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

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **Public document** | |
| **N521** | 2022/01/26 |
| **Source** | Leonardo Chiariglione |
| **Title** | MPAI-MCS Use Cases and Functional Requirements WD0.5 |
| **Target** | MPAI Community |

[1 Introduction 2](#_Toc90211505)

[2 Scope of proposed standard 3](#_Toc90211506)

[3 Terms and Definitions 5](#_Toc90211507)

[4 References 6](#_Toc90211508)

[4.1 Normative references 6](#_Toc90211509)

[4.2 Informative references 6](#_Toc90211510)

[5 Use Cases 7](#_Toc90211511)

[5.1 Client-Based Ambient 7](#_Toc90211512)

[5.1.1 TX Client AIW 7](#_Toc90211513)

[5.1.2 Server AIW 10](#_Toc90211514)

[5.1.3 RX Client AIW 12](#_Toc90211515)

[5.1.4 AI Modules 13](#_Toc90211516)

[5.2 Server-Based Ambient 14](#_Toc90211517)

[5.2.1 TX Client AIW 14](#_Toc90211518)

[5.2.2 AIW of Server 15](#_Toc90211519)

[5.2.3 RX Client AIW 18](#_Toc90211520)

[6 AI Modules 18](#_Toc90211521)

[6.1 AIMs and their data 18](#_Toc90211522)

[6.1.1 Client-Based Ambient 18](#_Toc90211523)

[6.1.2 Server-Based Ambient 19](#_Toc90211524)

[6.2 Data Formats 20](#_Toc90211525)

[6.2.1 Ambient Descriptors 21](#_Toc90211526)

[6.2.2 Ambient Model 21](#_Toc90211527)

[6.2.3 Audio-Visual Object 22](#_Toc90211528)

[6.2.4 Audio-Visual Object Action 22](#_Toc90211529)

[6.2.5 Audio-Visual Scene 22](#_Toc90211530)

[6.2.6 Avatars Descriptors 22](#_Toc90211531)

[6.2.7 Avatar Model 22](#_Toc90211532)

[6.2.8 Emotion (Speech) 23](#_Toc90211533)

[6.2.9 Emotion (Text) 23](#_Toc90211534)

[6.2.10 Emotion (Video) 23](#_Toc90211535)

[6.2.11 Face Descriptors 23](#_Toc90211536)

[6.2.12 Face Descriptors (Auth) 23](#_Toc90211537)

[6.2.13 Fused Emotion 24](#_Toc90211538)

[6.2.14 Fused Meaning 24](#_Toc90211539)

[6.2.15 Gesture Descriptors 24](#_Toc90211540)

[6.2.16 Head Descriptors 24](#_Toc90211541)

[6.2.17 Input Audio 24](#_Toc90211542)

[6.2.18 Input Video 24](#_Toc90211543)

[6.2.19 Language Preferences 25](#_Toc90211544)

[6.2.20 Meaning (Text) 25](#_Toc90211545)

[6.2.21 Meaning (Video) 25](#_Toc90211546)

[6.2.22 Output Audio 25](#_Toc90211547)

[6.2.23 Output Visual 25](#_Toc90211548)

[6.2.24 Participant IDs 25](#_Toc90211549)

[6.2.25 Recognised Text 25](#_Toc90211550)

[6.2.26 Speech Descriptors (Auth.) 26](#_Toc90211551)

[6.2.27 Viewpoint 26](#_Toc90211552)

[Annex 1 – MPAI-wide terms and definitions (Normative) 27](#_Toc90211553)

[Annex 2 - Notices and Disclaimers Concerning MPAI Standards (Informative) 30](#_Toc90211554)

[Annex 3 – The Governance of the MPAI Ecosystem (Informative) 32](#_Toc90211555)

[Annex 4 Example use cases 34](#_Toc90211556)

[1 Virtual e-learning (VEL) 34](#_Toc90211557)

[2 Local Avatar Videoconference 35](#_Toc90211558)

[3 Teleconsulting 36](#_Toc90211559)

# Introduction

In recent years, Artificial Intelligence (AI) and related technologies, applied to a broad range of applications, have started affecting the life of millions of people and they are expected to do so even 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. Indeed, research has shown that data coding with AI-based technologies is generally *more efficient* than with existing technologies for, e.g., compression and feature-based description.

However, some AI technologies may carry inherent risks, e.g., in terms of bias toward some classes of users. Therefore, the need for standardisation is more important and urgent than ever.

The international, unaffiliated, not-for-profit MPAI – Moving Picture, Audio and Data Coding by Artificial Intelligence Standards Developing Organisation has the mission to develop *AI-enabled data coding standards*. MPAI Application Standards enable the development of AI-based products, applications and services.

As a part of its mission, MPAI has developed standards operating procedures to enable a user of MPAI implementations to make informed decision about their applicability. Central to this is the notion of Performance, defined as a set of attributes characterising a reliable and trustworthy implementation.

For the aforementioned reasons, to fully achieve the MPAI mission, technical standards must be complemented by the creation and management of an ecosystem designed to underpin the life cycle of MPAI standards through the steps of specification, technical testing, assessment of product safety and security, and distribution.

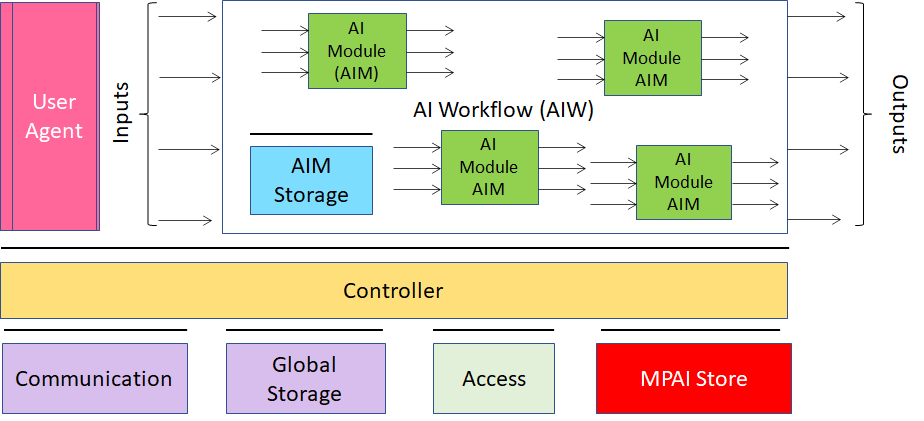
In the following, Terms beginning with a capital letter are defined in *Table 1* if they are specific to MPAI-MCS Standard and to *Table 18* if they are common to all MPAI Standards.

The MPAI Ecosystem is fully specified in [1]. It is composed of:

* MPAI as provider of Technical, Conformance and Performance Specifications.
* Implementers of MPAI standards.
* MPAI-appointed Performance Assessors.
* The MPAI Store which takes care of secure distribution of validated Implementations.

The common infrastructure enabling implementation of MPAI Application Standards is the AI Framework (AIF) Standard (MPAI-AIF), specified in this document.

*Figure 1* depicts the MPAI-AIF Reference Model [2] under which Implementations of MPAI Application Standards and user-defined MPAI-AIF conforming applications operate.



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

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

MPAI Application Standards normatively specify Semantics and Syntax 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 Implementation. MPAI also defines 3 Interoperability Levels of an AIW that executes an AIW. The AIW may have 3 Levels:

*Level 1 –* Implementer-specific and satisfying the MPAI-AIF Standard.

*Level 2 –* Specified by an MPAI Application Standard.

*Level 3 –* Specified by an MPAI Application Standard and certified by a Performance Assessor.

MPAI offers 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. Additional information on Interoperability Levels is provided in Annex 3.

# Scope of proposed standard

*Mixed-reality Collaborative Spaces* (MPAI-MCS) is an MPAI standard project containing proposed Use Cases, and Functional Requirements for AI Workflows, AI Modules, and Data Formats applicable to scenarios where geographically separated Humans – called Participants – have real-time collaborations in virtual-reality spaces called Ambients via speaking Avatars for the purpose of achieving goals generally defined by the Use Case and specifically carried out by Avatars as directed by the Participants.

