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

|  |
| --- |
| **Public Document** |

|  |  |
| --- | --- |
| **N412** | 2021/10/24 |
| **Source** | Requirements (MCS) |
| **Title** | MPAI-MCS Use Cases and Functional Requirements WD0.2.3.1 |
| **Target** | MPAI Community |

[1 Introduction 2](#_Toc86144592)

[2 Scope of proposed standard 3](#_Toc86144593)

[2.1 General 3](#_Toc86144594)

[2.2 MCS Features 4](#_Toc86144595)

[2.3 Technology support 4](#_Toc86144596)

[2.4 Virtual e-learning (VEL) 5](#_Toc86144597)

[2.4.1 Description 5](#_Toc86144598)

[2.4.2 Steps 5](#_Toc86144599)

[2.5 Local Avatar Videoconference (LAV) 6](#_Toc86144600)

[2.5.1 Description 6](#_Toc86144601)

[2.5.2 Steps 7](#_Toc86144602)

[2.6 Use Case #3 – Teleconsulting 8](#_Toc86144603)

[2.7 Use Case #4 – Multipoint videoconference 8](#_Toc86144604)

[3 Terms and Definitions 9](#_Toc86144605)

[4 References 9](#_Toc86144606)

[4.1 Normative references 9](#_Toc86144607)

[4.2 Informative references 10](#_Toc86144608)

[5 Use Cases 10](#_Toc86144609)

[5.1 Virtual e-Learning 10](#_Toc86144610)

[5.1.1 AIW of “Client TX” 10](#_Toc86144611)

[5.1.2 AIW of Server 12](#_Toc86144612)

[5.1.3 AIW of “VEL Client RX” 14](#_Toc86144613)

[5.1.4 AI Modules 15](#_Toc86144614)

[5.2 Use Case #2 – Local Avatar Videoconference (LAV) 16](#_Toc86144615)

[5.2.1 AIW of “Participant TX” 16](#_Toc86144616)

[5.2.2 AIW of Server 18](#_Toc86144617)

[5.2.3 AIW of “Participant RX” 21](#_Toc86144618)

[5.2.4 AI Modules 22](#_Toc86144619)

[6 AI Modules 22](#_Toc86144620)

[6.1 AIMs and their data 22](#_Toc86144621)

[6.1.1 Participant TX 22](#_Toc86144622)

[6.1.2 Server 22](#_Toc86144623)

[6.1.3 Participant RX 23](#_Toc86144624)

[6.2 Data Formats 23](#_Toc86144625)

[6.2.1 # of Participants 24](#_Toc86144626)

[6.2.2 3D Audio Navigation Info 24](#_Toc86144627)

[6.2.3 3D Audio Scene (O) 24](#_Toc86144628)

[6.2.4 3D AV Object 24](#_Toc86144629)

[6.2.5 3D AV Object Commands 24](#_Toc86144630)

[6.2.6 3D Video Scene (O) 24](#_Toc86144631)

[6.2.7 3D Visual Ambient Descriptors 24](#_Toc86144632)

[6.2.8 Ambient Audio 24](#_Toc86144633)

[6.2.9 Ambient Type 24](#_Toc86144634)

[6.2.10 Avatar Model 24](#_Toc86144635)

[6.2.11 Coded Message 24](#_Toc86144636)

[6.2.12 Emotion (Speech) 25](#_Toc86144637)

[6.2.13 Emotion (Video) 25](#_Toc86144638)

[6.2.14 Encoded Speech 25](#_Toc86144639)

[6.2.15 Face Descriptors 25](#_Toc86144640)

[6.2.16 Final Emotion 25](#_Toc86144641)

[6.2.17 Final Meanings 25](#_Toc86144642)

[6.2.18 Gesture Descriptors 25](#_Toc86144643)

[6.2.19 Head Descriptors 25](#_Toc86144644)

[6.2.20 Head Model 25](#_Toc86144645)

[6.2.21 ID’d Avatars Descriptors 25](#_Toc86144646)

[6.2.22 ID’d Decoded Speech 25](#_Toc86144647)

[6.2.23 Input Video 25](#_Toc86144648)

[6.2.24 Meaning (Speech) 26](#_Toc86144649)

[6.2.25 Meaning (Video) 26](#_Toc86144650)

[6.2.26 Output Audio Scene 26](#_Toc86144651)

[6.2.27 Output Visual Scene 26](#_Toc86144652)

[6.2.28 Participant IDs 26](#_Toc86144653)

[6.2.29 Recognised Text 26](#_Toc86144654)

[6.2.30 Separated Speech 26](#_Toc86144655)

[6.2.31 Speaker Descriptors 26](#_Toc86144656)

[6.2.32 Visual Navigation 26](#_Toc86144657)

[Annex 1 – MPAI-wide terms and definitions (Normative) 27](#_Toc86144658)

[Annex 2 - Notices and Disclaimers Concerning MPAI Standards (Informative) 30](#_Toc86144659)

[Annex 3 – The Governance of the MPAI Ecosystem (Informative) 32](#_Toc86144660)

# Introduction

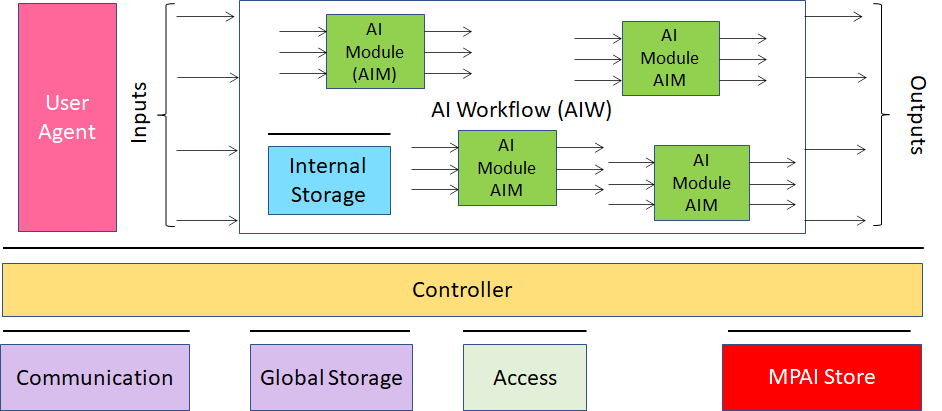
Moving Picture, Audio and Data Coding by Artificial Intelligence (MPAI) is an [international Standards Developing Organisation](http://mpai.community/) with the mission to develop *AI-enabled data coding standards*. Research has shown that data coding with AI-based technologies is generally *more efficient* than with existing technologies. Compression and feature-based description are notable examples of coding.

In the following, Terms beginning with a capital letter are defined in *Table 1* if they are specific to MPAI-MCS Standard and to *Table 18* if they are common to all MPAI Standards.

MPAI Application Standards enable the development of AI-based products, applications and services. The MPAI AI Framework (AIF) Standard (MPA-AIF) [2] provides the foundation on which the technologies defined by MPAI Application Standards operate.

*Figure 1* depicts the MPAI-AIF Reference Model. MPAI-AIF provides the foundation on which Implementations of MPAI Application Stan­dards operate.

An AIF Implementation allows execution of AI Workflows (AIW), composed by basic processing elements called AI Modules (AIM).



*Figure 1 – The AI Framework (AIF) Reference Model and its Components*

MPAI Application Standards normatively specify Semantics and Format 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.

In particular, an AIM is defined by its Function and Data, but not by its internal architecture, which may be based on AI or data processing, and implemented in software, hardware or hybrid software and hardware technologies.

MPAI defines Interoperability as the ability to replace an AIW or an AIM Implementation with a functionally equivalent AIW or AIM Implementation. An AIW *executed in an AIF* may have one of the following MPAI-defined Interoperability Levels:

1. Interoperability Level 1, if the AIW is proprietary and composed of AIM with proprietary functions using any proprietary or standard data Format.
2. Interoperability Level 2, if the AIW is composed of AIMs having all their Functions, Formats and Connections specified by an MPAI Application Standard.
3. Interoperability Level 3, if the AIW has Interoperability Level 2, and the AIW and its AIMs are certfified by an MPAI-appointed Assessor to hold the attributes of Rel­iability, Robustness, Replicability and Fairness – collectively called Performance (*Level 3*).

