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

MPAI Technical Specification

Object and Scene Description

V1.0

WARNING

Use of the technologies described in this Technical Report 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 Report.

Readers are invited to review Annex 3 - Notices and Disclaimers.

Technical Specification
Object and Scene Description V1.0

1 Introduction

Technical Specification: Object and Scene Description (MPAI-OSD) – in the following also called MPAI-OSD – has been developed by MPAI – Moving Picture, Audio, and Data Coding by Artificial Intelligence, the international, unaffiliated, non-profit organisation developing standards for Artificial Intelligence (AI)-based data coding with clear Intellectual Property Rights licensing frameworks in compliance with the rigorous MPAI Process [9,10] in pursuit of the following policies:

1. Be friendly to the AI context but, to the extent possible, agnostic to the technology – AI or Data Processing – used in an implementation.
2. Be attractive to different industries, end users, and regulators.
3. Address three levels of standardisation all exposing standard interfaces with an aggregation level decided by the implementer:
   3.1. data types.
   3.2. Components called AI Modules (AIM).
   3.3. Configurations of AIMs called AI Workflows (AIW).
4. Specify the data exchanged by AIMs with as clear a semantic as possible.

As manager of the MPAI Ecosystem specified by Governance of MPAI Ecosystem (MPAI-GME) [1] and ensures that a user can:

1. Operate a reference implementation of the Technical Specification, by providing a Reference Software Specification with annexed software.
2. Test the conformance of an implementation with the Technical Specification, by providing the Conformance Testing Specification.
3. Assess the performance of an implementation of a Technical Specification, by providing the Performance Assessment Specification.
4. Get conforming implementations possibly with a performance assessment report from a trusted source through the MPAI Store.

Technical Specification: AI Framework (MPAI-AIF) [2] specifies a standard AI Framework (AIF) that enables dynamic configuration, initialisation, and control of AIWs depicted in Figure 1.

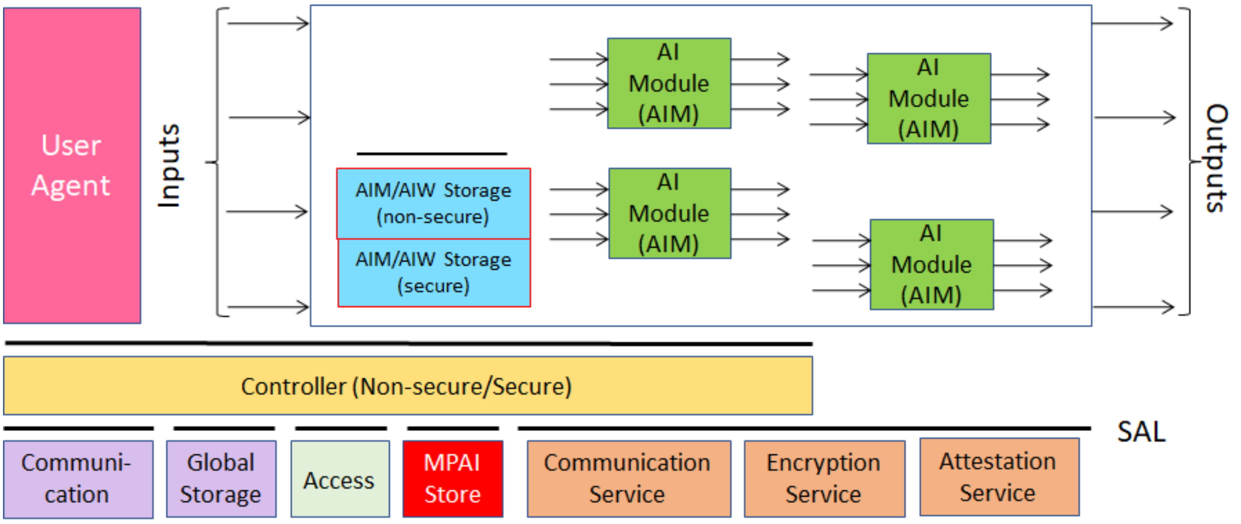

Figure 1 - The AI Framework (MPAI-AIF) V2 Reference Model

MPAI-AIF enabling the secure execution of AI Workflows (AIW) that can be constituted by AI Modules (AIM). Thus, users can have machines whose internal operation they understand to some degree, rather than machines that are just “black boxes” resulting from unknown training with unknown data and component developers can provide components with standard interfaces that can have improved performance compared to other implementations.

An AIW and its AIMs may have 3 interoperability levels:
Level 1 – Implementer-specific and satisfying the MPAI-AIF Standard.
Level 2 – Specified by an MPAI Application Standard.
Level 3 – Specified by an MPAI Application Standard and certified by a Performance Assessor.
Users are free to adopt any of the three levels.

AIM can execute data processing or Artificial Intelligence algorithms and can be implemented in hardware, software, or hybrid hardware/software. AI Module can be Composite if they include connected AI Modules.

The MPAI-MMC V2 Technical Specification can be implemented in one of the following modalities:

1. As a specific AIW implementing a Use Case, as specified in this document.
2. As a specific AIM, as specified in this document.
3. As a specific data type, as specified in this document.

However, MPAI does not mandate the choice of modality, which remains the sole decision of the implementer.

In many MPAI Technical Specifications there are data types that refer to Objects and Scenes that can be uni- and multimodal and possibly refer to locations that may be in a physical or virtual space.

MPAI values the consistent use of data types across its Technical Specifications. Therefore, MPAI-OSD has been developed to be the reference point for the consistent use of data types across MPAI standards. When consistency is not possible because different usages are consolidated, such usages are clearly identified, and individual specific usages recorded.

Currently, there are no MPAI-OSD specific Use Cases. Therefore, only the Scope, Reference Models, and I/O Data of relevant Use Cases from other Technical Specifications are reported here. The full Use Case specification can be found in the Technical Specifications owning the Use Cases. All Use Cases are assumed to be implemented according to the MPAI-AIF.

MPAI-OSD will be accompanied by the Reference Software, Conformance Testing, and Performance Assessment Specifications. Conformance Testing specifies methods enabling users to ascertain whether a data type generated by an AIM, an AIM, or an AIW conform with this Technical Specification.

2 Scope

Technical Specification: Object and Scenes Description (MPAI-OSD) specifies Data Formats to enable description and localisation of uni- and multi-modal Objects and Scenes in a Virtual Space for uniform use across MPAI Technical Specifications.

MPAI-OSD has been developed by the Context-based Audio Enhancement (MPAI-CAE), Multimodal Conversation (MPAI-MMC), and Portable Avatar Format (MPAI-PAF) Development Committees, and by the Connected Autonomous Vehicle (CAV) group of the Requirements Standing Committee.

3 Definitions

Terms beginning with a capital letter have the meaning defined in Table 1. Terms beginning with a small letter have the meaning commonly defined for the context in which they are used. For instance, Table 1 defines *Object* and *Scene* but does not define *object* and *scene*.

A dash "-" preceding a Term in Table 1 indicates the following readings according to the font:

1. Normal font: the Term in the table without a dash and preceding the one with a dash should be read before that Term. For example, "Avatar" and "- Model" will yield "Avatar Model."
2. *Italic* font: the Term in the table without a dash and preceding the one with a dash should be read after that Term. For example, "Avatar" and "- Portable" will yield "Portable Avatar."

Table 1 - General MPAI-HMC terms

Term	Definition
Audio	Digital representation of an analogue audio signal sampled at a frequency between 8-192 kHz with a number of bits/sample between 8 and 32, and non-linear and linear quantisation. Data with characteristics of Audio may be synthetically produced.
Avatar	An Object rendered to represent a Human of a Machine in a virtual space.
- Model	An inanimate Avatar exposing animation interfaces.
- *Portable*	A Data Type including Avatar ID, Time, Audio-VisualScene Descriptors, Spatial Attitude, Avatar Model, Body Descriptors, Face Descriptors, Language Preference, Speech Coding, Speech Data, Text, and Personal Status [5].
Centre Point	The point of an Object selected to have coordinates (0,0,0).
Context	Additional information about a communication emitted by an Entity, such as language, culture etc..
Data	Information in digital form.
- Format	The standard digital representation of Data.

- Type	An instance of Data with a specific Data Format.
Descriptor	The Digital Representation of a feature of an Object.
- *Body*	A Data Type including the digital representation of the features of the body of a real or digital human.
- *Face*	A Data Type including the digital representation of a feature of the face of a real or digital human.
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.
- *Digital*	A Digitised or a Virtual Human.
- *Digitised*	An Object that has the appearance of a specific human when rendered.
- *Virtual*	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 – Objects, Digital Humans etc. – belonging to some levels in a hierarchical classification (taxonomy).
- *Audio*	The instance of an Audio Object.
- *Visual*	The instance of a Visual Object.
Object	A data structure that can be rendered to cause an Experience.
- *Audio*	An Object described by Audio Descriptors.
- *Audio-Visual*	An Object described by Audio-Visual Descriptors.
- *Body*	A digital representation of the body of a Human or a Machine.
- *Descriptor*	The digital representation of the feature of an Object.
- *Digital*	A Digitised or a Virtual Object.
- *Digitised*	The digital representation of a real object.
- *Face*	The digital representation of the face of a Human or a Machine.
- *Speech*	An Object described by Speech Descriptors.
- *Text*	A string of Text.
- *Virtual*	An Object not representing an object in the real environment.
- *Visual*	An Object described by Visual Descriptors.
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 a Virtual Space as a human-perceptible entity.
Scene	A composition of Objects located according to a Scene Geometry.
- *Audio*	A Scene composed of Audio Objects.
- *Digital*	A digitised scene or a Virtual Scene
- *Audio-Visual*	A Scene composed of Audio Objects, Visual Objects and co-located Audio-Visual Objects.
- *Visual*	A Scene composed of Visual Objects.
Scene Descriptors	The digital representation of a feature of a scene.
- *Audio*	A Data Type including the digital representation of the audio features of a digital scene.
- *Audio-Visual*	A Data Type combining the Audio or Visual Scene Descriptors.
- *Visual*	A Data Type including the digital representation of the visual features of a digital scene.
Scene Geometry	The digital representation of the Object arrangement of a Scene.