An MCS can be implemented with N Clients, where N is the number of Participants. In general a Server is also required. For the following analysis, it is convenient to split a Client in the transmitting (TX) and receiving (RX) parts.

The functions of the Client TX, Server and Client RX identified are

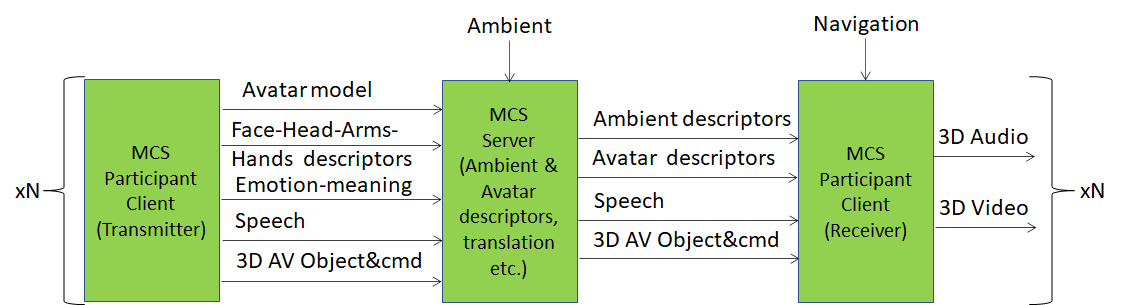
1. *Client TX* generates the following types of information:
   1. Text.
   2. Speech.
   3. Audio – mono- or multi-channel.
   4. Video – 2D or volumetric.
   5. 3D audio-visual objects (in real time or retrieved).
2. *Client TX (or Server)* generates:
   1. An animated Avatar that realistically represents the Participant so that it can perform the functions proper of the role in the collaboration.
   2. Participant’s speech transmitted *as is* or synthesised from input text.
3. *Server*
   1. Authenticate the Avatars in the Ambient.
   2. Translates input languages into to the languages preferred by each Participant.
4. *Client RX* composes and renders the Ambient and its components to the extent permitted by the format of data received (e.g., individual object or a packages scene).

There are several possible ways to split the MCS functionalities among Client TX, Server and Client RX in order to implement an MCS:

1. *Client-based*
   1. Client TX sends Participants’ media data:
      1. Speech
      2. Animated Avatar
      3. 3D audio-visual objects.
   2. Server
      1. Authenticates Participants
      2. Translates Participants’ speech
      3. Sends Ambient and original/translated Speech to all Clients RX
      4. Forwards Avatars and 3D AV object received from Participants to all Clients RX
   3. Client RX renders MCS directly as received from Server or after composing a scene using the individual objects received from Server.
2. *Server-based*
   1. Client TX sends Participant’s text, speech, video and 3D audio-visual objects
   2. Server
      1. Authenticates Participants.
      2. Creates synthetic Speech from text (if any present).
      3. Translates Participants’ speech.
      4. Creates Animated Avatars.
      5. Sends Ambient, original/translated Speech, and Avatars to all Clients RX.
   3. Client RX renders MCS directly as received from Server or after composing a scene using the individual objects received from Server.

In between these two extreme configurations, there is a variety of ways to allocate the functions of Clients and Server.

*Figure 2* depicts a possible intermediate arrangement where extraction of personal information – descriptors of face, head, arm and hands of the user – is done by the client, while other functions – such as provision of the static Ambient, processing of audio information and creation of Avatar descriptors – are executed by the server. In general, how much of the computing power is pushed to the edge is decided by the context in which the MCS is created. The functions of Client RX are independent of how the functions between Client TX and Server are allocated.



*Figure 2 – The 3 components of an MPAI-MCS Use Case*

Examples of applications target of the MPAI-MCS standard are:

1. *Local Avatar Videoconference* where realistic virtual twins (Avatars) of physical twins (Participants) sit around a table and hold a conference. In this case, as many functions as practically possible are assigned to clients because there may be strict security issues in terms of participant identity and avoidance of clear text information (e.g, voice and emotions of participants) sent to the server.
2. *Virtual eLearning* where avatars may still be realistic virtual twins of humans but there may be less concerns about identity and information transmitted to the server. In this case most functions of Client TX can be delegated to the Server.

With reference to *Figure 2*, MPAI-MCS has the the following features:

1. Ambients, are 3D virtual spaces representing actual, realistic or fictitious physical spaces, populated by 3D objects representing actual, realistic or fictitious Visual Objects with specified affordances and Audio Objects propagating according to the specific Visual Object affordances (e.g., walls, objects).
2. Avatars move around in Ambients, express emotions and perform gestures corresponding to actual, realistic or fictitious Humans.
3. Humans generate media information captured by TX Clients generating and transmitting coded representations of Text, Human Features (e.g., emotion extracted from speech and face), Audio and Video.
4. The Participant is sensed by different types of sensors:
   1. Microphones
   2. Cameras, possibly volumetric
   3. Sensors of human commands (mouse, haptic, etc.)
5. Commands are agnostic of the specific type of sensor.
6. Ambients may be supported by Servers whose goal is to create digital representations of Ambient and possibly of their components and distribute them to RX Clients in:
   1. Final form in which case all participating Humans share exactly the same user experience
   2. Component form in which case participating Humans may have different user experience depending on how they assemble the components.
7. Ambients are populated by 3D Audio-Visual Objects generated by a possibly human-animated file or by a device operating in real time.
8. Humans may use Commands to act on 3D Audio-Visual Objects performing such actions as:
   1. Manually 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. Assign physical properties to (different parts) of the 3D AV object.
   5. Annotate portions of the 3D AV object.
   6. Deform/sculpt the 3D AV object.
   7. Combine 3D AV objects.
   8. Call an anomaly detector on a portion with a criterion.
   9. Create links between different parts of the 3D AV object.
   10. Follow a link to another portion of the 3D AV object.
   11. 3D print (portions of) the visual part 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.

The structure of this document is:

|  |  |
| --- | --- |
| Chapter 2 | Defines the characteristics of Mixed-reality Collaborative Spaces and 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 AIWs implementing the Use Cases and their compon­ents |
| Chapter 6 | Identifies and describes   1. AIMs (Section 6.1) 2. AIMs’ Data Formats (Section 6.1.1). |

It is expected that, once the MPAI-MCS standardisation process has reacheds the appropriate stage, this document will become an attachment to the MPAI-MCS Call for Technologies requesting technologies satisfying the requirements specified in Section 6.1.1.

# 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. |
| Audio | Digital representation of an analogue audio signal sampled at a frequency between 8-192 kHz with a number of bits/sample between 8 and 32. |
| Audio Object | Audio information with its metadata. |
| Audio Scene | The Audio component of an AV Scene. |
| Audio-Visual Scene | (AV Scene) The structured composition of Audio and Visual Objects in an Ambient. |
| 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 | The portion of a video containing the face of a human. |
| Gesture | The coordinate movement of hands and possibly arms |
| Identity | The label associated to a human in a unique way. |
| Input Audio | Audio entering an MPAI-MCS Client (TX). |
| Intention | The result of analysis of the goal of an input question. |
| Meaning | Information extracted from the input such as syntactic and semantic information. |
| Separated Speech | Speech extracted from Input Audio. |
| Speech | Digital representation of analogue speech sampled at a frequency between 8 kHz and 96 kHz with a number of bits between 16 bits/sample and 24 bits/sample PCM values. |
| Visual Object | Visual information with its metadata. |
| Visual Scene | The Visual component of an AV Scene. |
| Viewpoint | The audio and visual point in the MCS selected by a human. It needs not coincide with the point in which an avatar is located |

# References

## Normative references

MPAI-MCS normatively references the following documents:

1. MPAI Technical Specification; The Governance of the MPAI Ecosystem (MPAI-GME) V1.
2. MPAI Technical Specification; AI Framework (MPAI-AIF) V1.
3. MPAI Technical Specification; Context-based Audio Enhancement (MPAI-CAE) V1.
4. MPAI Technical Specification; Multimodal Conversation (MPAI-CAE) V1.
5. ISO 639-1:2002; Codes for the representation of names of languages — Part 1: Alpha-2 code
6. ISO/IEC 10646; Information technology – Universal Coded Character Set (UCS)

