



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

MPAI Technical Specification

Portable Avatar Format (MPAI-PAF)

V1.5

WARNING

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

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

Readers are invited to review [Notices and Disclaimers](#).

Technical Specification

Portable Avatar Format (MPAI-PAF)

V1.5

Contents

1	Foreword	3
2	Introduction (informative).....	5
3	Scope	6
4	Definitions	6
5	References	7
5.1	Normative References	7
5.2	Informative References	7
6	AI Workflows.....	8
6.1	Technical Specifications.....	8
6.1.1	Videoconference Client Transmitter	10
6.1.2	Virtual Meeting Secretary	14
6.1.3	Avatar Videoconference Server	18
6.1.4	Videoconference Client Receiver.....	21
6.2	Reference Software	24
6.3	Conformance Testing	24
6.4	Performance Assessment.....	24
7	AI Modules.....	25
7.1	Technical Specifications.....	25
7.1.1	Audio-Visual Scene Creation.....	25
7.1.2	Portable Avatar Multiplexing.....	27
7.1.3	Audio-Visual Scene Rendering.....	30
7.1.4	PS-Face Interpretation.....	32
7.1.5	Avatar and Speech Rendering	33
7.1.6	PS-Gesture Interpretation.....	35
7.1.7	Entity Body Description.....	36
7.1.8	Personal Status Display	38
7.1.9	Entity Face Description.....	40
7.1.10	Response and Scene Rendering.....	42
7.1.11	Face Identity Recognition	45
7.1.12	Scene and Avatar Rendering	48
7.1.13	SubAIMs	49
7.1.14	JSON Metadata	49
7.1.15	Profiles	49
7.1.16	Conformance Testing	49
7.1.17	Face Personal Status Extraction	50
7.1.18	Service Participant Authentication.....	52
7.1.19	Gesture Personal Status Extraction	54
7.1.20	Visual Scene Creation	56
7.1.21	Portable Avatar Demultiplexing.....	58
7.2	Reference Software	60
7.3	Conformance Testing	60
7.4	Performance Assessment.....	61

8	Data Types.....	61
8.1	Technical Specifications.....	61
8.1.1	Avatar.....	61
8.1.2	Body Descriptors Object.....	62
8.1.3	Face Descriptors Object.....	63
8.1.4	Gesture Descriptors Object.....	66
8.1.5	Portable Avatar.....	67
8.2	Conformance testing.....	68
8.3	Performance Assessment.....	69

1 Foreword

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

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

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

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

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

MPAI operates on four solid pillars:

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

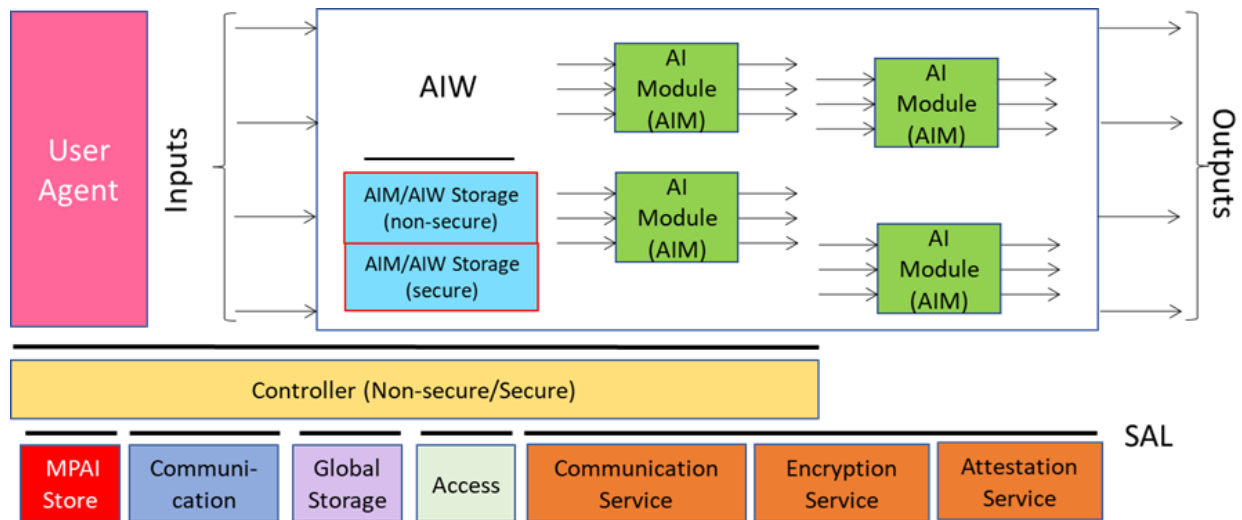


Figure 1 – The AI Framework (MPAI-AIF) V2 Reference Model

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

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

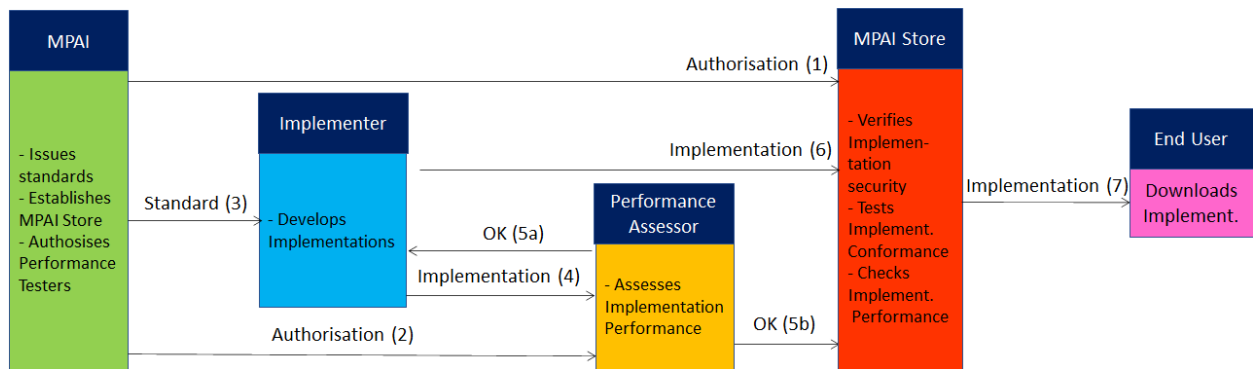


Figure 2 – The MPAI Ecosystem

2 Introduction (informative)

There is a long history of computer-created objects called “digital humans”, i.e., digital objects having a human appearance when rendered. In most cases the underlying assumption of these objects has been that creation, animation, and rendering is done in a closed environment, thus with little or no need for standards.

In a communication and more so in a metaverse context, there are many cases where a digital human is not constrained within a closed environment thus requiring levels of standardisation.

Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5 – in the following also called MPAI-PAF V1.5 or MPAI-PAF – is a response to the requirements of newer contexts where avatars are used. MPAI-PAF specifies a standard for Portable Avatar Format (PAF) enabling a receiving party to render a digital human as intended by the sending party.

MPAI-PAF V1.5 specifies the Avatar-Based Videoconference (PAF-ABV) AI Workflow where:

1. **Client Transmitters** send PAFs containing:
 - At the beginning: Avatar Models, Language Selector, and Speech Object and Face Object for participant authentication.
 - Continuously: Avatar Descriptors, and Speech Objects to a Server.
2. **Avatar Videoconference Server:**
 - At the beginning:
 - Selects an Environment, i.e., a meeting room and equips it with objects, i.e., meeting table and chairs.
 - Places Avatar Models around the table.
 - Distributes for each participant a PAF containing Environment, Avatar Models, and their positions to all receiving clients.
 - Continuously sends to receiving clients:
 - Translated Speech from participants according to Language Selectors.
 - Sends PAFs containing Avatar Descriptors and translated Speech.
3. **Client Receivers:**
 - At the beginning: receive Environment and PAFs containing Avatar Models and Language Selectors from the server.
 - Continuously from the server:
 - Receive PAFs containing Avatar Descriptors and translated Speech.
 - Create Audio and Visual Scene Descriptors.
 - Render the Audio-Visual Scene as seen from the human participant-selected Point of View.

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

3 Scope

Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5 specifies:

1. The *Portable Avatar Format* and related *Data Types* enabling a receiver to decode and render an Avatar and its environment as intended by the sender.
2. The *Personal Status Display* and other AI Modules enabling the interoperable handling of avatars and related data.
3. The AI Framework (MPAI-AIF)-conforming *AI Workflows and AI Modules* of the Avatar-Based Videoconference AI Workflow (AIW-ABV) also using Data Types and AI Modules specified by other MPAI Technical Specifications.

MPAI-PAF V1.5 includes the following chapters:

1. Scope (This chapter)
2. [Definitions](#)
3. [References](#)
4. [AI Workflows](#)
5. [AI Modules](#)
6. [Data Types](#)

The current version of MPAI-PAF has been developed by the Portable Avatar Format Development Committee (PAF-DC). MPAI may issue new versions of MPAI-PAF that extend or replace the scope of the current Technical Specification.

4 Definitions

Capitalised Terms have the meaning defined in [Table 1](#). All MPAI-defined Terms are accessible [online](#). Low-case terms have the meaning commonly defined for the context in which they are used or represent an entity in the real world as opposed to the corresponding Capitalised Term. For instance, Table 1 defines *Body* but does not define *body*

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

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

Table 1 – Terms and Definitions

Term	Definition
3D Model	Data representing the surface and relevant features of objects from the real world, or Media Data or a mixture of the two.
Avatar	A 3D Model representing a real or fictitious person or Data representing a real person.
- <i>Portable</i>	A Data Type conveying information related to an Avatar such as Avatar ID, Avatar Model, Body Descriptors, Face Descriptors, Speech Data, and Text, and Context information such as Time, Audio-Visual Scene, Spatial Attitude, Language Preference, and Personal Status.
Body	A digital representation of a human body, head included, face excluded.
- Descriptors	A Data Type representing the motion of the body of an Entity.

Face	The digital representation of a human face.
- Descriptors	A Data Type representing the motion and conveying information on the Personal Status of the face of an Entity.
- Identifier	The Identifier of a human inferred from analysing the face of that human.
Gesture	A Data Type representing the movement of the body or part of it, such as the head, arm, hand, and finger, often as a complement to a vocal utterance.
Human	A human being in a real space.
- <i>Digital</i>	A Digitised or a Virtual Human.
- <i>Digitised</i>	Data that has the appearance of a specific human when rendered.
- <i>Virtual</i>	Data created by a computer that has a human appearance when rendered but is not a Digitised Human.
Visual	A Data Type whose instance represents analogue signals – or is rendered to be perceived – in the human-visible range (380 to 700 nanometres).

5 References

5.1 Normative References

Technical Specification: Portable Avatar Format (MPAI-PAF) normatively references the following documents, both from MPAI and other standards organisations. Referenced MPAI standards are publicly available at the URL indicated in the reference.

1. MPAI; Technical Specification: [Governance of the MPAI Ecosystem](#) (MPAI-GME) V2.0.
2. MPAI; Technical Specification: [Artificial Intelligence Framework](#) (MPAI-AIF) V2.2.
3. MPAI; Technical Specification: [Context-based Audio Enhancement](#) - [Use Cases](#) (MPAI-CAE) V2.4.
4. MPAI; Technical Specification: [Object and Scene Description](#) (MPAI-OSD) V1.4.
5. MPAI; Technical Specification: [Profiles](#) (MPAI-PRF) - [AI Modules](#) (MPAI-PRF) V1.1.
6. MPAI; Technical Specification: [Data Types, Formats, and Attributes](#); V1.4.

5.2 Informative References

These references are provided for information purposes, e.g., because they normatively reference MPAI-PAF V1.5.

7. MPAI; [The MPAI Statutes](#).
8. MPAI; [The MPAI Patent Policy](#).
9. MPAI; Technical Specification: [Connected Autonomous Vehicles](#) (MPAI-CAV) – [Technologies](#) (CAV-TEC) V1.1.
10. MPAI; Technical Specification: [Human and Machine Communication](#) (MPAI-HMC) V2.1.
11. MPAI; Technical Specification: [Multimodal Conversation](#) (MPAI-MMC) V2.4.
12. MPAI; Technical Specification: [MPAI Metaverse Model](#) (MPAI-MMM) – [Technologies](#) V2.1.
13. MPAI; Technical Specification: [Object and Scene Description](#) (MPAI-OSD) V2.4.

