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

Public document

N1360 2022/08/12

Source MPAI-35

Title Use Cases and Functional Requirements – Object and Scene Description (MPAI-OSD)

Target MPAI Members

Abstract

Object and Scene Description (MPAI-OSD) is an MPAI project addressing a multiplicity of use cases where “objects” and “scenes” play a role. Objects and scene can be unimodal, i.e., only be composed of one type of media – audio or visual – or multimodal.

This document, part of the MPAI-OSD Call for Technologies set of documents collects Use Cases and Functional Requirements of technologies required to implement the Use Cases

Responses to the Call should be received by the MPAI Secretariat by 2023/09/15T23:59 UTC.

1 Introduction

This Use Case and Functional Requirements: Object and Scenes Description (MPAI-OSD) document describes a set of Use Cases and identifies a subset of the Functional Requirements where uni- and multimodal objects and scenes pay a role. This document is part of the MPAI Object and Scenes Description (MPAI-OSD) project.

The purpose of the planned Technical Specification targets the specification of object description and their localisation in a space. The Call for Technologies [1] seeks to obtain technologies that support some of and preferably all the Functional Requirements identified in this document that MPAI intends to use in the development of the planned Technical Specification. Those proposing technologies in response to the Call are requested to state their availability to license their technologies, if adopted by MPAI, in conformity with the Framework Licence [2].

2 Terms and Definitions

The terms used in this standard whose first letter is capital have the meaning defined in *Table 1*.

Table 1 – Table of terms and definitions

Term	Definition
Audio Object	Coded representation of Audio information with its metadata.
Descriptor	Coded representation of text, audio, speech, or visual feature.
Environment	A Physical or Virtual Space containing a Scene.
Identifier	The label uniquely associated with a human or an avatar or an object.
Instance	An element of a set of entities – Physical Objects, users etc. – belonging to some levels in a hierarchical classification (taxonomy).
Object Descriptor	An individual attribute of the coded representation of an object in a Scene, including its Spatial Attitude.
Orientation	The 3 yaw, pitch, and roll (α,β,γ) angles of a representative point of an object in the Real and Virtual Space.
Point of View	The Position and Orientation of a human or avatar looking at an Environment.
Position	The 3 coordinates (x,y,z) of a representative point of an object in the Real and Virtual Space.
Scene	An Environment populated by objects whether real or virtual.
Scene Presentation	The format used by an audio-visual renderer to render the Audio-Visual Scene internal to the machine from a selected Point of View.
Spatial Attitude	Position and Orientation and their velocities and accelerations of a Human and Physical Object in a Real or Virtual Environment.
Visual Object	Coded representation of Visual information with its metadata.

3 References

1. MPAI; Call for Technologies: Object and Scene Description (MPAI-OSD); N1359;https://mpai.community/standards/mpai-osd/call-for-technologies/
2. MPAI; Framework Licence: Object and Scene Description (MPAI-OSD); N1361; https://mpai.community/standards/mpai-osd/framework-licence/
3. MPAI; Patent Policy; https://mpai.community/about/the-mpai-patent-policy/
4. MPAI; Technical Specification: The governance of the MPAI ecosystem (MPAI-GME), V1.1; https://mpai.community/standards/mpai-gme/
5. MPAI; Technical Specification: AI Framework (MPAI-AIF) V2; https://mpai.community/standards/mpai-aif/
6. MPAI; Technical Specification: Avatar Representation and Animation (MPAI-ARA) V2; https://mpai.community/standards/mpai-ara/
7. MPAI; Technical Specification: Context-based Audio Enhancement (MPAI-CAE) V2; https://mpai.community/standards/mpai-cae/
8. Technical Specification: Connected Autonomous Vehicles (MPAI-CAV) – Architecture V1; https://mpai.community/standards/mpai-cav/
9. MPAI; Technical Specification: Multimodal Conversation (MPAI-MMC) V2; https://mpai.community/standards/mpai-mmc/
10. Khronos; Graphics Language Transmission Format (glTF); October 2021; https://registry.khronos.org/glTF/specs/2.0/glTF-2.0.html
11. MPAI; The MPAI Statutes; N421; https://mpai.community/statutes/
12. MPAI; The MPAI Patent Policy; https://mpai.community/about/the-mpai-patent-policy/.

1. Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Architecture V1; https://mpai.community/standards/mpai-mmm/

4 Use Cases

4.1 Conversation About a Scene (CAS)

Note: the full specification of this Use Case is provided in [8].

4.1.1 Scope of Conversation About a Scene

This Use Case addresses the case of a human holding a conversation with a machine:

1. The machine perceives (sees and hears) an Environment containing a speaking human and some scattered objects.
2. The machine recognises the human's Speech and obtains the human's Personal Status by capturing Speech, Face, and Gesture.
3. The human converses with the machine indicating the object in the Environment s/he wishes to talk to or ask questions about using Speech, Face, and Gesture.
4. The machine understands which object the human is referring to and generates an avatar that:
    a. Utters Speech conveying a synthetic Personal Status that is relevant to the human's Personal Status as shown by his/her Speech, Face, and Gesture, and
    b. Displays a face conveying a Personal Status that is relevant to the human's Personal Status and to the response the machine intends to make.
5. The machine displays the Scene Presentation corresponding to how it perceives the Environment. The objects in the scene are labelled with the machine's understanding of their semantics so that the human can understand how the machine sees the Environment.

4.1.2 Reference Architecture of Conversation About a Scene

Figure 1 gives the Conversation About a Scene Reference Model including the input/output data, the AIMs, and the data exchanged between and among the AIMs.

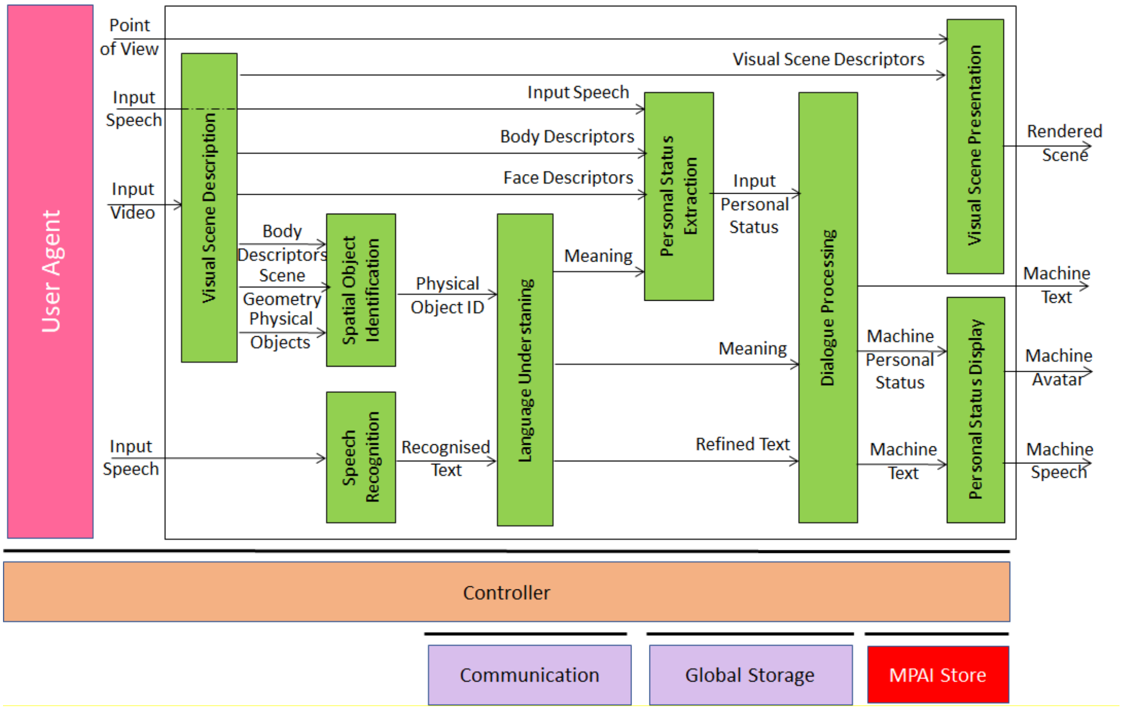


Figure 1 – Reference Model of Conversation About a Scene

The Machine operates according to the following workflow:

1. Visual Scene Description produces Body Object and Physical Objects from Input Video.
2. Speech Recognition produces Recognised Text from Input Speech.

3. Spatial Object Identification produces Object ID from Physical Object and Body Object.
4. Language Understanding produces Meaning and Refined Text from Recognised Text and Physical Object ID.
5. Personal Status Extraction produces Input Personal Status from Input Speech, Face Descriptors, Body Descriptors, and Meaning.
6. Dialogue Processing produces Machine Text and Machine Personal Status from Input Personal Status, Meaning, and Refined Text.
7. Personal Status Display produces Machine Text, Machine Speech, Machine Avatar from Machine Text, and Machine Personal Status.
8. Scene Presentation uses the Visual Scene Descriptors to produce the Rendered Scene as seen from the user-selected Point of View. The rendering is constantly updated as the machine improves its understanding of the scene and its objects.