## Informative references

Ambient

1. ISO 16739-1:2018 Industry Foundation Classes (IFC) For Data Sharing In The Construction And Facility Management Industries — Part 1: Data Schema

1. <https://technical.buildingsmart.org/standards/ifc/>
2. Khronos’s [glTF](https://www.khronos.org/gltf/): glTF defines an extensible, transmission-efficient publishing format that streamlines authoring workflows and interactive services by enabling the interoperable use of 3D content across the industry.
3. Pixar's [Universal Scene Description](https://developer.nvidia.com/usd): USD is an easily extensible, open-source 3D scene description and file format developed by Pixar for content creation and interchange among different tools. As a result of its power and versatility, it’s being widely adopted, not only in the visual effects community, but also in architecture, design, robotics, manufacturing, and other disciplines.
4. Blender; [www.blender.org](http://www.blender.org)

Face description

1. [FACS-based Facial Expression Animation in Unity](https://github.com/dccsillag/unity-facs-facial-expression-animation)

1. [Facial Action Coding System](http://web.cs.wpi.edu/~matt/courses/cs563/talks/face_anim/ekman.html)

1. [Facial Action Coding System](https://www.paulekman.com/facial-action-coding-system/)
2. https://www.noldus.com/applications/facial-action-coding-system
3. <https://3d.kalidoface.com/>
4. [List of motion and gesture file formats](https://en.wikipedia.org/wiki/List_of_motion_and_gesture_file_formats)
5. <https://docs.readyplayer.me/>
6. https://zivadynamics.com/zrt-face-trainer

Audio objects

1. <https://en.m.wikipedia.org/wiki/Ambisonics>
2. <https://docs.enklu.com/docs/Assets/Audio>
3. https://docs.unity3d.com/Manual/AmbisonicAudio.html

Web of Things

1. https://www.w3.org/WoT/
2. <http://model.webofthings.io/>
3. <https://iot.mozilla.org/wot/>

Presentation/rendering

1. Unity
2. WebXR
3. UnrealEngine
4. [Unreal Engine – Audio to Facial Animation SDK](https://www.unrealengine.com/marketplace/en-US/product/digital-avatar-service-link)
5. <https://developer.nvidia.com/nvidia-omniverse-platform>
6. [Microsoft Mesh](https://www.microsoft.com/en-us/mesh)

Companies

MetaVRse Alan Smithson

Gabriel René

Pascal BORNET - Chief Data Officer - Aera Technology

3d animation skinning bone based animation web 3d coding mesh compression

DensePose (1 & 2), OpenPose

# Use Cases

## Client-Based Ambient

This use case assumes that Clients have sufficient capabilities to process the data generated by humans and build the Ambient according to the instructions of the human.

### TX Client AIW

#### Function

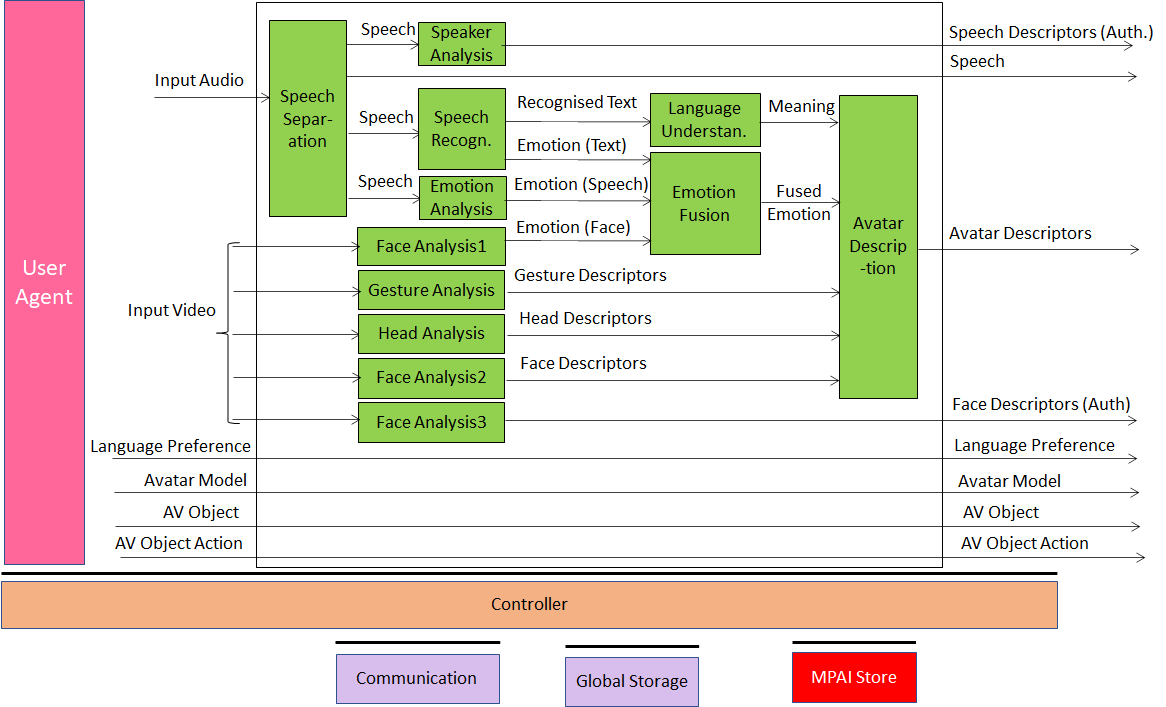
The function of the AIW is to

1. Receive:
   1. Local Audio (that includes Audio field created by RX Client) from microphone (array).
   2. Video of participant and from a camera (array).
   3. Participant’s Avatar Model.
   4. Participant’s colour and style of synthetic voice (e.g., for use in speech translation).
   5. Participant’s spoken language preferences (e.g., EN-US, IT-CH).
   6. Non-real-time Audio-Visual Objects from a storage.
   7. Real-time Audio-Visual Objects from a device.
   8. Actions on AV Objects.
2. Send to server for:
3. Distribution to Participants, possibly in encrypted form:
   1. Speech
   2. Participant’s Avatar Model.
   3. Participant’s colour and style of synthetic voice (e.g., for use in speech translation).
   4. Participant’s spoken language preferences (e.g., EN-US, IT-CH).
   5. Avatar Descriptors
   6. AV Objects
   7. Actions on AV Objects.
4. Authentication
5. Speaker Authentication Descriptors
6. Face Authentication Descriptors.

#### Architecture

Each participant

1. At the start, sends to Server
   1. For distribution to Participants:
      1. Language Preference
      2. Avatar model.
      3. Non-real time AV Objects.
   2. For authentication
      1. Speaker Descriptors for Authentication
      2. Face Descriptors for Authentication
2. During the meeting:
   1. Speech Separation produces Participant’s Speech
   2. Speech Recognition
      1. Receives Speech
      2. Produces Recognised Text and Emotion (Text)
   3. Emotion Analysis
      1. Receives Speech
      2. Produces Emotion (Speech)
   4. Face Analysis1
      1. Receives Video of Face
      2. Produces Emotion (Video) and Meaning (Video)
   5. Face Analysis2
      1. Receives Video of Face
      2. Produces Face Descriptors
   6. Head Analysis
      1. Receives Video of Face
      2. Produces Head Descriptors
   7. Gesture Analysis
      1. Receives Video of Arms and Hands
      2. Produces Gesture Descriptors
   8. Face Analysis3
      1. Receives Video of Face
      2. Produces Face Authentication Descriptors
   9. Language Understanding
      1. Receives Text and Recognised Text
      2. Produces Meaning (Text) and Meaning (Speech)
   10. Emotion Fusion
       1. Receives Emotion (Text), Emotion (Speech) and Emotion (Video)
       2. Produes Fused Emotion
   11. Question Analysis
       1. Receives Meaning (Text) and Meaning (Video)
       2. Produes Fused Meaning
   12. Avatar Description
       1. Receives, Fused Emotion, Fused Meaning, Face Descriptors, Head Descriptors and Gesture Descriptors
       2. Produces Avatar Descriptors
   13. Participant sends to Server
       1. Real Time AV Object
       2. Real Time AV Object Actions
3. TX Clients sends to Server for distribution
   1. Speech.
   2. Avatar Descriptors.
   3. Real time 3D AV Object.
   4. Actions on AV Objects.