MPAI is the root of trust of the MPAI Ecosystem [1] offering Users access to the promised benefits of AI with a guarantee of increased transparency, trust and reliability as the Interoperability Level of an Implementation moves from 1 to 3.

# Scope of proposed standard

## General

*Mixed-reality Collaborative Spaces* (MPAI-MCS) is an MPAI standard project containing Use Cases, AI Modules, AI Workflows and Data Formats supporting scenarios where geographically separated Humans collaborate in real time with Avatars in a virtual-reality space to achieve goals generally defined by the Use Case and specifically carried out by Humans and Avatars.

The structure of this document is:

|  |  |
| --- | --- |
| Chpter 2 | Defines the characteristics of Mixed-reality Collaborative Spaces.  Describes the currently supported Use Cases. |
| Chapter 3 | Defines the Terms used in this document. |
| Chapter 4 | Lists normative and informative references. |
| Chapter 5 | Identifies and describes the AI Workflows implementing the Use Cases |
| Chapter 6 | Identifies and describes   1. AI Modules (Section 6.1) 2. AIMs’ Data Formats (Section 6.2). |

It is expected that, once the MPAI-MCS standardisation process has read the appropriate stage, this document will become an attachment to a future MPAI-MCS Call for Technologies requesting technologies conforming to the requirements specified in Section 6.2.

## MCS Features

The MPAI MCS Use Cases share the following features:

1. The MCS virtual spaces, called Ambients, are 3D graphical spaces representing an actual, realistic or fictitious physical space with specified affordance.
2. Ambients are populated by 3D objects representing actual, realistic or fictitious Visual Objects with specified affordance and Audio Objects propagating according to the specific object affor­dances.
3. Avatars move around in Ambients, express emotions and perform gestures corresponding to actual, realistic or fictitious Humans.
4. Humans generate media information captured by devices called MCS TX Clients generating and transmitting coded representations of Text, Audio and Video.
5. Physical spaces are sensed by different types of sensors
   1. Audio
   2. Visual
   3. Kinetic tracker
   4. Haptic
6. MCSs are typically supported by MCS Servers whose goal is to create representations of MCSs and their components and distribute them to MCS RX Clients.
7. MCS Servers may create and send packaged digital representations of the MCSs or they just create and send descriptions to be composed by MCS RX Clients according to the needs of the receiving human.
8. Ambients are populated by 3D Audio-Visual Objects generated by a possibly human-animated file or by a device operating in real time.
9. Humans may act on 3D Audio-Visual Objects performing such actions as:
   1. Manual or automatically define a portion of the 3D AV object.
   2. Count objects per assigned volume size.
   3. Detect structures in a (portion of) the 3D AV object.
   4. Deform/sculpt the 3D AV object.
   5. Combine 3D AV objects.
   6. Call an anomaly detector on a portion with a criterion.
   7. Follow a link to another portion of the object.
   8. 3D print (portions of) the 3D AV object.
10. Humans may create and attach metadata to the 3D AV object:
    1. Define a portion of the object – manual or automatic.
    2. Assign physical properties to (different parts) of the 3D AV object.
    3. Annotate a portion of the 3D AV object.
    4. Create links between different parts of the 3D AV object.

We assume that 3D AV Objects have a standard format (e.g., glTF) at least for the purpose of acting on the Object.

## Technology support

The state of the art of technologies and standards required to implement the use cases are:

1. Software such as Microsoft Mesh, Cesium or Teleport may be used to develop the visual part of Ambients.
2. What about the Audio part?
3. Software to animate the face – lips, eyes, muscles – of an avatar [].
4. Software to generate gesture description for sign language (country/language dependent)

## Virtual e-learning (VEL)

### Description

In the Virtual E-Learning (VEL) Use Case, a teacher holds a lecture to N students. The teacher and the students attending the lecture – called Participants – reside at their own locations each having a microphone set and a camera set in sufficient numbers and features to support the requirements described in the following. They can also activate devices capable to produce 3D AV Objects to be shared, or to retrieve and share 3D AV Objects.

The school or cultural institution under whose aegis the lecture is held (Hosting Organisation) runs a VEL Server capable to provide Ambients populated by speaking avatars representing the Participants arranged in different styles, e.g., classroom or other. Avatars can make limited movements in the Ambient technically implemented as Teleportation.

### Steps

The activities in MCS-VEL unfold as follows:

1. Participants provide and communicate their own personae (Avatar models) (or select from a list of models) to the VEL Server:
   1. Avatar model capabilities and objects with their affordances.
   2. Initial position in the MCS-VEL space.
   3. Colour and style of synthetic voice (e.g., used in speech translation).
   4. Participants’ spoken language preferences (e.g., EN-US, IT-CH).
2. The VEL Server makes available:
   1. *Ambients* arranged as:
      1. Classroom style.
      2. An evocative place, e.g., the Stoa of Athens, or a chemistry laboratory.
   2. *Other visual objects*, e.g., furnishings of the Ambient.
3. The VEL Server can
   1. Convert:
      1. Speech, text and gesture; e.g., speech to gesture, text to gesture etc.
      2. Input speech from the language of a speaker to the languages of the intended listeners.
   2. Position Audio Objects using spatial Ambisonic audio:
      1. Participants’ speeches at the intended spatial locations.
      2. Audio objects of a 3D AV Object at the intended spatial locations.
4. During the lecture:
   1. The camera sset of each participant extracts and sends to VEL server:
      1. Head: movement.
      2. Face: lips, eyes.
      3. Torso.
      4. Arms and hands.
   2. The microphone set of each participant
      1. Captures the 3D Audio field of the participant’s room.
      2. Separates the speech from the rest of the 3D Audio field
      3. Cancels the participant’s voice received back from the server.
      4. Extracts Speech Descriptors.
      5. Sends own 3D Audio field, separated speech, speech descriptors to VEL Server.
   3. Participants issue teleportation commands.
   4. The VEL Server:
      1. Combines head motions, facial features, hand gestures and speech descriptors received from the individual VEL TX Clients.
      2. Translates the Speech of a Participant to the specific languages of other Partic­ipants.
   5. The teacher
      1. Locates and connects devices capable to produce 3D AV Objects []
      2. Sends 3D AV Objects generated by devices
         1. 3D Visual Output of a microscope.
         2. Molecules captured as 3D objects by an electronic microscope.
      3. Calls synthetic 3D AV Objects from a DB in support of the lecture, e.g.,:
         1. 3D model of the brain of a mouse.
      4. Enter, navigate and act on 3D audio-visual objects (Blender?)
      5. Starts an experiment, e.g.:
         1. The principles of optics.
         2. Gravity and its effects.
         3. The docking of molecules in chemistry.
         4. The inside of an atom.
      6. Places physical (moving) objects on their desk for reproduction as 3D objects at participants’ locations and interactive engagement.
      7. Acts on the object by using the teacher’s own sensing and actuating devices.

## Local Avatar Videoconference (LAV)

### Description

Today’s videoconference falls short from a satisfactory supplement to a physical meeting. Meeting participants are able to hear the voice selected by the videoconference server and see the full screen face of a participant who is speaking but cannot have similar details for other speakers at the same time. Participants can hear the voice of the speaker, but are generally unable to hear what other participants are seeing and are unable to have an audio-visual experience of the participants that is comparable to the experience they would have at a physical meeting. In particular, today’s experi­ence is regularly chopped up in separate moments where one participant talks and the others listen, unlike a real meeting which is a collective experience where one participant may well speak but the others have reactions, not necessarily vocal. Chat between participants plays a role, but it is far from the experience of a real meeting, where the collections of all interactions and their fusions by individual participantsis what makes a physical meeting irreplaceable.

