpaceTime	Space Time of Visual Object.
Gravity	Value of Gravity in Scene measured in m/s^2
Rights	Rights that may be exercised on the Basic Visual Scene.
DescrMetadata	Descriptive Metadata

8.1.71.5 Conformance Testing

A Data instance Conforms with Basic Visual Scene Descriptors (OSD-BVS) if:

1. The Data validates against the Basic Visual Scene Descriptors' JSON Schema.
2. All Data in the Basic Visual Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Visual Data Qualifiers.

8.1.72 Visual Scene Descriptors

8.1.72.1 Definition

A Data Type including the Visual Objects of a scene, their sub-scenes, and their arrangement in the scene.

8.1.72.2 Functional Requirements

Visual Scene Descriptors include

1. Visual Objects
2. The Descriptors of the Visual Scenes includes in the Visual Scene called Visual Sub-Scenes.
3. Rights that may be exercised on the Visual Scene.

Scenes may be hierarchical, i.e., they may contain Objects and Scenes.

8.1.72.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/VisualSceneDescriptors.json>

8.1.72.4 Semantics

Label	Description
Header	Visual Scene Descriptors Header
- Standard-VisualSceneDescriptors	The characters "OSD-VSD-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
VisualSceneDescriptorsID	Identifier of Visual Scene Descriptors.
ObjectCount	Number of Visual Objects in Scene.
SubSceneCount	Number of Visual Scenes in Scene.
VisualSceneDescriptorsSpaceTime	Space and Time of Visual Scene Descriptors.
VisualSceneObjects[]	Set of Visual Objects.
- VisualSceneObject	Visual Object.
- VisualSceneObjectSpaceTime	Space Time of Visual Object.
VisualSceneSubScenes[]	Set of Visual Sub-Scenes.
- VisualSceneSubScene	Visual Sub-Scene.

- VisualSceneSubSceneSpaceTime	Space Time of Visual Sub-Scene.
DescrMetadata	Descriptive Metadata

8.1.72.5 Conformance Testing

A Data instance Conforms with Visual Scene Descriptors (OSD-VSD) if:

1. The Data validates against the Visual Scene Descriptors' JSON Schema.
2. All Data in the Visual Scene Descriptors' JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.73 Basic Visual Scene Geometry

8.1.73.1 Definition

A Data Type including the arrangement of the Basic Visual Scene Geometry Objects in a scene. In the following, Data, Objects, Qualifiers, and Scenes should be read as Visual Data, Visual Objects, Visual Qualifiers, and Visual Scenes.

8.1.73.2 Functional Requirements

Basic Scene Geometry includes the Qualifiers and the Space-Time of the Objects.

8.1.73.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BasicVisualSceneGeometry.json>

8.1.73.4 Semantics

Label	Description
Header	Basic Visual Scene Geometry Header
- Standard-BasicVisualSceneGeometry	The characters "OSD-BVG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
- SceneObjectQualifiers	Qualifiers of Object.
- SceneObjectSpaceTime	Space Time of Object.
DescrMetadata	Descriptive Metadata

8.1.73.5 Conformance Testing

A Data instance Conforms with Basic Visual Scene Geometry (OSD-BVG) if:

1. The Data validates against the Scene Geometry's JSON Schema.
2. All Data in the Scene Geometry's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.74 Visual Scene Geometry

8.1.74.1 Definition

A Data Type including the arrangement of the Visual Objects in a scene with their Visual Qualifiers.

In the following, Data, Objects, Qualifiers, and (Sub-)Scenes should be read as Visual Data, Visual Objects, Visual Qualifiers, and Visual (Sub-)Scenes

8.1.74.2 Functional Requirements

Scene Geometry includes the arrangements of the Scenes - called Sub-Scenes - in addition to the arrangement of Objects.

8.1.74.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/VisualSceneGeometry.json>

8.1.74.4 Semantics

Label	Description
Header	Visual Scene Geometry Header
- Standard-VisualSceneGeometry	The characters "OSD-VSG-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
SceneGeometryID	Identifier of Scene Geometry.
ObjectCount	Number of Objects in Scene.
SubSceneCount	Number of Sub-Scenes in Scene.
SceneGeometrySpaceTime	Space and Time of Scene Geometry.
SceneObjects[]	Set of Data related to Objects.
– SceneObjectID	ID of Object.
– SceneObjectSpaceTime	Space Time of Object.
SceneSubScenes[]	Set of Sub-Scenes.
– SceneSubSceneID	ID of Sub-Scene.
– SceneSubSceneSpaceTime	Space Time of Sub-Scene.

DescrMetadata	Descriptive Metadata.
----------------------	-----------------------

8.1.74.5 Conformance Testing

A Data instance Conforms with Visual Scene Geometry (OSD-VSG) if:

1. The Data validates against the Scene Geometry's JSON Schema.
2. All Data in the Scene Geometry's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers.