- *Audio*	A Data Type describing the Spatial arrangement of the Visual Objects of a Scene.
- *Audio-Visual*	A Data Type describing the Spatial arrangement of the Audio, Visual, and Audio-Visual Objects of a Scene.
- *Visual*	A Data Type describing the Spatial arrangement of the Visual Objects of a Scene.
Attitude	
- *Spatial*	Position and Orientation and their velocities and accelerations of a Human and Visual Object in a Virtual Environment.
Virtual Space	A space generated and maintained by a computing platform that can be rendered.
Speech	Digital representation of analogue speech sampled at a frequency between 8 kHz and 96 kHz with a number of bits/sample of 8, 16 or 24, and non-linear and linear quantisation or compressed. Data with characteristics of Speech may be synthetically produced.

4 References

4.1 Normative Reference

1. MPAI; Technical Specification: Governance of the MPAI ecosystem (MPAI-GME), V1.1; https://mpai.community/standards/mpai-gme/
2. MPAI; Technical Specification: AI Framework (MPAI-AIF) V2.0; https://mpai.community/standards/mpai-aif/
3. MPAI; Technical Specification: Context-based Audio Enhancement (MPAI-CAE) V2.1; https://mpai.community/standards/mpai-cae/
4. Technical Specification: Connected Autonomous Vehicles (MPAI-CAV) – Architecture V1.0; https://mpai.community/standards/mpai-cav/
5. MPAI; Technical Specification: Multimodal Conversation (MPAI-MMC) V2.1; https://mpai.community/standards/mpai-mmc/
6. Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Architecture V1.1; https://mpai.community/standards/mpai-mmm/
7. MPAI; Technical Specification: Portable Avatar Format (MPAI-PAF) V1.1; https://mpai.community/standards/mpai-paf/
8. Khronos; Graphics Language Transmission Format (glTF); October 2021; https://registry.khronos.org/glTF/specs/2.0/glTF-2.0.html

4.2 Informative References

9. MPAI; The MPAI Statutes; N421; https://mpai.community/statutes/
10. MPAI; Patent Policy; https://mpai.community/about/the-mpai-patent-policy/
11. MPAI; Framework Licence: Object and Scene Description; https://mpai.community/wp-content/uploads/2023/08/N1361-Framework-Licence-Object-and-Scene-Description-MPAI-OSD.pdf

5 Use Cases

Technical Specification: Object and Scene Description (MPAI-OSD) V1 – in the following MPAI-OSD – contributes to enabling Use Cases developed in the context of other MPAI Technical Specifications. Conversation About a Scene (MMC-CAS)

6 Composite AI Modules

Composite AIMs are AI Modules composed of multiple AI Modules. They are used in several MPAI Use Cases. Each Composite AI Module is specified by including the following elements:

1	Functions of the Composite AIM	The functions performed by the Composite AIM.
2	Reference Model of the AIW	The Topology of AIMs in the Composite AIM.
3	Input and Output Data of the AIW	Input and Output Data of the Composite AIM.
4	Functions of the AIMs	Functions performed by all AIMs of the Composite AIM.
5	Input and Output Data of the AIMs	Input and Output Data of the AIMs of Composite AIM.
6	Specification of AIMs and JSON Metadata	Links to summary specification on the web of each AIM used in the standard and the corresponding JSON Metadata as specified by MPAI-AIF [7].

6.1 Audio-Visual Scene Description (OSD-AVD)

6.1.1 Functions of Audio-Visual Object Description

The Audio-Visual Scene Description (OSD-AVD) Composite AIM receives two independently developed Audio Scene Descriptors and Visual Scene Descriptors in the same Virtual Space and produces Audio-Visual Scene Descriptors whose co-located Audio Objects and Visual Objects have the same or related identifiers.

6.1.2 Reference Architecture of Audio-Visual Object Description

Figure 2 gives the Reference Model of Audio-Visual Scene Description.

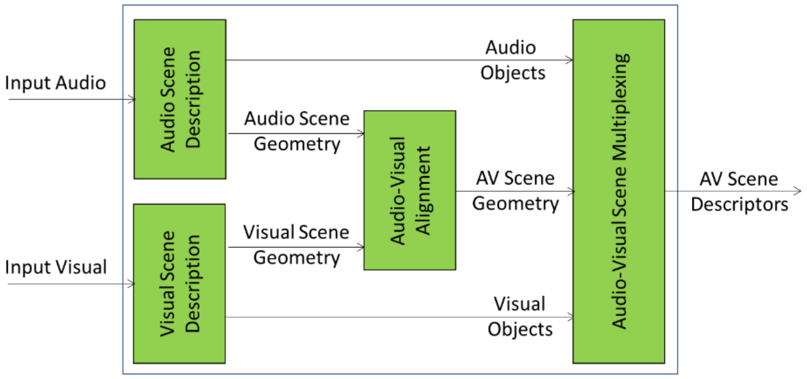

Figure 2 - Reference Model of Audio-Visual Scene Description

6.1.3 Input/output data of Audio-Visual Object Description

Table 2 gives the input/output data of Audio-Visual Scene Description.

Table 2 – I/O data of Audio-Visual Scene Description

Input data	From	Comment
Input Audio	A real environment	The Input Audio and Input Visual are from the

		same scene
Input Visual	A real environment	The Input Audio and Input Visual are from the same scene
Output data	**To**	**Comments**
Audio-Visual Scene Descriptors	Downstream AIM	The co-located Audio and Visual Objects in the Scene convey the same or related identifiers.

6.1.4 Functions of Audio-Visual Object Description AI Modules

Table 3 gives functions of the AIMs.

Table 3 - AI Modules Audio-Visual Scene Description

AIM	Modules
Audio Scene Description	Produces the Audio Scene Descriptors (Geometry+Objects).
Visual Scene Description	Produces the Visual Scene Descriptors (Geometry+Objects).
Audio-Visual Alignment	Identifies co-located Audio and Visual Objects. Assigns the same or related Identifiers to the co-located Audio and Visual Objects. Updates the Audio-Visual Scene Geometry.
Audio-Visual Scene Multiplexing	Multiplexes the new Audio-Visual Scene Geometry and the Audio and Visual Objects.

6.1.5 I/O Data of Audio-Visual Object Description AI Modules

Table 4 gives the list of the AIMs with their functions.

Table 4 - AI Modules of Audio-Visual Scene Description

AIM	Receives	Produces
Audio Scene Description	Input Audio	Audio Objects Audio Scene Geometry
Visual Scene Description	Input Visual	Visual Objects Visual Scene Geometry
Audio-Visual Alignment	Audio Scene Geometry Visual Scene Geometry	Audio-Visual Scene Geometry
Audio-Visual Scene Multiplexing	Audio Objects Visual Objects Audio-Visual Scene Geometry	Audio-Visual Scene Descriptors

6.1.6 Specification of Audio-Visual Object Description AIMs and JSON Metadata

Table 5 – AIM and JSON Metadata

- OSD-AVS	Audio-Visual Scene Description	X
- CAE-ASD	Audio Scene Description	X
- CAE-AAT	Audio Analysis Transform	X
- CAE-ASL	Audio Source Localisation	X
- CAE-ASE	Audio Separation and Enhancement	X
- CAE-AST	Audio Synthesis Transform	X
- CAE-AMX	Audio Descriptor Multiplexing	X
- OSD-VSD	Visual Scene Description	X

- OSD-AVA	Audio-Visual Alignment	X
- OSD-AMX	Audio-Visual Scene Multiplexing	X

6.2 Visual Object Identification (OSD-VOI)

6.2.1 Functions of Visual Object Identification

The purpose of the Visual Object Identification (OSD-VOI) AIM is to provide the Identifier of a Visual Object in an Environment containing a plurality of Objects one of which a human indicates by pointing at it with a finger.

6.2.2 Reference Architecture of Visual Object Identification

Figure 3 depicts the AIM implementing the Visual Object Identification AIM.