At a typical meeting, people stay at a table – if that is the arrangements. They do not move and, if they do, it is for reasons that have not much to do with the meeting. Someone may be leaving, temporarily or permanently and someone may be joining. Coffee imay be brought in and people go to a table and have a coffee break. Today, a blackboard is rarely used, but a person may be standing near a screen and illustrate projected material or may – more likely – stay seated and use a laser pointer.

It is technologically still unfeasible to capture participants seating in their rooms and combine a virtual meeting room with the 3D representations of the participants in a visually satisfactory way. Technology makes it possible, however, to capture the main features of a human torso – face, lips, head, arms and hands – and use them to animate an avatar sitting at a table in a realistic way.

The Local Avatar Videoconference (MCS-LAV) Use Case is designed to offer conference participants the ability to enjoy the collective experience of a physical meeting represented by the following features of the participants: speech and facial expression, and movements of head, arms and hands.

The 3D scene is represented by descriptors which are sent to participants in lieu of a bandwidth consuming full 3D AV scene.

Depending on the specifics of the use case, the number of passive participants may be larger than the number of active participants. An example is a city hall meeting that is broadcasted to the citizenship who has the right to attend but not to speak.

### Steps

An MCS-LAV is attended by personae each representing a participants with:

1. Their actual speech.
2. Avatars having:
   1. Static bodies.
   2. Heads animated according to the participants’ head movement.
   3. Faces
      1. Animated by emotion and meaning extracted from the participants’ faces.
      2. Corroborated by emotions and meanings extracted from participants’ speeches.
   4. Hands animated according to participants’ gestures.
3. One or more devices capable to produce 3D AV Objects. We assume that the Objects have a standard format at least for the purpose of acting on them (e.g., present, rotate, select a sub-object etc.).

The AIW implementing this Use Case is distributed in the sense that there are:

1. N “Participant TX” AIW instances.
2. One Server AIW instance.
3. M “Participant RX” instances.

The LAV system is composed of

1. N “Participant TX” clients to
   1. Extract and send
      1. Speech, and its emotion and meaning.
      2. Emotion and meaning, and descriptors of face.
      3. Descriptors of arms and hands.
   2. Send special messages, e.g., I am leaving, I am joining, I am requesting the floor
   3. Locate and connect 3D AV Object-generating devices to the Participant TX.
   4. Send 3D AV Objects from above or from a database.
   5. Issue commands to act on 3D AV Objects.
2. An MCS-LAV server
   1. Adopts a meeting Ambient
   2. Describes the 3D visual scene by using all participants’ information and the visual component of the 3D AV Objects.
   3. Translates speeches from the language of the speakers of the individual languages of the speakers.
   4. Adds the 3D Audio component of the 3D AV Objects.
   5. Sends the descriptors of the 3D AV space to participants (not the full 3D AV scene).
3. M “Participant RX” clients to
   1. Create their *personal* 3D visual spaces by using:
      1. The (static) visual descriptors of the Ambient of the virtual meeting.
      2. The (dynamic) visual descriptors of the Avatars.
      3. The 3D Visual Object resulting from actions at their intended location.
   2. Create their *personal* 3D audio spaces by using
      1. The participants’ speeches located at the corresponding Avatars’ positions.
      2. The 3D Audio Object resulting from actions at their intended location.
   3. Navigate the resulting *personal* virtual 3D AV space (without moving their Avatar).
   4. Teleport their Avatars, e.g., to stand close to the 3D AV object.

## Use Case #3 – Teleconsulting

An entrepreneur (E) offers teleconsulting services on a class of objects of particularly difficult use. A Customer (C) contacts E for advice on how to use a particular machine.

This is how the envisaged MCS teleconsulting service can take place:

1. C contacts E
2. E requests C to provide a 3D scan of the object
3. C provides the requested scan
4. E starts its MCS composed by
   1. the virtual representation of the object placed, e.g., on a table, or movable
   2. the avatar of E sitting in front of the object
   3. the avatar of C sitting next to the avatar of E
5. While speaking, the avatar of E manipulates the object , e.g.,
   1. rotates it
   2. touches a particular point of the object
   3. uses a virtual tool to indicate a type of operation
6. C and E see their own and the other avatars’ actions as if they were sitting in the virtual position of the avatar
7. While speaking, C acts on the physical object and the actions are reflected on the avatar and the virtual object
8. Avatars can move around the object (e.g., in the case of a large object)

## Use Case #4 – Multipoint videoconference

The N participants in the conference reside at their locations, in their cultural environment. Their avatars sit around a virtual conference table located in a virtual room in an agreed cultural environment. A relevant quote is Marshall McLuhan’s “the medium is the message”.

This is how such a virtual shared-cultural conference could be managed:

1. The participants agree on and describe a shared cultural and/or context environment which can be real (representative of a physical space) or imagined (the components in the environment do not have a correspondence with the physical world):
   1. Conference style (board meeting, conference meeting, MPAI meeting etc.)
   2. Language that will be used in the shared space
   3. Room setting, furnishing, table and chairs, a CAV, outdoor
   4. Etc.
2. The organiser selects the multiconference service provider implementing the agreed setting
3. Participants provide/select and communicate to the multiconference service provider their own “personae”
   1. Avatar model
   2. Position in the meeting space
   3. Voice colour and style or own/synthetic
   4. Spoken language preference (e.g., EN-US, IT-CH) of the persona
4. Participant ensures that their own personae are authenticated
5. During the conference
   1. The camera of each participant
      1. Detects the participant’s body movements and extracts facial features and hand gestures
      2. Sends body movements and facial features to the multiconfer­ence unit
   2. The microphone set of a participant
      1. Captures the 3D field of the participant’s environment
      2. Separates the voice from the rest of the sound field
      3. Extracts and sends the sound field with descriptors of the speech
      4. Displays a choice of which sound field components should be preserved
   3. The multiconference unit
      1. Animates avatars at their assigned position using their body motions, facial features, hand gestures and speech descriptors
      2. Translates the cultural/context setting (speech etc.) of a participant to the agreed common setting
      3. Merges and sends to participants all sound fields as specified by each participant
      4. Sends participants an attendance table with metadata
   4. Participants
      1. Use the attendance table to, e.g., mute or reduce the influence of a particular source
      2. Place objects on their desks which are shown in front of them at the meeting or placed in the space for individual participants to engage, e.g., rotate etc.

# Terms and Definitions

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

*Table 1* *– Table of terms and definitions*

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Affordance | Quality or property of an object that defines its possible uses or makes clear how it can or should be used. |
| Ambient | The physical space of a participant and the shared virtual space. |
| Avatar | An animated 3D object representing a particular person in a virtual space. |
| Avatar Model | An inanimate avatar |
| Emotion | An attribute that indicates an emotion out of a finite set of Emotions |
| Face |  |
| Gesture |  |
| Identity |  |
| Meaning | Information extracted from the input text such as syntactic and semantic information |
| Navigation |  |

# References

## Normative references

MPAI-MCS normatively references the following documents:

1. MPAI Standard; The Governance of the MPAI Ecosystem; N341.
2. MPAI Technical Specification; AI Framework (MPAI-AIF); N324.
3. MPAI Technical Specification; Context-based Audio Enhancement (MPAI-CAE); N326.
4. MPAI Technical Specification; Multimodal Conversation (MPAI-CAE); N328.
5. ISO/IEC 10646; Information technology – Universal Coded Character Set (UCS)