6 AI Workflows

6.1 Technical Specifications

Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5, jointly with other MPAI Technical Specifications, provides technologies for the digital representation of 3D Model Data that enable the **Avatar-Based Videoconference**, a form of videoconference held in a Virtual Environment populated by speaking Avatars and implemented as an AI Workflow specified by [Technical Specification: AI Framework \(MPAI-AIF\) V2.2](#).

Table 1 lists the AIWs specified by MPAI-PAF V1.5. Links lead to the individual MPAI-PAF V1.5 AIW specifications. They include AIW Function, Reference Model, and I/O Data and, for each AIM, Functions, I/O Data, and links to the AIMs and JSON metadata related to the AI Workflow.

All previously specified MPAI-PAF AI-Workflows are superseded by those specified by V1.5 but may be used if their version is explicitly mentioned.

Table 1 - AIWs specified by MPAI-PAF V1.5

Acronym	Names and Specifications of AI Workflows	JSON
PAF-CTX	Videoconference Client Transmitter	X
MMC-VMS	Virtual Meeting Secretary	X
PAF-AVS	Avatar Videoconference Server	X
PAF-CRX	Videoconference Client Receiver	X

Figure 1 depicts the Avatar-Based Videoconference system composed of four subsystems (AI Workflows).

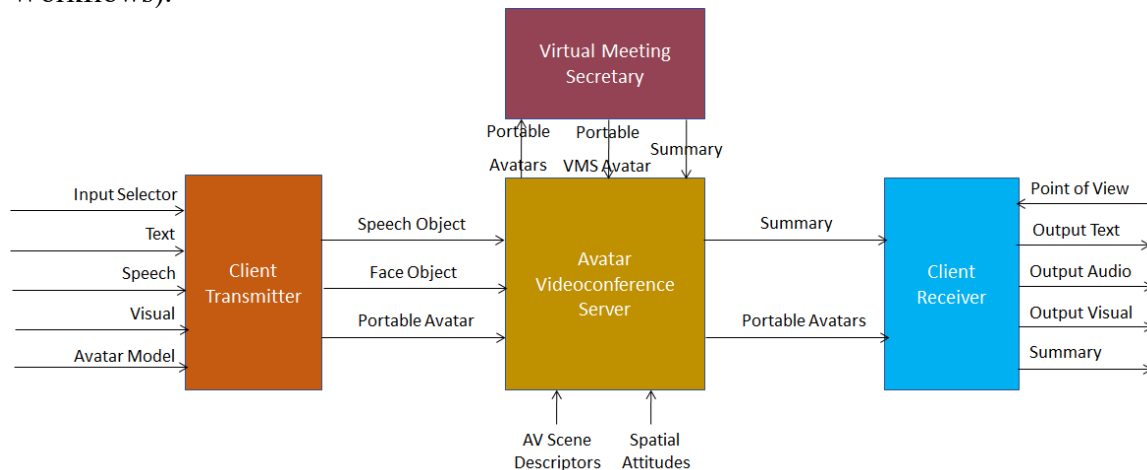


Figure 1 – Avatar-Based Videoconference end-to-end diagram

The components of the PAF-ABV system:

1. **Participant:** a human joining an ABV either individually or as a member of a group of humans in the same physical space.
2. **Audio-Visual Scene:** a Virtual Audio-Visual Environment populated by Visual Objects such as a Table, chairs etc., and Audio Objects. The Environment is described by Audio-Visual Scene Descriptors.
3. **Portable Avatar:** an MPAI-PAF-specified Data Type including data representing a human participant.
4. **Videoconference Client Transmitter (VCT):**

- At the beginning of the conference, VCT:
 - Receives from Participants and sends to the Server Portable Avatars containing Avatar Model(s) and Language Selectors.
 - Sends to the Server Speech Object and Face Object for Authentication.
 - Continuously updates the Server with Portable Avatars containing Avatar Descriptors and Speech.
5. The **Avatar Videoconference Server (AVS)**
- At the beginning of the conference, AVS:
 - Selects specific Audio-Visual Descriptors, representing the meeting room.
 - Equips the room with objects, e.g., table and chairs.
 - Places Avatar Models around the Table with a specific Spatial Attitude.
 - Distributes Portable Avatars containing Audio-Visual Scene Descriptors, Avatar Models, their Speech Objects and Spatial Attitudes to all Videoconference Client Receivers.
 - Assigns IDs to Avatars after authenticating Speech and Face Objects.
 - Sets the common conference language.
 - Continuously:
 - Translates and sends to Participants Speech according to their Language Selectors.
 - Sends Portable Avatars containing Avatar Descriptors, Speech, and Spatial Attitude of Participants and Virtual Meeting Secretary to all Receiving Clients and Virtual Meeting Secretary.
6. **Virtual Meeting Secretary** is represented as an Avatar but does not correspond to any Participant and continuously:
- Uses the common meeting language.
 - Understands Text Objects and Speech Objects of all Avatars and extracts their Personal Statuses.
 - Drafts a Summary of its understanding of Avatars' Text Objects, Speech Objects, and Personal Statuses.
 - Sends the Summary outside of the Virtual Environment for participants to read and edit directly, and/or displays the Summary in the Visual Space for Avatars to comment, e.g., via Text Objects.
 - Refines the Summary based on comments received.
 - Sends its Portable Avatar containing its Avatar Descriptors to the Server.
7. **Videoconference Client Receiver (VCR)**:
- At the beginning of the conference, the VCR:
 - Receives Audio-Visual Scene Descriptors and Portable Avatars containing Avatar Models with their Spatial Attitudes.
 - Continuously:
 - Receives Portable Avatars with Avatar Descriptors and Speech.
 - Produces Visual Scene Descriptors and Audio Scene Descriptors.
 - Renders the Audio-Visual Scene by spatially adding the Avatars' Speech Objects to the Spatial Attitude of the respective Avatars' Mouths. Rendering may be done from a Point of View, that is possibly different from the Position assigned to their Avatars in the Visual Scene. This is selected by participants who use a device of their choice (Head Mounted Display or 2D display/earpad) to experience the Audio-Visual Scene.

Each component of the Avatar-Based Videoconference Use Case is implemented as an AI Workflow (AIW) composed of AI Modules (AIMs). Each AIW includes the following elements:

1	Functions of the AIW	The functions performed by the AIW implementing the Use Case.
2	Reference Model of the AIW	The Topology of AIMS in the AIW.
3	Input and Output Data of the AIW	Input and Output Data of the AIW.
4	Functions of the AIMS	Functions performed by the AIMS.
5	Input and Output Data of the AIMS	Input and Output Data of the AIMS.
6	AIW, AIMS, and JSON Metadata	Links to summary specification on the web of the AIMS and corresponding JSON Metadata [2].

6.1.1 Videoconference Client Transmitter

6.1.1.1 Functions

The functions of the Videoconference Client Transmitter are to:

1. Receive from a Participant:
 - Input Audio from the microphone.
 - Input Visual from the camera.
 - Participant's Avatar Model.
 - Participant's Language Selector (e.g., EN-US, IT-CH).
2. Send to the Server:
 - Speech Object (for Authentication).
 - Face Object (for Authentication).
 - Input Portable Avatars containing:
 - Language Selector (at the start).
 - Avatar Model (at the start).
 - Input Speech.
 - Avatar Descriptors.

6.1.1.2 Reference Model

Figure 1 gives the Reference Model of Client Transmitter AIW. Red text refers to data sent at the start of meeting session.

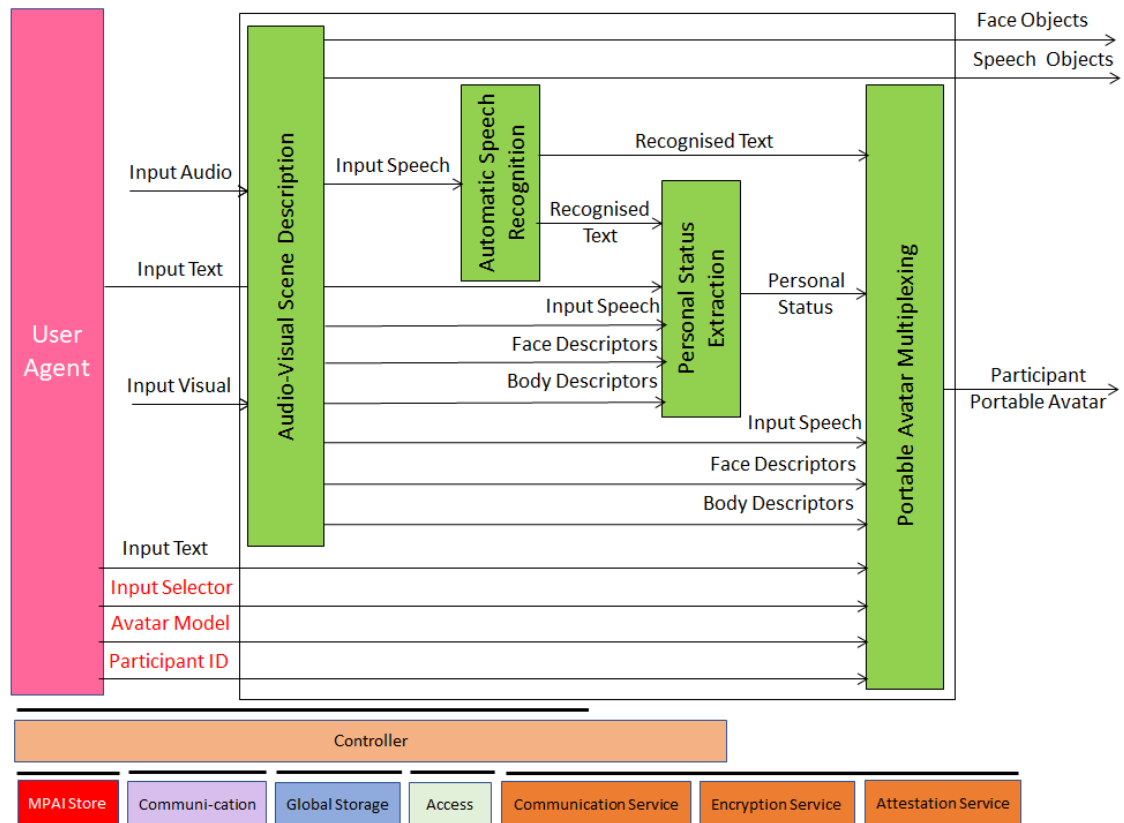


Figure 1 – Reference Model of Videoconference Client Transmitter (PAF-ABV)

At the start, each participant provides:

1. Language Selector
2. Avatar Model.
3. Speech Object (for Authentication).
4. Face Object (for Authentication).
5. Participant ID

During the videoconference:

1. *Audio-Visual Scene Description* produces Speech Objects, Face Objects, Face Descriptors, Body Descriptors and Audio-Visual Scene Geometry.
2. *Automatic Speech Recognition* produces Recognised Text.
3. *Personal Status Extraction* produces Personal Status.
4. *Portable Avatar Multiplexing* multiplexes Recognised Text, Personal Status, Input Speech, Face Descriptors, Body Descriptors, Language Selector, Avatar Model, and Participant ID.
5. *Videoconference Client Transmitter* sends Portable Avatars to Avatar Videoconference Server that the Server processes and re-distributes to Client Receivers.

6.1.1.3 Input and Output Data

Table 1 gives the input and output data of the Client Transmitter AIW:

Table 1 – Input and output data of Client Transmitter AIW

Input	Description
Input Text	Chat text used by a participant to communicate with Virtual Meeting Secretary or other participants

input Selector	The language(s) a participant wishes to speak and hear.
Input Audio	Audio of a participants' Speech in a meeting room.
Input Visual	Visual of participants in a meeting room.
Avatar Model	The avatar model selected by the participant.
Output	Description
Speech Object	A participant's utterance used by Server for authentication.
Participant Portable Avatar	Portable Avatar produced by Client Transmitter.
Face Object	Participant's face used by Server for authentication.