*Figure 3 – Client-Based Ambient TX Client Reference Model*

*Table 2* gives the input and output data of TX Client AIW:

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

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Language Preference | The language Participant wishes to hear in the Ambient |
| Input Audio | Audio including Participant’s Speech, RX Client Audio and other audio |
| Input Video | Video of Participant’s torso (hade, face, arms, hands). |
| Avatar Model | The avatar model selected by Participant. |
| Audio-Visual Object | A non-real-time Audio-Visual Object a Participant may send in advance. |
| Audio-Visual Object Actions | Commands that an originator can and any participants may (if authorised) send to act on an Audio-Visual Object. |
| **Output** | **Comments** |
| Language Preference | As in input. |
| Speaker Descriptors (Auth.) | Participant’s Speech Descriptors for Authentication |
| Speech | Participant’s Speech as separated from Ambient Audio. |
| Avatar Descriptors | As in input. |
| Face Descriptors (Auth.) | Participant’s Face Descriptors for Authentication |
| Audio-Visual Object | As in input. |
| Audio-Visual Object Action | As in input. |

#### AI Modules

*Table 3* gives the AI Modules of TX Client AIW.

*Table 3 – AI Modules of TX Client AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Speech Separation** | Provides Speech from non-speech Sound in Input Audio. |
| **Speaker Analysis** | Provides Speaker’s Speech Descriptors for Authentication |
| **Speech Recognition** | Provides Text and Emotion from Separated Speech. |
| **Emotion Analysis** | Provides Emotion from Speech. |
| **Face Analysis1** | Provides Emotion and Meaning from Input Video (face). |
| **Face Analysis2** | Provides Face Descriptors for reproduction of face on avatar. |
| **Head Analysis** | Provides Head Descriptors for reproduction of head on avatar. |
| **Gesture Analysis** | Provides Gesture Descriptors for reproduction of arms and hands on avatar. |
| **Face Analysis3** | Provides Face Descriptors for Authorisation. |
| **Language Understanding** | Provides Meaning from Recognised Text and Input Text. |
| **Emotion Fusion** | Provides the Fused Emotion from Speech and Face Emotion. |
| **Question Analysis** | Produces Fused Meaning. |
| **Avatar Description** | Produces the full set of Avatar Descriptions. |

### Server AIW

#### Function

The function of the Server AIW is:

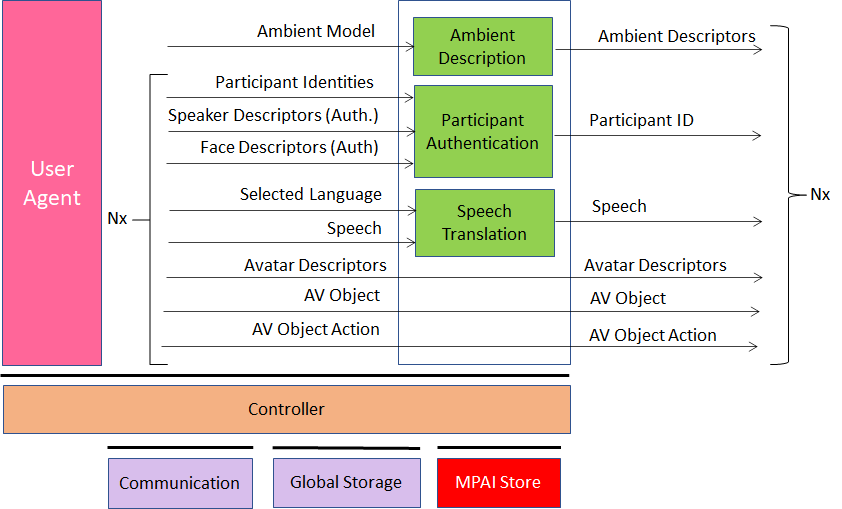
1. At the start, to create the Ambient based on
   1. Hosting Organisation inputs (Ambient).
   2. Participant inputs (static Avatars, AV Objects).
2. During the lecture:
   1. to create and send the full set of AV dynamic Descriptors of the Avatars in the Ambient and their movements.
   2. to forward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

#### Architecture

The Server:

1. Receives from:
   1. Server manager:
      1. Selected Ambient.
      2. Participants’ Identities.
   2. Each Participant:
      1. Speaker Descriptors (Auth.)
      2. Face Descriptors (Auth.)
      3. Selected Language
      4. Speech.
      5. Avatar Descriptors.
      6. AV Objects.
      7. AV Object Actions.
2. Authenticates Participants
3. Sends:
   1. Ambient Descriptors.
   2. All Participant IDs
   3. Speeches.
   4. Avatar visual Descriptors.
   5. AV Objects.
   6. Actions on AV Objects.

*Figure 4* gives the architecture of Server AIW.



*Figure 4 – Client-Based Ambient Server Reference Model*

#### I/O data

*Table 4* gives the input and output data of Server AIW.

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

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Ambient Selection | From Hosting Organisation |
| Participant Identities (xN) | From Hosting Organisation |
| Speaker Descriptors (Auth.) (xN) | Participant’s Speech Descriptors for Authentication |
| Face Descriptors (Auth.) (xN) | Participant’s Face Descriptors for Authentication |
| Selected Language (xN) | From all participants |
| Speech (xN) | From all participants |
| Avatar Descriptors (xN) | From all participants |
| AV Object (xN) | From participants wishing to do so |
| AV Object Actions (xN) | From participants wishing to do so |
| **Outputs** | **Comments** |
| Ambient Descriptors (xN) | Static Ambient Descriptors |
| Participant ID (xN) | Participant ID |
| Speech (xN) | Participant Speech |
| Avatar Descriptors (xN) | Avatars Descriptors |
| AV Objects (xN) | Real time AV objects |
| AV Object Actions (xN) | Commands to act on AV objects |

#### AI Modules

*Table 5* gives the AI Modules of Server AIW.

*Table 5 – AI Modules of Server AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Ambient Description** | Creates all static Ambient Descriptors. |
| **Participant Authentication** | Authenticates Participants using their Speech and Face Descriptors |
| **Translation** | 1. Selects an active speaker. 2. Translates the Speech of that speaker to the set of translated Speeches of the Selected Languages. 3. Assigns a translated Speech to the appropriate set of Participants. |

### RX Client AIW

#### Function

The Function of the RX Client AIW is to:

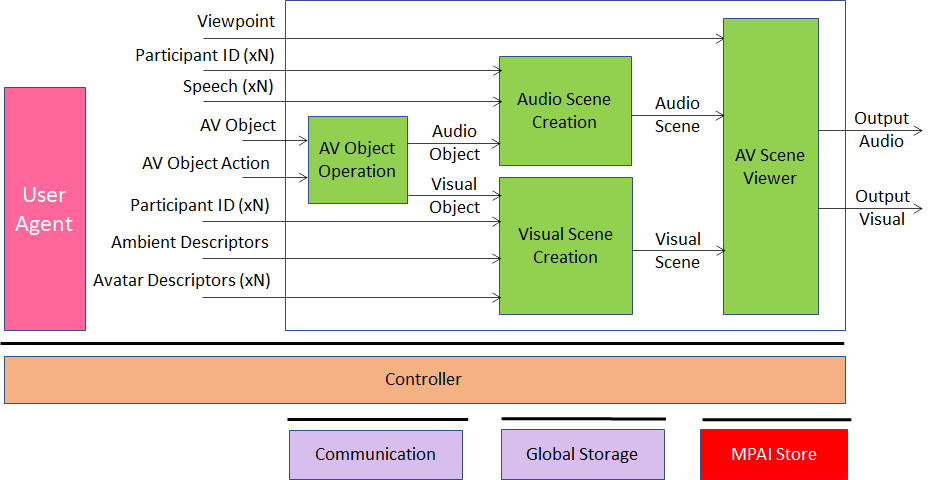
1. Create the virtual AV e-learning scene.
2. Let Participant act on AV Object(s).
3. Merge the virtual AV e-learning scene with the AV Object(s).
4. Present the merged AV scene.
5. Let Participant navigate the merged AV scene.