## Informative references

1. https://www.w3.org/WoT/
2. <http://model.webofthings.io/>
3. <https://iot.mozilla.org/wot/>
4. <https://developer.nvidia.com/nvidia-omniverse-platform>
5. Facial Action Coding System <http://web.cs.wpi.edu/~matt/courses/cs563/talks/face_anim/ekman.html>
6. Facial Action Coding System <https://www.paulekman.com/facial-action-coding-system/>
7. Blender; [www.blender.org](http://www.blender.org)
8. https://docs.unity3d.com/Manual/AmbisonicAudio.html
9. https://docs.enklu.com/docs/Assets/Audio
10. https://techcommunity.microsoft.com/t5/mixed-reality-blog/microsoft-mesh-app-august-2021-update-new-features/ba-p/2746856
11. <https://3d.kalidoface.com/>
12. List of motion and gesture file formats, <https://en.wikipedia.org/wiki/List_of_motion_and_gesture_file_formats>
13. FACS-based Facial Expression Animation in Unity; https://github.com/dccsillag/unity-facs-facial-expression-animation

# Use Cases

## Virtual e-Learning

### AIW of “Client TX”

#### Function

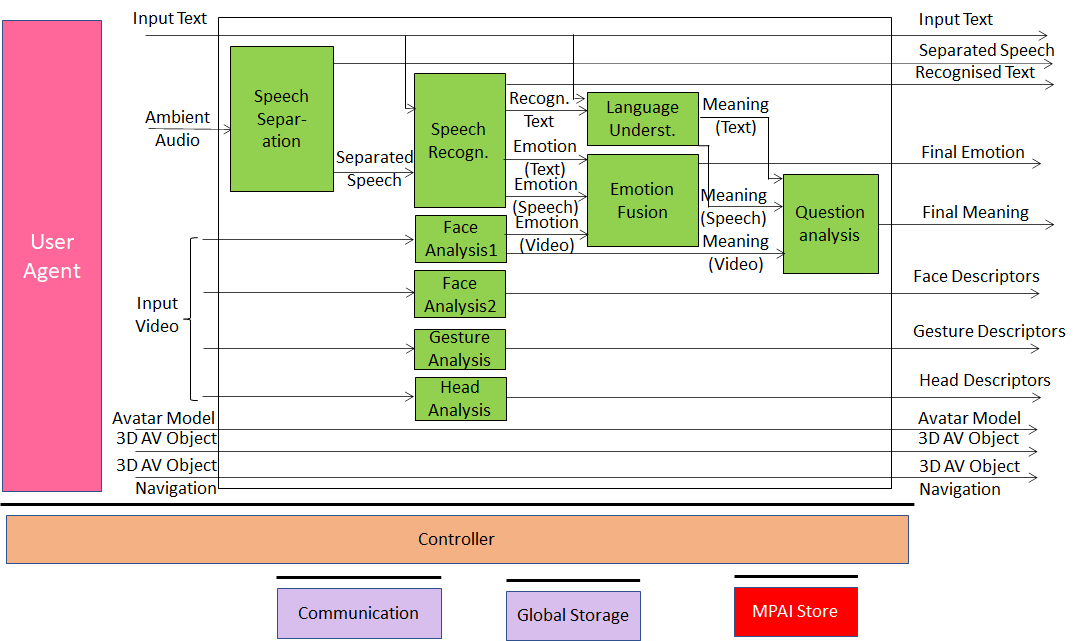
The function of the AIW is:

1. At the start, to send participant’s:
   1. Avatar model.
   2. Non-real time 3D AV Objects.
2. During the meeting, to continuously detect and send participant’s:
   1. Input Text (text entered via keyboard).
   2. Separated Speech (speech separated from Ambient Audio).
   3. Recognised Text (text produced by Speech Recogniser from Separated Speech).
   4. Final Emotion.
   5. Final Meaning.
   6. Face Descriptors.
   7. Real time 3D AV object.
   8. Commands acting on 3D AV object.

#### Architecture

Each participant (sending side)

1. Has the following devices:
   1. Microphone (array)
   2. Camera (array)
   3. A device separating participant’s Speech from Ambient Audio including the 3D Audio field created by Participant Rx.
2. Sends before the meeting:
   1. Avatar model (or selection from a choice).
   2. Files containing any 3D audio-visual presentation.
3. Sends during the meeting:
   1. Final Emotion and Meaning
   2. Text recognised from Speech
   3. Head, Face and Gesture Descriptors
   4. Speech
   5. Real time 3D AV Object
   6. Commands to act on 3D AV Objects.



*Figure 2 – Reference Model of the “Client TX” client*

The input and output data are given by *Table 1*:

*Table 2 – Input and output data of “Client TX” AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Ambient Audio | Audio including Participant’s Speech, Client RX Audio and other audio |
| Input Video | Video of Participant’s torso |
| Avatar model | The model or the ID of the avatar model selected by Participant |
| Head model | Each Participant sends their own head model or selects one from those offered by the Hosting Organisation. |
| 3D AV object | Each Participant may send a 3D AV objects to the server for later access |
| 3D AV object commands | The originator can and any participants may (if authorised) send commands that act on a 3D AV object |
| **Output** | **Comments** |
| Separated Speech | Participant’s Speech |
| Text | Recognised from Separated Speech.. |
| Final Emotion | Emotion. |
| Final Meaning | Meaning. |
| Face Descriptors | For Face reproduction on avatar. |
| Gesture Descriptors | For Gesture reproduction on avatar. |
| Head Descriptors | For Head reproduction on avatar. |
| Avatar Model | As in input. |
| 3D AV Object | As in input. |
| 3D AV Object Navigation | As in input. |

#### AI Modules

The AI Modules of “Participant TX” are given in *Table 3*.

*Table 3 – AI Modules of “Client TX” AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Speech Separation** | Provides Speech separates from non-speech Sound in Ambient Audio. |
| **Speech Recognition** | Provides Text and Emotion from Separated Speech. |
| **Face Analysis1** | Provides Emotion and Meaning from Input Video (face). |
| **Face Analysis2** | Provides Face Descriptors for reproduction of face on avatar. |
| **Gesture Analysis** | Provides Gesture Descriptors for reproduction of arms and hands on avatar |
| **Head Analysis** | Provides Head Descriptors for reproduction of head on avatar. |
| **Language Understanding** | Provides Meaning from Recognised Text and Input Text. |
| **Emotion Fusion** | Provides the Final Emotion from Speech and Face Emotion. |
| **Question Analysis** | Produces Final Meaning. |

### AIW of Server

#### Function

The function of VEL Server AIW is:

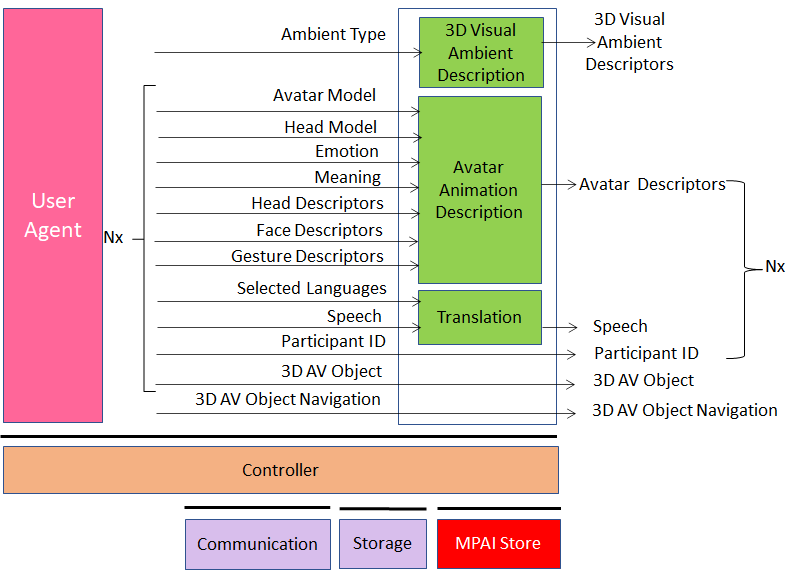
1. At the start, to create the Ambient and the Avatars based on inputs from the Hosting Organ­isation and the Participants.
2. During the lecture:
   1. to create and the send the full set of 3D AV dynamic descriptors of the Avatars in the Ambient and their movements.
   2. to forward other data from one Participant to all Participants.