6.1.1.4 Functions of AI Modules

Table 2 gives the functions of AI Modules of the Client Transmitter AIW.

Table 2 – AI Modules of Client Transmitter AIW

AIM	Function
Audio-Visual Scene Description	1. Receives Input Audio and Input Visual. 2. Provides Input Speech, Speech Object, Face Descriptors, Body Descriptors, Face Object.
Automatic Speech Recognition	1. Receives Input Speech. 2. Provides Recognised Text.
Personal Status Extraction	1. Receives Recognised Text, Speech, Face Descriptors, Body Descriptors. 2. Provides the Participant's Personal Status.
Portable Avatar Multiplexing	1. Receives Language Selector, Avatar Model, Input Text, Input Speech, Recognised Text, Personal Status, Participant ID, Face Descriptors, Body Descriptors. 2. Provides Participant Portable Avatars.

6.1.1.5 I/O Data of AI Modules

Table 3 gives the AI Modules of Client Transmitter AIW.

Table 3 – AI Modules of Client Transmitter AIW

AIM	Input	Output
Audio-Visual Scene Description	Input Audio Input Visual	Input Speech Speech Object Face Object Face Descriptors Object Body Descriptors Object
Automatic Speech Recognition	Input Speech	Recognised Text
Personal Status Extraction	Recognised Text Input Speech Face Descriptors Object Body Descriptors	Personal Status

Portable Avatar Multiplexing	Recognised Text Personal Status Input Speech Face Descriptors Object Body Descriptors Object Input Text Language Selector Avatar Model Participant ID	Portable Avatar.
--	---	----------------------------------

6.1.1.6 AIW, AIM, and JSON Metadata

Table 4 - AIW, WIMs, and JSON Metadata

AIW	AIMs/1	AIMs/2	AIMs/3	Name	JSON
PAF-CTX				Videoconference Client Transmitter	X
	OSD-AVS			Audio-Visual Scene Description	X
		OSD-ASD		Audio Scene Description	X
			CAE-AAT	Audio Analysis Transform	X
			CAE-ASL	Audio Source Localisation	X
			CAE-ASE	Audio Separation and Enhancement	X
			CAE-AST	Audio Synthesis Transform	X
			CAE-AMX	Audio Descriptors Multiplexing	X
		OSD-VSD		Visual Scene Description	X
		OSD-AVA		Audio-Visual Alignment	X
	MMC-ASR			Automatic Speech Recognition	X
	MMC-PSE			Personal Status Extraction	X
		MMC-ETD		Entity Text Description	X
		MMC-ESD		Entity Speech Description	X
		PAF-EFD		Entity Face Description	X
		PAF-EBD		Entity Body Description	X
		MMC-PTI		PS-Text Interpretation	X
		MMC-PSI		PS-Speech Interpretation	X
		PAF-PFI		PS-Face Interpretation	X
		PAF-PGI		PS-Gesture Interpretation	X
		MMC-PMX		Personal Status Multiplexing	X

	PAF-PMX			Portable Avatar Multiplexing	X
--	---------	--	--	--	-------------------

6.1.1.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-CTX AIM.

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

Table 2 – Conformance Testing Method for PAF-CTX AIM

Receives	Input Text	Shall validate against Text Object Schema. Text Data shall conform with Text Qualifier.
	input Selector	Shall validate against Selector Schema.
	Input Audio	Shall validate against Audio Object Schema. Speech Data shall conform with Audio Qualifier.
	Input Visual	Shall validate against Visual Object Schema. Visual Data shall conform with Visual Qualifier.
	Avatar Model	Shall validate against 3D Model Object Schema. Speech Data shall conform with 3D Model Qualifier.
Produces	Speech Object	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Participant Portable Avatar	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.
	Face Object	Shall validate against Visual Object Schema. Face Data shall conform with Visual Qualifier.

6.1.2 Virtual Meeting Secretary

6.1.2.1 Functions

In Virtual Meeting applications, such as in the [Avatar Based Videoconference](#), i.e., a video-conference where avatars participate realistically impersonating the human participants, the Virtual Secretary:

1. Listens to the Speech of each avatar.
2. Monitors their Personal Status.
3. Drafts a Summary using the avatars' Personal Status and Text obtained from Automatic Speech Recognition or directly via Text input in the meeting's common language.
4. The Summary can be made available to participants in two different ways:
 - o Transferred to an external application so that participants can edit the Summary.
 - o Displayed to avatars where:
 - Avatars make Speech or Text comments (e.g., offline via chat).
 - The Virtual Secretary edits the Summary by understanding Text, Speech, and the avatars' Personal Statuses.

6.1.2.2 Reference Model

Figure 1 specifies the Reference Model of the Virtual Secretary AIW. It is assumed that Meaning represents both meaning of Input Text and meaning of Refined Text.

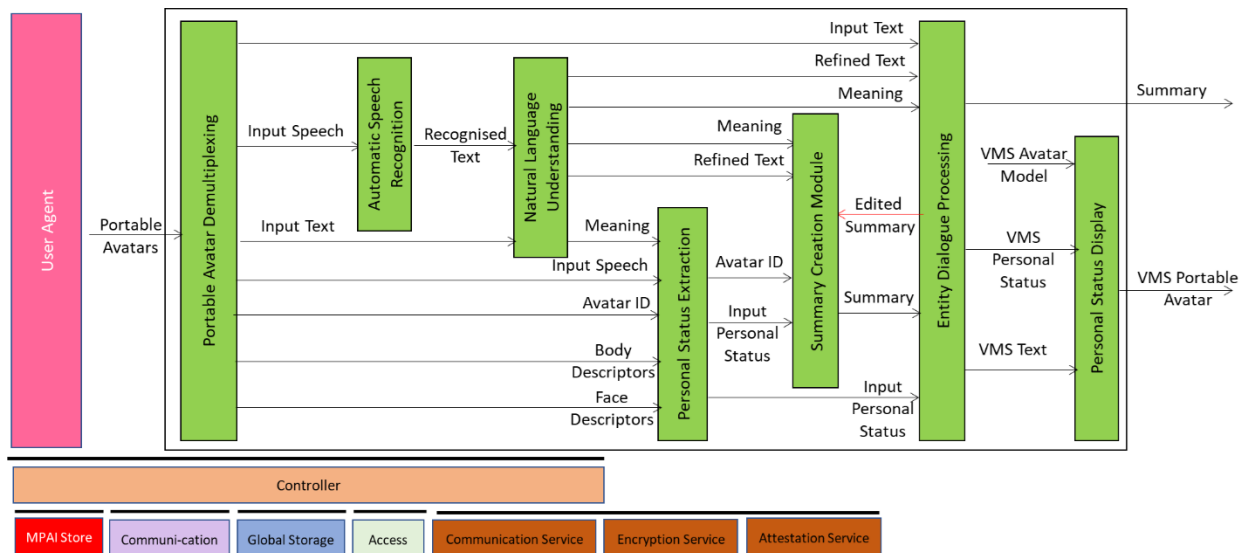


Figure 1 – Reference Model of the Virtual Meeting Secretary (MMC-VMS) AIW

The Virtual Secretary processes one avatar at a time according to the following workflow:

1. Portable Avatar De-multiplexing produces Input Text, Input Speech, Avatar ID, Body Descriptors, and Face Descriptors.
2. Automatic Speech Recognition extracts Text from Avatar Speech.
3. Natural Language Understanding:
 - Receives Recognised Text.
 - Produces Refined Text (of Recognised Text) and Meaning.
4. Personal Status Extraction:
 - Receives Meaning, Speech, and Body and Face Descriptors.
 - Produces the Personal Status of the avatar it is interacting with.
5. Summary Creation Module:
 - Receives Refined Text, Personal Status, and Meaning
 - Produces Summary using Personal Status and Text in the meeting's common language.
 - Receives Edited Summary from Entity Dialogue Processing.
6. Entity Dialogue Processing:
 - Sends Summary to external application.
 - Sends Edited Summary produced from Refined Text (from Speech), Avatar's Text (from chat), Meaning, and Summary back to Summary Creation Module.
 - Produces VMS Text and VMS Personal Status.
7. Personal Status Display produces VMS Portable Avatar containing VMS Avatar Model, VMS Text, VMS Speech, and VMS Avatar Descriptors.

6.1.2.3 I/O Data

Table 1 gives the input/output data of Virtual Meeting Secretary.

Table 1 – I/O data of Virtual Meeting Secretary

Input data	From	Description
Portable Avatar	Server	Participants' Portable Avatars as re-multiplexed by Server
Output data	To	Descriptions
VMS Portable Avatar	Server	VMS Portable Avatar to Server

Summary	Server	Summary of avatars' interventions
-------------------------	--------	-----------------------------------

6.1.2.4 Functions of AI Modules

Table 2 gives the functions of Virtual Meeting Secretary AIMS.

Table 2 – Functions of Virtual Meeting Secretary AI Modules

AIM	Functions
Portable Avatar Demultiplexing	1. Receives Portable Avatar. 2. Provides the Data required by Virtual Secretary's AIMS.
Automatic Speech Recognition	1. Receives Speech. 2. Provides Recognised Text.
Natural Language Understanding	1. Refines Recognised Text. 2. Extracts Meaning.
Personal Status Extraction	1. Receives Meaning, Input Speech, Body Descriptors, Face Descriptors. 2. Extracts Personal Status.
Summary Creation Module	1. Receives Meaning, Refined Text, Avatar ID, Input Personal Status of Avatar ID, and Edited Summary (from Entity Dialogue Processing.) 2. Produces and refines Summary using Edited Summary.
Entity Dialogue Processing	1. Receives Input Text, Refined Text, Meaning, Summary, Input Personal Status. 2. Produces Text, Virtual Secretary Personal Status, and Edited Summary.
Personal Status Display	1. Receives Virtual Secretary's Avatar Model, Personal Status, and Text. 2. Shows Virtual Secretary as Virtual Secretary Portable Avatar.

6.1.2.5 I/O Data of AI Modules

Table 3 gives the AI Modules of the Virtual Secretary depicted in Figure 4.

Table 3 – AI Modules of Virtual Secretary

AIM	Receives	Produces
Portable Avatar Demultiplexing	Portable Avatar	1. Input Text 2. Input Speech 3. Avatar ID 4. Body Descriptors Object 5. Face Descriptors Object
Automatic Speech Recognition	Speech	Recognised Text
Natural Language Understanding	Recognised Text	1. Refined Text 2. Meaning

Personal Status Extraction	1. Meaning 2. Speech 3. Face Descriptors Object 4. Body Descriptors Object	Personal Status
Summary Creation Module	1. Meaning 2. Refined Text 3. Edited Summary	Summary
Entity Dialogue Processing	1. Refined Text 2. Portable Avatar 3. Meaning 4. Summary	1. VMS Portable Avatar 2. VMS Text 3. Edited Summary
Personal Status Display	1. VMS Text 2. VMS Avatar Model 3. VMS Personal Status	VMS Portable Avatar

6.1.2.6 AIW, AIMs, and JSON Metadata

Table 4 – AIMs and JSON Metadata

Note: AIM1/s are Composite AIMs, AIM/2s are Basic AIMs.

AIW	AIMs/1	AIMs/2	Name	JSON
MMC-VMS			Virtual Meeting Secretary	X
	PAF-PDX		Portable Avatar Multiplexing	X
	MMC-ASR		Automatic Speech Recognition	X
	MMC-NLU		Natural Language Understanding	X
	MMC-PSE		Personal Status Extraction	X
		MMC-ETD	Entity Text Description	X
		MMC-ESD	Entity Speech Description	X
		PAF-EFD	Entity Face Description	X
		PAF-EBD	Entity Body Description	X
		MMC-PTI	PS-Text Interpretation	X
		MMC-PSI	PS-Speech Interpretation	X
		PAF-PFI	PS-Face Interpretation	X
		PAF-PGI	PS-Gesture Interpretation	X
		MMC-PMX	Personal Status Multiplexing	X
	MMC-SCM		Summary Creation Module	X

	MMC-EDP		Entity Dialogue Processing	X
	PAF-PSD		Personal Status Display	X
		MMC-TTS	Text-To-Speech	X
		PAF-EFD	Entity Face Description	X
		PAF-EBD	Entity Body Description	X
		PAF-PMX	Portable Avatar Multiplexing	X

6.1.2.7 Conformance Testing

Table 5 provides the Conformance Testing Method for MMC-VMS AIM.