8.1.75 Visual Event Descriptors

8.1.75.1 Definition

An Item including a series of Visual Scene Descriptors for a certain duration.

8.1.75.2 Functional Requirements

Visual Event Descriptors contains Visual Scene Descriptors for a Time.

8.1.75.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/UltrasoundEventDescriptors.json>

8.1.75.4 Semantics

Label	Description
Header	Visual Event Descriptors Header
- Standard-VisualEventDescriptors	The characters "OSD-VIEE-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
EventID	Identifier of the Event.
EventSpaceTime	Data about start and end Space-Time.
SceneDescriptors[]	Set of Scene Descriptors
- SceneDescriptors	Set of AV Scene Descriptors of IDs.
DescrMetadata	Descriptive Metadata

8.1.75.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Visual Event Descriptors (OSD-VIE) if:

1. The Data validates against the Annotation's JSON Schema.
2. All Data in the Annotation's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.76 Annotation

8.1.76.1 Definition

Annotation is Data attached to an Object or a Scene. As opposed to Qualifier that describes intrinsic properties of an Object, an Annotation is spatially and temporally local and changeable.

8.1.76.2 Functional Requirements

Elements of an Annotation are:

1. M-Instance ID
2. Annotation ID
3. Annotation Space-Time
4. Annotation Data
 1. JSON Text Objects
 2. Annotation Space-Time in Object or Scene
 3. Permitted Actions on Annotated Data

Annotation Data is text containing the JSON code conforming to the JSON Schema of the Item intended as Annotation. Examples of such Items are Perceptible Entities, Intention, Meaning, and Personal Status and Its components.

8.1.76.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Annotation.json>

8.1.76.4 Semantics

Label	Description
Header	Annotation Header
- Standard-Annotation	The characters “OSD-ANN-V”
- Version	Major version – 1 or 2 Bytes
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 Bytes
MInstanceID	The Virtual Space whose Object or Scene contains Annotations.
AnnotationID	Identifier of Annotation.
Annotation[]	The actual Annotation.
- AnnotationJSONText	Text of the JSON representing the Data Type used in the Annotation.
- AnnotationSpaceTime	Where/when Annotation is attached.
- ProcessActions[]	What is possible to do with the Annotation
- ProcessActionID	List of possible Process Actions
DescrMetadata	Descriptive Metadata

8.1.76.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Annotation (OSD-ANN) if:

1. The Data validates against the Annotation’s JSON Schema.
2. All Data in the Annotation’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas

3. Conform with their Data Qualifiers if present.

8.1.77 Perceptible Entity

8.1.77.1 Definition

Perceptible Entity is one of

1. 3D Model, Audio, Audio-Visual, Speech, Text, and Visual Object.
2. 3D Model, Audio, Audio-Visual, Speech, and Visual Scene.
3. 3D Model, Audio, Audio-Visual, Speech, and Visual Event.

8.1.77.2 Functional Requirements

A Perceptible Entity

1. Inherits the Functional requirements of Objects, Scenes, and Events listed above.
2. May include Rights that are Granted to certain Process to perform certain Actions at certain Times and Locations on the Perceptible Entity.

8.1.77.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/PerceptibleEntity.json>

8.1.77.4 Semantics

Label	Description
Header	Perceptible Entity Header
- Standard-PerceptibleEntity	The characters “OSD-PCE-V”
- Version	Major version – 1 or 2 Bytes
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 Bytes
PerceptibleEntityID	Identifier of Perceptible Entity.
PerceptibleEntity	Anyone of the following Objects, Scenes, or Events.
- Object	Intended Object
- Scene	Intended Scene
- Event	Intended Event
- RightsID or Rights	Individual Rights ID
DescrMetadata	Descriptive Metadata

8.1.77.5 Conformance Testing

A Data instance Conforms with Perceptible Entity (OSD-PCE) if:

1. The Data validates against the Perceptible Entity’s JSON Schema.
2. All Data in the Perceptible Entity’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.78 Audio Source

8.1.78.1 Definition

A Data Type representing an audio source and its characteristics.

8.1.78.2 Functional Requirements

An Audio Source may include:

1. The ID of the M-Instance where it is located.
2. The Audio Source's
 1. ID.
 2. Time-Space information.
 3. Type.
 4. Annotation.
 5. Rights.