4.1.3 I/O Data of Conversation About a Scene

Table 2 gives the input/output data of Conversation About a Scene

Table 2 – I/O data of Conversation About a Scene

Input data	**From**	**Comment**
Input Video	Camera	Points to human and scene.
Input Speech	Microphone	Speech of human.
Point of View	Human	The point of view of the scene displayed by Scene Presentation.
Output data	**To**	**Comments**
Machine Speech	Human	Machine’s speech.
Machine Avatar	Human	Portion of machine’s avatar (e.g., face).
Rendered Scene	Human	Reproduction of the scene perceived by machine containing labelled objects as seen from the Point of View.

4.2 Human-Connected Autonomous Vehicle (CAV) Interaction (HCI)

4.2.1 Scope of Human-CAV Interaction Subsystem

A Connected Autonomous Vehicle (CAV) is a system able to execute a command to move itself based on 1) capture of data sensed by a range of onboard sensors exploring the environment and 2) analysis, and interpretation of the data captured and transmitted by other sources in range, such as other CAVs, traffic lights and roadside units. Chapter 5 of Annex 1 - Connected Autonomous Vehicle describes the four Subsystems of a CAV. **Human-CAV interaction (HCI)** is one such Subsystem having the function to recognise the human owner or renter, respond to humans’ commands and queries, converse with humans during the travel and activate the Autonomous Motion Subsystem in response to humans’ requests. Reference Architecture of Human-CAV Interaction.

4.2.2 Reference Architecture of Human-CAV Interaction Subsystem

Figure 2 represents the Human-CAV Interaction (HCI) Reference Model.

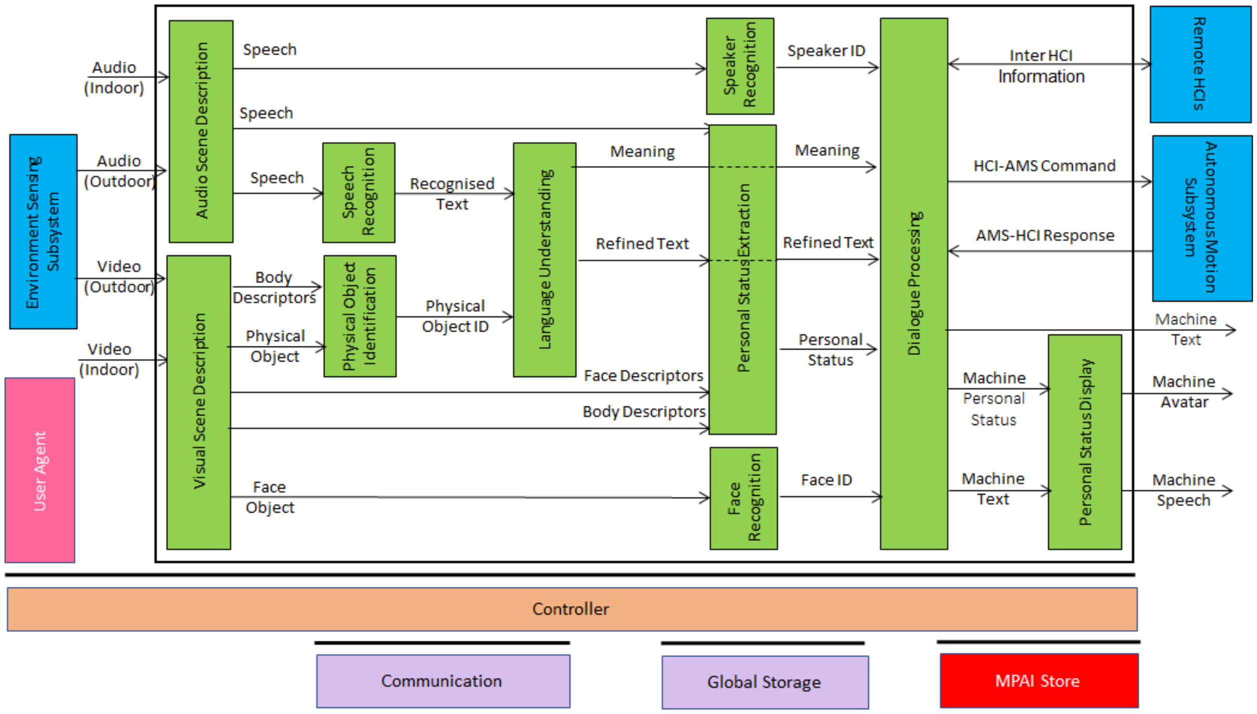


Figure 2 – Human-CAV Interaction Reference Model

The operation of HCI involves the following functions:

1. A group of humans approaches the CAV outside the CAV:
   1.1. The Audio Scene Description AIM creates the Audio Scene Description in the form of Audio (Speech) Objects corresponding to each speaking human in the Environment (close to the CAV).
   1.2. The Visual Scene Description creates the Visual Scene Descriptors in the form of Body Objects with the possibility of extracting the Head and Face corresponding to each human in the Environment (close to the CAV).
   1.3. The Speaker Recognition and Face Recognition AIMs authenticate the humans that the HCI is interacting with using Speech and Face Descriptors.
   1.4. The Speech Recognition AIM recognises the speech of each human.
   1.5. The Language Understanding AIM extracts Meaning and produces the refined Refined Text.
   1.6. The Personal Status Extraction AIM extracts the Personal Status of the humans.
   1.7. The Dialogue Processing AIM validates the human Identities, produces the response and displays the HCI Personal Status, and issues commands to the Autonomous Motion Subsystem.
2. A group of humans sits in the seats inside the CAV:
   2.1. The Audio Scene Description AIM creates the Audio Scene Descriptions in the form of Audio (Speech) Objects corresponding to each speaking human in the cabin.
   2.2. The Visual Scene Description creates the Visual Scene Descriptors in the form of Body Objects with the possibility of extracting the Head and Face corresponding to each human in the cabin.
   2.3. The Speaker Recognition and Face Recognition AIMs identify the humans the HCI is interacting with using Speech and Face Descriptors.
   2.4. The Speech Recognition AIM recognises the speech of each human.
   2.5. The Language Understanding AIM extracts Meaning and produces the refined Refined Text.
   2.6. The Personal Status Extraction AIM extracts the Personal Status of the humans.

2.7. The Dialogue Processing AIM recognises the human Identities, produces the response, displays the HCI Personal Status, and issues commands to the Autonomous Motion Subsystem.

3. The HCI interacts with the humans in the cabin in several ways:

3.1. By responding to commands/queries from one or more humans at the same time, e.g.:

- Commands to go to a waypoint, park at a place, etc.
- Commands with an effect in the cabin, e.g., turn off air conditioning, turn on the radio, call a person, open window or door, search for information etc.

Note: For completeness, Figure 2 includes the interaction of HCI with AMS (e.g., commands and responses regarding selection of Route by human) and with remote HCIs. However, this document does not address the format in which these interactions are performed.

3.2. By conversing with and responding to questions from one or more humans at the same time about travel-related issues (in-depth domain-specific conversation), e.g.:

- Humans request information, e.g., time to destination, route conditions, weather at destination, etc.
- CAV offers alternatives to humans, e.g., long but safe way, short but likely to have interruptions.
- Humans ask questions about objects in the cabin.

3.3. By following the conversation on travel matters held by humans in the cabin if 1) the passengers allow the HCI to do so, and 2) the processing is carried out inside the CAV.

Note that the version of the Audio Scene Description provides all the Speech Objects in the Audio Scene, removing all other audio sources. The Speaker Recognition and Speech Recognition AIMs support multiple Speech Objects as input. Each Speech Object has an identifier to enable the Speaker Recognition and Speech Recognition AIMs to provide labelled Speaker IDs and Recognised Texts. If the Face Recognition AIM provides Face IDs corresponding to the Speaker IDs, the Dialogue Processing AIM can correctly associate the Speaker IDs (and the corresponding Recognised Texts) with the Face IDs.

4.2.3 I/O Data of Human-CAV Interaction

Table 3 gives the input/output data of Human-CAV Interaction.

Table 3 – I/O data of Human-CAV Interaction

Input data	**From**	**Comment**
Audio (ESS)	Users in the outside Environment	User authentication User command User conversation
Audio	Cabin Passengers	User’s social life Commands/interaction with CAV
Text	Cabin Passengers	User’s social life Commands/interaction with CAV
Video (ESS)	Users in the outside Environment	Commands/interaction with CAV
Video	Cabin Passengers	User’s social life Commands/interaction with CAV
AMS Response	Motion Actuation Subsystem	AMS Response about Command execution
Output data	**To**	**Comments**

Output Speech	Cabin Passengers	CAV’s response to passengers
Output Avatar	Cabin Passengers	Portion of CAV’s Avatar (e.g., heand & face).
Output Text	Cabin Passengers	CAV’s response to passengers
AMS Commands	Motion Actuation Subsystem	Command to AMS to actuate wheels, brakes, etc.

4.3 Environment Sensing Subsystem in a Connected Autonomous Vehicle

4.3.1 Functions of Environment Sensing Subsystem

The Environment Sensing Subsystem (ESS) of a Connected Autonomous Vehicle (CAV):

1. Uses all Subsystem devices to acquire as much as possible information from the Environment – electromagnetic and acoustic data.
2. Receives an initial estimate of the Ego CAV’s Spatial Attitude and Environment Data (e.g., temperature, pressure, humidity, etc.) from the Motion Actuation Subsystem.
3. Produces a sequence of Basic Environment Representations (BER) for the duration of the travel.
4. Passes the Basic Environment Representations to the Autonomous Motion Subsystem.

4.3.2 Reference Architecture of Environment Sensing Subsystem

Figure 3 gives the Environment Sensing Subsystem Reference Model.