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

Table 5 – Conformance Testing Method for MMC-VMS AIM

Receives	Portable Avatars	Shall validate against Portable Avatar Schema. Speech Data shall conform with Summary Qualifier.
Produces	VMS Portable Avatar	Shall validate against Portable Avatar Schema. Speech Data shall conform with Summary Qualifier.
	Summary	Shall validate against Summary Schema. Speech Data shall conform with Summary Qualifier.

6.1.2.8 Performance Assessment

6.1.3 Avatar Videoconference Server

6.1.3.1 Functions

The Server:

1. *At the start:*
 - Receives Speech Object and Speech Objects of each participant.
 - Authenticates participant's Avatar.
 - Receives Portable Avatars each containing Language Preference and Avatar Model.
 - Selects an Audio-Visual Scene.
 - Selects the Spatial Attitudes of the Avatar Models in the Audio-Visual Scene.
 - Selects the common meeting Language.
 - Distributes all Portable Avatars each containing: Audio-Visual Scene, Language Preference, Avatar Model, and Spatial Attitude.
2. *During the videoconference:*
 - Receives participant Avatars' and Virtual Meeting Secretary's Avatar Descriptors.
 - Translates participants' Speech Objects according to their Language Preferences.
 - Sends Portable Avatars containing Avatar ID, Text, Speech translated to the common meeting Language, Face Descriptors and Gesture Descriptors to Virtual Meeting Secretary.
 - Receives Virtual Meeting Secretary's Portable Avatar containing Avatar ID, Text, Speech in the common meeting Language, Face Descriptors and Gesture Descriptors.

- Translates Virtual Meeting Secretary's Speech according to each participant's Language Preferences.
- Sends participant Avatars and Virtual Meeting Secretary's Portable Avatars containing Avatar ID, Text, Translated Speech, Face Descriptors and Gesture Descriptors to Client Receivers.

6.1.3.2 Reference Model

Figure 1 gives the Reference Model of Avatar Videoconference Server AIW. Red text refers to Data sent at meeting start.

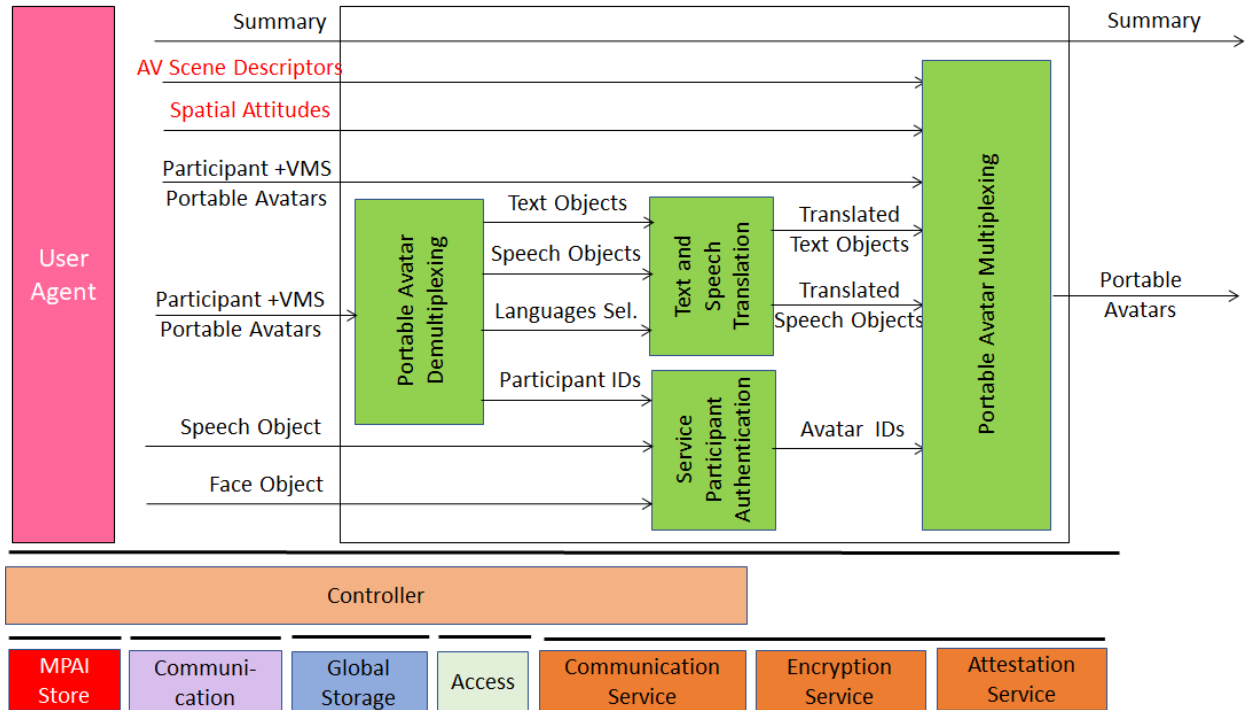


Figure 1 – Reference Model of Avatar Videoconference Server

6.1.3.3 I/O Data

Table 1 gives the input and output data of Avatar Videoconference Server AIW.

Table 1 – Input and output data of Avatar Videoconference Server AIW

Input	Description
Summary	From Virtual Meeting Secretary
AV Scene Descriptors	Initial AV Scene Descriptors set by Server
Spatial Attitudes	Initial Spatial Attitudes set by Server
Input and VMS Portable Avatars	From Transmitting Clients and Virtual Meeting Secretary
Speech Objects	Participants' Speech Object for Authentication
Face Objects	Participants' Face Object for Authentication
Outputs	Description

Summary	As above
Portable Avatars	As re-multiplexed by Server

6.1.3.4 Functions of AI Modules

Table 2 gives the functions of the AI Modules of the Avatar Videoconference Server AIW.

Table 2 - AI Modules of Avatar Videoconference Server AIW

AIM	Functions
Portable Avatar Demultiplexing	Makes available Input Text, Input Speech, Language Preferences and participant ID from Input and VS Portable Avatars.
Text and Speech Translation	Translates active Speech and Text of all Participants to the Selected Languages.
Service Participant Authentication	- Authenticates participants using Speech and Face Objects. - Connects participant ID to Avatar ID.
Portable Avatar Multiplexing	Multiplexes components of Portable Avatars for transmission to Receiving Clients.

6.1.3.5 I/O Data of AI Modules

Table 3 gives the Input/Output Data of the AI Modules of the Avatar Videoconference Server AIW.

Table 3 - AI Modules of Avatar Videoconference Server AIW

AIM	Input	Output
Portable Avatar Demultiplexing	Input and VMS Portable Avatars	1. Text Object 2. Speech Object 3. Language Selector 4. Participant ID
Text and Speech Translation	1. Language Selector 2. Text Object 3. Speech Object	1. Translated Text Object 2. Translated Speech
Service Participant Authentication	1. Participant ID 2. Speech Object 3. Face Object	Avatar Identifier
Portable Avatar Multiplexing	1. Audio-Visual Scene Descriptors 2. Spatial Attitudes 3. Portable Avatars 4. Translated Text Object 5. Translated Speech 6. Avatar Identifier	Portable Avatars

6.1.3.6 AIW, AIM, and JSON Metadata

Table 4 – AIW, AIMs and JSON Metadata

AIW	AIMs	Name	JSON
PAF-AVS		Avatar Videoconference Server	X
	PAF-PDX	Portable Avatar Demultiplexing	X
	MMC-TST	Text and Speech Translation	X
	PAF-SPA	Service Participant Authentication	X
	PAF-PMX	Portable Avatar Multiplexing	X

6.1.3.7 Conformance Testing

Table 2 provides the Conformance Testing Method for the PAF-AVS AIW.

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

Table 2 – Conformance Testing Method for PAF-AVS AIM

Receives	Summary	Shall validate against Summary schema.
	AV Scene Descriptors	Shall validate against AV Scene Descriptors Schema.
	Spatial Attitudes	Shall validate against Spatial Attitude Schema.
	Portable Avatars	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.
	Speech Objects	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Face Objects	Shall validate against Visual Schema. Speech Data shall conform with Visual Qualifier.
Produces	Summary	Shall validate against Summary Schema.
	Portable Avatars	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.

6.1.4 Videoconference Client Receiver

6.1.4.1 Functions

The Function of the Videoconference Client Receiver is to:

- Create Local Visual Scene by placing and animating the Avatar Models with their Spatial Attitudes.
- Local Audio-Visual Scene by adding Speech to Avatars' Mouths.
- Render Audio-Visual Scene as seen from the Participant-selected Point of View.

6.1.4.2 Reference Model

Figure 1 depicts the Reference Model of the Videoconference Client Receiver. Red Text for Data received at the start. This is the operation:

1. *At the start*
 - Receives Portable Avatars containing:
 - Audio-Visual Scene Descriptors
 - Avatar Models
 - Spatial Attitudes

- Creates the initial Audio-Visual Scene.
- 3. *During the Videoconference:*
 - Receives the Avatar Models containing:
 - Speech
 - Body Descriptors
 - Face Descriptors
 - Creates the running Audio-Visual Scene using each Avatar's:
 - Body and Face Descriptors.
 - Speech Objects.
- 4. Renders the Audio-Visual Scene based on the selected Point of View.

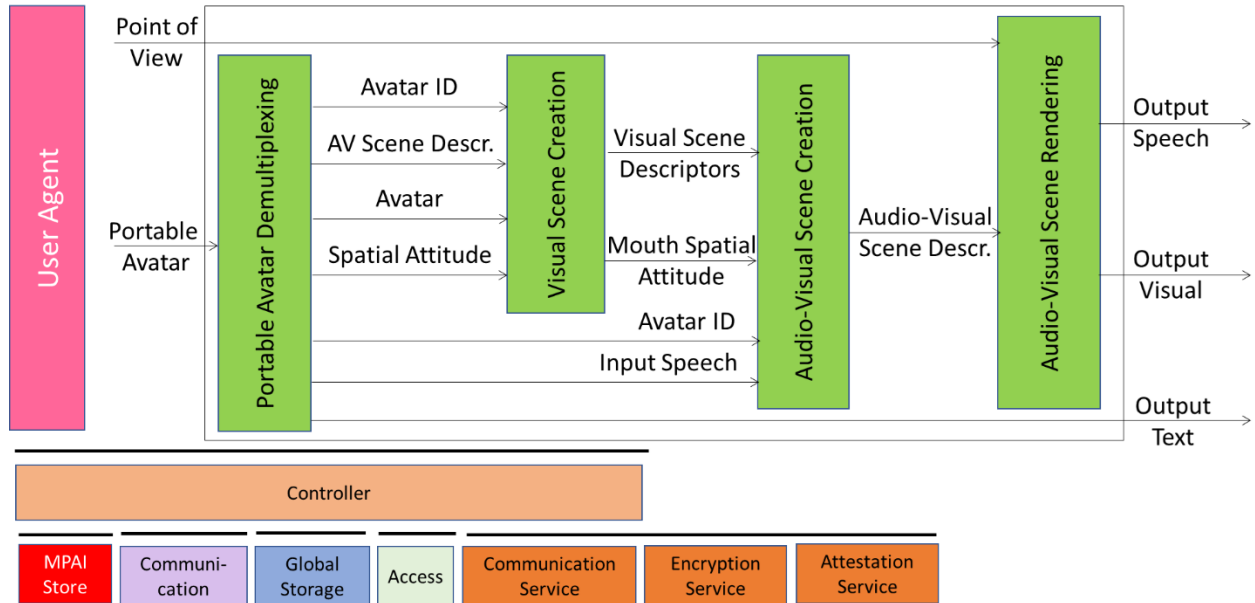


Figure 1 – Reference Model of Videoconference Client Receiver

6.1.4.3 I/O Data

Table 1 gives the input and output data of Videoconference Client Receiver.