8.1.78.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/AudioSource.json>

8.1.78.4 Semantics

Label	Description
Header	Audio Source Header
– Standard-AudioSource	The characters “OSD-AUS-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
AudioSourceID	Identifier of the Audio Source.
AudioSourceSpaceTime	Space-Time information of the Audio Source.
AudioSourceType	Type of source.
- Diffuseness	- Matrix of Frequency (Hz) - Normalised Intensity in dB.
- DirectionalPatterns	- Tensor of Frequency (Hz) - Directivity (Azimuth-Elevation in degrees) - Normalised Intensity in dB.
- ShapeAndSize	- Tensor of Frequency (Hz) - Directivity (Azimuth-Elevation in degrees) - Normalised Intensity in dB. - Shape of Source and its Size.
AudioSourceAnnotation	Annotations of Audio Source.
– Annotation	ID of Annotation.
– AnnotationSpaceTime	Where/when Annotation is attached.
– RightsID	Rights that may be exercised on the Annotation.
AudioSourceRights	Rights that may be exercised on the Audio Source.
DescrMetadata	Descriptive Metadata.

8.1.78.5 Conformance Testing

A Data instance Conforms with Audio Source if:

1. The Data validates against the Audio Source's JSON Schema.
2. All Data in the Audio Source's JSON Schema

1. Have the specified type
2. Validate against their JSON Schemas
3. Conform with their Data Qualifiers, if present.

8.1.79 Point of View

8.1.79.1 Definition

Position and Orientation of an Object in a Virtual Environment excluding velocity and acceleration.

8.1.79.2 Functional Requirements

- An Object may have one of the following attributes: Speech, Audio; Visual; 3D Model, Audio-Visual; Haptic; Smell; RADAR; LiDAR; Ultrasound.
- Accuracy is the estimated absolute difference between the measured spatial and angular values of each of CartPosition, SpherPosition, Orientation, and their true value.

8.1.79.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/PointOfView.json>

8.1.79.4 Semantics

Table 1 provides the semantics of the components of Point of View. The following should be noted:

1. Each of Position, Velocity, and Acceleration is provided either in Cartesian (X,Y,Z) or Spherical (r,φ,θ) Coordinates.
2. The Euler angles are indicated by (α,β,γ).

Table 1 – Semantics of Point of View

Label	Description
Header	Point of View Header
- Standard-Point of View	The characters “OSD-OPV-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstance	ID id Virtual space Orientation refers tu
PointOfViewID	Identifier of Object Point of View.
General	Set of general data.
- CoordType	One of Cartesian, Spherical, Geodesic, Toroidal.
- ObjectType	One of Digital Human, Generic.
- MediaType	One of Speech, Audio, Visual, Audio-Visual, Haptic, Smell, RADAR, LiDAR, Ultrasound.
PositionAndOrientation	
- CartPosition (X,Y,Z)	Array (in metres)
- CartPositionAccuracy (X,Y,Z)	Array Of CartPositionAccuracy
- SpherPosition (r,φ,θ)	Array (in metres and degrees)

- SpherPositionAccuracy (r,φ,θ)	Array of - SpherPositionAccuracy
- Orient (α,β,γ)	Array (in degrees)
- OrientAccuracy (α,β,γ)	Array of OrientAccuracy
DescrMetadata	Descriptive Metadata

8.1.79.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Point of View (OSD-OPV) if:

1. The Data validates against the Point of View's JSON Schema.
2. All Data in the Point of View's JSON Schema.
 1. Have the specified type.
 2. Validate against their JSON Schemas.

8.1.80 Bounding Box

8.1.80.1 Definition

A Data Type representing a rectangle or right parallelepiped containing a 2D or 3D Visual Object, respectively.

8.1.80.2 Functional Requirements

The rectangle or right parallelepiped is defined, respectively, by

1. Rectangle (2D): 3 vertices not on a straight line.
2. Right Parallelepiped (3D): 4 vertices not on a plane.

The Visual Object (Content) may be absent. If present, it may fit exactly in the rectangle/parallelepiped and have the same axes of the rectangle/parallelepiped.

8.1.80.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/BoundingBox.json>

8.1.80.4 Semantics

Label	Description
Header	Bounding Box Header
- Standard-BoundingBox	The characters "OSD-BBX-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
BoundingBoxID	Identifier of Bounding Box.
Dimension	One of 2D, 3D
VisualData	Visual Data in the Bounding Box.
VisualDataQualifier	Qualifier of Visual Data in the Bounding Box.
DescrMetadata	Descriptive Metadata

8.1.80.5 Conformance Testing

A Data instance Conforms with Bounding Box (OSD-BBX) if:

1. The Data validates against the Bounding Box's JSON Schema.
2. All Data in the Bounding Box's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.81 Position

8.1.81.1 Definition

A Data Type representing an Object's position, velocity, and acceleration.

8.1.81.2 Functional Requirements

- The Position of an Object is given by the Coordinates of a representative point of the Object.
- Cartesian and Polar Coordinate Systems are supported.
- An Object may have one of the following attributes: Speech, Audio; Visual; 3D Model, Audio-Visual; Haptic; Smell; RADAR; LiDAR; Ultrasound.
- Accuracy is the estimated absolute difference between the measured spatial values of each of CartPosition, SpherPosition, CartVelocity, SpherVelocity, CartAccel, SpherAccel and their true value.