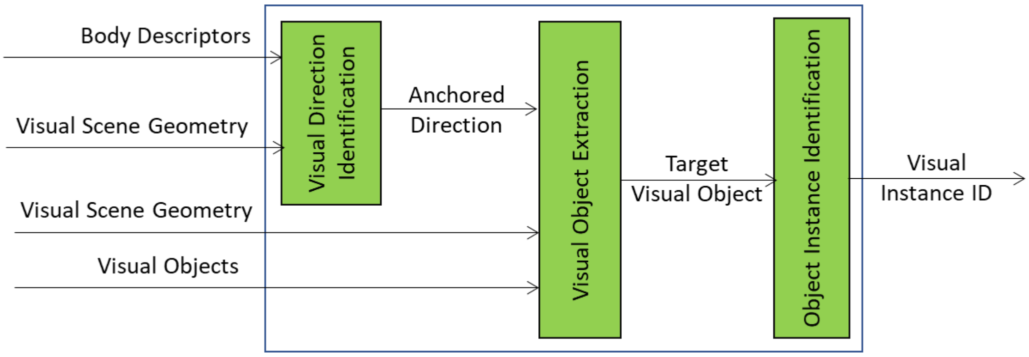

Figure 3 – Reference Model of the Visual Object Identification AIM

The workflow of Visual Spatial Object Identification unfolds as follows:

1. Direction Identification provides the (ϕ,θ) angles obtained by analysing the finger of the human.
2. Object Extraction uses the Visual Scene Geometry and the Direction to find the Object intersected by the line identified by (ϕ,θ) passing through the finger. It is assumed that one and only one Object is found.
3. Object Instance Identification provides the ID of the Object Instance.

6.2.3 Input/output data of Visual Object Identification

Table 6 gives the input/output data of Visual Object Identification.

Table 6 – I/O data of Visual Object Identification

Input data	**From**	**Comment**
Body Descriptors	Visual Scene Description	There is a human pointing to an object
Visual Objects	Visual Scene Description	There are many scene objects
Visual Scene Geometry	Visual Scene Description	Provides description of the scene
Output data	**To**	**Comments**
Visual Object Instance ID	Human or another AIM	Human points to one object only

6.2.4 Functions of AI Modules of Visual Object Identification

Table 7 gives the functions of the AIMs.

Table 7 – Functions of Visual Object Identification AI Modules

AIM	Modules
Visual Direction Identification	Produces Anchored Direction with the following features: 1. The Point belongs to the forefinger of the Entity. 2. Azimuth and Elevation of the line enable it to cross the Object in-tended by the Entity.
Visual Object Extraction	Singles out the Visual Object indicated by the Entity.
Visual Instance Identification	Produces an Instance ID identifying an element of a set of Visual Objects belonging to a level in a taxonomy.

6.2.5 I/O Data of AI Modules of Visual Object Identification

Table 8 gives the list of the AIMs with their functions.

Table 8 - AI Modules of Visual Object Identification

AIM	Receives	Produces
Visual Direction Identification	Visual Scene Geometry Body Descriptors	Anchored Direction
Visual Object Extraction	Visual Scene Geometry Anchored Direction	Target Visual Object
Visual Instance Identification	Target Visual Object	Instance Identifier

6.2.6 Specification of Visual Object Identification AIMs and JSON Metadata

Table 9 – AIM and JSON Metadata

AIM	Name	JSON
OSD-VOI	Visual Object Identification	X
- OSD-VDI	Visual Direction Identification	X
- OSD-VOE	Visual Object Extraction	X
- OSD-VII	Visual Instance Identification	X

7 Data Formats

Table 10 provides the list of Data Formats target of the Call for Technologies.

Table 10 – Data formats

Name of Data Format	Subsection	Use Case
Coordinates, Angles, and Objects	7.1	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
Spatial Attitude	7.2	ARA-ABV
		MMC-CAS
		MMC-HCI

		MPAI-CAV
		MPAI-MMM
Audio Scene Geometry	7.3	ARA-ABV
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
Audio Scene Descriptors	7.4	ARA-ABV
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
Visual Scene Geometry	7.6	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
		MMC-HCI
Visual Scene Descriptors	7.7	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
		MMC-HCI
Audio-Visual Scene Geometry	7.9	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
		MMC-HCI
Audio-Visual Scene Descriptors	7.10	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM
		MMC-HCI

The following Sections specify of the data formats.

7.1 Coordinates, Angles, and Objects

Figure 4 depict the regular way of defining Cartesian. *Figure 5* depicts the Cartesian Coordinates applicable to a visual capture device such as camera or LiDAR placed in an Environment with the (x,y) plane perpendicular and crossing the Device’s sensors. The z axis is perpendicular to the (x,y) plane and pointing to the captured scene.

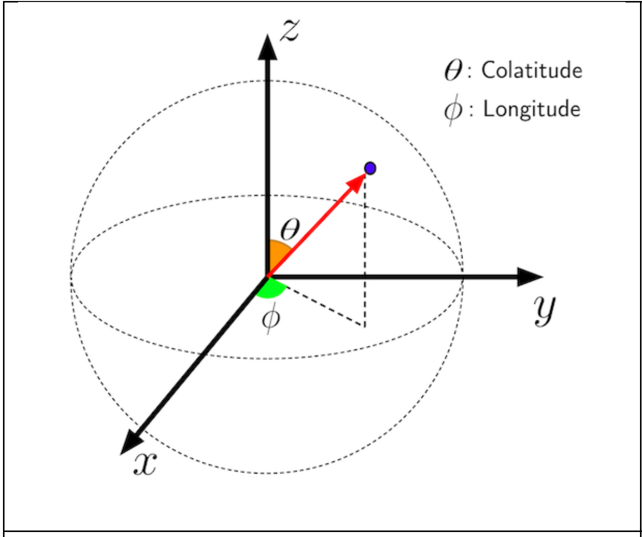	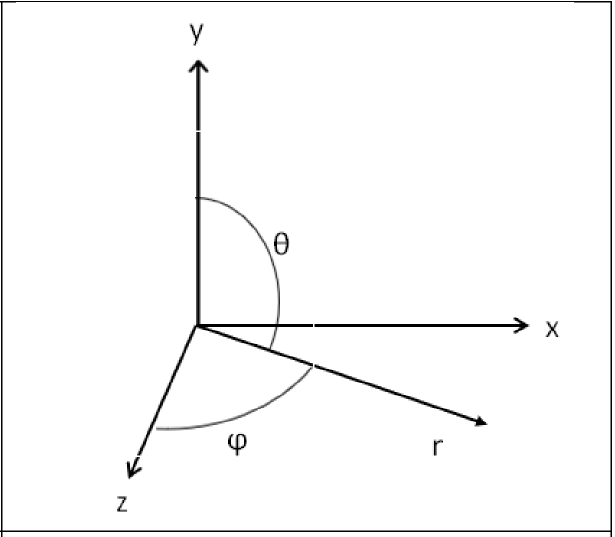
Figure 4 – Cartesian and Spherical Coordinates	*Figure 5 – Cartesian Coordinates of a capture device*

Figure 6, *Figure 7*, and *Figure 8* graphically represent how different applications associate the local (x,y,z) coordinates with the roll, pitch, and yaw rotations.

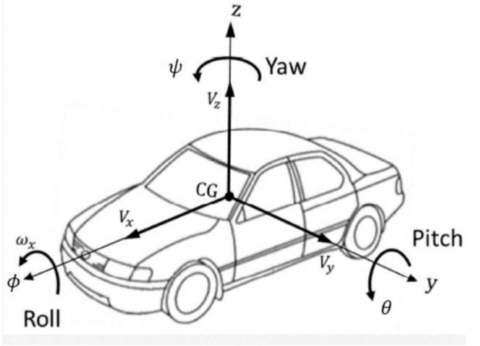	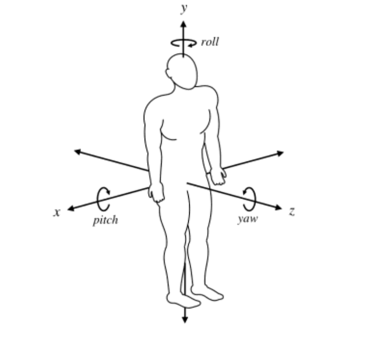	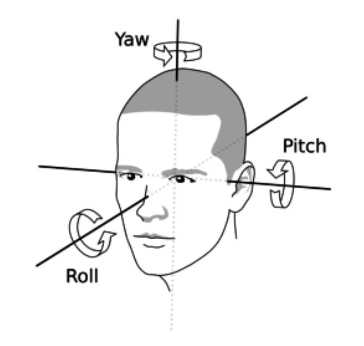
Figure 6 – Car	*Figure 7 – Human*	*Figure 8 – Head*

7.2 Spatial Attitude

Table 11 gives the components of the Spatial Attitude of an Object. The Position of an Object is that of a representative point in the Object.