Table 1 – Input and Output Data of Videoconference Client Receiver AIW

Input	Description
Point of View	Avatar-selected Position to see the Audio-Visual Scene.
Portable Avatars	Portable Avatars from Videoconference Avatar Server.
Output	Description
Output Speech	Rendered by Audio-Visual Scene Rendering.
Output Visual	Rendered by Audio-Visual Scene Rendering.

6.1.4.4 Functions of AI Modules

Table 2 gives the AI Modules of Videoconference Client Receiver AIW.

Table 2 – Functions of Videoconference Client Receivers' AI Modules

AIM	Input
-----	-------

Portable Avatar Demultiplexing	Extracts Avatar ID, Audio-Visual Scene Descriptors, Avatar Model, Spatial Attitude, Body Descriptors, Face Descriptors, and Input Speech from Portable Avatars.
Visual Scene Creation	Creates the Visual Scene and provides the Spatial Attitudes of the Mouths of all Avatars.
Audio-Visual Scene Creation	Creates the Audio Scene.
Audio-Visual Scene Rendering	Provides a ready-to-rendered Audio-Visual Scene.

6.1.4.5 I/O Data of AI Modules

Table 2 gives the AI Modules of Videoconference Receiving Client AIW.

Table 3 – I/O Data of Videoconference Client Receivers' AI Modules

AIM	Input	Output
Portable Avatar Demultiplexing	Portable Avatars	1. Avatar IDs 2. Audio-Visual Scene Descriptors 3. Avatars 4. Spatial Attitudes 5. Input Speech Objects 6. Output Text Objects
Visual Scene Creation	1. Avatar IDs 2. Audio-Visual Scene Descriptors 3. Avatars 4. Spatial Attitude 5. Input Speech Objects	1. Visual Scene Descriptors 2. Mouth Spatial Attitudes
Audio-Visual Scene Creation	1. Avatar ID 2. Input Speech Objects 3. Mouth Spatial Attitudes	1. Audio Scene Descriptors
Audio-Visual Scene Rendering	1. Audio Scene Descriptors 2. Visual Scene Descriptors 3. Point of View	1. Output Speech 2. Output Visual

6.1.4.6 AIW, AIM, and JSON Metadata of Videoconference Client Receiver

Table 4 – AIMs and JSON Metadata

AIW	AIMs	Name	JSON
PAF-CRX		Videoconference Client Receiver	X
-	PAF-PDX	Portable Avatar Demultiplexing	X
-	PAF-VSC	Visual Scene Creation	X
-	PAF-AVC	Audio-Visual Scene Creation	X
-	PAF-AVR	Audio-Visual Scene Rendering	X

6.1.4.7 Conformance Testing

Table 2 provides the Conformance Testing Method for the PAF-CRX AIM as a Basic AIM.

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

Table 2 – Conformance Testing Method for CAE-CRX AIW

Receives	Point of View	Shall validate against Point of View Schema.
	Portable Avatars	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.
Produces	Output Speech	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Output Visual	Shall validate against Visual Object Schema. Visual Data shall conform with Visual Qualifier.

6.2 Reference Software

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

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

Note that at this stage only part of the AIMs required to operate the MPAI-PAF AIWs have a Reference Software Implementation.

6.3 Conformance Testing

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

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

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

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

6.4 Performance Assessment

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

1. Quality: the Performance of a [Videoconference Client Transmitter](#) AIW can measure how well the AIW represents the human Participant.
2. Bias: Performance of a [Videoconference Client Receiver](#) AIW can reproduce the avatar videoconference.

3. Legal compliance: the Performance of an AIW can measure the compliance of the AIW to a regulation, e.g., the European AI Act.
4. Ethical compliance: the Performance Assessment of an AIW can measure the compliance of an AIW to a target ethical standard.

Note that at this stage MPAI-PAF AIWs do not have a Performance Assessment Specification.

7 AI Modules

7.1 Technical Specifications

Table 1 provides the links to the specifications and the JSON syntax of all AIMs specified by **Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5**. MPAI-PAF V1.5 AI-Modules supersede those previously specified which may still be used if their version is by explicitly signaled. AIMs in bold are Composite.

Table 1 – Specifications and JSON syntax of AIMs used by MPAA-PAF V1.5

Acronym	AIM Name	JSON	Acronym	AIM Name	JSON
PAF-AVC	Audio-Visual Scene Creation	X	PAF-PMX	Portable Avatar Multiplexing	X
PAF-AVR	Audio-Visual Scene Rendering	X	PAF-PFI	PS-Face Interpretation	X
PAF-ASR	Avatar and Speech Rendering	X	PAF-PGI	PS-Gesture Interpretation	X
PAF-EBD	Entity Body Description	X	PAF-PSD	Personal Status Display	X
PAF-EFD	Entity Face Description	X	PAF-RSR	Response and Scene Rendering	X
PAF-FIR	Face Identity Recognition	X	PAF-SAR	Scene and Avatar Rendering	X
PAF-FPS	Face Personal Status Extraction	X	PAF-SPA	Service Participant Authentication	X
PAF-GPS	Gesture Personal Status Extraction	X	PAF-VSC	Visual Scene Creation	X
PAF-PDX	Portable Avatar Demultiplexing	X			

7.1.1 Audio-Visual Scene Creation

7.1.1.1 Functions

The Audio-Visual Scene Creation (PAF-AVC) AIM receives the Descriptors of a Visual Scene that includes Avatars, the Speech Objects uttered by the Avatars and the Spatial Attitudes of the Avatars, and provides the Descriptors of an Audio-Visual Scene where the Speech Objects associated with each Avatars are attached to the Mouth of each Avatar using the Mouths' Spatial Attitudes:

Receives	<i>Visual Scene Descriptors</i>	Descriptors of a Visual Scene.
----------	---------------------------------	--------------------------------

	<i>Avatars</i>	Avatars in the Visual Scene.
	<i>Mouth Spatial Attitude</i>	Spatial Attitudes of the Avatars in the Scene.
	<i>Speech Objects</i>	Speech Objects uttered by the Avatars
Produces	<i>Audio-Visual Scene Descriptors</i>	New AV Scene Descriptors where Speech Objects are located at Avatars' mouths.

7.1.1.2 Reference Model

Figure 1 depicts the Reference Model of the Audio-Visual Scene Creation (PAF-AVC) AIM.

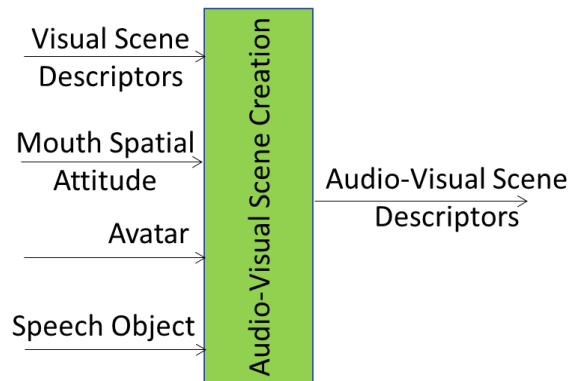


Figure 1 Audio-Visual Scene Creation (PAF-AVC) AIM

7.1.1.3 Input/Output Data

Table 1 specifies the Input and Output Data of Audio-Visual Scene Creation (PAF-AVC) AIM.

Table 1 – I/O Data of the Audio-Visual Scene Creation (PAF-AVC) AIM

Input	Description
Visual Scene Descriptors	Input Visual Scene Descriptors
Mouth Spatial Attitudes	Spatial Attitude of Avatars' Mouths
Avatars	IDs of Avatars
Speech Objects	Speech Objects associated to Avatars
Output	Description
Audio-Visual Scene Descriptors	Output AV Scene Descriptors

7.1.1.4 SubAIMs

No SubAIMs.

7.1.1.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.4/AIMs/AudioVisualSceneCreation.json>

7.1.1.6 Profiles

No profiles.

7.1.1.7 Conformance Testing

Table 2 provides the Conformance Testing Method for the PAF-AVC AIM as a Basic AIM.

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

Table 2 – Conformance Testing Method for PAF-AVC AIM

Receives	Visual Scene Descriptors	Shall validate against Visual Scene Descriptors Schema.
	Mouth Spatial Attitudes	Shall validate against Spatial Attitude Schema.
	Avatars	Shall validate against Avatar Schema. Avatar Model Data shall conform with 3D Model Qualifier.
	Speech Objects	Shall validate against Avatar Schema. Speech Data shall conform with Speech Qualifier.
Produces	Audio-Visual Scene Descriptors	Shall validate against Visual Scene Descriptors Schema.

7.1.2 Portable Avatar Multiplexing

7.1.2.1 Functions

The Portable Avatar Multiplexing (PSD-PMX) AIM Multiplexes the input component elements of a Portable Avatar - Portable Avatar ID, Avatar Space-Time, Avatar, Language Selector, Speech Object, Text Object, Speech Model, Personal Status, Audio-Visual Scene Descriptors, and Audio-Visual Scene Space Time into a Portable Avatar:

Receives	An arbitrary number of elements in Portable Avatar out of:
	- Portable Avatar ID
	- Avatar Space-Time
	- Avatar
	- Language Selector
	- Text Object
	- Speech Model
	- Speech Object
	- Personal Status
	- Audio- Visual Scene Space-Time
	- Audio-Visual Scene Descriptors

	- An existing Portable Avatar
Changes	Existing with Input Data
Adds	Input Data that is not in the Input Portable Avatar
Produces	Portable Avatar

7.1.2.2 Reference Model

Figure 1 specifies the Reference Model of the Portable Avatar Multiplexing (PSD-PMX) AIM.

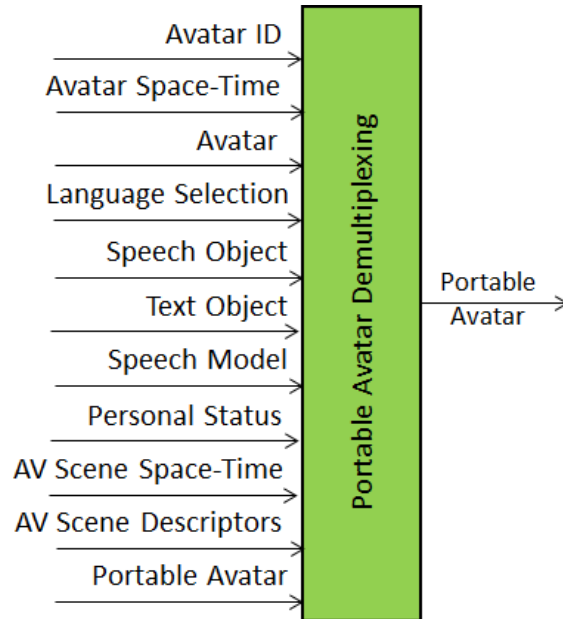


Figure 1 – The Portable Avatar Multiplexing (PSD-PMX) AIM

7.1.2.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Portable Avatar Multiplexing (PSD-PMX) AIM.

Table 1 – I/O Data of the Portable Avatar Multiplexing (PSD-PMX) AIM

Input	Description
AvatarID	Avatar ID.
Avatar Space-Time	Portable Avatar Time.
Avatar	Avatar in Portable Avatar.
Language Selector	Language of Avatar.
Speech Object	The Speech in the time when the PA is valid.
Text Object	The Time in the time when the PA is valid.
Speech Model	The NN Model used to synthesise text.
Avatar Personal Status	The Avatar's Personal Status.

AV Scene Descriptors	Descriptors of AV Scene.
AV Scene Space-Time	Space-Time info of AV Scene.
Portable Avatar	The input Portable Item.
Output	Description
Portable Avatar	The output Portable Item.

7.1.2.4 SubAIMs

No SubAIMs.

7.1.2.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/PortableAvatarMultiplexing.json>

7.1.2.6 Profiles

No Profiles.

7.1.2.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PMX AIM.