8.1.81.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Position.json>

8.1.81.4 Semantics

Table 1 provides the semantics of Position. It should be noted that each of Position, Velocity, and Acceleration can be expressed either in Cartesian (X,Y,Z) or Spherical (r,φ,θ) Coordinates.

Table 1 – Semantics of the Spatial Attitude

Label	Description
Header	Position Header
- Standard-Position	The characters "OSD-OPS-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	ID of Virtual Space Position refers to.
ObjectPositionID	Identifier of Object Position.
General	Set of general data
- CoordinateType	One of Cartesian, Spherical, Geodesic, Toroidal.
- ObjectType	One of Digital Human, Generic.
- MediaType	One of Speech, Audio, Visual, 3D Model, Audio-Visual, Haptic, Smell, RADAR, LiDAR, Ultrasound.
Position	
- CartPosition (X,Y,Z)	Array (in metres)

- CartPositionAccuracy (X,Y,Z)	Array of CartPositionAccuracy
- SpherPosition (r,φ,θ)	Array (in metres and degrees)
- SpherPositionAccuracy (r,φ,θ)	Array of SpherPositionAccuracys
Velocity of Position	
- CartVelocity (X,Y,Z)	Array (in metres)
- CartVelocityAccuracy (X,Y,Z)	Array of - CartVelocityAccuracys (X,Y,Z)
- SpherVelocity (r,φ,θ)	Array (in metres and degrees)
- SpherVelocityAccuracy (r,φ,θ)	Array of SpherVelocityAccuracys
Acceleration of Position	
- CartAccel (X,Y,Z)	Array (in metres)
- CartAccelAccuracy (X,Y,Z)	Array of CartAccelAccuracys
- SpherAccel (r,φ,θ)	Array (in metres and degrees)
- SpherAccel (r,φ,θ)	Array (in metres and degrees)
DescrMetadata	Descriptive Metadata

8.1.81.5 Conformance Testing

A Data instance Conforms with MPAI-OSD V1.3 Position (OSD-OPS) if:

1. The Data validates against the Position 's JSON Schema.
2. All Data in the Position 's JSON Schema have the specifies type.

8.1.82 Colour Object

8.1.82.1 Definition

The representation of colour information with added optional Qualifier.

8.1.82.2 Functional Requirements

Colour is expressed as a set of number depending on the Colour Qualifier.

8.1.82.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/ColourObject.json>

8.1.82.4 Semantics

Label	Description
Header	Colour Object Header
– Standard-ColourObject	The characters “MMM-OSD-V”
– Version	Major version – 1 or 2 characters

– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	Identifier of M-Instance.
ColourData	The Data of the Colour.
ColourQualifier	The Qualifier of the Colour Data.
DescrMetadata	Descriptive Metadata

8.1.83 Right Parallelepiped

8.1.83.1 Definition

A Data Type representing a rectangle (2D or right parallelepiped containing a 2D or 3D Visual Object, respectively.

8.1.83.2 Functional Requirements

The rectangle or right parallelepiped is defined, respectively, by

1. Rectangle (2D): 3 vertices not on a straight line.
2. Right Parallelepiped (3D): 4 vertices not on a plane.

8.1.83.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/RightParallelepiped.json>

8.1.83.4 Semantics

Label	Description
Header	Right Parallelepiped Header
- Standard-RightParallelepiped	The characters “OSD-RPP-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
RightParallelepipedID	Identifier of Right Parallelepiped.
Dimensions	One of 2D, 3D
RightParallelepipedData	Data about Right Parallelepiped.
- Precision	Simple or double precision
- CartPositions	The Positions of the selected vertices of the rectangle/parallelepiped.
DescrMetadata	Descriptive Metadata

8.1.83.5 Conformance Testing

A Data instance Conforms with MPAI-OSD Right Parallelepiped (OSD-RPP) if:

1. The Data validates against the Right Parallelepiped's JSON Schema.
2. All Data in the Right Parallelepiped's JSON Schema have the specified type.

8.1.84 Coordinates

8.1.84.1 Definition

A set of numbers used to indicate the position of a point in a space.

8.1.84.2 Functional Requirements

All points in the space shall have a set of numbers representing them.

The coordinate systems supported so far are:

1. Cartesian
2. Spherical
3. Geodesic
4. Cylindrical
5. Toroidal

8.1.84.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Coordinates.json>

8.1.84.4 Semantics

Label	Description
Header	Time Header
- Standard-Object	The characters "OSD-CRD-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
CoordinatesID	Identifier of Coordinates.
CoordinateTypes	One of Cartesian, Spherical, Geodesic, Cylindrical, Toroidal.
CoordinateData	Three numbers
DescrMetadata	Descriptive Metadata

8.1.84.5 Conformance Testing

A Data instance Conforms with Coordinates (OSD-CRD) if all the Data:

1. Have the specified type.
2. Validate against the Coordinates' JSON Schema.

8.1.85 Selector

8.1.85.1 Definition

A Data Type used to indicate specific operating values of an AIW or AIM.