7.2.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Object Spatial Attitude",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
```

```
},
"OSAID": {
  "type": "string"
},
"General": {
  "type": "object",
  "properties": {
    "CoordType": {
      "type": "number"
    },
    "ObjectType": {
      "type": "number"
    },
    "Precision": {
      "type": "number"
    },
    "MediaType": {
      "type": "number"
    }
  }
},
"CartPosition": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"SpherPosition": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"Orientation": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"CartVelocity": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"SpherVelocity": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"OrientVelocity": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
    "type": "number"
  }
},
"CartAccel": {
  "type": "array",
  "minItems": 3,
  "maxItems": 3,
  "items": {
          "type": "number"
        }
      },
      "SpherAccel": {
        "type": "array",
        "minItems": 3,
        "maxItems": 3,
        "items": {
          "type": "number"
        }
      },
      "OrientAccel": {
        "type": "array",
        "minItems": 3,
        "maxItems": 3,
        "items": {
          "type": "number"
        }
      }
    }
}
```

7.2.2 Semantics

Table 11 provides the semantics of the components of the Spatial Attitude. The following should be noted:

1. The first byte is always present.
2. Each of the other components is optional.
3. Each of Position, Velocity, and Acceleration is provided either in Cartesian (X,Y,Z) or Spherical (r,φ,θ) Coordinates.
4. The Euler angles are indicated by (α,β,γ).

Table 11 – Components of the Spatial Attitude

<table>
<tr><td><b>HEADER</b></td><td>9 Bytes</td><td></td></tr>
<tr><td>• Standard</td><td>7 Bytes</td><td>The string OSD-OSA</td></tr>
<tr><td>• Version</td><td>1 Byte</td><td>Major version</td></tr>
<tr><td>• Subversion</td><td>1 Byte</td><td>Minor version</td></tr>
<tr><td><b>OSAID</b></td><td>16 Bytes</td><td>UUID Identifier of Object Spatial Attitude.</td></tr>
<tr><td><b>General</b></td><td></td><td></td></tr>
<tr><td>• CoordType</td><td>bit 0</td><td>0: Cartesian, 1: Spherical</td></tr>
<tr><td>• ObjectType</td><td>bit 1-2</td><td>00: Digital Human
01: Generic
10 and 11: reserved</td></tr>
<tr><td>• Precision</td><td>bit 3</td><td>0: single precision; 1: double precision</td></tr>
<tr><td>• MediaType</td><td>bit 4-6</td><td>000: Audio; 001: Visual; 010: Haptic; 011: Smell; 100: RADAR; 101: LiDAR; 110: Ultrasound; 111: reserved</td></tr>
<tr><td>• Reserved</td><td>bit 6-7</td><td>reserved</td></tr>
<tr><td>• SpatialAttitudeMask</td><td>2 Bytes</td><td>3*3 matrix of booleans (by rows)
<table>
<tr><td></td><td>Position</td><td>Velocity</td><td>Acceleration</td></tr>
<tr><td>Cartesian</td><td></td><td></td><td></td></tr>
<tr><td>Spherical</td><td></td><td></td><td></td></tr>
<tr><td>Orientat.</td><td></td><td></td><td></td></tr>
</table></td></tr>
<tr><td colspan="3"><b>Position and Orientation</b></td></tr>
<tr><td>• CartPosition (X,Y,Z)</td><td>12/24 Bytes</td><td>Array (in metres)</td></tr>
<tr><td>• SpherPosition (r,φ,θ)</td><td>12/24 Bytes</td><td>Array (in metres and degrees)</td></tr>
</table>

• Orient (α,β,γ)	12/24 Bytes	Array (in degrees)
Velocity of Position and Orientation		
• CartVelocity (X,Y,Z)	12/24 Bytes	Array (in metres)
• SpherVelocity (r,φ,θ)	12/24 Bytes	Array (in metres and degrees)
• OrientVelocity (α,β,γ)	12/24 Bytes	Array (in degrees)
Acceleration of Position and Orientation		
• CartAccel (X,Y,Z)	12/24 Bytes	Array (in metres)
• SpherAccel (r,φ,θ)	12/24 Bytes	Array (in metres and degrees)
• OrientAccel (α,β,γ)	12/24 Bytes	Array (in degrees)

7.3 Audio Scene Geometry

The Audio Scene Geometry format is specified in [3]. It is reported here for convenience.

7.3.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Audio Scene Geometry",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "ASGID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "AudioObjectCount": {
      "type": "integer"
    },
    "AudioObjectsData": {
      "type": "object",
      "properties": {
        "AudioObjectID": {
          "type": "string"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        }
      }
    }
  }
}
```

7.3.2 Semantics

Table 12 provides the semantics of the Audio Scene Geometry.

Table 12 – Audio Scene Geometry Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string CAE-ASG
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor version
ASGID	16 Bytes	UUID Identifier of Audio Scene Geometries set.
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start of current Audio Scene Geometry (in μs).
• EndTime	8 Bytes	End of current Audio Scene Geometry (in μs).
AudioObjectCount	1 Byte	Number of Audio Objects in the Audio Scene.
AudioObjectsData	N1 Bytes	Data associated to each Audio Object.
• AudioObjectID	1 Byte	ID of a specific Audio Object in the Audio Scene.
• Reserved	6-7 bits	
• Spatial Attitude	N2 Bytes	Spatial Attitude of each Audio Object.

7.4 Audio Object

7.4.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "AudioObject",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "AOBID": {
      "type": "string"
    },
    "AudioObjectsData": {
      "type": "object",
      "properties": {
        "AudioObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInObject": {
              "$ref": "https://schemas.mpai.community/CAE/V2.1/data/AudioObject.json"
            }
          }
        }
      }
    }
  }
}
```

7.4.2 Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string CAE-ASD
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor version
AOBID	16 Bytes	UUID Identifier of the Audio Object.
AudioObjectData	N1 Bytes	Data associated to each Audio Object.
• Reserved	6-7 bits	
• AudioObject	N2 Bytes	
○ FormatID	1 Byte	Audio Object Format Identifier
○ ObjectLength	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N3 Bytes	Data of Audio Object

7.5 Audio Scene Descriptors

The Audio Scene Descriptors format is specified in [3]. It is reported here for convenience.

7.5.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Audio Scene Descriptors",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "ASDID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "AudioObjectCount": {
      "type": "integer"
    },
    "AudioObjectsData": {
      "type": "object",
      "properties": {
        "AudioObjectID": {
          "type": "string"
        },
        "SamplingRate": {
          "type": "number"
        },
        "SamplingType": {
          "type": "number"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        },
        "AudioObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInObject": {
              "$ref": "https://schemas.mpai.community/CAE/V2.1/data/AudioObject.json"
            }
          }
        }
      }
    }
  }
}
```

7.5.2 Semantics

Table 13 provides the semantics of Audio Scene Descriptors.

Table 13 – Audio Scene Descriptors Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string CAE-ASD
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor
ASDID	16 Bytes	UUID Identifier of Audio Scene Descriptors set.
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start of current Audio Scene Descriptors (in µs).
• EndTime	8 Bytes	End of current Audio Scene Descriptors (in µs).
AudioObjectCount	1 Byte	Number of Audio Objects in the Audio Scene.
AudioObjectsData	N1 Bytes	Data associated to each Audio Object.
• AudioObjectID	1 Byte	ID of a specific Audio Object in the Audio Scene.

• SamplingRate	0-3 bits	0:8, 1:16, 2:24, 3:32, 4:44.1, 5:48, 6: 64, 7: 96, 8: 192 (all kHz)
• SampleType	4-5 bits	0:16, 1:24, 2:32, 3:64 (all bits/sample)
• Reserved	6-7 bits	
• Spatial Attitude	N2 Bytes	According to MPAI-OSD V1
• AudioObject	N3 Bytes	
○ FormatID	1 Byte	Audio Object Format Identifier
○ ObjectLength	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N4 Bytes	Data of Audio Object

7.6 Visual Scene Geometry

7.6.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Visual Scene Geometry",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "VSGID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "VisualObjectCount": {
      "type": "integer"
    },
    "VisualObjectsData": {
      "type": "object",
      "properties": {
        "VisualObjectID": {
          "type": "string"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        }
      }
    }
  }
}
```

7.6.2 Semantics

Table 14 provides the semantics of Visual Scene Geometry.

Table 14 – Visual Scene Geometry Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string OSD-VSD
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor
VSGID	16 Bytes	UUID Identifier of the total set of Visual Scene Geometries (uuid).
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start time of current Visual Scene Descriptors (in microseconds).
• EndTime	8 Bytes	End time of current Visual Scene Descriptors (in microseconds).
VisualObjectCount	1 Byte	Number of Visual Objects in Visual Scene.
VisualObjectsData	N1 Bytes	Data associated to each Visual Object.
• VisualObjectID	1 Byte	ID of a specific Visual Object in a Visual Scene.
• Reserved	1 Byte	
• SpatialAttitude	N2 Bytes	Spatial Attitude of each Object

7.7 Visual Object

7.7.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "VisualObject",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "VOBID": {
      "type": "string"
    },
    "VisualObjectsData": {
      "type": "object",
      "properties": {
        "FormatID": {
          "type": "integer"
        },
        "ObjectLength": {
          "type": "integer"
        },
        "DataInObject": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/VisualObject.json"
        }
      }
    }
  }
}
```

7.7.2 Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string CAE-ASD
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor version
VOBID	16 Bytes	UUID Identifier of the Visual Object.
VisualObjectData	N1 Bytes	Data associated to each Visual Object.
• VisualObject	N2 Bytes	
○ FormatID	1 Byte	Audio Object Format Identifier
○ ObjectLength	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N3 Bytes	Data of Audio Object