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

Table 2 – Conformance Testing Method for PAF-PMX AIM

Receives	AvatarID	Shall be string or validate against Instance ID Schema
	Avatar Space-Time	Shall validate against Space-Time Schema
	Avatar	Shall validate against Avatar Schema. Avatar Model Data shall conform with 3D Model Qualifier.
	Language Selector	Shall validate against Selector Schema
	Speech Object	Shall validate against Speech Object Schema. Text Data shall conform with SpeechQualifier.
	Text Object	Shall validate against Text Object Schema. Speech Data shall conform with Text Qualifier.
	Speech Model	Shall validate against Machine Learning Model Schema. Speech Model Data shall conform with Machine Learning Model Qualifier.
	Avatar Personal Status	Shall validate against Personal Status Schema.
	AV Scene Descriptors	Shall validate against AV Scene Descriptors Schema.

	AV Scene Space-Time	Shall validate against Space-Time Schema.
Produces	Portable Avatar	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.

7.1.3 Audio-Visual Scene Rendering

7.1.3.1 Functions

The Audio-Visual Scene Rendering (PAF-AVR) AIM

1. Receives an input Point of View and all or some of the following input data: an input Portable Avatar, input Audio-Visual Scene Descriptors, and an input Spatial Attitude.
2. Produces the Audio, Speech, and Visual components resulting from the rendering from the input Point of View of one of:
 1. The input Audio-Visual Scene Descriptors - if no input Portable Avatar is present.
 2. A speaking avatar constructed according to the data of the input Portable Avatar embedded in the input Potable Avatar's Audio-Visual Scene Descriptors with the input Spatial Attitude - if the input Audio-Visual Scene Descriptors are not present.
 3. The input Audio-Visual Scene Descriptors that include an avatar constructed according to the data of the input Portable Avatar and embedded in the input Audio-Visual Scene Descriptors with the input Spatial Attitude - if both input Portable Avatar and input Audio-Visual Scene Descriptors are present.

Receives	Portable Avatar	Jointly with or alternatively with AV Scene Descriptors.
	Audio-Visual Scene Descriptors	Alternative to or superseding that of the Portable Avatar.
	Spatial Attitude	Spatial Attitude of the Avatar in the Audio-Visual Scene.
	Point of View	To be used in rendering the scene and its objects.
Transforms	Portable Avatar	Into generic Audio-Visual Scene Descriptors if input Portable Avatar is present.
Produces	Portable Avatar's Output Speech	Always integrated in the Audio-Visual Scene. Output Speech results from the rendering of Audio Scene Descriptors from human-selected Point of View.
	Output Audio	Resulting from the rendering of Audio Scene Descriptors from human-selected Point of View.
	Output Visual	Resulting from the rendering of Audio Scene Descriptors from human-selected Point of View. View Selector tells the OSD-AVR AIM where the visual components of the Portable Avatar should also be integrated.

7.1.3.2 Reference Model

Figure 1 specifies the Reference Model of the Audio-Visual Scene Rendering (PAF-AVR) AIM.

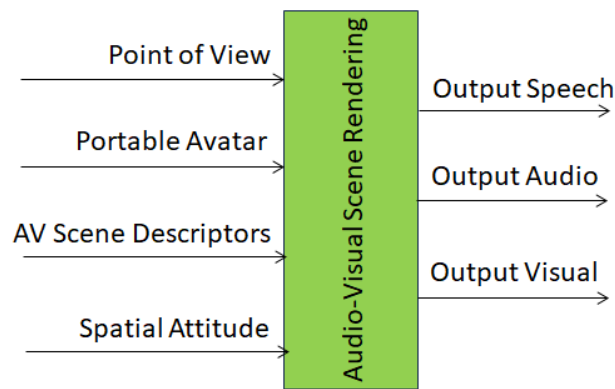


Figure 1 – The Audio-Visual Scene Rendering (PAF-AVR) AIM

7.1.3.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio-Visual Scene Rendering (PAF-AVR) AIM.

Table 1 – I/O Data of the Audio-Visual Scene Rendering (PAF-AVR) AIM

Input	Description
Portable Avatar	Data produced, e.g., by Personal Status Display.
AV Scene Descriptors	Audio-Visual Scene Descriptors.
Point of View	Point from where an Entity perceives the Audio-Visual Scene
Spatial Attitude	of the Avatar in the Audio-Visual Scene.
Output	Description
Output Speech Object	The Speech components of the Audio-Visual Scene.
Output Audio Object	The Audio components of the Audio-Visual Scene.
Output Visual Object	The Visual components of the Audio-Visual Scene.

7.1.3.4 SubAIMs

No SubAIMs.

7.1.3.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/AudioVisualSceneRendering.json>

7.1.3.6 Profiles

The Profiles of Audio-Visual Scene Rendering are [specified](#).

7.1.3.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-AVR AIM.

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

Table 2 – Conformance Testing Method for PAF-AVR AIM

Receives	Portable Avatar	Shall validate against Point of View Schema.
	AV Scene Descriptors	Shall validate against AV Scene Descriptors Schema.
	Point of View	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.
	Spatial Attitude	Shall validate against Spatial Attitude Schema.
Produces	Output Speech Object	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Output Audio Object	Shall validate against Audio Object Schema. Audio Data shall conform with Audio Qualifier.
	Output Visual Object	Shall validate against Visual Object or 3D Model Schema. Visual Data shall conform with Visual Object.

7.1.4 PS-Face Interpretation

7.1.4.1 Functions

The PS-Face Interpretation (PAF-PFI) AIM receives input Faces Descriptors and produces the Face Personal Status from the input Face Descriptors:

Receives	<i>Face Descriptors Object</i>	from Face Description or as input to PS-Face Interpretation
Produces	<i>Face Personal Status</i>	the Personal Status of the Face Modality

7.1.4.2 Reference Model

Figure 1 specifies the Reference Architecture of the PS-Face Interpretation (PAF-PFI) AIM.

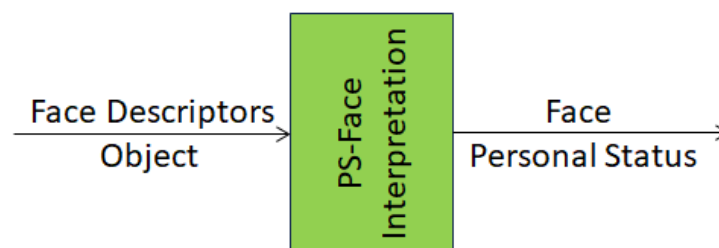


Figure 1– The PS-Face Interpretation (PAF-PFI) AIM Reference Model

7.1.4.3 Input/Output Data

Table 1 specifies the Input and Output Data of the PS-Face Interpretation (PAF-PFI) AIM.

Table 1– I/O Data of the PS-Face Interpretation (PAF-PFI) AIM

Input	Description
Face Descriptors Object	Descriptors of Face
Output	Description
Face Personal Status	Personal Status of Face

7.1.4.4 SubAIMs

No SubAIMs.

7.1.4.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/PSFaceInterpretation.json>

7.1.4.6 Profiles

No Profiles.

7.1.4.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PFI AIM.

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

Table 2 – Conformance Testing Method for PAF-PFI AIM

Receives	Face Descriptors Object	Shall validate against Face Descriptors Object Schema
Produces	Face Personal Status	Shall validate against Face Personal Status Schema

7.1.5 Avatar and Speech Rendering

7.1.5.1 Functions

The Avatar and Speech Rendering (PAF-ASR) AIM produces the Visual Object corresponding to an avatar synthesised from an Avatar Module animated by Face and Body Descriptors and the Speech Object from input Face Descriptors, input Body Descriptors, input Avatar Model, and input Speech (which is rendered unchanged as Output Speech).

Receives	Input Speech	Speech uttered by the avatar.
	Face Descriptors	Avatar Face Descriptors.
	Body Descriptors	Avatar Body Descriptors.
	Avatar Model	When a specific Avatar Model is externally provided.
Produces	Output Speech	The input Speech Object .
	Output Visual	The rendering of the animated avatar.

7.1.5.2 Reference Model

Figure 1 depicts the Reference Model of the Avatar and Speech Rendering (PAF-ASR) AIM.

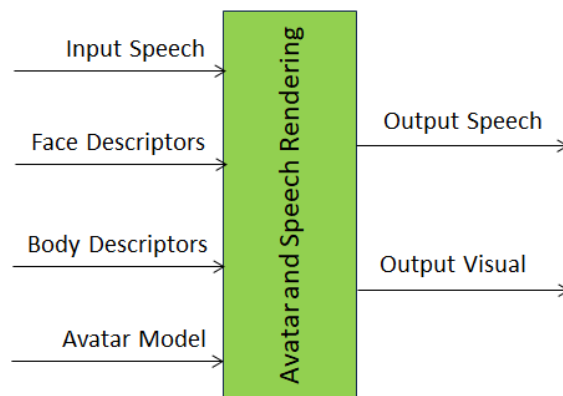


Figure 1– The Avatar and Speech Rendering (PAF-ASR) AIM

7.1.5.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Avatar and Speech Rendering (PAF-ASR) AIM.

Table 1– I/O Data of the Avatar and Speech Rendering (PAF-ASR) AIM

Input	Description
Input Speech	Speech uttered by the avatar.
Face Descriptors Object	Avatar Face Descriptors.
Body Descriptors Object	Avatar Body Descriptors.
Avatar Model	To be animated.
Output	Description
Output Speech	Speech from input.
Output Visual	Rendering of animated avatar.

7.1.5.4 SubAIMs

No SubAIMs.

7.1.5.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/AvatarAndSpeechRendering.json>

7.1.5.6 Profiles

No Profiles

7.1.5.7 Conformance Testing

Table 2 provides the Conformance Testing Method for the Avatar and Speech Rendering (PAF-ASR) AIM.

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

Table 2 – Conformance Testing Method for Avatar and Speech Rendering (PAF-ASR) AIM

Input	Description
Input Speech	Shall validate against Speech Object Schema.
Face Descriptors Object	Shall validate against Face Descriptors Schema.
Body Descriptors Object	Shall validate against Body Descriptors Schema.
Avatar Model	Shall validate against 3D Model Object Schema.
Output	Description
Output Speech	Shall validate against Speech Object Schema.
Output Visual	Shall validate against Visual Object Schema.

7.1.6 PS-Gesture Interpretation

7.1.6.1 Functions

The PS-Gesture Interpretation (PAF-PGI) AIM receives input Gesture Descriptors and produces the Gesture Personal Status from the input Gesture Descriptors:

Receives *Gesture Descriptors Object* from Gesture Description or as input to PS-Gesture Interpretation
Produces *Gesture Personal Status* the Personal Status of the Gesture Modality

7.1.6.2 Reference Model

Figure 1 specifies the Reference Architecture of the PS-Gesture Interpretation (PAF-PGI) AIM.

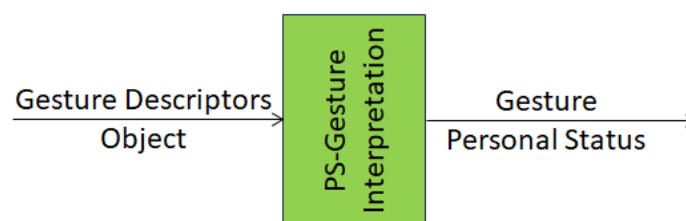


Figure 1– The PS-Gesture Interpretation (PAF-PGI) AIM Reference Model

7.1.6.3 Input/Output Data

Table 1 specifies the Input and Output Data of the PS-Gesture Interpretation (PAF-PGI) AIM.

Table 1– I/O Data of the PS-Gesture Interpretation (PAF-PGI) AIM

Input	Description
Gesture Descriptors Object	Descriptors of Gesture
Output	Description
Gesture Personal Status	Personal Status of Gesture

7.1.6.4 SubAIMs

No SubAIMs.

7.1.6.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/PSGestureInterpretation.json>

7.1.6.6 Profiles

No Profiles.

7.1.6.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PGI AIM.

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

Table 2 – Conformance Testing Method for PAF-PGI AIM

Receives	Gesture Descriptors Object	Shall validate against Gesture Descriptors Schema
Produces	Gesture Personal Status	Shall validate against Gesture Personal Status Schema

7.1.7 Entity Body Description

7.1.7.1 Functions

The Entity Body Description (PAF-EBD) AIM receives an input Visual Object representing a Body and produces the Descriptors of the input Visual Object:

Receives	<i>Body Visual Object</i>	Body of Entity or from upstream AIM.
Produces	<i>Body Descriptors Object</i>	Descriptors of Body Visual Object

7.1.7.2 Reference Model

Figure 1 specifies the Reference Model of the Entity Body Description (PAF-EBD) AIM.