8.1.85.2 Functional Requirements

Selector informs an AIW/AIM that a communicating Entity uses/requests to use:

1. Specific media – Text, Speech, Visual, or Gesture – as input or output.
2. Specific Language – as input or output.
3. Media or their Descriptors.
4. View an Avatar or a Scene

8.1.85.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Selector.json>

8.1.85.4 Semantics

Label	Description
Header	Selector Header
- Standard-Selector	The characters “OSD-SEL-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
InputMedia	One or more of Text, Speech, Visual, or Gesture.
OutputMedia	One or more of Text, Speech, Visual, or Gesture.
InputLanguage	One of a list of languages.
OutputLanguage	One of a list of languages.
MediaOrDescriptors	One of Text, Speech, Face, Body for MMC-TST
SpeechDescriptors	One of No, Yes for MMC-PSE
View	One of Avatar or Scene
DescrMetadata	Descriptive Metadata

8.1.85.5 Conformance Testing

A Data instance Conforms with Selector (OSD-SEL) if:

1. The Data validates against the Selector’s JSON Schema.
2. All Data in the Selector’s JSON Schema have the specified types.

8.1.86 Device Scene Geometry

8.1.86.1 Definition

A Data Type including the Point of View of the Devices capturing media information, possibly grouped according to the characteristics of Devices.

8.1.86.2 Functional Requirements

A Device Scene Geometry may include:

1. The ID of a Virtual Space (M-Instance) the Device Geometry may refer to.
2. Groups of Devices sharing the same characteristics including:
 1. Characteristic Name.

2. Data about the individual Devices of the Group, including:
 1. Device ID.
 2. Point of View.

8.1.86.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/DeviceSceneGeometry.json>

8.1.86.4 Semantics

Label	Description
Header	Device Scene Geometry Header
– Standard-DeviceSceneGeometry	The characters “OSD-DSG-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
CaptureTime	Duration of media information capture.
DeviceEnsembleList[]	Identifier of the Audio Object.
- CharacteristicName	A Name in an Enumeration.
- DeviceEnsemble[]	Data of the Devices in the group.
- DeviceID	Device ID.
- PointOf View	Point of View of the Device.
DescrMetadata	Descriptive Metadata

8.1.86.5 Conformance Testing

A Data instance Conforms with Data Scene Geometry (OSD-DSG) if:

1. The Data validates against the Data Scene Geometry’s JSON Schema.
2. All Data in the Data Scene Geometry’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.87 Simple Location

8.1.87.1 Definition

A Data Type providing a simple format to represent a real and virtual region with Space-Time attributes.

8.1.87.2 Functional Requirements

Simple location represents a space as a combination of Rights Parallelepipeds.

8.1.87.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SimpleLocation.json>

8.1.87.4 Semantics

Label	Description
Header	Simple Location Header
- Standard-SimpleLocation	The characters “SLC-SLC-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of a virtual space including the location represented by Simple Location.
SimpleLocationID	Identifier of the Simple Location.
SimpleLocationData[]	Data included in Simple Location.
- SimpleLocationSpaceTime	SpaceTime of a right parallelepiped.
- RightParallelepiped	Right Parallelepiped Data.
DescrMetadata	Descriptive Metadata

8.1.87.5 Conformance Testing

A location instance Conforms with Simple Location (OSD-SLC) if:

1. The Data validates against the Simple Location’s JSON Schema.
2. All Data in the Simple Location’s JSON Schema have the specified types.

8.1.88 Instance Identifier

8.1.88.1 Definition

A Data Type associating a string (Identifier) with an element of a set of entities – Speech, Objects, Visual Objects, User IDs etc. – belonging to some levels in a hierarchical classification (taxonomy).

8.1.88.2 Functional Requirements

Instance Identifier includes:

1. ID of Virtual Space (M-Instance)
2. Instance Label
3. Confidence level of the association between Instance Label and Instance.
4. Taxonomy
5. Confidence level of the association between Taxonomy and the Instance.

8.1.88.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/InstanceIdentifier.json>

8.1.88.4 Semantics

Label	Description
Header	Instance Identifier Header
– Standard-InstanceIdentifier	The characters “OSD-IID-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance
InstanceID	Identifier of Instance.
InstanceSpaceTime	Data about Space-Time
InstanceIdentifierData	Data set of Instance Identifier.
InstanceLabel	Instance identified by Instance Identifier.
LabelConfidenceLevel	Confidence of Instance Label and Instance association.
TaxonomyLabel	Taxonomy Instance Identifier belongs to.
TaxonomyConfidenceLevel	Confidence of Taxonomy Label .
TaxonomyDataLength	Number of Bytes
TaxonomyDataURI	URI of Taxonomy.
DescrMetadata	Descriptive Metadata

8.1.88.5 Conformance Testing

A Data instance Conforms with Instance Identifier (OSD-IID) if:

1. The Data validates against the Instance Identifier’s JSON Schema.
2. All Data in the Instance Identifier’s JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers if present.