7.8 Visual Scene Descriptors

7.8.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Visual Scene Descriptors",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "number"
        }
      }
    },
    "VSDID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "VisualObjectCount": {
      "type": "integer"
    },
    "VisualObjectsData": {
      "type": "object",
      "properties": {
        "VisualObjectID": {
          "type": "string"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        },
        "VisualObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInObject": {
              "$ref": "https://schemas.mpai.community/OSD/V1.0/data/VisualObject.json"
            }
          }
        }
      }
    }
  }
}
```

7.8.2 Semantics

Table 15 provides the semantics of Visual Scene Descriptors.

Table 15 – Visual Scene Descriptors Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string OSD-VSD
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor
VSDID	16 Bytes	UUID Identifier of the total set of Visual Scene Descriptors (uuid).
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start time of current Visual Scene Descriptors (in microseconds).
• EndTime	8 Bytes	End time of current Visual Scene Descriptors (in microseconds).
VisualObjectCount	1 Byte	Number of Visual Objects in Visual Scene.
VisualObjectsData	N1 Bytes	Data associated to each Visual Object.
• VisualObjectID	1 Byte	ID of a specific Visual Object in a Visual Scene.
• Reserved	1 Byte	
• SpatialAttitude	N2 Bytes	According to MPAI-OSD V1
• VisualObject	N3 Bytes	
○ FormatID	1 Byte	Visual Object Format Identifier
○ Length	4 Bytes	Number of Bytes in Visual Object
○ DataInObject	N4 Bytes	Data of Visual Object

7.9 Audio-Visual Scene Geometry

7.9.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Audio-Visual Scene Geometry",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "AVGID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "AVObjectCount": {
      "type": "integer"
    },
    "AVObjectsData": {
      "type": "object",
      "properties": {
        "AVObjectID": {
          "type": "string"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        }
      }
    }
  }
}
```

7.9.2 Semantics

Table 16 provides the semantics of the Audio-Visual Scene Geometry.

Table 16 – Audio-Visual Scene Geometry

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string OSD-AVG
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor

AVGID	16 Bytes	UUID Identifier of the total set of Audio-Visual Scene Geometries.
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start time of current Audio-Visual Scene Descriptors (in microseconds).
• EndTime	8 Bytes	End time of current Audio-Visual Scene Descriptors (in microseconds).
AVObjectCount	1 Byte	Number of Objects in Scene.
AVObjectData	N1 Bytes	Data associated to each Object.
• AVObjectID	1 Byte	ID of a specific Object in the Scene.
• SpatialAttitude	N2 Bytes	

7.10 Audio-Visual Object

7.10.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "AudioVisualObject",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "AVOID": {
      "type": "string"
    },
    "AudioObjectsData": {
      "type": "object",
      "properties": {
        "AudioObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInObject": {
              "$ref": "https://schemas.mpai.community/CAE/V2.1/data/AudioObject.json"
            }
          }
        },
        "VisualObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInObject": {
              "$ref": "https://schemas.mpai.community/OSD/V1.0/data/VisualObject.json"
            }
          }
        }
      }
    }
  }
  }
```

7.10.2 Semantics

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string OSD-AVO
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor version
AVOID	16 Bytes	UUID Identifier of the Audio-Visual Object.
AVObjectData	N1 Bytes	Data associated to each Audio-Visual Object.
• AudioObject	N2 Bytes	
○ FormatID	1 Byte	Audio Object Format Identifier
○ ObjectLength	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N3 Bytes	Data of Audio Object
• VisualObject	N4 Bytes	
○ FormatID	1 Byte	Visual Object Format Identifier
○ Length	4 Bytes	Number of Bytes in Visual Object
○ DataInObject	N5 Bytes	Data of Visual Object

7.11 Audio-Visual Scene Descriptors

7.11.1 Syntax

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "title": "Audio-Visual Scene Descriptors",
  "type": "object",
  "properties": {
    "Header": {
      "type": "object",
      "properties": {
        "Standard": {
          "type": "string"
        },
        "Version": {
          "type": "integer"
        },
        "Subversion": {
          "type": "integer"
        }
      }
    },
    "AVSID": {
      "type": "string"
    },
    "Time": {
      "type": "object",
      "properties": {
        "TimeType": {
          "type": "boolean"
        },
        "StartTime": {
          "type": "number"
        },
        "EndTime": {
          "type": "number"
        }
      }
    },
    "AVObjectCount": {
      "type": "integer"
    },
    "AVObjectsData": {
      "type": "object",
      "properties": {
        "AVObjectID": {
          "type": "string"
        },
        "SamplingRate": {
          "type": "number"
        },
        "SamplingType": {
          "type": "number"
        },
        "SpatialAttitude": {
          "$ref": "https://schemas.mpai.community/OSD/V1.0/data/SpatialAttitude.json"
        },
        "AVObject": {
          "type": "object",
          "properties": {
            "FormatID": {
              "type": "integer"
            },
            "ObjectLength": {
              "type": "integer"
            },
            "DataInAObject": {
              "$ref": "https://schemas.mpai.community/CAE/V2.1/data/AudioObject.json"
            },
            "DataInVObject": {
              "$ref": "https://schemas.mpai.community/OSD/V1.0/data/VisualObject.json"
            }
          }
        }
      }
    }
  }
}
```

7.11.2 Semantics

Table 17 provides the semantics of the Audio-Visual Scene Descriptors.

Table 17 – Audio-Visual Scene Descriptors

Label	Size	Description
HEADER	9 Bytes	
• Standard	7 Bytes	The string OSD-AVS
• Version	1 Byte	Major version
• Subversion	1 Byte	Minor
AVDID	16 Bytes	UUID Identifier of the total set of Audio-Visual Scene Descriptors.
Time	17 Bytes	Collects various data expressed with bits
• TimeType	0 bit	0=Relative: time starts at 0000/00/00T00:00 1=Absolute: time starts at 1970/01/01T00:00.
• Reserved	1-7 bits	reserved
• StartTime	8 Bytes	Start time of current Audio-Visual Scene Descriptors (in microseconds).

• EndTime	8 Bytes	End time of current Audio-Visual Scene Descriptors (in microseconds).
AVObjectCount	1 Byte	Number of Objects in Scene.
AVObjectData	N1 Bytes	Data associated to each Object.
• AVObjectID	1 Byte	ID of a specific Object in the Scene.
• SamplingRate	0-3 bits	0: 8kHz, 1: 16kHz, 2: 24kHz, 3: 32kHz, 4: 44.1kHz, 5: 48kHz, 6: 64kHz, 7: 96kHz, 8: 192kHz
• SampleType	4-5 bits	0:16bit, 1:24bit, 2:32bit, 3:64bit)
• Reserved	6-7 bits	
• SpatialAttitude	N2 Bytes	According to MPAI-OSD V1
• *AudioObject*	N3 Bytes	
○ FormatID	1 Byte	Audio Object Format Identifier
○ Length	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N4 Bytes	Data of Audio Object
• *VisualObject*	N5 Bytes	
○ FormatID	1 Byte	Visual Object Format Identifier
○ Length	4 Bytes	Number of Bytes in Audio Object
○ DataInObject	N6 Bytes	Data of Visual Object

Annex 1 - MPAI Basics

1 General

In recent years, Artificial Intelligence (AI) and related technologies have been introduced in a broad range of applications affecting the life of millions of people and are expected to do so much more in the future. As digital media standards have positively influenced industry and billions of people, so AI-based data coding standards are expected to have a similar positive impact. In addition, some AI technologies may carry inherent risks, e.g., in terms of bias toward some classes of users making the need for standardisation more important and urgent than ever.

The above considerations have prompted the establishment of the international, unaffiliated, not-for-profit Moving Picture, Audio and Data Coding by Artificial Intelligence (MPAI) organisation with the mission to develop *AI-enabled data coding standards* to enable the development of AI-based products, applications, and services.

As a rule, MPAI standards include four documents: Technical Specification, Reference Software Specifications, Conformance Testing Specifications, and Performance Assessment Specifications. The last – and new in standardisation – type of Specification includes standard operating procedures that enable users of MPAI Implementations to make informed decision about their applicability based on the notion of Performance, defined as a set of attributes characterising a reliable and trustworthy implementation.

2 Governance of the MPAI Ecosystem

Technical Specification: Governance of the MPAI Ecosystem lays down the foundations of the MPAI Ecosystem. MPAI develops and maintains the following documents the following technical documents:

1. Technical Specification.
2. Reference Software Specification.
3. Conformance Testing.
4. Performance Assessment.
5. Technical Report

An MPAI Standard is a collection of a variable number of the 5 document types.

Figure 9 depicts the operation of the MPAI ecosystem generated by MPAI Standards.