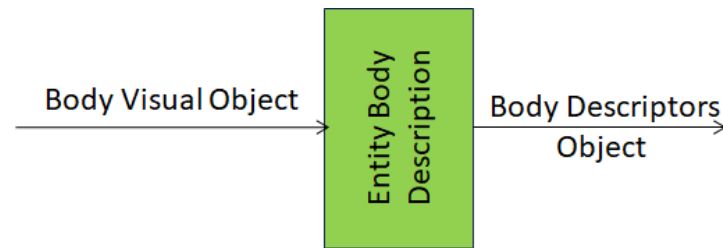


Figure 1 Entity Body Description (PAF-EBD) AIM

7.1.7.3 Input/Output Data

Table 1 specifies the Input and Output Data of Entity Body Description (PAF-EBD) AIM.

Table 1 – I/O Data of the Entity Body Description (PAF-EBD) AIM

Input	Description
Body Visual Object	Visual Object representing the body of an Entity.
Output	Description
Body Descriptors Object	Body Descriptors of Visual Object.

7.1.7.4 SubAIMs

No SubAIMs.

7.1.7.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/EntityBodyDescription.json>

7.1.7.6 Profiles

No profiles.

7.1.7.7 Conformance Testing

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

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

Table 2 – Conformance Testing Method for MMC-EBD AIM

Receives	Body Visual Object	Shall validate against Visual Object Schema. Body Data shall conform with Visual Object.
Produces	Body Descriptors Object	Shall validate against Body Descriptors XML Schema.

7.1.8 Personal Status Display

7.1.8.1 Functions

The Personal Status Display (PAF-PSD) AIM receives an input Identifier of a conversational machine, an input Avatar Model, and an input Text Object and produces a Portable Avatar that includes the input Identifier, an Avatar uttering a Speech Object synthesised from the input Text Object and based on the input Avatar Model. The PAF-PSD AIM may also receive an input Personal Status and Speech Model that can be used to synthesise the Speech Object from the input Text Object and produce a speaking Avatar that displays the input Personal Status:

Receives	Machine ID	ID to be used to identify the Avatar in Portable Avatar.
	Text Object	Text associated to Avatar in Portable Avatar.
	Personal Status	Personal Status associated to Avatar in Portable Avatar.
	Avatar Model	3D Model associated to Avatar in Portable Avatar.
	Speech Model	Speech Model Associated to Avatar in Portable Avatar.
Produces	Portable Avatar	Output Portable Avatar.
Enables	PAF-AVR	To render the Portable Avatar produced by PAF-PSD.

7.1.8.2 Reference Model

Figure 1 depicts the AIMs implementing the Personal Status Display (PAF-PSD) AIM.

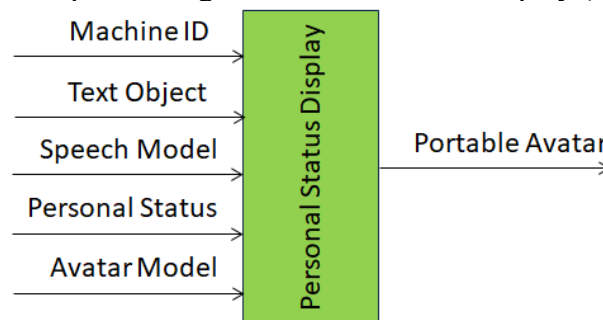


Figure 1 – Reference Model of Personal Status Display (PAF-PSD) AIM

7.1.8.3 Input/Output Data

Table 1 gives the Input/Output Data of Personal Status Display (PAF-PSD).

Table 1 – I/O Data of Personal Status Display

Input data	Description
Avatar ID	Portable Avatar's ID from Upstream AIM
Avatar Model	Part of Portable Avatar from Upstream AIM or embedded in PSD
Text Object	Texts of Portable Avatar from Keyboard or upstream AIM

Personal Status	To add PS to Speech, Face, and Gesture from Personal Status Extraction or Machine
Speech Model	Neural Network from Upstream AIM or embedded in PSD
Output data	Description
Portable Avatar	Output Portable Avatar to downstream AIM or renderer.

7.1.8.4 SubAIMs

Figure 2 gives the Reference Model of the the Personal Status Display Composite AIM.

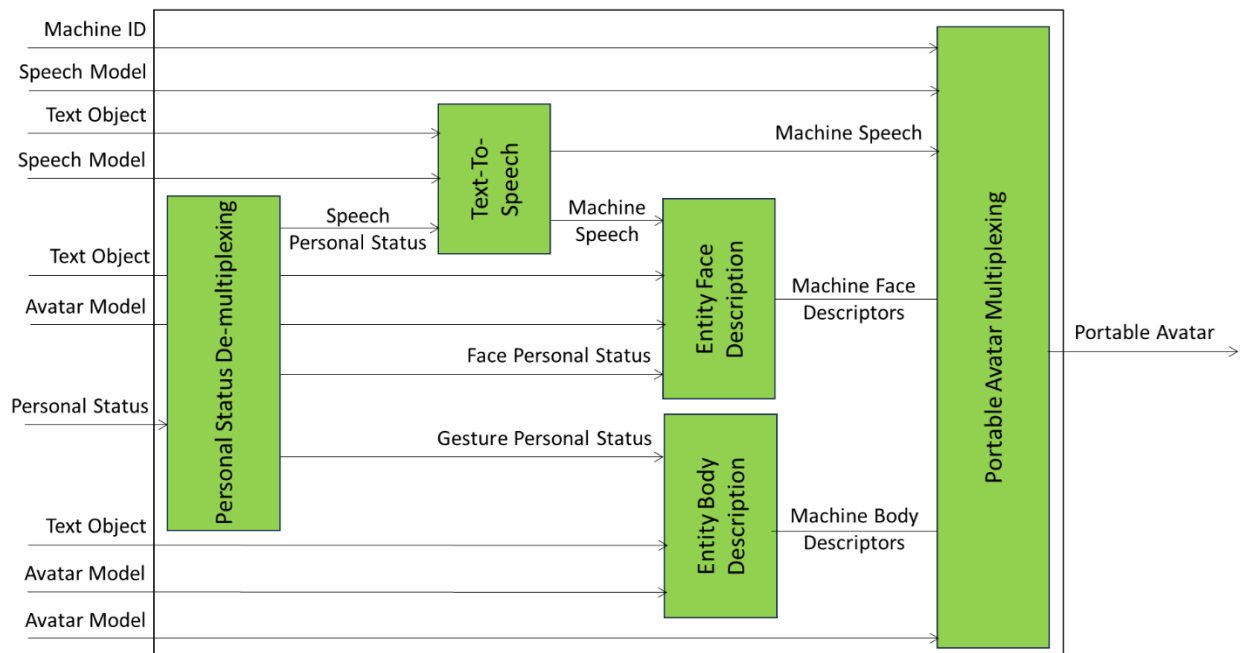


Figure 2 – Reference Model of Personal Status Display Composite AIM

The Personal Status Display Composite AIM operates as follows:

1. Avatar ID is the ID of the Portable Avatar.
2. Personal Status Demultiplexing makes available the component PS-Speech, PS-Face, and PS-Gesture Modalities.
3. Machine Text is synthesised as Speech using a Speech Model in a format specified by NN Format and the Personal Status provided by PS-Speech.
4. Machine Speech and PS-Face are used to produce the Machine Face Descriptors.
5. PS-Gesture and Text are used for Machine Body Descriptors using the Avatar Model.
6. Portable Avatar Multiplexing produces the Portable Avatar.

Table 2 gives the list of PSD AIMs with their input and output Data.

Table 2 –AIMs of Personal Status Display Composite AIM and JSON Metadata

AIW	AIMs	Name and Specification	JSON
PAF-PSD		Personal Status Display	X
	MMC-PDX	Personal Status Demultiplexing	X
	MMC-TTS	Text-to-Speech	X
	PAF-EFD	Entity Face Description	X

	PAF-EBD	Entity Body Description	X
	PAF-PMX	Portable Avatar Multiplexing	X

7.1.8.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/PersonalStatusDisplay.json>

7.1.8.6 Profiles

The Profiles of Personal Status Display are [specified](#).

7.1.8.7 Conformance Testing

The Conformance Testing Method for the PAF-PSD Basic AIM is provided here. The Conformance Testing Method for the individual Basic AIMs of the PAF-PSD Composite AIM is provided by the individual Basic AIMs.

Table 2 provides the Conformance Testing Method for PAF-PSD AIM.

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

Table 2 – Conformance Testing Method for PAF-PSD AIM

Receives	Machine ID	Shall be string or validate against Instance ID Schema
	Text Object	Shall validate against Text Object Schema. Text Data shall conform with Speech Qualifier.
	Personal Status	Shall validate against Personal Status Schema.
	Avatar Model	Shall validate against 3D Model Schema. Avatar Model Data shall conform with 3D Model Qualifier.
	Speech Model	Shall validate against Machine Learning Model Schema. Speech Model Data shall conform with Machine Learning Model Qualifier.
Produces	Portable Avatar	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.

7.1.9 Entity Face Description

7.1.9.1 Functions

The Entity Face Description (PAF-EFD) AIM receives an input Visual Object representing a Face and produces the Descriptors of the input Visual Object:

Receives	<i>Face Visual Object</i>	Face of Entity or from <i>upstream AIM</i> .
Produces	<i>Face Descriptors Object</i>	<i>Descriptors of Entity Face.</i>

7.1.9.2 Reference Model

Figure 1 specifies the Reference Model of the Entity Face Description (PAF-EFD) AIM.

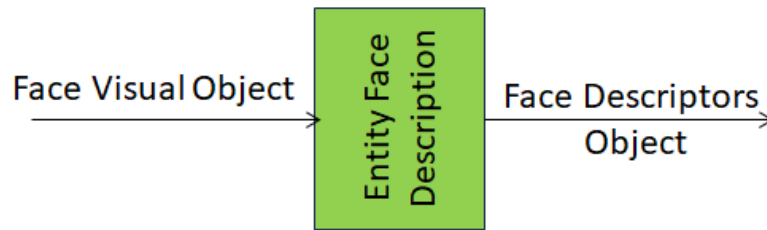


Figure 1 - Entity Face Description (PAF-EFD) AIM

7.1.9.3 Input/Output Data

Table 1 specifies the Input and Output Data of Entity Face Description (PAF-EFD) AIM.

Table 1 – I/O Data of the Entity Face Description (PAF-EFD) AIM

Input	Description
Face Visual Object	Entity Face to be Described.
Output	Description
Face Descriptors Object	Descriptors of Face.

7.1.9.4 SubAIMs

No SubAIMs.

7.1.9.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/EntityFaceDescription.json>

7.1.9.6 Profiles

No profiles.

7.1.9.7 Reference Software

The open-source Reference Software is [available](#). Send an email to the [MPAI Secretariat](#) to access the code.

7.1.9.8 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-EFD AIM.

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

Table 2 – Conformance Testing Method for PAF-EFD AIM

Receives	Face Visual Object	Shall validate against Visual Object Schema. Face Data shall conform with Visual Qualifier.
Produces	Face Descriptors Object	Shall validate against Face Descriptors Schema.

7.1.10 Response and Scene Rendering

7.1.10.1 Functions

The Response and Scene Rendering (PAF-RSR) AIM produces Audio, Speech, and Visual data representing a speaking avatar in an Audio-Visual Scene from a selection of input AV Scene Descriptors, input Point of View of the viewer, input Speech or Text, input Speech Model, input Avatar Model, and input Personal Status:

Receives	Point of View	Selected by viewer to see/hear the scene.
	AV Scene Descriptors	Describing the Audio-Visual Scene.
	Avatar PoV	The Point of View of the Avatar in the AV Scene.
	Input Speech	Speech alternative to Speech synthesised from Text Object.
	Speech Model	The Speech Model used to synthesise Speech.
	Text Object	To be used to: 1. Synthesise into Speech using a provided or an internal Speech Model 2. Generate Face Descriptors and Body Descriptors.
	Personal Status	To be used as additional information to synthesise Speech and generate Face and Body Descriptors.
	Avatar Model	When a specific Avatar Model is externally provided.
Produces	Output Speech	The output Speech Object .
	Audio Speech	The Audio of the Audio-Visual Scene Descriptors.
	Output Visual	The rendering of the animated avatar and the visual scene.