#### Architecture

The MCS-VEL Server

1. Receives from
   1. Hosting Organisation:
      1. Selected Ambient.
      2. Number if Participants (N).
   2. Each Participant:
      1. Speech.
      2. Fused Emotion and Meaning
      3. Head, Face and Gesture Descriptors.
      4. 3D AV Objects.
      5. Commands to act on 3D AV Objects.
2. Computes and sends:
   1. 3D Visual Ambient descriptors.
   2. All Participant IDs
   3. All Speeches with their IDs.
   4. Avatar visual Descriptors.
3. Sends:
   1. 3D AV Objects.
   2. Commands to act on 3D AV Objects.

The architecture of “VEL Server” AIW is given by *Figure 7*.



*Figure 3 – Reference Model of the “VEL Server”*

#### I/O data

The input and output data are given by *Table 10*.

*Table 4 – Input and output data of LEV Server AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| # of Participants (N) | From Hosting Organisation |
| Ambient Selection | From Hosting Organisation |
| Selected Languages | From all participants |
| Speech (xN) | From all participants |
| Avatar Model (xN) | From all participants |
| Head Model (xN) | From all participants |
| Emotion & Meaning (xN) | From all participants |
| Head Descriptors (xN) | From all participants |
| Face Descriptors (xN) | From all participants |
| Gesture Descriptors (xN) | From all participants |
| 3D AV Object (xN) | From participants wishing to do so |
| 3D AV Object Commands (xN) | From participants wishing to do so |
| **Outputs** | **Comments** |
| 3D Visual Ambient Descriptors | Static descriptors of Ambient |
| Participant ID (xN) | Static participant IDs |
| Identified Speech (xN) | Participants’ Speeches and IDs |
| Identified Avatar Descriptors (xN) | Avatars Descriptors with Participant IDs |
| 3D AV Objects | Real time 3D AV objects |
| 3D AV Object Commands (xN) | Commands to act on 3D AV objects |

#### AI Modules

The AI Modules of “Server” are given in *Table 5*.

*Table 5 – AI Modules of VEL Server AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **3D Visual Ambient Description** | Creates all static 3D Visual Ambient Descriptors. |
| **Avatar Animation Description** | Creates Avatar Descriptors and associates Identity of all Participants to all Avatars and Speeches. |
| **Translation** | 1. Selects an active speaker 2. Translates the Speech of that speaker to the set of speeches translated to the set of Selected Languages 3. Assigns a translated speech to the appropriate set of Participants |

### AIW of “VEL Client RX”

#### Function

The Function of the “VEL Client RX” AIW is to:

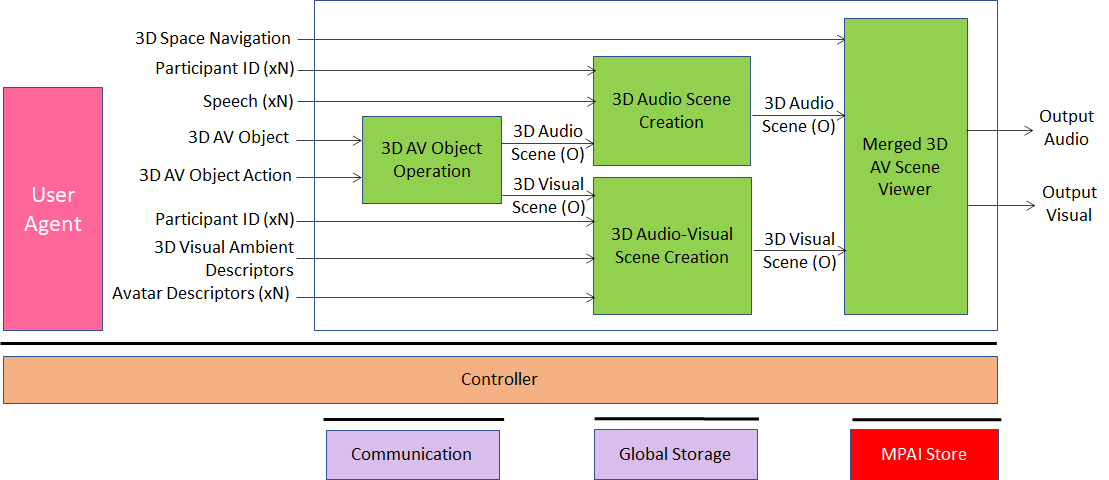
1. Create the virtual 3D AV e-learning scene.
2. Let Participant act on 3D AV Object(s)
3. Merge the virtual 3D AV e-learning scene with the 3D AV Object(s)
4. Present the merged 3D AV scene.
5. Let Participant navigate the merged 3D AV scene.

#### Architecture

The “VEL Client RX” AIW:

1. Creates the 3D visual space using:
   1. The 3D Visual Ambient descriptors.
   2. The Avatars descriptors.
   3. The visual output of the 3D AV Object Viewer.
2. Allows participant to have an AV experi­ence from a selected point in the virtual 3D AV space.
3. Synthesises the 3D audio space with sound sources at:
   1. Each Avatar position.
   2. Location of 3D AV Object.
4. Presents audio information congruent with the position of the Participant’s selected viewpoint in the virtual 3D Visual scene.

The architecture of “Client RX” AIW is given by *Figure 4*.



*Figure 4 – Reference model of the “VEL Client RX”*

#### I/O data

The input and output data of the “VEL Client RX” are given by *Table 6*.

*Table 6 – Input and output data of “Participant RX” AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| 3D Space Navigation | Participant’s commands to navigate the 3D Visual scene |
| Participants’ IDs (xN) | Unique Participants’ IDs |
| Speeches (xN) | Participants’ speeches with ID |
| 3D AV Object | Stored and real-time 3D AV Objects |
| 3D AV Object Action | Standard commands acting on 3D AV Object |
| 3D Visual Ambient Descriptors | Static Descriptors of Ambient |
| Avatar Descriptors (xN) | Descriptors of Avatars bodies with participant IDs |
| **Output** | **Comments** |
| Output 3D Audio | To be reproduced with loudspeaker array |
| Output 3D Video | To be reproduced with 2D or 3D display |

### AI Modules

The AI Modules of “Participant RX” are given in *Table 9*.

*Table 7 – AI Modules of Local Avatar Videoconference*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **3D AV Object Operation** | 1. Receives Action Commands on 3D AV Object. 2. Provides resulting 3D Audio and Visual components. |
| **3D Audio Scene Creation** | Creates 3D Audio Scene resulting from speaking Avatars at the respec­tive locations of the Scene. |
| **3D Visual Scene Creation** | Creates 3D Visual Scene composed of static 3D Visual Scene Descriptors and Avatars. |
| **3D AV Object Viewer** | Dispalys Participant’s audio-visual scene of the merged 3D AV Scene. |

## Use Case #2 – Local Avatar Videoconference (LAV)

### AIW of “Participant TX”

#### Function

The function of the AIW is:

1. At the start, to send participant’s:
   1. Avatar model selection.
   2. Head model.
   3. 3D AV Objects.
2. During the meeting, to continuously detect and send:
   1. Speech Separated from Ambient Audio
   2. Face Descriptors (for face recognition)
   3. Speaker Descriptors (for speaker identification).
   4. Emotion & meaning.
   5. Head motion.
   6. Coded messages (I have to leave etc.).
   7. Real time 3D AV object.
   8. Commands acting on 3D AV object.

#### Architecture

The architecture of “Participant TX” AIW is given by *Figure 5*.

Chart, waterfall chart

Description automatically generated

*Figure 5 – Reference Model of the “Participant TX” client*

Each participant (sending side)

1. Has the following devices:
   1. Microphone (array)
   2. Camera (array)
   3. An acoustic device capable to separate participant’s Speech from Ambient Aound and the 3D Audio field created by Participant Rx.
2. Sends before the meeting:
   1. Selection of the avatar body model.
   2. Own avatar head and face of the model or selection of one head and face.
   3. Files containing any 2D- or 3D audio-visual presentation.
3. Sends during the meeting:
   1. Face and Speech Descriptors for identification
   2. Final Emotion and Meaning of face and speech
   3. Text recognised from Speech
   4. Movement of head and face
   5. Gesture Descriptors
   6. Encoded Speech
   7. Real time 3D AV Object
   8. Commands to act on 3D AV Object.