8.1.89 Space-Time

8.1.89.1 Definition

Data Type representing the Spatial Attitude and Time information.

8.1.89.2 Functional Requirements

Space-Time includes Spatial Attitude and Time.

8.1.89.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpaceTime.json>

8.1.89.4 Semantics

Label	Description
Header	Space-Time Header
- Standard-Object	The characters “OSD-SPT-V”

- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstance	Identifier of Virtual Space.
SpaceTimeID	Identifier of Space-Time.
Space	Spatial Attitudes at T ₀ and T ₁
Time	Time interval between T ₀ and T ₁
DescrMetadata	Descriptive Metadata

8.1.89.5 Conformance Testing

A Data instance Conforms with Space-Time (OSD-SPT) if:

1. The Data validates against the Space-Time’s JSON Schema.
2. All Data in the Space-Time’s JSON Schema
 1. Have the specified type.
 2. Validate against their JSON Schemas.
 3. Conform with their Data Qualifiers if present.

8.1.90 Light Source

8.1.90.1 Definition

A Data Type representing a light source and its characteristics.

8.1.90.2 Functional Requirements

A Light Source may include:

1. The ID of the M-Instance where it is located.
2. The Light Source's
 1. ID
 2. Time-Space information
 3. Type (shape)
 4. Colour information
 5. Intensity
 6. Annotations
 7. Rights

8.1.90.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/LightSource.json>

8.1.90.4 Semantics

Label	Description
Header	Light Source Header
– Standard-LightSource	The characters “OSD-LTS-V”
– Version	Major version – 1 or 2 characters

– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
MInstanceID	Identifier of M-Instance.
LightSourceID	Identifier of the Light Source.
LightSourceSpaceTime	Space-Time information of the Light Source.
LightSourceType	One of the following:
- Point	Point-like source
- Directional	- Azimuth - Elevation
- SkyLight	- Azimuth - Elevation
- Spotlight	- Azimuth - Elevation - Angle - Blend ($0 \leq 1$) representing the fall out rate of the light at the border of the basis of the cone.
- Area	Characterised by Shape
ColourInformation	Selected from Colour Space.
Intensity	The light source's intensity measured in Watt.
LightSourceAnnotation	Annotations of Visual Data
– Annotation	ID of Annotation
– AnnotationSpaceTime	Where/when Annotation is attached.
– RightsID	Rights that may be exercised on the Annotation
LightSourceRights	Rights that may be exercised on the Light Source.
DescrMetadata	Descriptive Metadata

8.1.90.5 Conformance Testing

A Data instance Conforms with Light Source if:

1. The Data validates against the Light Source's JSON Schema.
2. All Data in the LightSource's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.91 Spatial Attitude

8.1.91.1 Definition

An Item representing the Position and Orientation of an Object, and their velocities and accelerations.

8.1.91.2 Functional Requirements

The Spatial Attitude is defined as the combination of Position and orientation, the Functional Requirements are defined by Position and Orientation.

8.1.91.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/SpatialAttitude.json>

8.1.91.4 Semantics

Table 1 provides the semantics of the components of the Spatial Attitude.

Table 1 – Semantics of the Spatial Attitude

Label	Description
Header	Spatial Attitude Header
- Standard-SpatialAttitude	The characters “OSD-OSA-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
MInstanceID	ID of Virtual Space Object refers to.
ObjectSpatialAttitudeID	Identifier of Object Spatial Attitude.
General	Set of general data
- CoordinateType	One of Cartesian, Spherical, Geodesic, Toroidal.
- ObjectType	One of Digital Human, Generic.
- MediaType	One of Speech, Audio, Visual, Audio-Visual, Haptic, Smell, RADAR, LiDAR, Ultrasound.
Position	As specified by Position
Orientation	As specified by Orientation
DescrMetadata	Descriptive Metadata

8.1.91.5 Conformance Testing

A Data instance Conforms with V1.2 Spatial Attitude (OSD-OSA) if:

1. The Data validates against the Spatial Attitude’s JSON Schema.
2. All Data in the Spatial Attitude ’s JSON Schema have the specified type.

8.1.92 Object Audio Characteristics

8.1.92.1 Definition

Data describing the characteristics of an object as pertains Audio.