#### Architecture

The RX Client AIW:

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

*Figure 5* gives the architecture Client RX AIW.



*Figure 5 – Client-Based Ambient RX Client Reference Model*

#### I/O data

*Table 6* gives the input and output data of RX Client AIW.

*Table 6 – Input and output data of RX Client AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Viewpoint | Participant-selected Viewpoint to see and hear Ambient |
| Participants’ IDs (xN) | Unique Participants’ IDs |
| Speech (xN) | Participant’s Speech |
| AV Object | Nor-real-time and real-time AV Object |
| AV Object Action | Commands to act on AV Object |
| Ambient Descriptors | Static Descriptors of Ambient |
| Avatar Descriptors (xN) | Descriptors of Avatars bodies |
| **Output** | **Comments** |
| Output Audio | To be reproduced with loudspeaker array |
| Output Visual | To be reproduced with 2D or 3D display |

### AI Modules

*Table 7* gives the AI Modules of RX Client AIW.

*Table 7 – AI Modules of Client-Based Ambient*

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

## Server-Based Ambient

### TX Client AIW

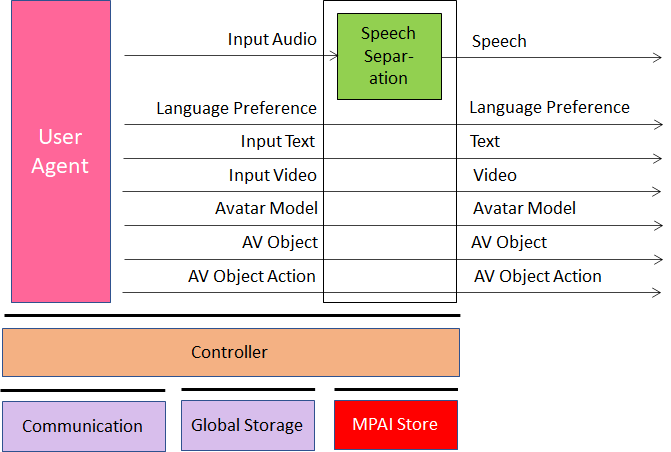
#### Function

The function of the AIW is to

1. Receive:
   1. Local Audio (that includes Audio field created by RX Client) from microphone (array).
   2. Video of participant and from a camera (array).
   3. Participant’s Avatar Model.
   4. Participant’s spoken language preferences (e.g., EN-US, IT-CH).
   5. Non-real-time Audio-Visual Objects from a storage.
   6. Real-time Audio-Visual Objects from a device.
   7. Actions acting on AV Objects.
2. Send to server for distribution to Participants:
3. Speech
4. Participant’s Avatar Model.
5. Participant’s spoken language preferences (e.g., EN-US, IT-CH).
6. Avatar Descriptors
7. AV Objects
8. Actions acting on AV Objects.

#### Architecture

*Figure 6* gives the architecture of TX Client AIW.



*Figure 6 – Server-Based Ambient TX Client Reference Model*

TX Client:

1. Receives Local Audio
2. Receives and sends
   1. Speech
   2. Language Preference
   3. Video
   4. Avatar Model.
   5. AV Objects.
   6. AV Objects Actions.

#### I/O data

*Table 8* gives the input and output data of TX Client AIW:

*Table 8 – Input and output data of TX Client AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Language Preference | The language Participant wishes to hear from a Participant. |
| Input Audio | Audio including Participant’s Speech, RX Client Audio and other audio |
| Input Video | Video of Participant’s torso (hade, face, arms, hands). |
| Avatar model | The avatar model selected by Participant. |
| AV object | A non-real-time AV objects a Participant may send in advance. |
| AV object Actions | Commands that an originator can and any participants may (if authorised) send to act on a AV object. |
| **Output** | **Comments** |
| Language Preference | As in input. |
| Speech | Participant’s Speech as separated from Ambient Audio. |
| Avatar Model | As in input. |
| AV Object | As in input. |
| AV Object Actions | As in input. |

#### AI Modules

*Table 9* gives the AI Modules of TX Client.

*Table 9 – AI Modules of TX Client AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Speech Separation** | Provides Speech separated from non-speech Audio in Input Audio. |

### AIW of Server

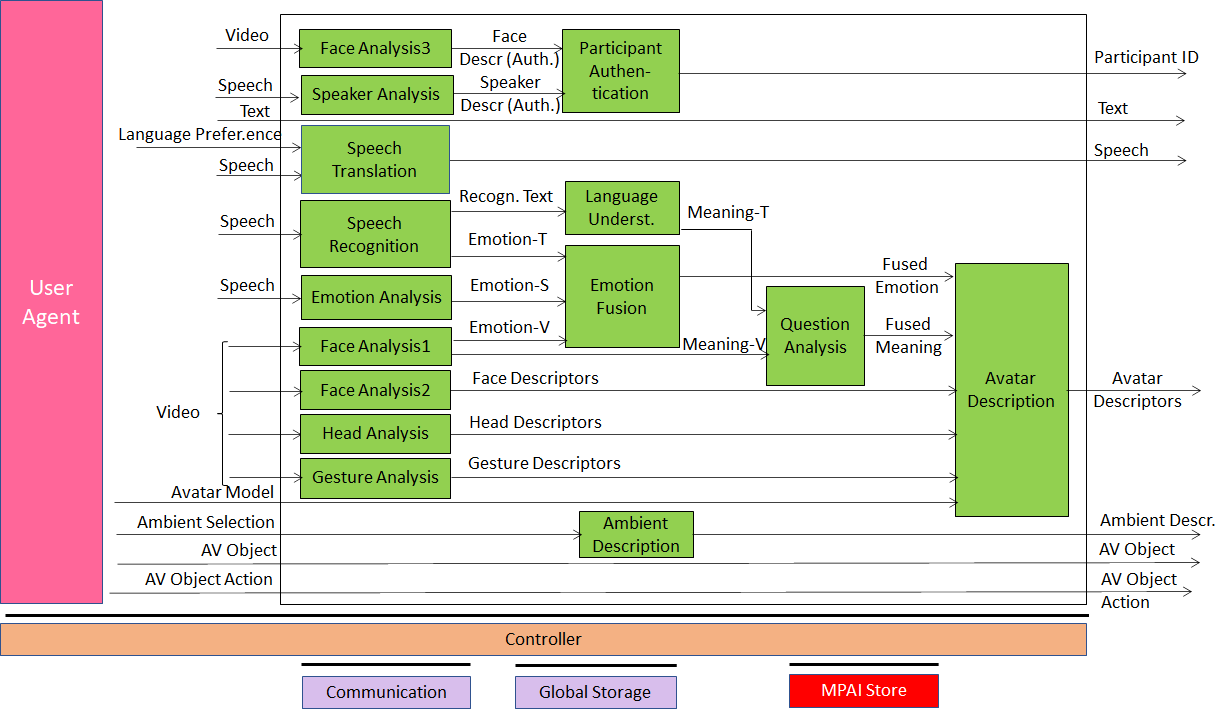
#### Function

The function of Server AIW is:

1. At the start:
   1. Receives:
      1. Ambient selection
      2. Language Preference
      3. Speaker Description (Authentication)
      4. Face Description (Authentication)
      5. Avatar Models (xN).
      6. AV Objects.
   2. Produces and sends Participant Indentities
2. Continuously performs
   1. Speech Translation
   2. Participant Authentication
   3. Avatar Descriptor Creation
3. Continuously sends:
   1. Speech
   2. AV Objects.
   3. AV Actions.

#### Architecture

*Figure 7* gives the architecture of Server AIW.