#### I/O data

The input and output data are given by *Table 8*:

*Table 8 – Input and output data of “Participant TX” AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Ambient Audio | Audio including participant’s Speech and other audio |
| Input Video | Video of participant’s torso |
| Coded messages | Each participants may send coded messages representing “I want to speak”, “I need to leave” etc. |
| Avatar model | The ID of the avatar model selected by the participant |
| Head model | Each participant sends their own head model or selects one from those offered by the videoconference service |
| 3D AV object | Each participant may send a 3D AV objects to the server for real time distribution |
| 3D AV object commands | The originator can and any participants may (if authorised) send commands that act on a 3D AV object (presentation of real time) |
| **Output** | **Comments** |
| Speaker Descriptors | For speaker identification by server. |
| Encoded Speech | The compressed Speech |
| Text | Recognised from Separated Speech.. |
| Final Emotion | The Descriptors of Emotion. |
| Final Meaning | The Descriptors of Meaning. |
| Face Descriptors | For face identification by server. |
| Gesture Descriptors | The Descriptors of Gesture. |
| Head Motion | Head Motion Descriptors. |
| Coded Message | As in input. |
| Avatar Model | As in input. |
| Head Model | As in input. |
| 3D AV Object | As in input. |
| 3D AV Object Commands | As in input. |

#### AI Modules

The AI Modules of “Participant TX” are given in *Table 9*.

*Table 9 – AI Modules of Multimodal Question Answering*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Speech Separation** | Provides Speech separates from non-speech Sound in Ambient Audio. |
| **Speaker Analysis** | Provides Speaker Descriptors. |
| **Speech Encoder** | Provides Speech in compressed format. |
| **Speech Recognition** | Provides Text and Emotion from Separated Speech |
| **Face Analysis1** | Provides Emotion and Meaning from Input Video (face) |
| **Face Analysis2** | Provides Face Descriptors from Input Video (face). |
| **Gesture Analysis** | Provides Gesture Descriptors from Input Video (gesture) |
| **Head Analysis** | Provides the movement of the head of a Participant. |
| **Language Understanding** | Provides Meaning from Recognised Text. |
| **Emotion Fusion** | Provides the Final Emotion from Speech and Face Emotion. |
| **Question Analysis** | Produces Final Meaning. |

### AIW of Server

#### Function

The function of Server AIW is:

1. At the start:
   1. Receives:
      1. Ambient selection
      2. Number of participants (N).
      3. Avatar body selection (xN).
      4. Avatar head model (xN).
      5. 3D AV Objects.
   2. Creates and sends Static Descriptors containing:
      1. Ambient
      2. Avatar bodies.
2. Continuously:
   1. Receives:
      1. Speech (xN).
      2. Face and Speaker Descriptors (xN).
      3. Emotion and meaning (xN).
      4. Coded Messages (xN).
      5. 3D AV Objects.
      6. 3D AV commands.
   2. Performs:
      1. Monitoring of participants’ identity using Face and Speaker Descriptors.
      2. Creation of Avatar Descriptors: heads, faces, arms and hands.
      3. Associates Participant IDs to Speeches and Avatars
   3. Sends:
      1. Descriptors of dynamic objects and IDs: faces, heads, arms, hands.
      2. Speeches with coordinates of sources and IDs.
      3. 3D AV objects.
      4. 3D AV commands.

#### Architecture

The architecture of Server AIW is given by *Figure 6*.

Diagram

Description automatically generated with medium confidence

*Figure 6 – Reference Model of the MCS-LAV Server*

MCS-LAV Server

1. Receives from conference manager
   1. Selected Ambient.
   2. Number if Participants (N).
2. Receives from each participant:
   1. Face and Speaker Descriptors (for identification).
   2. Encodes Speech.
   3. Head movements.
   4. Fused emotion and meaning
   5. Face and Gesture Descriptors.
   6. 3D AV Objects.
   7. Commands to act on 3D AV Objects.
3. Creates Descriptors of:
   1. 3D Ambient (table, chairs and avatar bodies) (one shot).
   2. Avatars’ head, face, arms and hands.
4. Sends
   1. 3D Visual Ambient descriptors.
   2. All Participant IDs
   3. All Speeches with their IDs.
   4. Avatar descriptors.
   5. 3D AV Objects.
   6. Commands to act on 3D AV Objects.

#### I/O data

The input and output data are given by *Table 10*.

*Table 10 – Input and output data of MCS-AVL (Server) AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| # of Participants (N) | From Server manager |
| Ambient Selection | From Server manager |
| Face Descriptors (xN) | From all participants (for identification) |
| Speaker Descriptors (xN) | From all participants (for identification) |
| Rncoded Speech (xN) | From all participants |
| Avatar Model (xN) | From all participants |
| Head Model (xN) | From all participants |
| Head Motion (xN) | From all participants |
| Emotion & Meaning (xN) | From all participants |
| Face Descriptors (xN) | From all participants |
| Head Descriptors (xN) | From all participants |
| Gesture Descriptors (xN) | From all participants |
| Coded Message (xN) | From all participants wishing to do so |
| 3D AV Object (xN) | From all participants wishing to do so |
| 3D AV Object Commands (xN) | From all participants wishing to do so |
| **Outputs** | **Comments** |
| 3D Visual Ambient Descriptors | Static descriptors of Ambient |
| Participant ID (xN) | Static participant IDs |
| ID’d Encoded Speech (xN) | Participants’ Speeches and IDs |
| ID’d Avatar Descriptors (xN) | Descriptors of Avatars with Participant IDs |
| 3D AV Objects | Real time 3D AV objects |
| 3D AV Object Commands (xN) | Commands to act on 3D AV objects |

#### AI Modules

The AI Modules of “Server” are given in *Table 9*.

*Table 11 – AI Modules of Multimodal Question Answering*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **3D Visual Ambient Description** | Collects all 3D Visual Ambient Descriptors. |
| **Participant Identification and Speeches** | Determines and associates Identity of all Participants to their Speeches |
| **Avatar Animation Description** | Collects and associates Identity of all Participants to Visual Descriptors of all Avatars. |

### AIW of “Participant RX”

#### Function

The Function of the “Participant RX” AIW is to:

1. Create the 3D AV scene of the conference.
2. Allow the participant to have the 3D audio-visual experience of the AV scene.

#### Architecture

The architecture of “Participant RX” AIW is given by *Figure 7*.

Diagram

Description automatically generated with medium confidence

*Figure 7 – Reference model of the “Participant RX” client*

The “Participant RX” AIW:

1. Creates the visual 3D space using:
   1. The 3D Visual Ambient descriptors.
   2. The Avatars descriptors.
   3. The visual output of the 3D AV Object Viewer.
2. Allows participant to have an AV experi­ence from a selected point in the virtual 3D AV space.
3. Synthesises the 3D audio space with sound sources at:
   1. Each Avatar position.
   2. Location of 3D AV Object.
4. Presents audio information congruent with the position of the Participant’s selected viewpoint in the virtual 3D Visual scene.

#### I/O data

The input and output data are given by *Table 12*

*Table 12 – Input and output data of “Participant RX” AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| ID’d Encoded Speeches (xN) | Participants’ speeches with ID |
| 3D AV Object | Real time 3D AV objects |
| 3D AV Object Commands | Standard instructions to act on 3D objects |
| 3D Visual Ambient Descriptors | Static Descriptors of Ambient |
| Participants’ IDs (xN) | Static participants’ IDs |
| ID’d Avatar Descriptors (xN) | Descriptors of Avatars bodies with participant IDs |
| Visual Navigation | Participant’s commands to navigate the 3D Visual scene |
| **Output** | **Comments** |
| 3D Audio | To be reproduced with loudspeaker array |
| 3D Video | To be reproduced with 2D or 3D display |

### AI Modules

The AI Modules of “Participant RX” are given in *Table 9*.