8.1.92.2 Functional Requirements

Object Audio Characteristics includes

1. Reflectivity
 1. Early Reflections Time (s)
 2. Late Reflections Time (s)
2. Reverberation
 1. RT60 (frequency dependent)

2. RT30 (frequency dependent)
3. R20 (frequency dependent)
4. EDT (Time taken by audio from full scale to -10 dB)
3. Diffusion: Schroeder Diffusion Coefficient (number)
4. Absorption: Absorption Coefficient (number 0 to 1)

8.1.92.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/ObjectAudioCharacteristics.json>

8.1.92.4 Semantics

Label	Description
Header	Object Audio Characteristics Header
– Standard-ObjectAudioCharacteristics	The characters “OSD-OAC-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	Identifier of M-Instance.
ObjectAudioCharacteristicsID	Identifier of Texture.
ObjectAudioCharacteristics	Data of Object Audio Characteristics
- Reflectivity	The reflection of sound in the same medium after hitting a surface.
- EarlyReflectionTime	Measured in seconds
- LateReflectionTime	Measured in seconds
- Reverberation	The persistence of sound after it has been produced
- RT60	Table with frequency (Hz) and dB
- RT30	Table with frequency (Hz) and dB
- RT20	Table with frequency (Hz) and dB
- EDT	Time taken by audio from full scale to -10 dB (ms).
- Diffusion	The spreading of sound waves throughout a space, measured by the Schroeder Diffusion Coefficient (number).
- Absorption	The loss of energy affected by sound waves that have reached an absorbent material measured by the Absorption Coefficient ($1 \geq \text{number} \geq 0$).
ObjectAudioCharacteristicsRights	Rights to Act on the Object Audio Characteristics.
DescrMetadata	Descriptive Metadata

8.1.92.5 Conformance Testing

A Data instance Conforms with Object Audio Characteristics (OSD-OAC) if:

1. The Data validates against the Object Audio Characteristics's JSON Schema.
2. All Data in the Texture's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas

3. Conform with their Data Qualifiers, if present.

8.1.93 Texture Map

8.1.93.1 Definition

Texture map (OSD-TEM) is a 2D array containing Data used to modify the parameters of an object's surface points.

8.1.93.2 Functional Requirements

Different types of Texture Map (TM) can be used to represent the following parameters of an object's surface points.

Diffuse map	TM used to set the colour of the model and add shapes or details.
Specular map	TM used to set the degree of shininess or reflectivity of a surface.
Colour map	TM used to add colour, texture, and shading to a surface.
Bump maps	TM used to add small details that appear three-dimensional
Normal map	TM used to provide surface normal information in an RGB image
Displacement map	TM used to provide displacement information to the actual geometric position of points over the textured surface
Ambient occlusion map	TM used to provide lighting data used to calculate the exposure of each point in a scene is to ambient lighting.
Reflection map	TM used to define how and where a surface reflects the surrounding environment.
Roughness map	TM used to define how smooth or rough a surface is, affecting how light is scattered or reflected.

8.1.93.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/TextureMap.json>

8.1.93.4 Semantics

Label	Description
Header	Texture Header
– Standard-Texture	The characters “OSD-TEM-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	Identifier of M-Instance.
TextureID	Identifier of Texture.
TextureMap	Set of Visual Characteristics
- BumpMap	Greyscale value for each Texture pixel.
- DisplacementMap	Greyscale value for each Texture pixel.

- AmbientOcclusionMap	Greyscale value for each Texture pixel.
- RoughnessMap	Greyscale value for each Texture pixel.
- DiffuseMap	RGB values for each Texture pixel.
- ColourMap	RGB values for each Texture pixel.
- ReflectionMap	RGB values for each Texture pixel.
- SpecularMap	Intensity value or RGB values for each Texture pixel.
- NormalMap	The new normal vector for each Texture pixel.
TextureRights	Rights to Act on the Texture.
DescrMetadata	Descriptive Metadata

8.1.93.5 Conformance Testing

A Data instance Conforms with Texture (OSD-TEM) if:

1. The Data validates against the Texture Map's JSON Schema.
2. All Data in the Texture Map's JSON Schema
 1. Have the specified type
 2. Validate against their JSON Schemas
 3. Conform with their Data Qualifiers, if present.

8.1.94 Orientation

8.1.94.1 Definition

An Item representing an Object's orientation, orientation velocity, and orientation acceleration.

8.1.94.2 Functional Requirements

- The Orientation of an Object is that of the principal axis of an Object.
- The following media types are supported: Speech, Audio; Visual; 3D Model; Audio-Visual; Haptic; Smell; RADAR; LiDAR; Ultrasound.
- Accuracy is the estimated absolute difference between the measures of each of Orientation, OrientVelocity, and OrientAccel and their true values.

8.1.94.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Orientation.json>

8.1.94.4 Semantics

Table 1 provides the semantics of the components of Orientation. It should be noted that the Euler angles are indicated by (α, β, γ) .

Table 1 – Semantics of Orientation

Label	Description
Header	Orientation Header
- Standard-Orientation	The characters "OSD-OOR-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."