*Figure 7 – Server-Based Ambient Server Reference Model*

Server:

1. Receives from conference manager
   1. Selected Ambient.
2. Receives from each participant:
   1. Language Preference
   2. Speaker Description (Authentication)
   3. Face Description (Authentication)
   4. Avatar Models (xN).
   5. Speech.
   6. AV Objects.
   7. AV Object Actions.
3. Performs:
   1. Produces Ambient Descriptors.
   2. Authenticates Participants.
   3. Translates Speech to Requested Languages.
   4. Recognises Speech.
   5. Extracts Emotion (Text) from Text.
   6. Extract Emotion (Speech) from Emotion Analysis.
   7. Extracts Emotion (Video) from Video.
   8. Extracts Face Descriptors from Video.
   9. Extracts Head Descriptors from Video.
   10. Extracts Gesture Descriptors from Video.
   11. Extracts Meaning (Text) and Meaning (Speech) from Text.
   12. Fuses Emotion (Text), Emotion (Speech) and Emotion (Video).
   13. Fuses Meaning (Text), Meaning (Speech) and Meaning (Video).
   14. Produces Avatar Descriptors.
   15. Produces Ambient Descriptors.
4. Sends to all Participants the following entities of all Participants
   1. Speech.
   2. Participant ID.
   3. Ambient Descriptors.
   4. Avatar Descriptors.
   5. AV Objects.
   6. Actions on AV Objects.

#### I/O data

*Table 10* gives the input and output data.

*Table 10 – Input and output data of Server AIW*

|  |  |
| --- | --- |
| **Input** | **Comments** |
| Ambient Selection | From Server manager |
| Language Preference | From all participants |
| Speech (xN) | From all participants |
| Video (xN) | From all participants |
| Avatar Models (xN) | Avatar Model selected by Participant |
| AV Object (xN) | From all participants wishing to do so |
| AV Object Actions (xN) | From all participants wishing to do so |
| **Outputs** | **Comments** |
| Ambient Descriptors | Static Descriptors of Ambient |
| Speech (xN) | Participant’s or Translated Speech |
| Participant ID (xN) | Static Participant ID |
| Avatar Descriptors (xN) | Descriptors of Avatar |
| AV Objects | Real time AV Objects |
| AV Object Actions (xN) | Commands to act on AV Objects |

#### AI Modules

*Table 11* gives the AI Modules of Server AIM.

*Table 11 – AI Modules of Server AIW*

|  |  |
| --- | --- |
| **AIM** | **Function** |
| **Ambient Description** | Creates Ambient Descriptors for presentation. |
| **Speech Translation** | Translates a Speech to a Selected Language |
| **Emotion Analysis** | Provides Emotion from Speech. |
| **Speaker Analysis** | Provides Speaker’s Speech Descriptors for Authentication |
| **Face Analysis3** | Provides Face Descriptors for Authorisation. |
| **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. |
| **Head Analysis** | Provides Head Descriptors for reproduction of head on avatar. |
| **Gesture Analysis** | Provides Gesture Descriptors for reproduction of arms and hands on avatar. |
| **Language Understanding** | Provides Meaning from Recognised Text and Input Text. |
| **Participant Identification** | Determines Identity of all Participants to their Speeches |
| **Emotion Fusion** | Provides the Final Emotion from Speech and Face Emotion. |
| **Question Analysis** | Produces Fused Meaning. |
| **Ambient Description** | Produces the Descriptors of the Ambient. |
| **Avatar Description** | Produces the Descriptors of the its Avatars. |

### RX Client AIW

The RX Client AIW is the same as for Client-Based Ambient.

# AI Modules

## AIMs and their data

### Client-Based Ambient

#### TX Client

*Table 12 – AIMs and Data of TX Client AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
|  | Language Preference | Language Preference |
| **Speech Separation** | Input Audio | Separatated Speech |
| **Speaker Analysis** | Speech | Speech Descriptors (Auth.) |
| **Speech Recognition** | Text  Speech | Recognised Text  Emotion (Text) |
| **Emotion Analysis** | Speech | Emotion (Speech) |
| **Face Analysis1** | Input Video | Emotion (Video)  Meaning (Video) |
| **Face Analysis2** | Input Video | Face Descriptors |
| **Head Analysis** | Input Video | Head Descriptors |
| **Gesture Analysis** | Input Video | Gesture Descriptors |
| **Face Analysis3** | Input Video | Face Descriptors (Auth.) |
| **Language Understanding** | Text  Recognised Text | Meaning (Text)  Meaning (Speech) |
| **Emotion Fusion** | Emotion (Text)  Emotion (Speech)  Emotion (Video) | Fused Emotion |
| **Question Analysis** | Meaning (Text)  Meaning (Speech)  Meaning (Video) | Fused Meaning |
| **Avatar Description** | Fused Emotion  Fused Meaning  Face Descriptors  Head Descriptors  Gesture Descriptors | Avatar Descriptors |
|  | Avatar Model | Avatar Model |
|  | AV Object | AV Object |
|  | AV Object Action | AV Object Action |

#### Server

*Table 13 – AIMs and Data of Server AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **Ambient Description** | Ambient Type | Ambient Descriptors |
| **Participant Authentication** | Participant Identities  Speech Descriptors (Auth.)  Face Descriptors (Auth.) | Participant ID (xN) |
| **Translation** | Selected Languages  Speech | Speech |
|  | Avatar Descriptors | Avatar Descriptors |
|  | AV Object | AV Object |
|  | AV Object Navigation | AV Object Navigation |

#### RX Client

*Table 14 – AIMs and Data of RX Client AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input** | **Output Data** |
| **AV Object Operation** | AV Object | Audio Object |
| AV Object Action | Visual Object |
| **Audio Scene Creation** | Participant ID | Audio Scene |
| Speech |
| Audio Object |
| **Visual Scene Creation** | Visual Object | Visual Scene |
| Participant ID |
| Ambient Descriptors |
| Avatar Descriptors |
| **AV Scene Viewer** | Viewpoint | Output Audio  Output Visual |
|  | Audio Scene |
| Visual Scene |

### Server-Based Ambient

#### TX Client

*Table 15 – AIMs and Data of TX Client AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **Speech Separation** | Input Audio | Speech |
|  | Language Preference | Language Preference |
|  | Input Video | Input Video |
|  | Avatar Model | Avatar Model |
|  | AV Object | AV Object |
|  | AV Object Action | AV Object Action |

#### Server

*Table 16 – AIMs and Data of Server AIW*

|  |  |  |
| --- | --- | --- |
| **AIM** | **Input Data** | **Output Data** |
| **Face Analysis3** | Input Video | Face Descriptors (Auth.) |
| **Speaker Analysis** | Speech | Speech Descriptors (Auth.) |
| **Translation** | Language Preference  Speech | Speech |
| **Speech Recognition** | Text  Speech | Recognised Text  Emotion (Text) |
| **Emotion Analysis** | Speech | Emotion Speech |
| **Speech Separation** | Input Audio | Separatated Speech |
| **Face Analysis1** | Input Video | Emotion (Video)  Meaning (Video) |
| **Face Analysis2** | Input Video | Face Descriptors |
| **Head Analysis** | Input Video | Head Descriptors |
| **Gesture Analysis** | Input Video | Gesture Descriptors |
| **Participant Authentication** | Face Descriptors (Auth.)  Speech Descriptors (Auth.) | Participant ID |
| **Language Understanding** | Text  Recognised Text | Meaning (Text)  Meaning (Speech) |
| **Emotion Fusion** | Emotion (Text)  Emotion (Speech)  Emotion (Video) | Fused Emotion |
| **Question Analysis** | Meaning (Text)  Meaning (Speech)  Meaning (Video) | Fused Meaning |
| **Avatar Description** | Fused Emotion  Fused Meaning  Face Descriptors  Head Descriptors  Gesture Descriptors | Avatar Descriptors |
| **Ambient Description** | Ambient Selection | Ambient Descriptors |
|  | AV Object | AV Object |
|  | AV Object Action | AV Object Action |

#### Client RX

See 6.1.1.3.