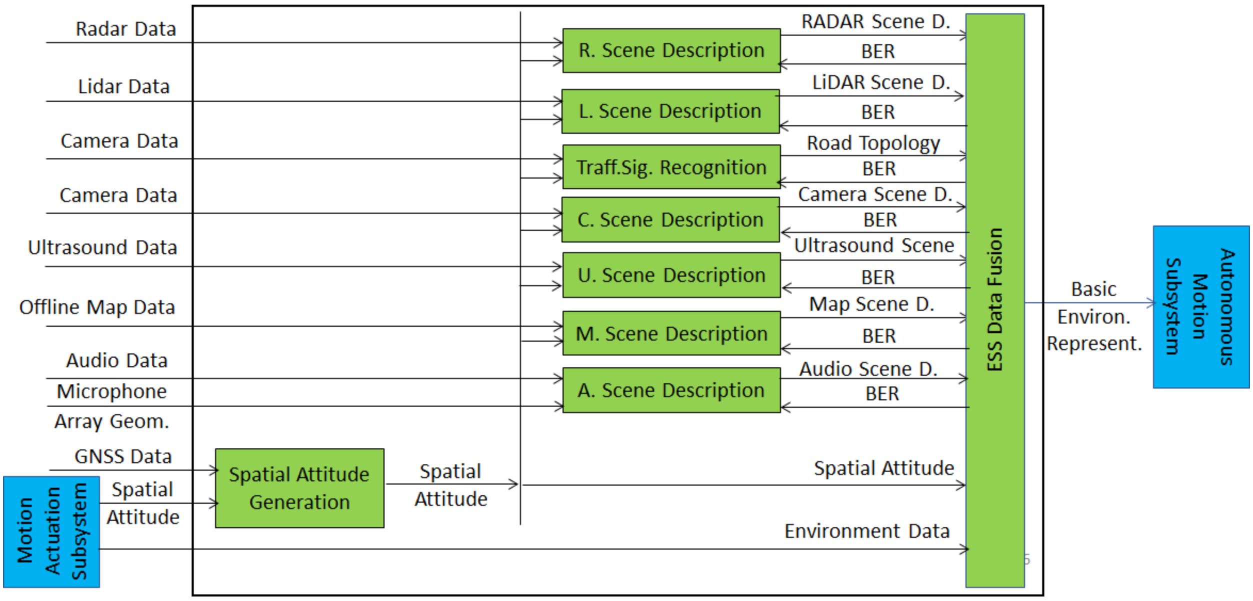


Figure 3 – Environment Sensing Subsystem Reference Model

The typical sequence of operations of the Environment Sensing Subsystem AIM is:

1. Compute the CAV’s Spatial Attitude using the initial Spatial Attitude provided by the Motion Actuation Subsystem and the GNSS.
2. Produce Environment Sensing Technology (EST)-specific Scene Descriptors, e.g., the RADAR Scene Descriptors, using snapshots of information (RADAR Data) provided by the RADAR EST.
3. Access the Basic Environment Representation at a previous time to produce the EST-specific Scene Description.
4. Integrate the Scene Descriptors from different Environment Sensing Technologies into the time-dependent Basic Environment Representation that includes Alert information.

Figure 3 assumes that Traffic Signalisation Recognition produces the Road Topology by analysing Camera Data. The model of *Figure 3* can easily by extended to the case where Data from other ESTs is processed to compute or help compute the Road Topology.

Figure 3 assumes that Environment Sensing Technologies are individually processed. An implementation may create a single Scene Descriptors for two or more ESTs.

4.3.3 I/O Data of Environment Sensing Subsystem

The currently considered Environment Sensing Technologies (EST) are:

1. Global navigation satellite system or GNSS (~1 & 1.5 GHz Radio).
2. Geographical position and orientation, and their time derivatives up to 2nd order (Spatial Attitude).
3. Visual Data in the visible range, possibly supplemented by depth information (400 to 700 THz).
4. LiDAR Data (~200 THz infrared).
5. RADAR Data (~25 & 75 GHz).
6. Ultrasound Data (> 20 kHz).
7. Audio Data in the audible range (16 Hz to 16 kHz).
8. Spatial Attitude (from the Motion Actuation Subsystem).
9. Other environmental data (temperature, humidity, ...).

Table 4 gives the input/output data of the Environment Sensing Subsystem.

Table 4 – I/O data of Environment Sensing Subsystem

Input data	From	Comment
Radar Data	~25 & 75 GHz Radio	Capture Environment with Radar
Lidar Data	~200 THz infrared	Capture Environment with Lidar
Camera Data (2/D and 3D)	Video (400-800 THz)	Capture Environment with Cameras
Ultrasound Data	Audio (>20 kHz)	Capture Environment with Ultrasound
Offline Mapa Data	Local storage	cm-level data at time of capture
Audio Data	Audio (16 Hz-16 kHz)	Capture Environment or cabin with Microphone Array
Microphone Array Geometry	Microphone Array	Microphone Array disposition
Global Navigation Satellite System (GNSS) Data	~1 & 1.5 GHz Radio	Get Pose from GNSS
Spatial Attitude	Motion Actuation Subsystem	To be fused with GNSS data
Other Environment Data	Motion Actuation Subsystem	Temperature etc. added to Basic Environment Representation
Output data	**To**	**Comment**
Alert	Autonomous Motion Subsystem	Critical last minute Environment Description from EST (in BER)
Basic Environment Representation	Autonomous Motion Subsystem	ESS-derived representation of external Environment

4.4 Autonomous Motion Subsystem in a Connected Autonomous Vehicle

4.4.1 Functions of Autonomous Motion Subsystem

The typical series of operations carried out by the Autonomous Motion Subsystem (AMS) is described below. Note that the sequential description does not imply that an operations can only be carried out after the preceding one has been completed.

1 Human-CAV Interaction requests Autonomous Motion Subsystem to plan and move the CAV to the human-selected destination. A dialogue between AMS-HCI-human may follow.
2 Computes the Route satisfying the human's request.
3 Receives the current Basic Environment Representation from Environment Sensing Subsystem.
4 While moving, the CAV:
4.1 Broadcasts a subset of the Basic Environment Representation and other data to CAVs in range.
4.2 Receives subsets of Basic Environment Representations and other data from other CAVs.
4.3 Produces the Full Environment Representation by fusing its own Basic Environment Representation with those from other CAVs in range.
4.4 Plans a Path connecting Poses.
4.5 Selects behaviour and motion to reach the next Pose acting on information about the Poses other CAVs in range intend to reach and the Objects between the current Pose and the next Pose.
4.6 Defines a Trajectory that:
4.6.1 Complies with general traffic regulations and local traffic rules.
4.6.2 Preserves passengers' comfort.
4.7 Refines Trajectory to avoid obstacles.
4.8 Sends Commands to the Motion Actuation Subsystem to take the CAV to the next Pose.
5 Stores the data resulting from a decision (Route Planner, Path Planner etc.)

4.4.2 Reference Architecture of Autonomous Motion Subsystem

Figure 4 gives the Autonomous Motion Subsystem Reference Model.

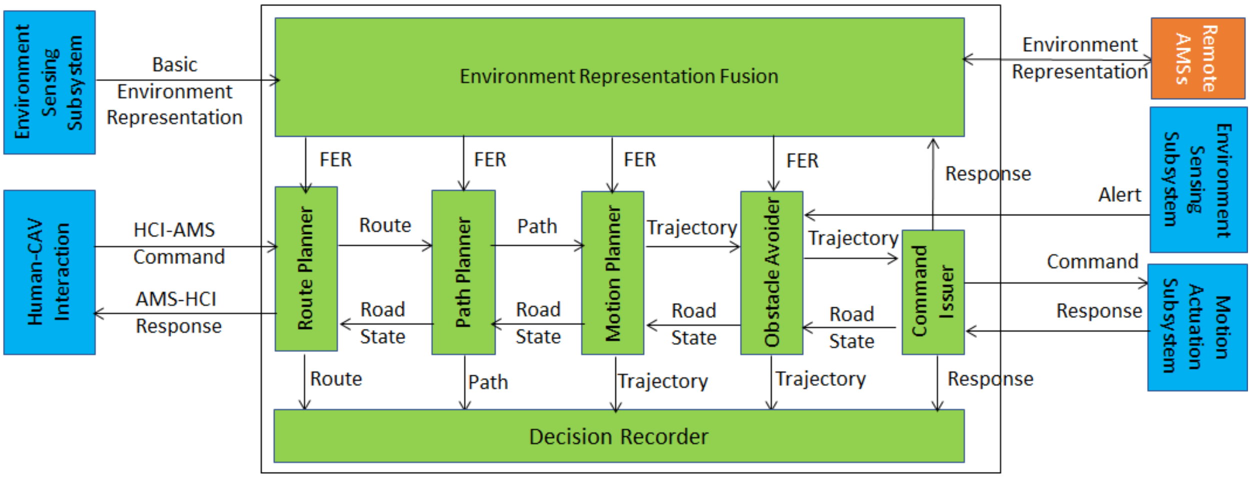


Figure 4 – Autonomous Motion Subsystem Reference Model

This is the operation according to the Reference Model:

1. A human requests the Human-CAV Interaction to the CAV transports them to a destination.
2. HCI interprets request and passes interpretation to the AMS.