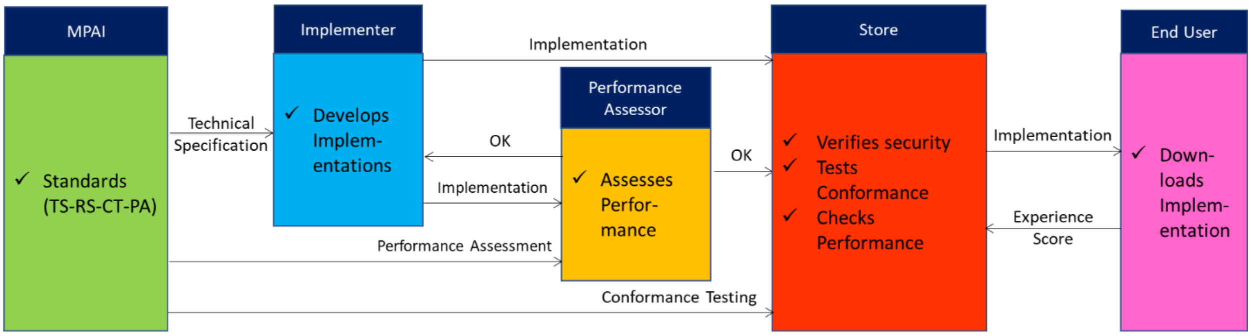

Figure 9 – The MPAI ecosystem operation

Table 18 identifies the following roles in the MPAI Ecosystem:

Table 18 - Roles in the MPAI Ecosystem

MPAI	Publishes Standards. Establishes the not-for-profit MPAI Store. Appoints Performance Assessors.
Implementers	Submit Implementations to Performance Assessors.
Performance Assessors	Inform Implementation submitters and the MPAI Store if Implementation Performance is acceptable.
Implementers	Submit Implementations to the MPAI Store.
MPAI Store	Assign unique ImplementerIDs (IID) to Implementers in its capacity as ImplementerID Registration Authority (IIDRA)[1]. Verifies security and Tests Implementation Conformance.
Users	Download Implementations and report their experience to MPAI.

3 AI Framework

In general, MPAI Application Standards are defined as aggregations – called AI Workflows (AIW) – of processing elements – called AI Modules (AIM) – executed in an AI Framework (AIF). MPAI defines Interoperability as the ability to replace an AIW or an AIM Implementation with a functionally equivalent Implementation.

Figure 10 depicts the MPAI-AIF Reference Model under which Implementations of MPAI Application Standards and user-defined MPAI-AIF Conforming applications operate [2].

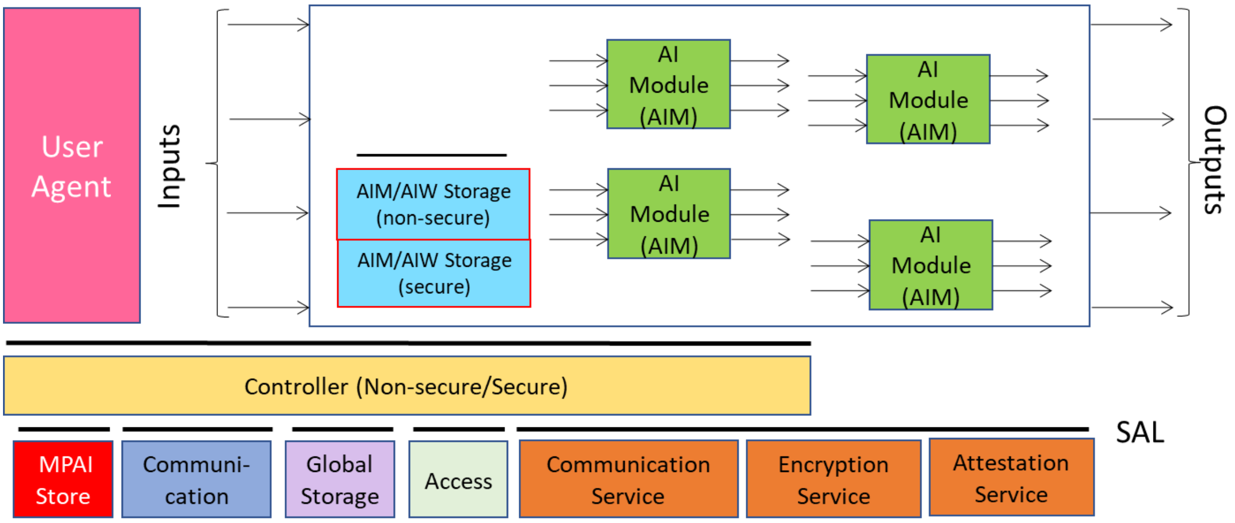

Figure 10 – The AI Framework (AIF) Reference Model

MPAI Application Standards normatively specify the Syntax and Semantics of the input and output data and the Function of the AIW and the AIMs, and the Connections between and among the AIMs of an AIW.

An AIW is defined by its Function and input/output Data and by its AIM topology. Likewise, an AIM is defined by its Function and input/output Data. MPAI standard are silent on the technology

[1] At the time of publication of this Technical Report, the MPAI Store was assigned as the IIDRA.

used to implement the AIM which may be based on AI or data processing, and implemented in software, hardware or hybrid software and hardware technologies.

MPAI also defines 3 Interoperability Levels of an AIF that executes an AIW. Table 19 gives the characteristics of an AIW and its AIMs of a given Level:

Table 19 - MPAI Interoperability Levels

Level	AIW	AIMs
1	An implementation of a use case	Implementations able to call the MPAI-AIF APIs.
2	An Implementation of an MPAI Use Case	Implementations of the MPAI Use Case
3	An Implementation of an MPAI Use Case certified by a Performance Assessor	Implementations of the MPAI Use Case certified by Performance Assessors

4 Audio Scene Description

The ability to describe (i.e., digitally represent) an audio-visual scene is a key requirement of several Use Cases. *Technical Specification: Context-based Audio Enhancement (MPAI-CAE) V2.1* [7] that includes the specification of Audio Scene Descriptors produced by the Composite Audio Scene Description AI Module (AIM) and depicted in Figure 16.

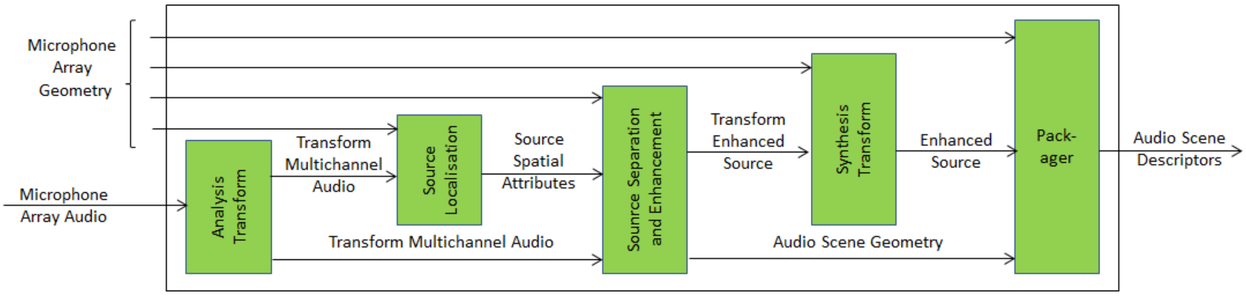
Microphone Array Geometry
Microphone Array Audio
Analysis Transform
Transform Multichannel Audio
Source Localisation
Source Spatial Attributes
Sounrce Separation and Enhancement
Transform Enhanced Source
Synthesis Transform
Enhanced Source
Pack-ager
Audio Scene Descriptors
Transform Multichannel Audio
Audio Scene Geometry

Figure 11 - The Audio Scene Description Composite AIM

5 Avatar-Based Videoconference

Technical Report: Avatar-Based Videoconference (MPAI-ARA) specifies AIWs and AIMs of a Use Case where geographically distributed humans hold a videoconference represented by their avatars having their visual appearance and uttering their real voice (Figure 12).

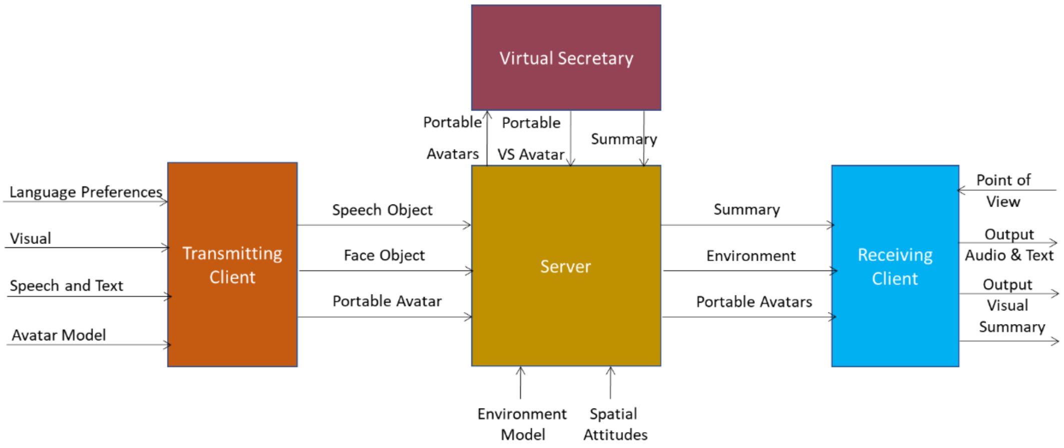

Figure 12 – Avatar-Based Videoconference end-to-end diagram

Figure 13 contains the reference architectures of the four AW Workflows constituting the Avatar-Based Videoconference: Client (Transmission side), Server, Virtual Secretary, and Client (Receiving side).