## 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** |
| Ambient Descriptors | 6.2.1 |  |
| Ambient Model | 6.2.2 |  |
| Audio-Visual Object | 6.2.3 |  |
| Audio-Visual Object Action | 6.2.4 |  |
| Audio-Visual Scene | 6.2.5 |  |
| Avatars Descriptors | 6.2.6 |  |
| Avatar Model | 6.2.7 |  |
| Emotion (Speech) | 6.2.8 |  |
| Emotion (Text) | 6.2.9 |  |
| Emotion (Video) | 6.2.10 |  |
| Face Descriptors | 6.2.11 |  |
| Face Descriptors (Auth) | 6.2.12 |  |
| Fused Emotion | 6.2.13 |  |
| Fused Meaning | 6.2.14 |  |
| Gesture Descriptors | 6.2.15 |  |
| Head Descriptors | 6.2.16 |  |
| Input Audio | 6.2.17 |  |
| Input Video | 6.2.18 |  |
| Meaning (Speech) | 6.2.19 |  |
| Meaning (Text) | 6.2.20 |  |
| Meaning (Video) | 6.2.20 |  |
| Output Audio | 6.2.21 |  |
| Output Visual | 6.2.22 |  |
| Participant IDs | 6.2.23 |  |
| Recognised Text | 6.2.24 |  |
| Speech Descriptors (Auth.) | 6.2.25 |  |
| Text | 6.2.26 |  |
| Viewpoint | 6.2.27 |  |

### Ambient Descriptors

The set of Descriptors required to represent and render the static components of the Ambient:

1. Visual Objects, e.g., table, swivel chair, walls, furniture, with their affordances.
2. Audio Objects: e.g., an audio source at given coordinates in the Ambient.

The Descriptors should allow:

1. Integration of animated avatars and AV Objects in the Ambient.
2. Addition/Removal/Displacement of Audio and Visual Objects to/from/in the Ambient.
3. Association of an Audio Object to a Visual Object.
4. Viewing and hearing the AV Scene from an arbitrarily selected Viewpoint.

**To Respondents**

MPAI Requests a set of Descriptors the would allow the digital representation and rendering of a static Ambient with the features described and with the ability to accommodate other dynamic Audio and Visual Descriptors of this Section.

### Ambient Model

The Ambient Model is the file format of the Ambient. The Ambient Model has

1. The visual component represented as Industry Foundation Classes (IFC)/BIM files (see [0]).
2. The Audio component populated with static audio sources.

**To Respondents**

MPAI requests respondents:

1. To comments on the formats for the visual component or to propose an alternative format.
2. To propose a format for the audio component.

### Audio-Visual Object

The coded representation of an object containing Audio and Visual Objects. Can be static or real time, i.e., generated by a device, e.g., a digital microscope.

**To Respondents**

MPAI requests respondents to propose an AV Object than can be streamed and recorded.

### Audio-Visual Object Action

An instruction to act on an AV Object that allows a user to execute actions on a AV Object.

**To Respondents**

MPAI requests proposals for the following set of instructions:

1. Manual or automatically define a portion of the AV object.
2. Count objects per assigned volume size.
3. Detect structures in a (portion of) the AV object.
4. Deform/sculpt the AV object.
5. Combine AV objects.
6. Assign a criterion to an anomaly detector on a portion of the AV object.
7. Create a link between a part to a different part of the AV object.
8. Follow a link to another portion of the object.
9. Print (a portion of) the visual part of the AV object.
10. Assign physical properties to (a part of) the AV object.
11. Annotate a portion of the AV object.

### Audio-Visual Scene

The structured composition of AV Objects in an Ambient.

**To Respondents**

MPAI requests respondents to propose an AV Scene format capable of representing both static and dynamic AV Objects in an Ambient.

### Avatars Descriptors

A set of Descriptors allowing animation of an Avatar Model. The Model should at least be composed of Head, Face, Arms and Hands.

**To Respondents**

MPAI requests respondents to propose a set of Descriptors supporting the stated requirements.

### Avatar Model

The static model of an Avatar capable to have its Head, Face, Arms and Hands (gesture) animated, based on the data formats requested in Subsections 6.2.11, 6.2.16 and 6.2.15.

**To Respondents**

MPAI requests respondents to propose an avatar model satisfying the stated requirements.

### Emotion (Speech)

The coded representation of the emotion embedded in a Speeech.

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

**To Respondents**

Respondents are requested:

1. To comment on the use of [4] to represent emotion embedded in speech in the MPAI-MCS context, or
2. To propose extensions or alternative formats for emotion embedded in speech.

### Emotion (Text)

The coded representation of the emotion embedded in a Text.

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

**To Respondents**

Respondents are requested:

1. To comment on the use of [4] to represent emotion embedded in ta ext in the MPAI-MCS context, or
2. To propose extensions or alternative formats for emotion embedded in text.

### Emotion (Video)

The coded representation of the emotion expressed by a Face.

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

**To Respondents**

Respondents are requested:

1. To comment on the use of [4] to represent emotion embedded in video in the MPAI-MCS context, or
2. To propose extensions or alternative emotion embedded in video.

### Face Descriptors

Descriptors of a face are used to animate the face of an Avatar such that it is a faithful representation of the Participant’s face, especially as far as emotion is concerned.

The Facial Action Coding System (FACS) [8, 13, 14, 15] is an anatomically based system capable of describing all facial movements. Facial expressions are broken down into Action Units, elementary components of muscle movement.

**To Respondents**

MPAI requests respondents to comment on the use of FACS in the MPAI-MCS context or to propose alternative formats capable of describing face.

### Face Descriptors (Auth)

The Descriptors of a face used to recognise the Participant’s identity.

**To Respondents**

MPAI requests respondents to propose a set of descriptors suitable to recognise a Participant’s face.

### Fused Emotion

Fused Emotion is the Emotion resulting by combining any of the following: Emotion (Text), Emotion (Speech) and Emotion (Video) in order to to improve the animation of an avatar face.

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

**To Respondents**

MPAI requests respondents to comment on the use of [4] in the MPAI-MCS context or to propose extensions or alternative Emotion representation formats.

### Fused Meaning

Fused Meaning is the meaning resulting from the fusion of any of the following: Meaning (Text) and Meaning (Video). This information is used to better animate the face of the avatar.

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

**To Respondents**

MPAI requests respondents to comment on the use of [4] in the MPAI-MCS context or to propose extensions or alternative Meaning representation formats.

### Gesture Descriptors

The coded representation of movement of hands and arms (gesture) suitable for:

1. Recognition of sign language.
2. Recognition of coded hand signs for navigation
3. Representation of arbitrary arm and hand motion.

**To Respondents**

MPAI requests respondents to propose a set of gesture descriptors satisfying the requirements.

### Head Descriptors

The coded representation of the movement of a human head to be used to animate the head of an avatar.

**To Respondents**

MPAI requests respondents to propose a set of head descriptors satisfying the requirements.

### Input Audio

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

MPAI-CAE has defined a digital representation of a microphone set [3].

**To Respondents**

MPAI requests respondents to

1. To comment on the use of [3] to capture audio from a Participant’s location in an MPAI-MCS context for the purpose of separating speech from the captured audio, or
2. To propose extensions to [3], or
3. To propose alternative representation formats.

### Input Video

Input Video is the digital visual representation of the Participant’s torso, i.e., the upper part of the body for the purpose of digitally representing head, face, arms and hands of a Participant.

**To Respondents**

MPAI requests respondents to propose an Input Video format that facilitates the optimal extraction of head, face, and gesture descriptors.

### Language Preferences

Languages are expressed as specified by [5].

**To Respondents**

MPAI requests respondents to comment on the choice.

### Meaning (Text)

The coded representation of the meaning embedded in a Text.

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

**To Respondents**

MPAI requests respondents to comment on the use of [4] in the MPAI-MCS context or to propose extensions or alternative Meaning (Text) representation formats.

### Meaning (Video)

The coded representation of the meaning embedded in a Video.

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

**To Respondents**

MPAI requests respondents to comment on the use of [4] in the MPAI-MCS context or to propose extensions or alternative Meaning (Video) representation formats.

### Output Audio

The Audio field generated by the RX Client.

MPAI-MMC has defined Multichannel Audio in [3].