3. The AMS activates the Route Planner to generate a set of Waypoints starting from the current Pose, obtained from the Full Environment Representation, up to the destination.
4. The Waypoints enter the Path Planner which generates a set of Poses to reach the next Waypoint.
5. For each Path, the Motion Planner generates a Trajectory to reach the next Pose.
6. The Obstacle Avoider AIM receives the Trajectory and checks if there is a last-minute change, detected from Alert.
7. If an Alert was received, the Obstacle Avoider AIM checks whether the implementation of the Trajectory creates a collision, especially with the Object creating the Alert.
    a. If a collision is indeed detected, the Obstacle Avoider AIM requests a new Trajectory from the Motion Planner.
    b. If no collision is detected, Obstacle Avoider AIM issues a Command to the Motion Actuation Subsystem.
8. The Motion Actuation Subsystem sends Feedback about the execution of the Command.
9. The AMS, based on the MAS-AMS Responses received and the potential discovery of changes in the Environment, can decide to discontinue the execution of the earlier Command and issue another AMS-MAS Command instead.
10. The decision of each element of the said chain may be recorded.

4.4.3 I/O Data of Autonomous Motion Subsystem

Table 5 gives the input/output data of Autonomous Motion Subsystem.

Table 5 – I/O data of Autonomous Motion Subsystem

Input data	**From**	**Comment**
Basic Environment Representation	Environment Sensing Subsystem	CAV's Environment representation.
HCI-AMS Command	Human-CAV Interaction	Human commands, e.g., "take me home"
Environment Representation	Other CAVs	Other CAVs and vehicles, and roadside units.
MAS-AMS Response	Motion Actuation Subsystem	CAV's response to AMS-MAS Command.
Output data	**To**	**Comment**
AMS-HCI Response	Human-CAV Interaction	MAS's response to AMS-MAS Command
AMS-MAS Command	Motion Actuation Subsystem	Macro-instructions, e.g., "in 5s assume a given State".
Environment Representation	Other CAVs	For information to other CAVs

4.5 Avatar-Based Videoconference – Client (Transmission side)

Avatar-Based Videoconference is a videoconference whose participants are avatars realistically impersonating human participants. See Chapter 5 of Annex 1 - MPAI Basics for more information on the Avatar-Based Videoconference Use Case. This is fully specified in [3].

4.5.1 Functions of Client (Transmission side)

The function of a Transmitting Client is to:

1. Receive:
    a. Input Audio from the microphone (array).
    b. Input Video from the camera (array).

    c. Participant's Avatar Model.
    d. Participant's spoken language preferences (e.g., EN-US, IT-CH).

2. Send to the Server:
    a. Speech Descriptors (for Authentication).
    b. Face Descriptors (for Authentication).
    c. Participant's spoken language preferences.
    d. Speech.
    e. Avatar Model.
    f. Compressed Avatar Descriptors.

4.5.2 Reference Architecture of Client (Transmission side)

Avatar-Based Videoconference is a videoconference whose participants are avatars realistically impersonating human participants. See Chapter 5 of Annex 1 - MPAI Basics for more information on the Avatar-Based Videoconference Use Case. This is fully specified in [3].

Figure 5 gives the architecture of the Transmitting Client AIW. Red text refers to data sent at meeting start.

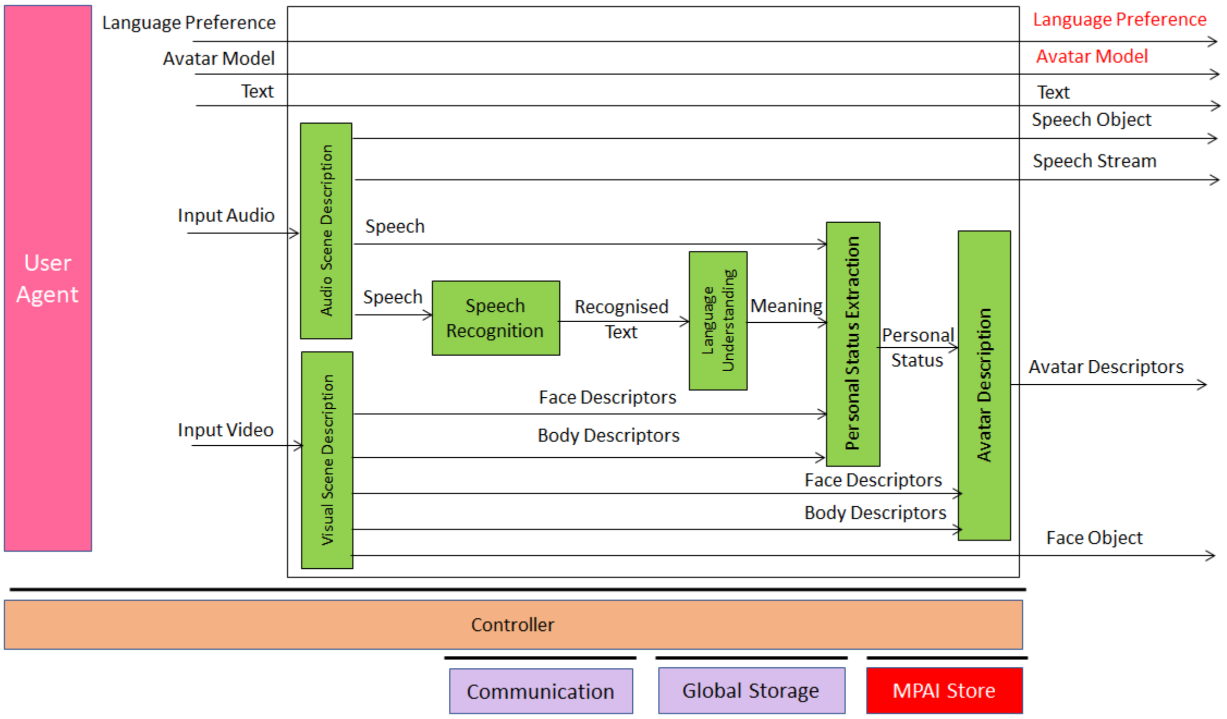


Figure 5 – Reference Model of Avatar Videoconference Transmitting Client

At the start each participant sends to the Server:
1. Language preferences
2. Avatar model.

During the meeting the following AIMs of the Transmitting Client produce:
1. Audio Scene Description: Audio Scene Descriptors.
2. Visual Scene Description: Visual Scene Descriptors.
3. Speech Recognition: Recognised Text.
4. Face Description: Face Descriptors.
5. Body Description: Body Descriptors.
6. Personal Status Extraction: Personal Status.
7. Language Understanding: Meaning.

8. Avatar Description: Compressed Avatar Descriptors.

During the meeting, the Transmitting Client of each participant sends to the Server for distribution to all participants:

1. Speech.
2. Compressed Avatar Descriptors.

4.5.3 Input and output data of Client (Transmission side)

Table 6 gives the input and output data of the Transmitting Client AIW:

Table 6 – Input and output data of Client Transmitting AIW

Input	Comments
Text	Chat text used to communicate with Virtual Secretary or other participants
Language Preference	The language participant wishes to speak and hear at the videoconference.
Input Audio	Audio of participant's Speech and Environment Audio.
Input Video	Video of participants' upper part of the body.
Avatar Model	The avatar model selected by the participant.
Output	**Comments**
Language Preference	As in input.
Participant's Speech	Speech as separated from Environment Audio.
Compressed Avatar Descriptors	Compressed Descriptors produced by Transmitting Client.

4.6 Avatar-Based Videoconference – Server

Avatar-Based Videoconference is a videoconference whose participants are avatars realistically impersonating human participants. See Chapter 5 of Annex 1 - MPAI Basics for more information on the Avatar-Based Videoconference Use Case. This is fully specified in [3].

4.6.1 Functions of Server

1. At the start, the Server:
1.1. Selects an Environment Model.
1.2. Selects the positions of the participants' Avatar Models.
1.3. Authenticates Participants.
1.4. Selects the common meeting language.
2. During the videoconference, the Server:
2.1. Receive participants' text, speech, and compressed Avatar Descriptors.
2.2. Translate participants' speech signals according to their language preferences.
2.3. Send participants' text, speech translated to the common meeting language, and compressed Avatar Descriptors to Virtual Secretary.
2.4. Receive text, speech, and compressed Avatar Descriptors from Virtual Secretary.
2.5. Translate Virtual Secretary's speech signal according to each participant's language preferences.
2.6. Send participants' and Virtual Secretary's text, translated speech, and compressed Avatar Descriptors to participants' clients.

4.6.2 Reference Architecture of Server

Figure 5 gives the architecture of Server AIW. Red text refers to data sent at meeting start.