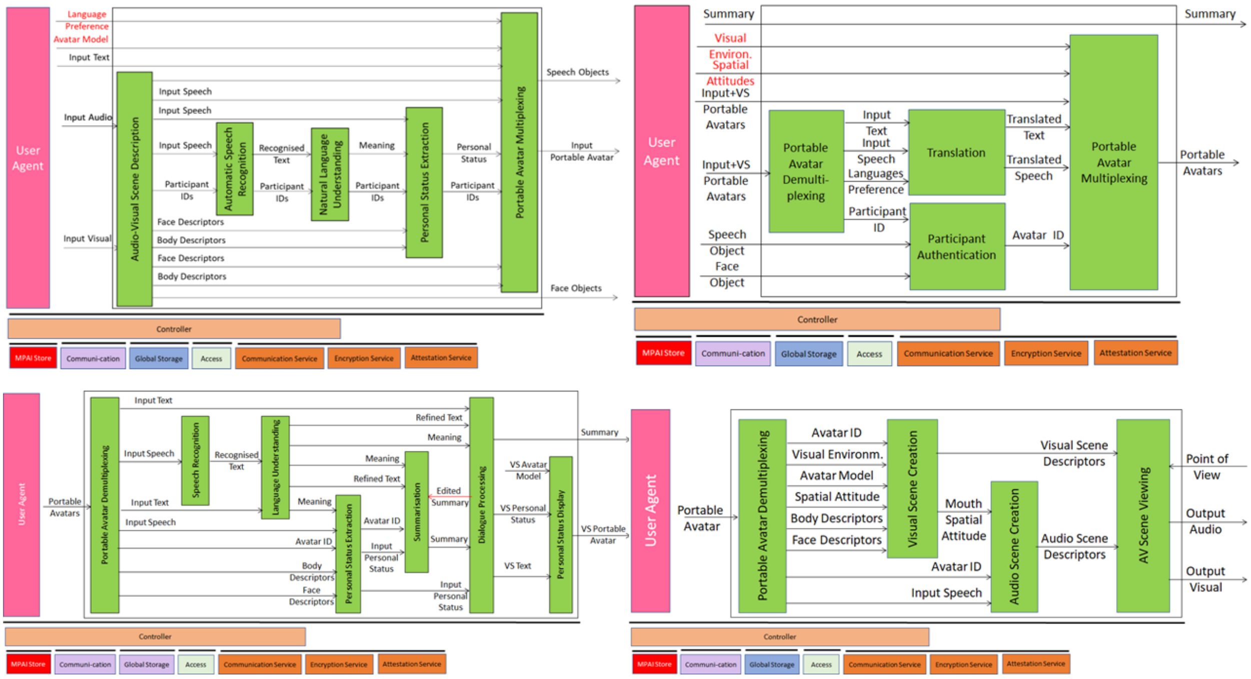

Figure 13 - The AIWs of Avatar-Based Videoconference

6 Connected Autonomous Vehicles

MPAI defines a Connected Autonomous Vehicle (CAV), as a physical system that:

1. Converses with humans by understanding their utterances, e.g., a request to be taken to a destination.
2. Acquires information with a variety of sensors on the physical environment where it is located or traverses like the one depicted in *Figure 14*.
3. Plans a Route enabling the CAV to reach the requested destination.
4. Autonomously reaches the destination by:

4.1. Moving in the physical environment.
4.2. Building Digital Representations of the Environment.
4.3. Exchanging elements of such Representations with other CAVs and CAV-aware entities.
4.4. Making decisions about how to execute the Route.
4.5. Acting on the CAV motion actuation to implement the decisions.

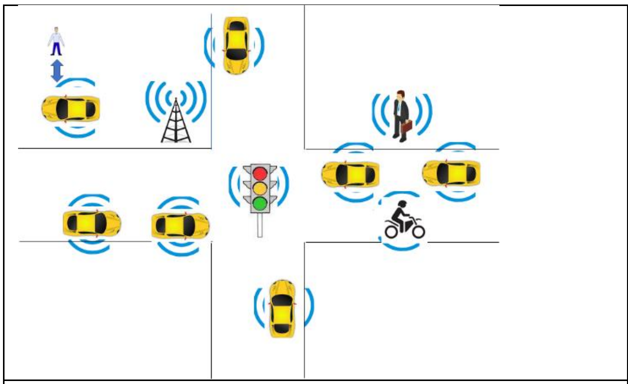

Figure 14 - An environment of CAV operation

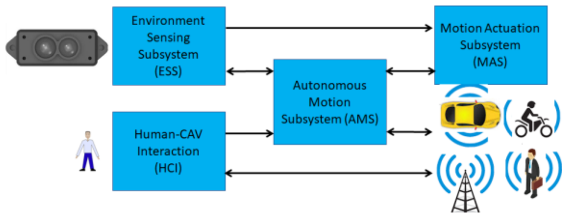
Environment Sensing Subsystem (ESS)
Human-CAV Interaction (HCI)
Autonomous Motion Subsystem (AMS)
Motion Actuation Subsystem (MAS)

Figure 15 – The MPAI-CAV subsystems

MPAI believes in the capability of standards to accelerate the creation of a global competitive CAV market and has published Technical Specification: Connected Autonomous Vehicle (MPAI-CAV) – Architecture that includes (see *Figure 15*):

1. A CAV Reference Model broken down into four Subsystems.
2. The Functions of each Subsystem.
3. The Data exchanged between Subsystems.
4. A breakdown of each Subsystem in Components (see Figure 16) of which the following is specified:

4.1. The Functions of the Components.
4.2. The Data exchanged between Components.
4.3. The Topology of Components and their Connections.

5. Functional Requirements of the Data exchanged (under development).
6. Standard technologies for the Data exchanged (in the future).

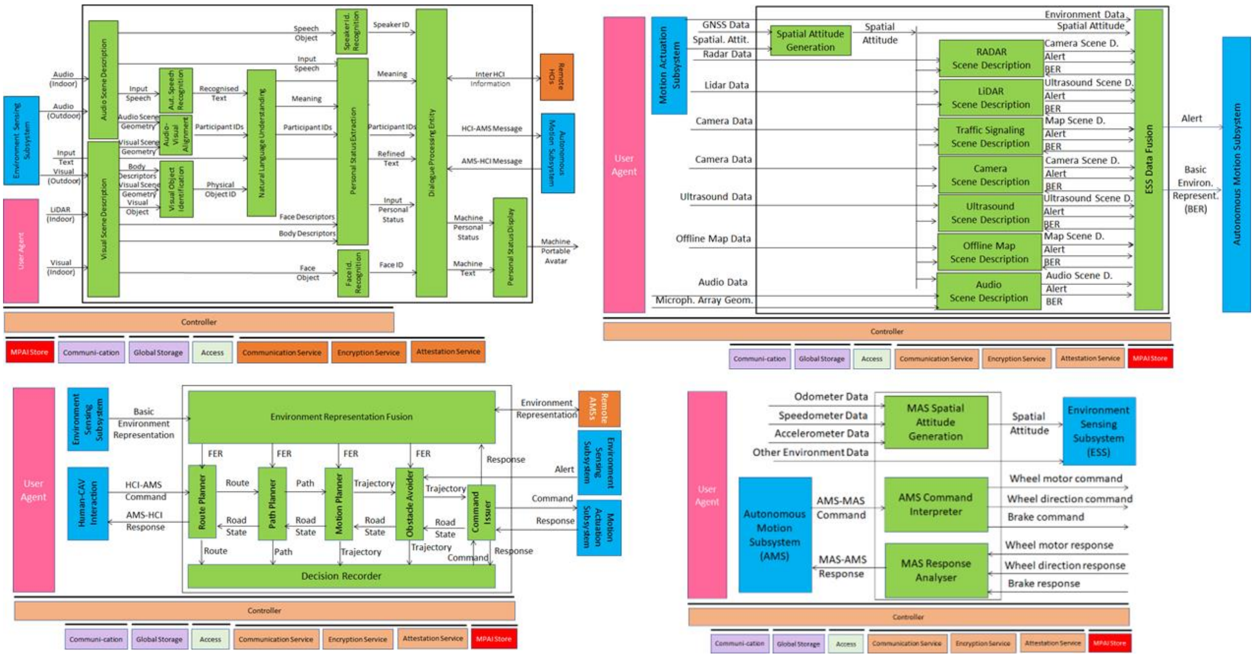

Figure 16 - The MPAI-CAV Subsystems with their Components (left-right & top bottom: Human-Cav Interaction, Environment Sensing Subsystem, Autonomous Motion Subsystem, and Motion Actuation Subsystem)

Subsystems are implemented as AI Workflows and Components as AI Modules according to Technical Specification: AI Framework (MPAI-AIF) [4].

Annex 2 - MPAI-wide terms and definitions

The Terms used in this standard whose first letter is capital and are not already included in Table 1 are defined in Table 20.