**To Respondents**

MPAI requests respondents to comment on the use of [3] in the MPAI-MCS context or to propose extensions or alternative Multichannel Audio representation formats.

### Output Visual

The Visual field generated by RX Client.

**To Respondents**

MPAI requests respondents to propose a data format appropriate to present a visual scene to a Participant.

### Participant IDs

Dynamic (session-by-session) Participant Identifier.

**To Respondents**

MPAI requests respondents to propose a session-dependent individual participant identification system based on participant identities known by the service.

### Recognised Text

Text should be digitally represented according to [6].

**To Respondents**

MPAI requests respondents to comment on the use of [6] in the MPAI-MCS context.

### Speech Descriptors (Auth.)

Descriptors of a speech used to recognise the Participant’s identity.

**To Respondents**

MPAI requests respondents to propose a set of descriptors suitable to recognise a Participant’s speech.

### Viewpoint

The coordinates in the Ambient from where a Participant looks at the Ambient.

**To Respondents**

MPAI requests respondents to propose a coordinate set suitable for MPAI-MCS purposes.

# 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 **Error! Reference source not found.**, 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”.

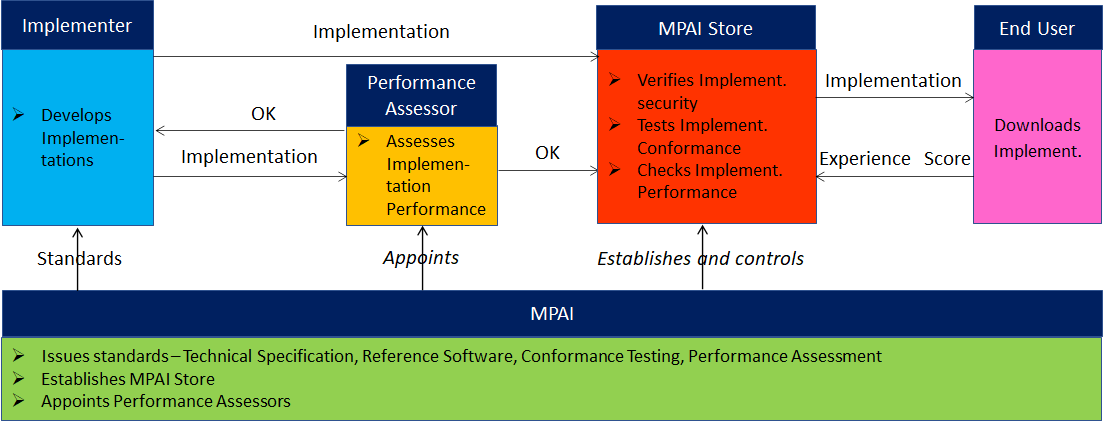
For Level 3, MPAI normatively specifies the process, the tools and the data or the characteristics of the data to be used to Assess the Grade of Performance of an AIM or an AIW.

|  |  |
| --- | --- |
| 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 *Figure 8* is a 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.
2. MPAI appoints Performance Assessors.
3. MPAI publishes Standards.
4. Implementers submit Implementations to Performance Assessors.
5. If the Implementation Performance is acceptable, Performance Assessors inform Implementers and MPAI Store.
6. Implementers submit Implementations to the MPAI Store
7. MPAI Store verifies security and Tests Confor­mance of Implementation.
8. Users download Implementations and report their experience to MPAI .



*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.

# Annex 4 Example use cases

# Virtual e-learning (VEL)

A teacher holds a lecture to N students. The teacher and the students – 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. They may also discover, connect and activate devices capable to produce AV Objects to be shared, or retrieve and share 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.

The activities unfold as follows:

1. Participants provide and communicate to the Server:
   1. Avatar models, model capabilities and objects with their affordances.
   2. Initial position of avatrs and objects in the MCS-VEL space.
   3. Preferences
      1. Colour and style of their synthetic speech (e.g., used in speech trans­lation).
      2. Spoken language (e.g., EN-US, IT-CH).
2. The 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., Ambient furnishings.
3. The Server can
   1. Convert:
      1. Speech, text and gesture; e.g., speech to gesture, text to gesture, gesture to speech etc.
      2. 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 AV Object at the intended spatial locations.
4. During the lecture:
   1. The camera set of each participant extracts and sends the Server Descriptors of head, face, arms and hands:
   2. The microphone set of each participant
      1. Captures the Audio field of the participant’s room.
      2. Separates the speech from the rest of the Audio field
      3. Cancels the participant’s voice received back from the server.
      4. Extracts Speech Descriptors.
      5. Sends own Audio field, separated speech, speech descriptors to the Server.
   3. Teacher
      1. Locates and connects devices capable to produce AV Objects
      2. Sends AV Objects generated by devices (e.g., Visual Output of a digital microscope, Molecules captured as objects by an electronic microscope).
      3. Calls synthetic AV Objects from a DB in support of the lecture, e.g., the model of a mouse brain.
      4. Enters, navigates and acts on audio-visual objects.
      5. Starts an experiment, e.g. the principles of optics, gravity and its effects, docking of molecules in chemistry, the inside of an atom.
      6. Places physical (moving) objects on their desk for reproduction as objects at participants’ locations and interactive engagement.
      7. Acts on the object by using the teacher’s own sensing and actuating devices.
   4. Students:
      1. Act on objects, if given permission
      2. Issue teleportation commands.
   5. The Server:
      1. Combines head, face, arms, hand and speech Descriptors received from the individual VEL TX Clients.
      2. Translates a Participant’s speech to the specific languages of other Participants.
   6. Participants:
      1. Receive visual descriptors of Ambient, Objects, Avatars, AV Objects
      2. Compose the information according to their needs
      3. Navigate the AV scene.

# Local Avatar Videoconference

Today’s videoconference falls short from a satisfactory replacement to a physical meeting. Videoconference 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 fusion by individual participantsi s what makes a physical meeting irreplaceable.

At a typical meeting, people stay at a table – if that is the arrangements. Typically, they do not move and, if they do, it is for reasons that have not much to do with the meeting, such a helping oneself a cup of coffe. Someone may be leaving, temporarily or permanently and someone may be joining. Coffee may 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 representations of the participants in a visually and audibly 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 features of the participants: speech and facial expression, and movements of head, arms and hands.

The scene is represented by descriptors which are sent to participants in lieu of a bandwidth-consuming full 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 may attend but not speak, unless specially admitted.

The Local Avatar Videoconference is attended by personae each representing a participants with:

1. Their actual speech.
2. Avatars having:
   1. Static bodies.
   2. Heads animated according to participants’ head movement.
   3. Faces
      1. Animated by emotion and meaning extracted from participants’ faces.
      2. Corroborated by emotions and meanings extracted from participants’ speeches.
   4. Arms and hands animated according to participants’ gestures.
3. One or more devices capable to produce 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 Local Avatar Videoconference is composed of

1. N “TX Clients” which
   1. Send speech
   2. Send emotion and meaning Descriptors extracted from
      1. Speech
      2. Head and face
      3. Arms and hands.
   3. Send special messages, e.g., I am leaving, I am joining, I am requesting the floor
   4. Locate and connect AV Object-generating devices to the TX Client.
   5. Send AV Objects from device or from a database.
   6. Issue Actions to act on AV Objects.
2. A Server which
   1. Selects a meeting Ambient
   2. Sends Ambient Descriptors to Participants.
   3. Integrates speech emotion and meaning Descriptors with visual avatar Descrip­tors.
   4. Sends visual avatar Descriptors.
   5. Sends AV Objects.
   6. Translates speech from the speaker language to Participants’ languages.
3. N RX Clients which
   1. Create their *personal* visual space by using:
      1. Ambient visual Descriptors.
      2. Avatar visual descriptors.
      3. Visual Object resulting from actions at their location.
   2. Teleport their Avatars, e.g., to stand close to the AV object.
   3. Create their *personal* audio space by using
      1. Participants’ speeches located at the corresponding Avatars’ positions.
      2. Audio Object resulting from actions at their location.
   4. View and hear the resulting *personal* virtual AV space from the selected viewpoint.

# 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 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)