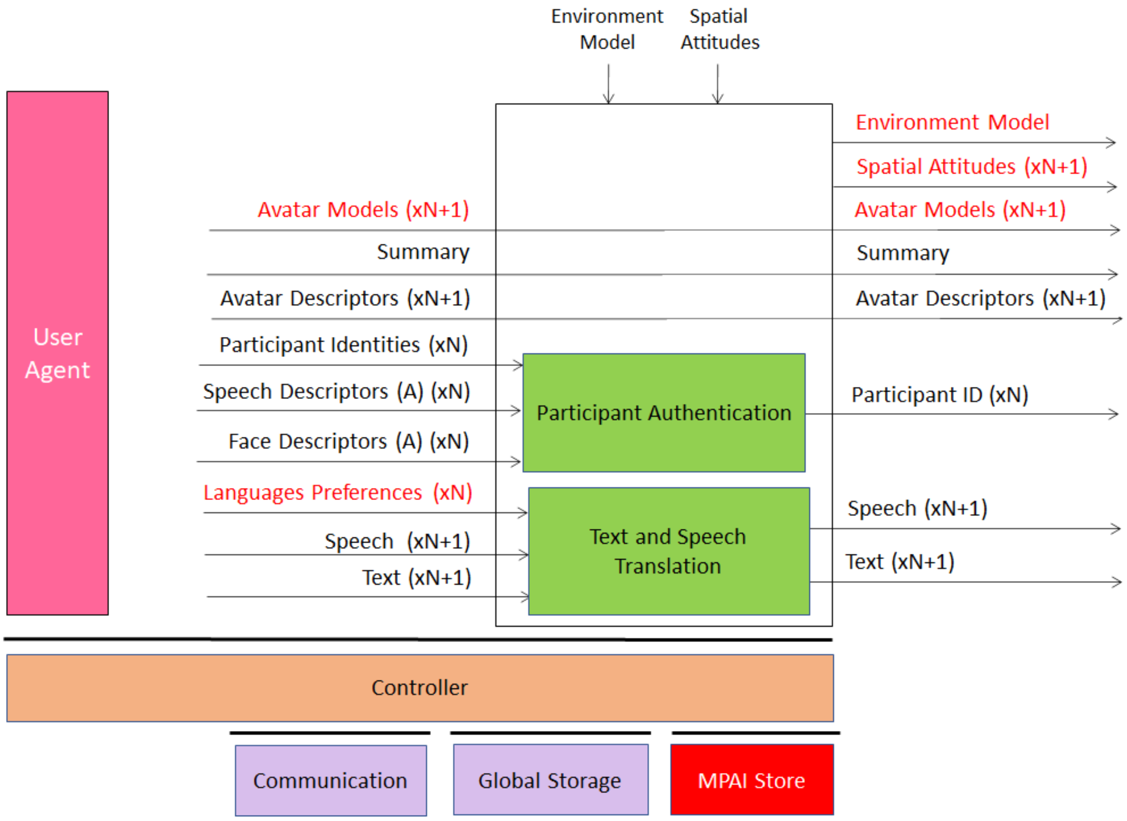


Figure 6 – Reference Model of Avatar-Based Videoconference Server

4.6.3 I/O Data of Server

Table 7 gives the input and output data of Server AIW.

Table 7 – Input and output data of Server AIW

Input	Comments
Participant Identities (xN)	Assigned by Conference Manager
Speech Descriptors (A) (xN)	Participant’s Speech Descriptors for Authentication
Face Descriptors (A) (xN)	Participant’s Face Descriptors for Authentication
Selected Languages (xN)	From all participants
Speech (xN+1)	From all participants and Virtual Secretary
Text (xN+1)	From all participants and Virtual Secretary
Avatar Model (xN+1)	From all participants and Virtual Secretary
Compressed Avatar Descriptors (xN+1)	From all participants and Virtual Secretary
Summary	From Virtual Secretary
Outputs	**Comments**
Environment Model	From Server Manager
Avatar Model (xN+1)	From all participants and Virtual Secretary
Compressed Avatar Descriptors (xN+1)	Participants + Virtual Secretary Compressed Avatar D.
Participant ID (xN+1)	Participants + Virtual Secretary IDs
Speech (xN+1)	Participants + Virtual Secretary Speech
Text (xN+1)	Participants + Virtual Secretary Speech

4.7 Avatar-Based Videoconference – Client (Receiving side)

Participants in Avatar-Based Videoconference are avatars realistically impersonating human participants at remote locations. See Chapter 5 of Annex 1 - MPAI Basics for more information on the Avatar-Based Videoconference Use Case. This is fully specified in [3].

4.7.1 Functions of Client (Receiving side)

The Function of the Client (Receiving Side) is to:

1. Create the Environment using the Environment Model.
2. Decompress Avatar Descriptors.
3. Place and animate the Avatar Models at their Spatial Attitudes.
4. Add the relevant Speech to each Avatar.
5. Present the Audio-Visual Scene as seen from the participant-selected Point of View.

4.7.2 Reference Architecture of Client (Receiving side)

The Receiving Client:

1. Creates the AV Scene using:
1.1. The Environment Model.
1.2. The Avatar Models and Compressed Avatar Descriptors.
1.3. The Speech of each Avatar.
2. Presents the Audio-Visual Scene based on the selected viewpoint in the Environment.

Figure 6 gives the architecture of Client Receiving AIW. Red text refers to data sent at the meeting start.

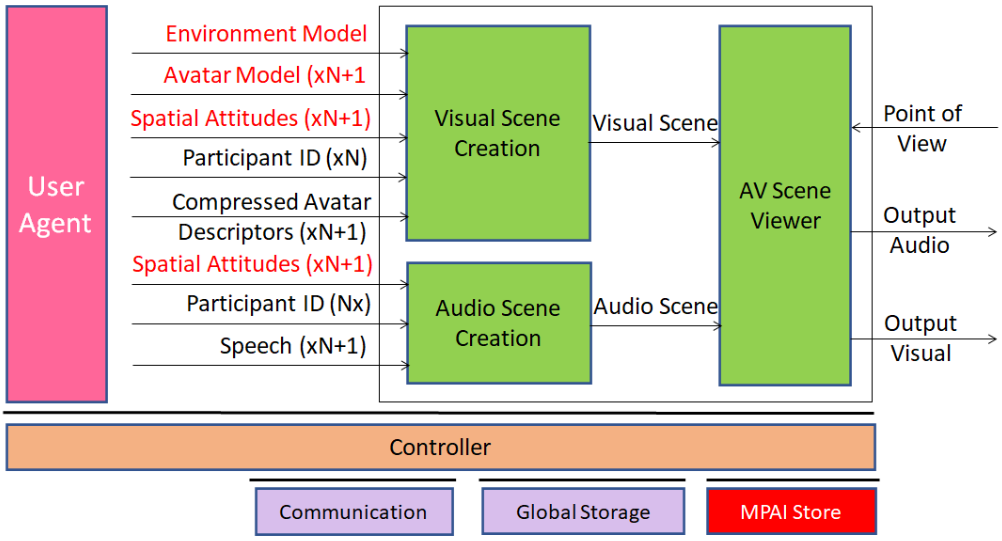


Figure 7 – Reference Model of Avatar-Based Videoconference Client (Receiving Side)

An implementation may decide to display text with the visual image for accessibility purposes.

4.7.3 I/O Data of Client (Receiving side)

Table 8 gives the input and output data of Client (Receiving Side) AIW.

Table 8 – Input and output data of Client (Receiving Side) AIW

Input	Comments
Point of View	Participant-selected point to see visual objects and hear audio objects in the Virtual Environment.

Spatial Attitudes (xN+1)	Avatars' Positions and Orientations in Environment.
Participant IDs (xN)	Unique Participants' IDs
Speech (xN+1)	Participant's Speech (e.g., translated).
Environment Model	Environment Model.
Compressed Avatar Descriptors (xN+1)	Descriptors of animated Avatars.
Output	**Comments**
Output Audio	Presented using loudspeaker (array)/earphones.
Output Visual	Presented using 2D or 3D display.

4.8 Spatial Object Identification (SOI)

4.8.1 Scope of the AIM

The purpose of the Spatial Object Identification AIM is to provide the Identifier of a Physical Object in an Environment with a plurality of Objects. The human indicates the intended object by pointing at it with a finger.

4.8.2 Reference Architecture

Figure 8 depicts the AIM implementing the Spatial Object Identification AIM.

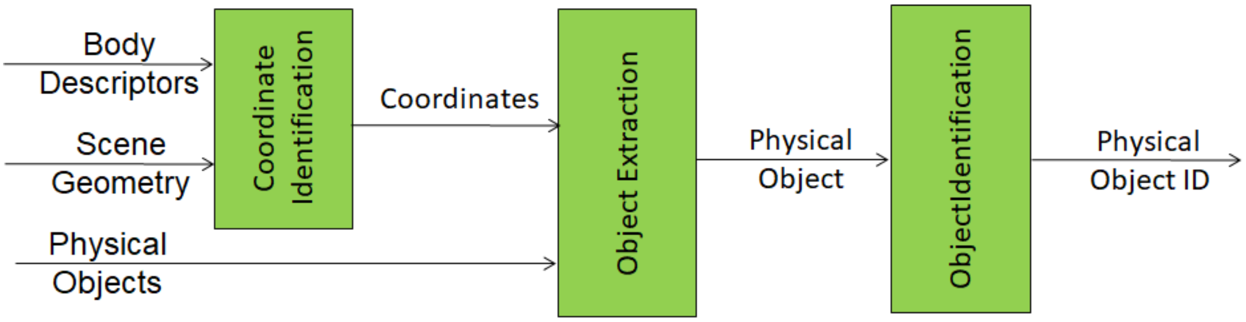


Figure 8 – Reference Model of the Spatial Object Identification AIM

4.8.3 Input/output data

Table 9 gives the input/output data of Spatial Object Identification.

Table 9 – I/O data of Spatial Object Identification

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

5 AI Modules

5.1 Visual Scene Description

5.1.1 Scope of the AIM

The scope of the Visual Scene Description AIM is to

1. Acquire a visual scene.
2. Provide the following output:
   a. The Face Descriptors of a human in the scene.
   b. The Body Descriptors of a human in the scene.
   c. The Physical Objects in the scene.
   d. The Scene Geometry.