Table 20 - MPAI-wide Terms

Term	Definition
Access	Static or slowly changing data that are required by an application such as domain knowledge data, data models, etc.
AI Framework (AIF)	The environment where AIWs are executed.
AI Modules (AIM)	A data processing element receiving AIM-specific Inputs and producing AIM-specific Outputs according to according to its Function. An AIM may be an aggregation of AIMs.
AI Workflow (AIW)	A structured aggregation of AIMs implementing a Use Case receiving AIW-specific inputs and producing AIW-specific outputs according to the AIW Function.
Application Standard	An MPAI Standard designed to enable a particular application domain.
Channel	A connection between an output port of an AIM and an input port of an AIM. The term "connection" is also used as synonymous.
Communication	The infrastructure that implements message passing between AIMs
Composite AIM	An AIM aggregating more than one AIM.
Component	One of the 7 AIF elements: Access, Communication, Controller, Internal Storage, Global Storage, Store, and User Agent
Conformance	The attribute of an Implementation of being a correct technical Implementation of a Technical Specification.
Conformance Tester	An entity Testing the Conformance of an Implementation.
Conformance Testing	The normative document specifying the Means to Test the Conformance of an Implementation.
Conformance Testing Means	Procedures, tools, data sets and/or data set characteristics to Test the Conformance of an Implementation.
Connection	A channel connecting an output port of an AIM and an input port of an AIM.
Controller	A Component that manages and controls the AIMs in the AIF, so that they execute in the correct order and at the time when they are needed
Data Format	The standard digital representation of data.
Data Semantics	The meaning of data.
Ecosystem	The ensemble of actors making it possible for a User to execute an application composed of an AIF, one or more AIWs, each with one or more AIMs potentially sourced from independent implementers.
Explainability	The ability to trace the output of an Implementation back to the inputs that have produced it.
Fairness	The attribute of an Implementation whose extent of applicability can be assessed by making the training set and/or network open to testing for bias and unanticipated results.
Function	The operations effected by an AIW or an AIM on input data.

Global Storage	A Component to store data shared by AIMs.
Internal Storage	A Component to store data of the individual AIMs.
Identifier	A name that uniquely identifies an Implementation.
Implementation	1. An embodiment of the MPAI-AIF Technical Specification, or 2. An AIW or AIM of a particular Level (1-2-3) conforming with a Use Case of an MPAI Application Standard.
Implementer	A legal entity implementing MPAI Technical Specifications.
ImplementerID (IID)	A unique name assigned by the ImplementerID Registration Authority to an Implementer.
ImplementerID Registration Authority (IIDRA)	The entity appointed by MPAI to assign ImplementerID's to Implementers.
Interoperability	The ability to functionally replace an AIM with another AIW having the same Interoperability Level
Interoperability Level	The attribute of an AIW and its AIMs to be executable in an AIF Implementation and to: 1. Be proprietary (Level 1) 2. Pass the Conformance Testing (Level 2) of an Application Standard 3. Pass the Performance Testing (Level 3) of an Application Standard.
Knowledge Base	Structured and/or unstructured information made accessible to AIMs via MPAI-specified interfaces
Message	A sequence of Records transported by Communication through Channels.
Normativity	The set of attributes of a technology or a set of technologies specified by the applicable parts of an MPAI standard.
Performance	The attribute of an Implementation of being Reliable, Robust, Fair and Replicable.
Performance Assessment	The normative document specifying the Means to Assess the Grade of Performance of an Implementation.
Performance Assessment Means	Procedures, tools, data sets and/or data set characteristics to Assess the Performance of an Implementation.
Performance Assessor	An entity Assessing the Performance of an Implementation.
Profile	A particular subset of the technologies used in MPAI-AIF or an AIW of an Application Standard and, where applicable, the classes, other subsets, options and parameters relevant to that subset.
Record	A data structure with a specified structure
Reference Model	The AIMs and theirs Connections in an AIW.
Reference Software	A technically correct software implementation of a Technical Specification containing source code, or source and compiled code.
Reliability	The attribute of an Implementation that performs as specified by the Application Standard, profile and version the Implementation refers to, e.g., within the application scope, stated limitations, and for the period of time specified by the Implementer.
Replicability	The attribute of an Implementation whose Performance, as Assessed by a Performance Assessor, can be replicated, within an agreed level, by another Performance Assessor.
Robustness	The attribute of an Implementation that copes with data outside of the stated application scope with an estimated degree of confidence.

Scope	The domain of applicability of an MPAI Application Standard
Service Provider	An entrepreneur who offers an Implementation as a service (e.g., a recommendation service) to Users.
Standard	The ensemble of Technical Specification, Reference Software, Conformance Testing and Performance Assessment of an MPAI application Standard.
Technical Specification	(Framework) the normative specification of the AIF. (Application) the normative specification of the set of AIWs belonging to an application domain along with the AIMs required to Implement the AIWs that includes: 1. The formats of the Input/Output data of the AIWs implementing the AIWs. 2. The Connections of the AIMs of the AIW. 3. The formats of the Input/Output data of the AIMs belonging to the AIW.
Testing Laboratory	A laboratory accredited to Assess the Grade of Performance of Implementations.
Time Base	The protocol specifying how Components can access timing information
Topology	The set of AIM Connections of an AIW.
Use Case	A particular instance of the Application domain target of an Application Standard.
User	A user of an Implementation.
User Agent	The Component interfacing the user with an AIF through the Controller
Version	A revision or extension of a Standard or of one of its elements.

Annex 3 - Notices and Disclaimers Concerning MPAI Standards (Informative)

The notices and legal disclaimers given below shall be borne in mind when downloading and using approved MPAI Standards.

In the following, “Standard” means the collection of four MPAI-approved and published documents: “Technical Specification”, “Reference Software” and “Conformance Testing” and, where applicable, “Performance Testing”.

Life cycle of MPAI Standards
MPAI Standards are developed in accordance with the MPAI Statutes. An MPAI Standard may only be developed when a Framework Licence has been adopted. MPAI Standards are developed by especially established MPAI Development Committees who operate on the basis of consensus, as specified in Annex 1 of the MPAI Statutes. While the MPAI General Assembly and the Board of Directors administer the process of the said Annex 1, MPAI does not independently evaluate, test, or verify the accuracy of any of the information or the suitability of any of the technology choices made in its Standards.

MPAI Standards may be modified at any time by corrigenda or new editions. A new edition, however, may not necessarily replace an existing MPAI standard. Visit the web page to determine the status of any given published MPAI Standard.

Comments on MPAI Standards are welcome from any interested parties, whether MPAI members or not. Comments shall mandatorily include the name and the version of the MPAI Standard and, if applicable, the specific page or line the comment applies to. Comments should be sent to the MPAI Secretariat. Comments will be reviewed by the appropriate committee for their technical relevance. However, MPAI does not provide interpretation, consulting information, or advice on MPAI Standards. Interested parties are invited to join MPAI so that they can attend the relevant Development Committees.

Coverage and Applicability of MPAI Standards
MPAI makes no warranties or representations of any kind concerning its Standards, and expressly disclaims all warranties, expressed or implied, concerning any of its Standards, including but not limited to the warranties of merchantability, fitness for a particular purpose, non-infringement etc. MPAI Standards are supplied “AS IS”.

The existence of an MPAI Standard does not imply that there are no other ways to produce and distribute products and services in the scope of the Standard. Technical progress may render the technologies included in the MPAI Standard obsolete by the time the Standard is used, especially in a field as dynamic as AI. Therefore, those looking for standards in the Data Compression by Artificial Intelligence area should carefully assess the suitability of MPAI Standards for their needs.

IN NO EVENT SHALL MPAI BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND

ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE PUBLICATION, USE OF, OR RELIANCE UPON ANY STANDARD, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE AND REGARDLESS OF WHETHER SUCH DAMAGE WAS FORESEEABLE.

MPAI alerts users that practicing its Standards may infringe patents and other rights of third parties. Submitters of technologies to this standard have agreed to licence their Intellectual Property according to their respective Framework Licences.

Users of MPAI Standards should consider all applicable laws and regulations when using an MPAI Standard. The validity of Conformance Testing is strictly technical and refers to the correct implementation of the MPAI Standard. Moreover, positive Performance Assessment of an implementation applies exclusively in the context of the MPAI Governance and does not imply compliance with any regulatory requirements in the context of any jurisdiction. Therefore, it is the responsibility of the MPAI Standard implementer to observe or refer to the applicable regulatory requirements. By publishing an MPAI Standard, MPAI does not intend to promote actions that are not in compliance with applicable laws, and the Standard shall not be construed as doing so. In particular, users should evaluate MPAI Standards from the viewpoint of data privacy and data ownership in the context of their jurisdictions.

Implementers and users of MPAI Standards documents are responsible for determining and complying with all appropriate safety, security, environmental and health and all applicable laws and regulations.

Annex 4 - Patent declarations (Informative)

Technical Specification: Object and Scene Description (MPAI-OSD) V1 has been developed according to the process outlined in the MPAI Statutes [9] and the MPAI Patent Policy [10], and following the prescriptions of *Framework Licence: Object and Scene Description (MPAI-OSD)* [11].

Table 21 will report the list of entities who will agree to licence their standard essential patents reading on *Technical Specification: Object and Scene Description (MPAI-OSD) V1* according to [11].:

Table 21 - Companies having submitted a patent declaration on (MPAI-OSD)

Entity	**Name**	**Email address**

The declarations will be published when patent declarations will be received in response to requests for declarations.