- Subversion	Minor version – 1 or 2 characters
MInstance	ID id Virtual space Orientation refers tu
ObjectOrientationID	Identifier of Object Orientation.
General	Set of general data
- ObjectType	One of Digital Human, Generic.
- MediaType	One of Speech, Audio, Visual, Audio-Visual, Haptic, Smell, RADAR, LiDAR, Ultrasound.
Orientation	
- Orient (α, β, γ)	Array (in degrees)
- OrientAccuracy	Array of (α, β, γ) Accuracy
Velocity of Orientation	
- OrientVelocity (α, β, γ)	Array (in degrees/s)
- OrientVelocityAccuracy	Array of (α, β, γ) Velocity Accuracy
Acceleration of Orientation	
- OrientAccel (α, β, γ)	Array (in degrees/s ²)
- OrientAcceAccuracy	Array of (α, β, γ) Acceleration Accuracy
DescrMetadata	Descriptive Metadata

8.1.94.5 Conformance Testing

A Data instance Conforms with Orientation (OSD-OOR) if:

1. The Data validates against the Orientation's JSON Schema.
2. All Data in the Orientation's JSON Schema have the specified types.

8.1.95 Trajectory

8.1.95.1 Definition

The sequence of start and end Spatial Attitudes SA ($SA_1, SA_2, \dots, SA_i$) and corresponding Times t (t_1, t_2, t_j) expected and actual of a series of segments.

8.1.95.2 Functional Requirements

A Trajectory is composed of Segments. Each Segment is described by the expected and actual start and end Spatial Attitudes and Times.

8.1.95.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Trajectory.json>

8.1.95.4 Semantics

Label	Description
Header	Trajectory Header
- Standard-Trajectory	The characters "CAV-TRJ-V"
- Version	Major version – 1 or 2 Bytes

- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 Bytes
TrajectoryID	Identifier of Trajectory.
TrajectoryData[]	Data in the Trajectory
- SpaceTime	Expected and/or actual Spatial Attitude and Time of a Trajectory segment.
DescrMetadata	Descriptive Metadata

8.1.95.5 Conformance Testing

A Data instance Conforms with Trajectory (OSD-TRJ) if:

1. The Data validates against the Trajectory ’s JSON Schema.
2. All Data in the Trajectory ’s JSON Schema
 1. Have the specified type
 2. Validate against JSON Schemas.
 3. Conform with their Data Qualifiers if present.

8.1.96 Path

8.1.96.1 Definition

Path is a sequence of Points of View.

8.1.96.2 Functional Requirements

Path is defined against an M-Instance or an OfflineMap.

8.1.96.3 Syntax

<https://schemas.mpai.community/OSD/V1.4/data/Path.json>

8.1.96.4 Semantics

Label	Description
Header	Path Header
- Standard-Path	The characters “OSD-PAT-V”
- Version	Major version – 1 or 2 Bytes
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 Bytes
PathID	Identifier of Path.
MInstanceID	ID of the Virtual Space where Path is defined.
OfflineMapID	ID of the referenced OfflineMap.
PathData[]	Path Dataset.
- PointOfView	Planned Individual Point of View in the Path.
DescrMetadata	Descriptive Metadata

8.1.96.5 Conformance Testing

A Data instance Conforms with Path MPAI-OSD (OSD-PAT) if the Data

1. Have the specified type
2. Validate against the Path's JSON Schema.

8.2 Conformance testing

A Data instance of a Data Type Conforms with MPAI-OSD V1.4 if the JSON Data validate against the relevant MPAI-OSD V1.4 JSON Schema and if the Data Conforms with the relevant Data Qualifier, if one is present. MPAI-OSD V1.4 does not provide method for testing the Conformance of the Semantics of the Data instance to the MPAI-OSD V1.3 specification.

Conformance testing can be performed by a human using a JSON Validator to verify the Conformance of the syntax of JSON Data to the relevant JSON Schema; and, if the Data has a Qualifier, to verify that the syntax of the Data conforms with the relevant values in the Data Qualifier. Alternatively, Conformance testing can be performed by software implementing the steps above.

8.3 Performance Assessment

Performance Assessment provides methods of assessing the performance of an Data instance. Performance may have various connotations, such as:

1. *Quality*: Performance Assessment measures the quality of the Data instance using a metric that depends on the nature of the Data, e.g., the accuracy of identification of Visual Sources in a Visual Scene Geometry.
2. *Bias*: Performance Assessment measures the disparity of treatment applied to the Data instance using a metric that depends on a bias related to certain attributes of the Data instance. For example, a systematic misidentification of an object.
3. *Legal compliance*: Performance Assessment uses an appropriate metric to measure how well the Data instance complies with with a certain legal standard.

Table 1 provides the Data Types specified by MPAI-OSD V1.2. MPAI-OSD AIWs and AIMs also utilise Data Types specified by other MPAI Technical Specifications. The linked list of all MPAI Data Types is [available](#).