5.1.2 Reference Architecture

Figure 9 depicts the AIM implementing the Visual Scene Description AIM.

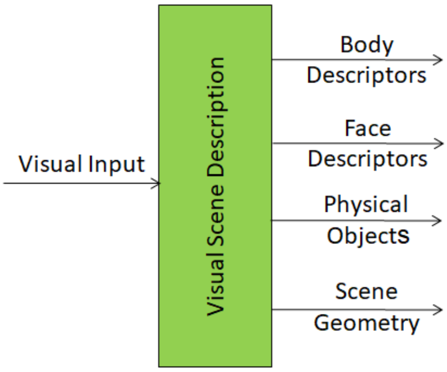


Figure 9 - Reference Model of the Visual Scene Description AIM

5.1.3 Input/output data

*Table 10*gives the input/output data of Spatial Object Identification.

Table 10 – I/O data of Spatial Object Identification

Input data	**From**	**Comment**
Visual Scene	Another AIM or a Device	
Output data	**To**	**Comments**
Body Descriptors	Downstream AIM	Interprets Body Descriptors
Face Descriptors	Downstream AIM	Interprets Face Descriptors
Physical Objects	Downstream AIM	Identifies Object
Scene Geometry	Downstream AIM	Used to localise human and objects

6 Data Formats

Table 11 provides the list of Data Formats target of the Call for Technologies.

Table 11 – Data formats

Name of Data Format	**Subsection**	**Use Case**
Virtual Environment	6.1	ARA-ABV
		MPAI-MMM
Coordinates, Angles, and Objects	6.2	ARA-ABV
		MMC-CAS
		MMC-HCI
		MPAI-CAV
		MPAI-MMM

Spatial Attitude and Point of View	6.3	ARA-ABV MMC-CAS MMC-HCI MPAI-CAV MPAI-MMM
Audio Scene Descriptors	6.4	ARA-ABV MMC-HCI MPAI-CAV MPAI-MMM
Visual Scene Descriptors	6.5	ARA-ABV MMC-CAS MMC-HCI MPAI-CAV MPAI-MMM MMC-HCI

The following Sections provide an initial specification of the data formats.

6.1 Virtual Environment

A Virtual Environment represents:

1. A bounded or unbounded space, e.g., a room, a square surrounded by buildings, an open space, etc.
2. The objects (e.g., table and chairs).

A Virtal Environment is represented according to the glTF syntax.

6.2 Coordinates, Angles, and Objects

A Capture Device, e.g., Audio, Visual such as camera or LiDAR placed in an Environment has 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. **Error! Reference source not found.** depicts the r elationship between the (x.y,z) and (r, φ, θ) coordinates.

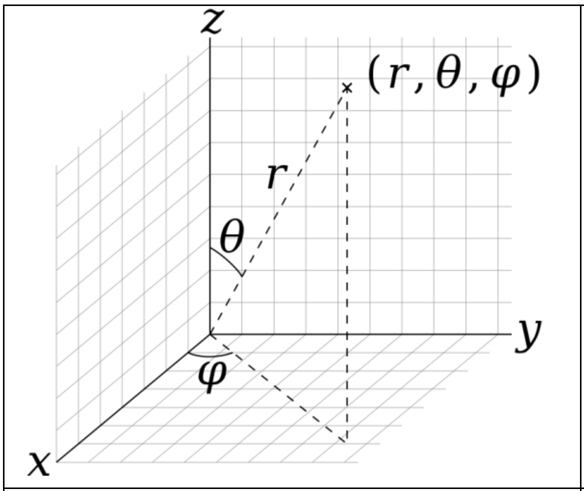
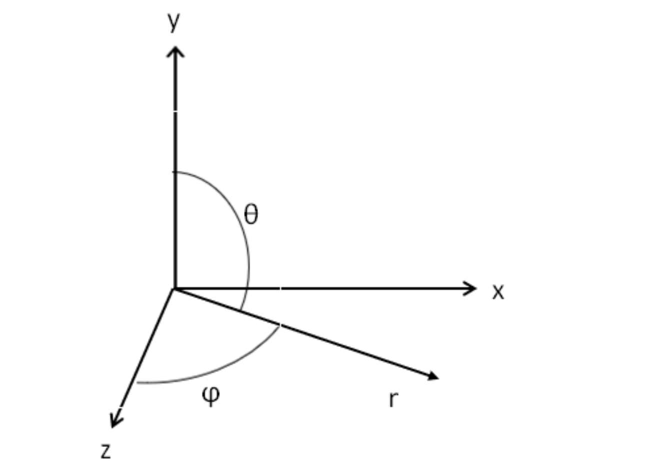
	

Figure 10 – Normal spatial coordinates	*Figure 11 - Spatial coordinates of capture device*

An object placed on a plane has:

- x as its principal axis parallel to the plane.

- y as the axis perpendicular to the x axis and to its left).
- z as the axis perpendicular to the plane pointing in the direction of the object.

The object can rotate around any of its three axes with the conventions of Table 12:

Table 12 - Axes and Angles

Axis	Angle	Name
x	φ	Roll
y	θ	Pitch
z	ψ	Yaw

Figure 12 and *Figure 13* graphically represents the elements of Table 12.

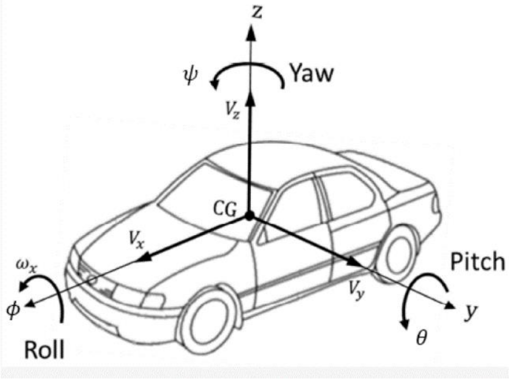


Figure 12 – Axes and Angles of a car

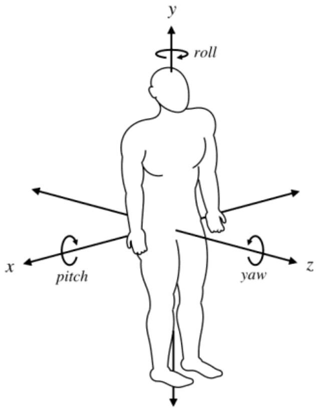


Figure 13 – Axes and Angles of a human

6.3 Spatial Attitude and Point of View

Spatial Attitude is a real vector Position and Orientation, and their Velocities and Accelerations, in the following order:

Table 13 – Components of Spatial Attitude

PosX	PosY	PosZ
OrientX	OrientY	OrientZ
PosXVel	PosYVel	PosZVel
OrientXVel	OrientYVel	OrientZVel
PosXAcc	PosYAcc	PosZAcc
OrientXAcc	OrientYAcc	OrientZAcc

The vector including only Position and Orientation is called Point of View.

Table 14 – Components of Point of View

PosX	PosY	PosZ
OrientX	OrientY	OrientZ

6.4 Audio Scene Descriptors

Table 15 – Audio Scene Descriptors

Variable name	Comments
Timestamp type	0: Absolute Time; 1: Relative Time
Timestamp value	In seconds
Space type	0: Absolute Space (from 0,0.0); 1: Relative Space
Coordinate system	0: Cartesian; 1: Polar
Space value	In meters; meters and degrees
Number of audio objects	Integer
Spatial Attitude1	MPAI-OSD
Audio Object1	MPAI-CAE
Spatial Attitude2	MPAI-OSD
Audio Object2	MPAI-CAE
…	

6.5 Visual Scene Descriptors

Visual Scene Descriptors are provided by the Visual Scene Description AIM (see Figure 9) contained in a real vector with the following structure of Table 16.

Table 16 – Visual Scene Descriptors

Variable name	
Timestamp type	0: Absolute Time; 1: Relative Time
Timestamp value	In seconds
Space type	0: Absolute Space (from 0,0.0); 1: Relative Space
Coordinate system	0: Cartesian; 1: Spherical
Space value	In meters
Number of humans	Integer
Spatial Attitude1	MPAI-OSD
Human1 Body Descriptors	MPAI-ARA
Human1 Face Descriptors	MPAI-ARA
Spatial Attitude2	MPAI-OSD
HumanID2 Body Descriptors	MPAI-ARA
HumanID2 Face Descriptors	MPAI-ARA
…	
Capture technology	0: mesh; 1:
Number of Objects	Integer
Spatial Attitude1	MPAI-OSD
Physical Object1	3D mesh?
Spatial Attitude1	MPAI-OSD
Physical Object2	3D mesh?
…	

6.6 Audio-Visual Scene Description

Audio, Body, Face, Objects are optional.

Variable name	

Timestamp type	0: Absolute Time; 1: Relative Time
Timestamp value	In seconds
Space type	0: Absolute Space (from 0,0.0); 1: Relative Space
Coordinate system	0: Cartesian; 1: Radial
Space value	In meters
Number of human AV Objects	Integer
Spatial Attitude$_1$	MPAI-OSD
Audio objectID$_1$	string
Body DescriptorsID$_1$	string
Face DescriptorsID1	string
Spatial Attitude$_2$	MPAI-OSD
Audio objectID2$_2$	string
Body DescriptorsID$_2$	string
Face DescriptorsID$_2$	string
…	
Number of non-human AV Objects	Integer
Spatial Attitude$_A$	MPAI-OSD
Audio ObjectID$_A$	string
Physical Object$_A$	string
Spatial Attitude$_B$	MPAI-OSD
Audio ObjectID$_B$	string
Physical Object$_B$	string
…	

Objects not having both an audio and a visual component are not recorded in the Audio-Visual Scene Description

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

The technical foundations of the MPAI Ecosystem are currently provided by the following documents developed and maintained by MPAI:

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 14 depicts the MPAI ecosystem operation for conforming MPAI implementations.