*Table 13 – AI Modules of Local Avatar Videoconference*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **3D Visual Scene Creation And Navigation** | Creates 3D Visual Scene corresponding to the selected point in the virtual 3D AV space, |
| **3D Audio Scene Creation And Navigation** | Creates 3D Audio Scene congruent witht the 3D Visual Scene |
| **3D AV Object Viewer** | Creates Participant’s view and audio of 3D AV Object |

# AI Modules

## AIMs and their data

### Participant TX

*Table 14 – AIMs and Data of Participant TX AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **Speech Separation** | Input Audio | Separatated Speech |
| **Speaker Recognition** | Separatated Speech | Speaker ID |
| **Speech Encoding** | Separatated Speech | Encoded Speech |
|  | Separatated Speech | Meaning (Speech) |
| Emotion (Speech) |
| **Face Analysis1** | Input Video | Meaning (Video) |
| Emotion (Video) |
| **Head Analysis** | Input Video | Head Motion |
| **Face Analysis2** | Input Video | Face ID |
|  | Coded Message | Coded Message |
|  | Avatar Model | Avatar Model |
|  | 3D AV Object | 3D AV Object |
|  | AV Object Commands | AV Object Commands |

### Server

*Table 15 – AIMs and Data of Server AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **3D Visual Ambient Description** | 3D Visual Ambient Descriptors | 3D Visual Ambient Descriptors |
| **Participant Identification and Speech** | Speech IDs | Participant IDs |
| Face IDs | Encoded Speeches |
| Encoded Speeches |  |
| **Avatar Animation Description** | Final Emotions | ID’d Avatar Descriptors |
| Final Meanings |
| Head Motions |
| Head Models |
| Avatar Models |
|  | 3D AV Object | 3D AV Object |
|  | AV Object Commands | AV Object Commands |

### Participant RX

*Table 16 – AIMs and Data of Participant RX AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **3D AV Object Viewer** | 3D AV Object | 3D Audio Scene (O) |
| 3D AV Object Commands | 3D Video Scene (O) |
| **3D Visual Ambient Creation And Navigation** | Participant IDs | Output Visual Scene |
| ID’s Avatars Descriptors |
| 3D Visual Ambient Descriptors |
| Visual Navigation |
| **3D Audio Ambient Creation And Navigation** | ID’d Encoded Speeches | Output Audio Scene |
| Navigation Command |
| 3D Visual Navigation Info |

## Data Formats

*Table 17* lists all data formats whose requirements are contained in this document. The first column gives the name of the data format, the second the subsection where the requirements of the data format are provided and the third the Use Case making use of it.

*Table 17 – Data formats*

|  |  |  |
| --- | --- | --- |
| **Name of Data Format** | **Subsection** | **Use Case** |
| # of Participants | 6.2.1 |  |
| 3D Audio Navigation Info | 6.2.2 |  |
| 3D Audio Scene (O) | 6.2.3 |  |
| 3D AV Object | 6.2.4 |  |
| 3D AV Object Commands | 6.2.5 |  |
| 3D Video Scene (O) | 6.2.6 |  |
| 3D Visual Ambient Descriptors | 6.2.7 |  |
| Ambient Audio | 6.2.8 |  |
| Ambient Descriptors | 6.2.8 |  |
| Ambient Type | 6.2.9 |  |
| Avatar Model | 6.2.10 |  |
| Coded Message | 6.2.11 |  |
| Emotion (Speech) | 6.2.12 |  |
| Emotion (Video) | 6.2.13 |  |
| Encoded Speech | 6.2.14 |  |
| Face Descriptors | 6.2.15 |  |
| Final Emotions | 6.2.16 |  |
| Final Meanings | 6.2.17 |  |
| Gesture Descriptors | 6.2.18 |  |
| Head Descriptors | 6.2.19 |  |
| Head Model | 6.2.20 |  |
| Head Motion | 6.2.21 |  |
| ID’d Avatar Descriptors | 6.2.21 |  |
| ID’d Encoded Speech | 6.2.24 |  |
| Input Video | 6.2.25 |  |
| Meaning (Speech) |  |  |
| Meaning (Video) | 6.2.26 |  |
| Navigation Command | 6.2.27 |  |
| Output Audio Scene | 0 |  |
| Output Visual Scene | 6.2.27 |  |
| Participant IDs | 6.2.30 |  |
| Recognised Text | 6.2.31 |  |
| Separated Speech | 6.2.32 |  |
| Speaker Descriptors | 6.2.31 |  |
| Visual Navigation | 6.2.32 |  |

### # of Participants

An integer corresponding to the number of Participants.

### 3D Audio Navigation Info

The coordinates of the Participant looking at the meeting from a particular viewpoint in the MCS.

### 3D Audio Scene (O)

The Audio component of 3D AV Objects.

### 3D AV Object

A description of a 3D Audio-Visual Object.

### 3D AV Object Commands

Instructions to navigate the 3D Audio-Visual Object.

### 3D Video Scene (O)

The Visual component of 3D AV Objects.

### 3D Visual Ambient Descriptors

The set of Descriptors required to represent the static components of the MCS: table, chair, walls, furniture etc.

### Ambient Audio

The digital representation of the audio captured from a Participant’s site.

MPAI-CAE has defined a digital representation of a microphone set [4].

### Ambient Type

An Identifier of a furnished MCS offered by the Service Provider.

### Avatar Model

1. An Identifier of a model of an avatar offered by the Service Provider or
2. An avatar model provided by a Participant.

### Coded Message

An Identifier of a message representing

1. Join the meeting
2. Ask for the floor
3. Leave the meeting
4. ...

### Emotion (Speech)

An emotion for speech out of a set.

### Emotion (Video)

An emotion set for a Face out of a set.

### Encoded Speech

Streamed compressed Speech.

### Face Descriptors

Descriptors for face recognition.

### Final Emotion

Emotion resulting from the fusion of Emotion (Speech) and Emotion (Video).

MPAI-MMC has defined a digital representation of Emotion [4].

### Final Meanings

Meaning resulting from the fusion of Meaning (Speech) and Meaning (Video).

MPAI-MMC has defined a digital representation of Meaning [4].

### Gesture Descriptors

Descriptors suitable to

1. Recognise sign language
2. Recognise coded hand signs for navigation
3. Animate arms and hands of a Participant’s avatar.

### Head Descriptors

Descriptors suitable to animate the head of the Participant’s avatar to reproduce the movements of the Participant’s head.

### Head Model

1. An Identifier of a model of an animation-ready avatar head offered by the Service Provider or
2. An animation-ready avatar model head provided by a Participant.

### ID’d Avatars Descriptors

A set of avatar Descriptors composed of torso, head, face, arms and hads, associated to a Participant ID.

### ID’d Decoded Speech

A continuous Speech stream associated to a Participant ID.

### Input Video

The digital visual representation of the Participant’s torso.

### Meaning (Speech)

MPAI-MMC defines how to digitally represent Meaning.

### Meaning (Video)

MPAI-MMC defines how to digitally represent Meaning.

### Output Audio Scene

The 3D Audio field generated by Participant RX.

### Output Visual Scene

The 3D Visual field generated by Participant RX.

### Participant IDs

Dynamic (session-by-session) Participant Identifier.

### Recognised Text

Text should be digitally represented according to [3].

### Separated Speech

Speech resulting from removal of non-speech information from Input Audio.

### Speaker Descriptors

Descriptors for Speaker recognition.

### Visual Navigation

Avatars participanting in a meeting are static, save for the specific coded messages that represent joining and leaving the meeting. Navigation Commands are used to define the viewpoint of the Participant in the 3D AV scene.