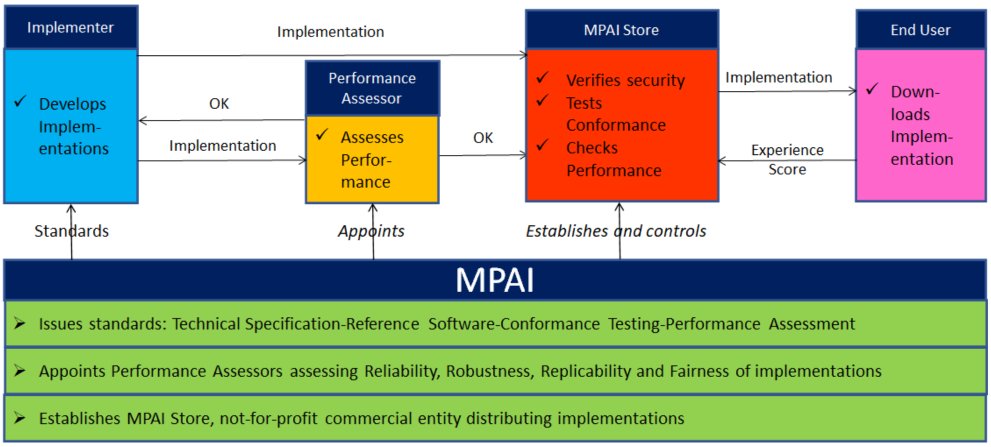

Implementer
✓ Develops Implem-entations
Implementation
OK
Implementation
Performance Assessor
✓ Assesses Perfor-mance
OK
MPAI Store
✓ Verifies security
✓ Tests Conformance
✓ Checks Performance
Implementation
Experience Score
End User
✓ Down-loads Implem-entation
Standards
Appoints
Establishes and controls
MPAI
Issues standards: Technical Specification-Reference Software-Conformance Testing-Performance Assessment
Appoints Performance Assessors assessing Reliability, Robustness, Replicability and Fairness of implementations
Establishes MPAI Store, not-for-profit commercial entity distributing implementations

Figure 14 – The MPAI ecosystem operation

Technical Specification: Governance of the MPAI Ecosystem Table 17 identifies the following roles in the MPAI Ecosystem:

Table 17 - 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 15 depicts the MPAI-AIF Reference Model under which Implementations of MPAI Application Standards and user-defined MPAI-AIF Conforming applications operate [5].

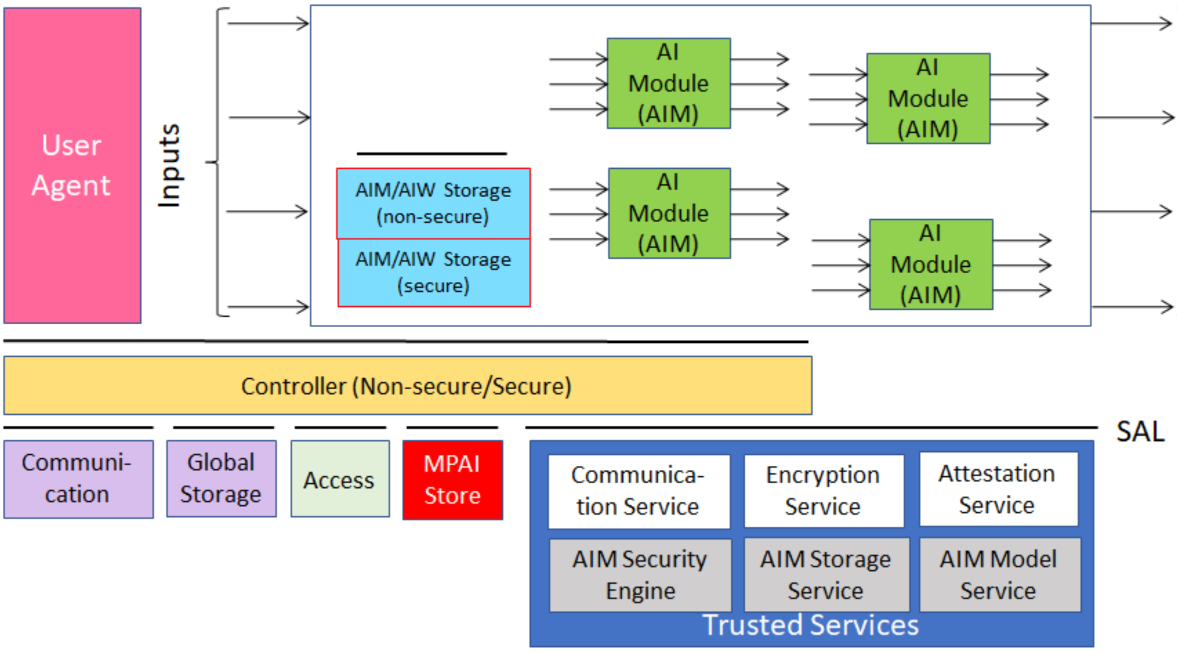


Figure 15 – The AI Framework (AIF) Reference Model

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

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 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 18 gives the characteristics of an AIW and its AIMs of a given Level:

Table 18 - 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-Visual Scene Description

The ability to describe (i.e., digitally represent) an audio-visual scene is a key requirement of several MPAI Technical Specifications and Use Cases. MPAI has developed Technical Specification: Context-based Audio Enhancement (MPAI-CAE) [3] that includes Audio Scene Descriptors and uses a subset of Graphics Language Transmission Format (glTF) [10] to describe a visual scene.

4.1 Audio Scene Descriptors

Audio Scene Description is a Composite AI Module (AIM) specified by Technical Specification: Context-based Audio Enhancement (MPAI-CAE) [3]. The position of an Audio Object is defined by Azimuth, Elevation, Distance.

The Composite AIM and its composing AIMs are depicted in Figure 21.

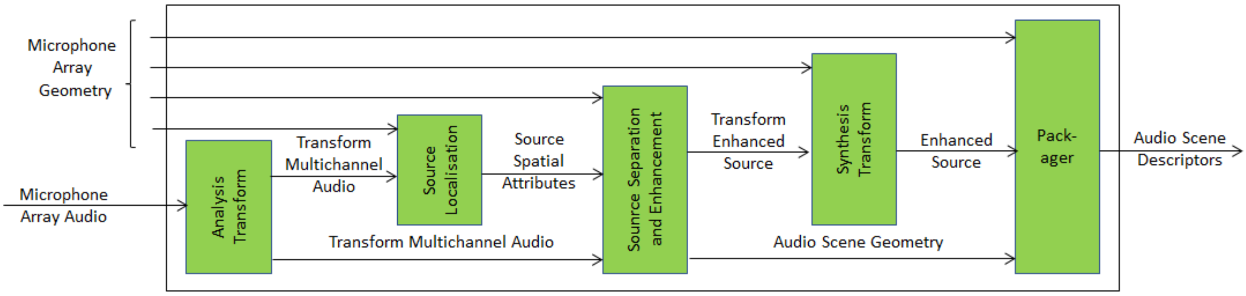

Microphone Array Geometry
Microphone Array Audio
Analysis Transform
Transform Multichannel Audio
Source Localisation
Source Spatial Attributes
Transform Multichannel Audio
Sounrce Separation and Enhancement
Transform Enhanced Source
Synthesis Transform
Enhanced Source
Audio Scene Geometry
Pack-ager
Audio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

Figure 16 - The Audio Scene Description Composite AIM

4.2 Visual Scene Descriptors

MPAI uses a subset of Graphics Language Transmission Format (glTF) [10] to describe a visual scene.

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. Figure 17 depicts the components of the system supporting the conference of a group of humans participating through avatars having their visual appearance and uttering the participants' real voice.

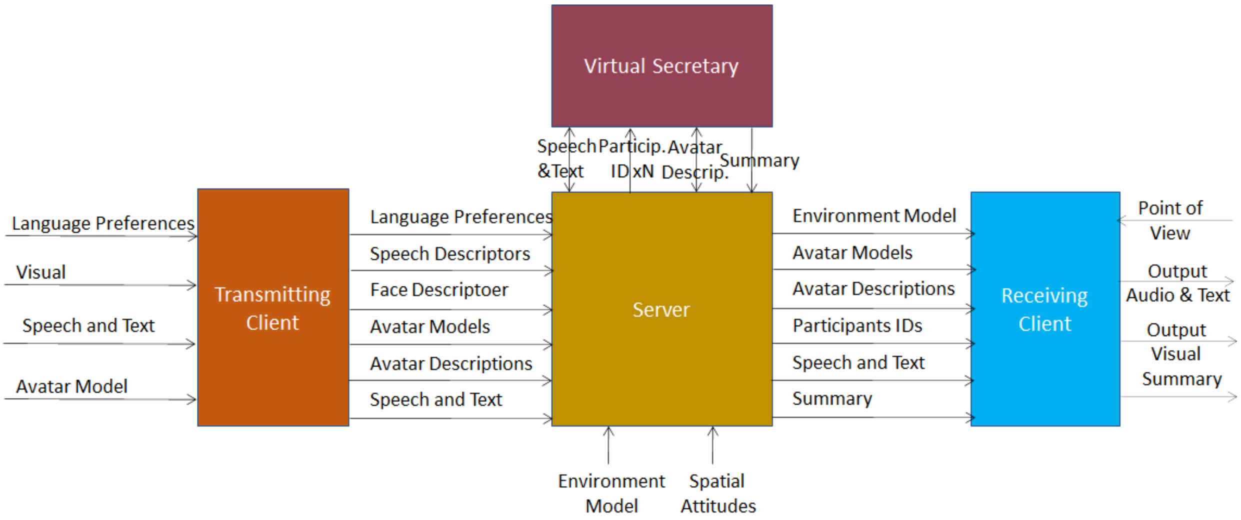


Figure 17 – Avatar-Based Videoconference end-to-end diagram

Figure 18 contains the reference architectures of the four AW Workflows constituting the Avatar-Based Videoconference: Client (Transmission side), Server, Virtual Secretary, and Client (Receiving side).

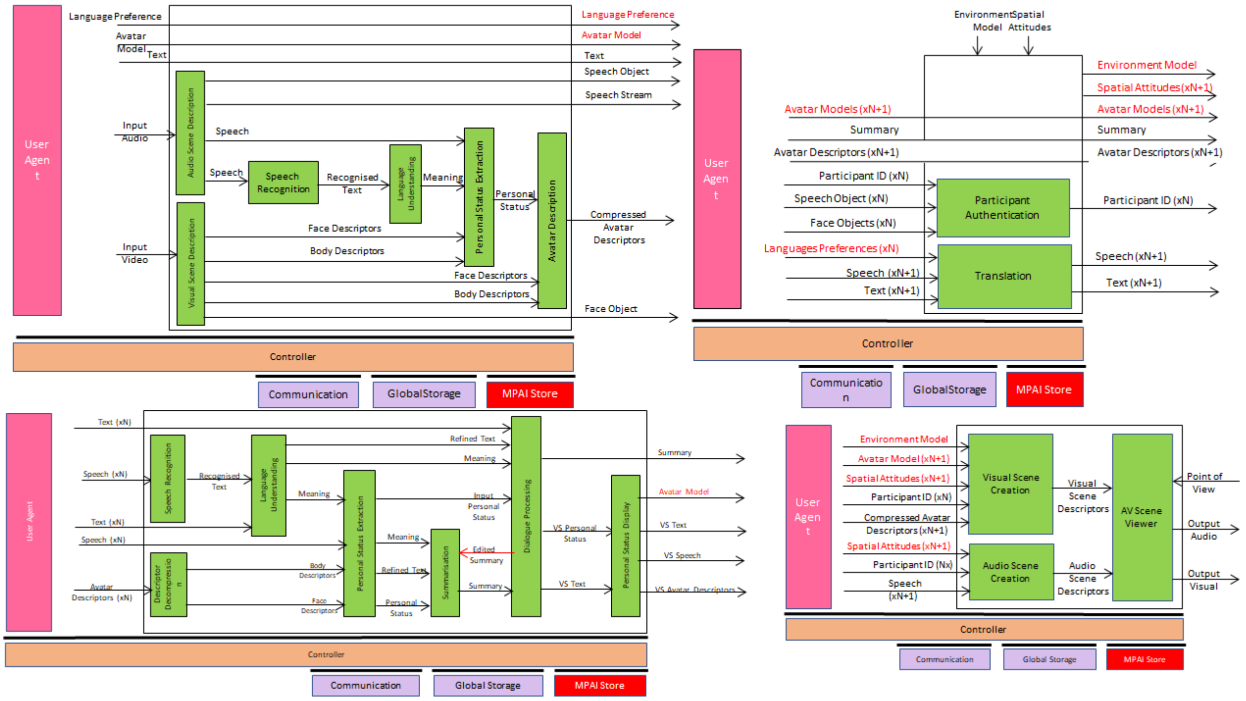


Figure 18 - The AIWs of Avatar-Based Videoconference

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 19*.
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.

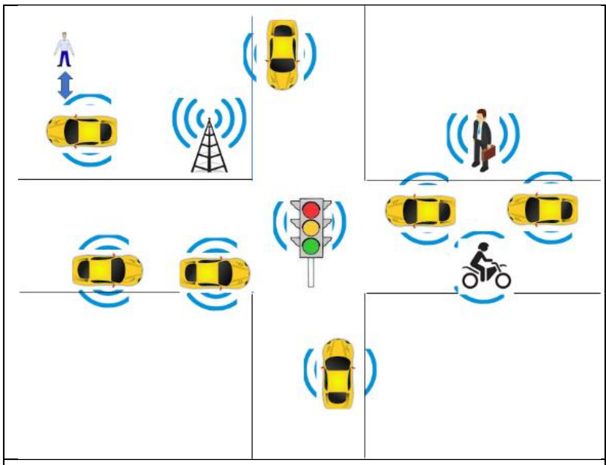


Figure 19 - An environment of CAV operation

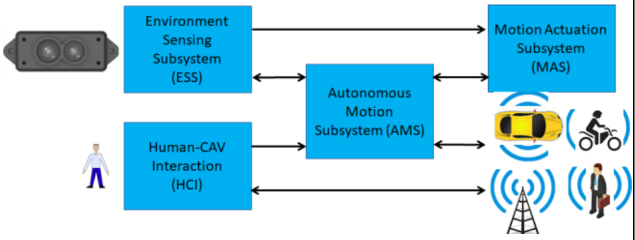

Environment Sensing Subsystem (ESS)
Motion Actuation Subsystem (MAS)
Autonomous Motion Subsystem (AMS)
Human-CAV Interaction (HCI)

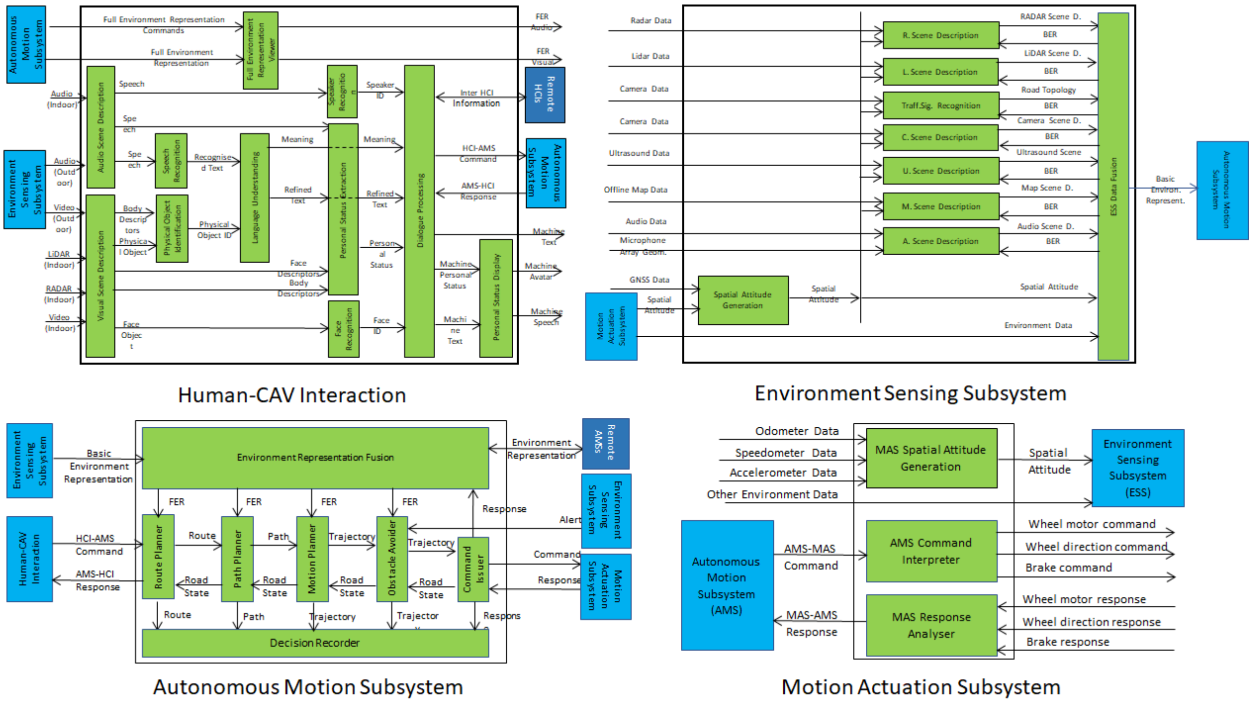
Figure 20 – The MPAI-CAV subsystems

MPAI believes in the capability of standards to accelerate the creation of a global competitive CAV market and has published Technical Specification:f Connected Autonomous Vehicle (MPAI-CAV) – Architecture that includes (see *Figure 20*):

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 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. Subsequently, Functional Requirements of the Data exchanged.
6. Eventually, standard technologies for the Data exchanged.



Human-CAV Interaction

Environment Sensing Subsystem

Autonomous Motion Subsystem

Motion Actuation Subsystem

Figure 21 - The MPAI-CAV Subsystems with their Components

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

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 19.

Table 19 - 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 AIMName (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.