# Annex 1 – MPAI-wide terms and definitions (Normative)

The Terms used in this standard whose first letter is capital and are not already included in *Table 1* are defined in *Table 18.*

*Table 18 – 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 Workflow (AIW) | An organised aggregation of AIMs implementing a Use Case receiving AIM-specific Inputs and producing AIM-specific Outputs according to its Function. |
| AI Module (AIM) | A processing element receiving AIM-specific Inputs and producing AIM-specific Outputs according to according to its 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 |
| Component | One of the 7 AIF elements: Access, Communication, Controller, Internal Storage, Global Storage, MPAI Store, and User Agent |
| Conformance | The attribute of an Implementation of being a correct technical Implem­entation of a Technical Specification. |
| Conformance Tester | An entity authorised by MPAI to Test 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 the following actors: MPAI, MPAI Store, Implementers, Conformance Testers, Performance Testers and Users of MPAI-AIF Im­plem­en­tations as needed to enable an Interoperability Level. |
| 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. |
| Interoperability | The ability to functionally replace an AIM with another AIM 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 procedures, the tools, the data sets and/or the data set characteristics 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 authorised by MPAI to Assess the Performance of an Implementation in a given Application domain |
| 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. |
| 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 by MPAI 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. |
| Zero Trust |  |

# Annex 2 - Notices and Disclaimers Concerning MPAI Standards (Informative)

The notices and legal disclaimers given below shall be borne in mind when [downloading](https://www.mpai.community/resources/) and using approved MPAI Standards.

In the following, “Standard” means the collection of four MPAI-approved and [published](https://www.mpai.community/resources/) 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](https://mpai.community/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](https://mpai.community/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](https://mpai.community/resources/) 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](mailto:secretariat@mpai.community). 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 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 MPAI standards 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](https://mpai.community/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.

Copyright

MPAI draft and approved standards, whether they are in the form of documents or as web pages or otherwise, are copyrighted by MPAI under Swiss and international copyright laws. MPAI Standards are made available and may be used for a wide variety of public and private uses, e.g., implementation, use and reference, in laws and regulations and standardisation. By making these documents available for these and other uses, however, MPAI does not waive any rights in copyright to its Standards. For inquiries regarding the copyright of MPAI standards, please contact the [MPAI Secretariat](mailto:secretariat@mpai.community).

The Reference Software of an MPAI Standard is released with the [MPAI Modified Berkeley Software Distribution licence](https://mpai.community/about/licence/). However, implementers should be aware that the Reference Software of an MPAI Standard may reference some third party software that may have a different licence.

# Annex 3 – The Governance of the MPAI Ecosystem (Informative)

**Level 1 Interoperability**

With reference to *Figure 1*, MPAI issues and maintains a standard – called MPAI-AIF – whose components are:

1. An environment called AI Framework (AIF) running AI Workflows (AIW) composed of inter­connected AI Modules (AIM) exposing standard interfaces.
2. A distribution system of AIW and AIM Implementation called MPAI Store from which an AIF Implementation can download AIWs and AIMs.

|  |  |
| --- | --- |
| Implementers’ benefits | Upload to the MPAI Store and have globally distributed Implementations of   * AIFs conforming to MPAI-AIF. * AIWs and AIMs performing prop­rietary functions executable in AIF. |
| Users’ benefits | Rely on Implementations that have been tested for security. |
| MPAI Store | * Tests the Conformance of Implementations to MPAI-AIF. * Verifies Implementations’ security, e.g., absence of malware. * Indicates unambiguously that Implementations are Level 1. |

**Level 2 Interoperability**

In a Level 2 Implem­entation, the AIW must be an Implementation of an MPAI Use Case and the AIMs must con­form with an MPAI Applicati­on Standard.

|  |  |
| --- | --- |
| Implementers’ benefits | Upload to the MPAI Store and have globally distributed Implementations of   * AIFs conforming to MPAI-AIF. * AIWs and AIMs conforming to MPAI Application Standards. |
| Users’ benefits | * Rely on Implementations of AIWs and AIMs whose Functions have been reviewed during standardisation. * Have a degree of Explainability of the AIW operation because the AIM Func­tions and the data Formats are known. |
| Market’s benefits | * Open AIW and AIM markets foster competition leading to better products. * Competition of AIW and AIM Implementations fosters AI innovation. |
| MPAI Store’s role | * Tests Conformance of Implementations with the relevant MPAI Standard. * Verifies Implementations’ security. * Indicates unambiguously that Implementations are Level 2. |

**Level 3 Interoperability**

MPAI does not generally set standards on how and with what data an AIM should be trained. This is an important differentiator that promotes competition leading to better solutions. However, the performance of an AIM is typically higher if the data used for training are in greater quantity and more in tune with the scope. Training data that have large variety and cover the spec­trum of all cases of interest in breadth and depth typically lead to Implementations of higher “quality”.

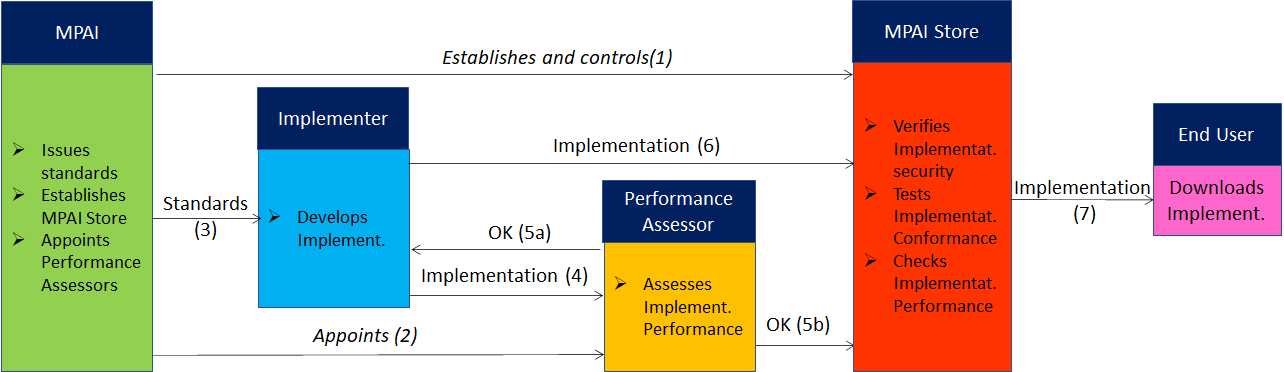
For Level 3, MPAI normatively specifies the process, the tools and the data or the characteristics of the data to be used to Assess the Grade of Performance of an AIM or an AIW.

|  |  |
| --- | --- |
| Implementers’ benefits | May claim their Implementations have passed Performance Assessment. |
| Users’ benefits | Get assurance that the Implementation being used performs correctly, e.g., it has been properly trained. |
| Market’s benefits | Implementations’ Performance Grades stimulate the development of more Performing AIM and AIW Implementations. |
| MPAI Store’s role | * Verifies the Implementations’ security * Indicates unambiguously that Implementations are Level 3. |

**The MPAI ecosystem**

The following is a **Error! Reference source not found.**high-level description of the MPAI ecosystem operation applicable to fully conforming MPAI implementations:

1. MPAI establishes and controls the not-for-profit MPAI Store (step 1).
2. MPAI appoints Performance Assessors (step 2).
3. MPAI publishes Standards (step 3).
4. Implementers submit Implementations to Performance Assessors (step 4).
5. If the Implementation Performance is acceptable, Performance Assessors inform Implementers (step 5a) and MPAI Store (step 5b).
6. Implementers submit Implementations to the MPAI Store (step 6); The Store Tests Confor­mance and security of the Implementation.
7. Users download Implementations (step 7).



*Figure 8 – The MPAI ecosystem operation*

The Ecosystem operation allows for AIW and AIF Implementations to be:

1. Proprietary: security is verified and Conformance to MPAI-AIF Tested (Level 1).
2. Conforming to an MPAI Application Standard: security is verified and Conformance to the relevant MPAI Application Standard Tested (Level 2).
3. Assessed to be Reliable, Robust, Fair and Replicable (Level 3).

and have their Interoperability Level duly displayed in the MPAI Store.