7.1.10.2 Reference Model

Figure 1 depicts the Reference Model of the Avatar Animation and Speech (PAF-AAS).

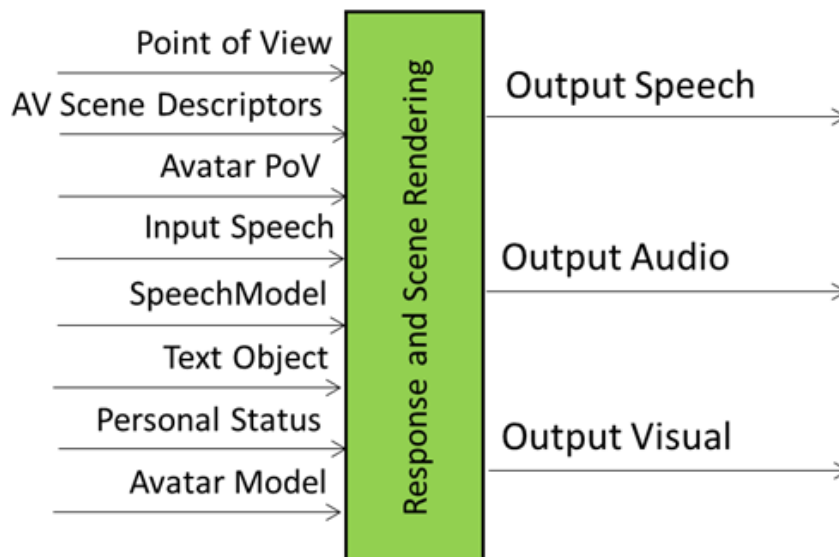


Figure 1– The Response and Scene Rendering (PAF-RSR) AIM

7.1.10.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Response and Scene Rendering (PAF-RSR) AIM.

Table 1– I/O Data of the Response and Scene Rendering (PAF-RSR) AIM

Input	Description
Point of View	Selected by viewer to see/hear the scene.
Audio-Visual Scene Descriptors	Describing the Audio-Visual Scene.
Avatar PoV	The Point of View of the Avatar in the AV Scene.
Input Speech	Speech alternative to Speech synthesised from Text Object.
Speech Model	The Speech Model used to synthesise Speech.
Text Object	For speech synthesis and Face and Body Descriptors.
Personal Status	additional information to synthesise Speech and generate Face and Body Descriptors.
Avatar Model	When a specific Avatar Model is externally provided.
Output	Description
Output Speech	The output Speech Object .
Output Audio	The Audio component of the Audio-Visual Scene Descriptors.
Output Visual	The rendering of the animated avatar and the visual scene.

7.1.10.4 SubAIMs

A Response and Scene Rendering (PAF-RSR) AIM may be implemented as a composite AIM depicted in Figure 2

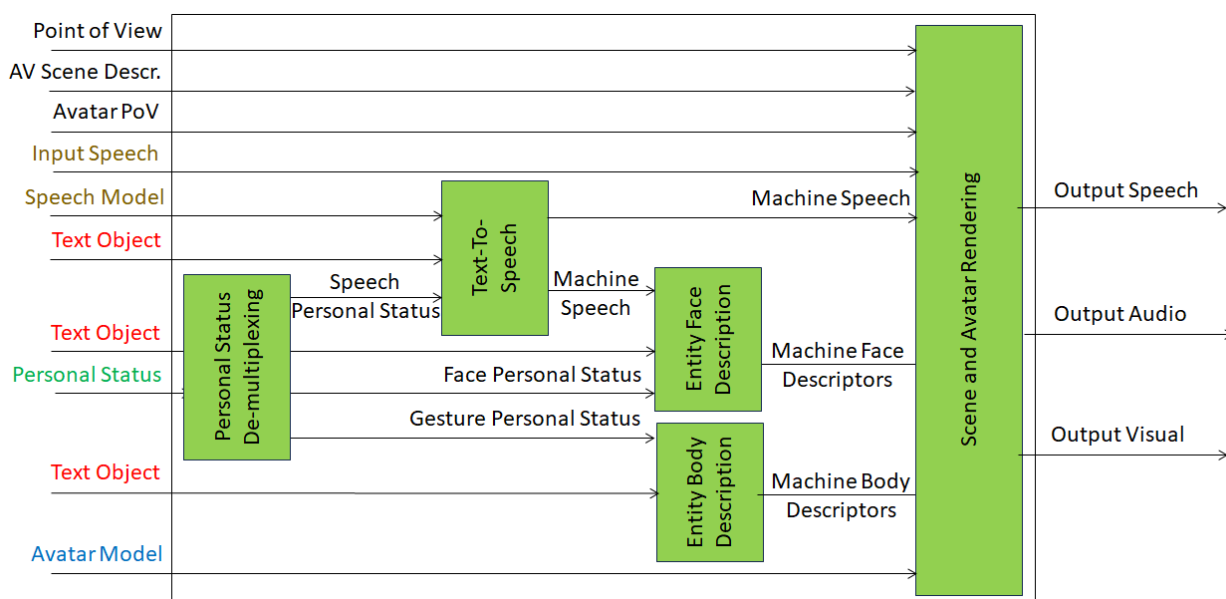


Figure 2 - Response and Scene Rendering (PAF-RSR) Composite AIM Reference Model

The AIMs composing the Response and Scene Rendering (PAF-RSR) Composite AIM are:

AIM	AIMs	Names	JSON
PAF-RSR		Response and Scene Rendering	X
	PAF-PSD	Personal Status Display	X
	MMC-TTS	Text-To-Speech	X
	PAF-EFD	Entity Face Description	X
	PAF-EBD	Entity Body Description	X
	PAF-SAR	Scene and Avatar Rendering	X

7.1.10.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/AvatarAnimationAndSpeech.json>

7.1.10.6 Profiles

No Profiles

7.1.10.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PSRAIM.

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

Table 2 – Conformance Testing Method for PAF-AAS AIM

Input	Description
Point of View	Shall validate against Point of View Schema.
Audio-Visual Scene Descriptors	Shall validate against Audio-Visual Scene Descriptors Schema.
Avatar PoV	Shall validate against Point of View Schema.
Input Speech	Shall validate against Speech Object Schema.
Speech Model	Shall validate against ML Model Schema.
Text Object	Shall validate against Text Object Schema.
Personal Status	Shall validate against Personal Status Schema.
Avatar Model	Shall validate against 3D Model Schema.
Output	Description
Output Speech	Shall validate against Speech Object Schema.
Output Audio	Shall validate against Audio Object Schema.
Output Visual	Shall validate against Visual Object Schema.

7.1.11 Face Identity Recognition

7.1.11.1 Functions

The Face Identity Recognition (PAF-FIR) AIM receives an input Visual Object representing a Face and produces a Bounding Box with the Face and the Identifier of the Face. The PAF-FIR AIM may also receive an input Visual Geometry of the Visual Scene, an input Time, and a Text Object related to the scene containing the input Visual Object:

Receives	<i>Text Object</i>	Text that is related with the Face to be identified.
	<i>Image Visual Object</i>	Image containing Face to be identified.
	<i>Face Time</i>	Time when the face should be identified.
	<i>Visual Scene Geometry</i>	Of the scene where the Face is located.
Searches for	<i>Bounding Boxes</i>	That include faces
Finds	best match	Between the Faces and those in a database.
Produces	<i>Face Identities</i>	Face Instance Identifiers.
	<i>Bounding Boxes</i>	Bounding Boxes that include faces.

7.1.11.2 Reference Model

Figure 1 depicts the Reference Model of the Face Identity Recognition AIM.

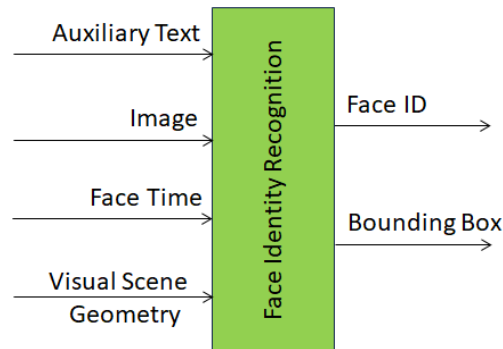


Figure 1 – Face Identity Recognition AIM

7.1.11.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Face Identity Recognition AIM.

Table 1 – I/O Data of the Face Identity Recognition AIM

Input	Description
Auxiliary Text Object	Text with a content related to Face ID.
Image Visual Object	An image containing the Face to be identified.
Face Time	The Time during which the Face should be identified.
Visual Scene Geometry	The Geometry of the Scene where the Face is located.
Output	Description
Face Identifiers	Associate strings to elements belonging to some levels in a hierarchical classification (taxonomy).
Bounding Boxes	The box containing the Face identified.

7.1.11.4 SubAIMs

No SubAIMs.

7.1.11.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/FaceIdentityRecognition.json>

7.1.11.6 Profiles

No Profiles.

7.1.11.6.1 Disclaimers

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

7.1.11.6.2 Guide to the PAF-FIR code

Use of this Reference Software for the PAF-FIR AI Module is for developers who are familiar with Python, Docker, RabbitMQ, and downloading models from HuggingFace. PAF-FIR performs face identity recognition with a pretrained FaceNet model; that is, it identifies the faces in a given number of frames per scene by comparison with a dataset of faces.

The PAF-FIR Reference Software is found at the MPAI [gitlab](https://gitlab.com/mpai-deepface) site. It contains:

1. src: a folder with the Python code implementing the AIM
2. Dockerfile: a Docker file containing only the libraries required to build the Docker image and run the container
3. requirements.txt: dependencies installed in the Docker image
4. README.md: where to find and save weights of face recognition model FaceNet512.

Library: <https://github.com/serengil/deepface>

7.1.11.6.3 Acknowledgements

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

7.1.11.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-FIR AIM.

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

Table 2 – Conformance Testing Method for PAF-FIR AIM

Receives	Text Object	Shall validate against Text Object Schema.
	Visual Object (Image)	Shall validate against Visual Object Schema. Image Data shall conform with Visual Qualifier.
	Face Time	Shall validate against Time Schema.
	Visual Scene Geometry	Shall validate against Visual Scene Geometry Schema.
Produces	Face Instance IDs	Shall validate against Instance ID Schema.
	Bounding Boxes	Shall validate against Bounding Box Schema. Bounding Box Data shall conform with Visual Qualifier.

7.1.11.8 Performance Assessment

Performance Assessment of a PAF-FIR AIM Implementation shall be performed using a dataset of faces for each face of which the Identity of the face is provided with reference to a Taxonomy. The Performance Assessment Report of an PAF-FIR AIM Implementation shall include:

1. The Identifier of the PAF-FIR AIM.
2. The identifier of the face dataset.
3. The identifier of the Taxonomy of face identifiers.
4. The Performance of the PAF-FIR AIM Implementation expressed by the Accuracy of the Identifiers provided by the output of the PAF-FIR AIM computed on all faces of the dataset referenced in 2 using the Taxonomy referenced in 3.

7.1.12 Scene and Avatar Rendering

7.1.12.1 Functions

The Scene and Avatar Rendering (PAF-SAR) AIM produces output Speech, Audio, Visual Data from input Audio-Visual Scene Descriptors, input Avatar Model, input Avatar Point of View, input Speech, input Face Descriptors and Body Descriptors:

Receives	Point of View	Selected by viewer to see/hear the scene.
	AV Scene Descriptors	Describing the Audio-Visual Scene.
	Avatar PoV	The Point of View of the Avatar in the AV Scene.
	Speech	Speech alternative to Speech synthesised from Text Object.
	Avatar Model	When a specific Avatar Model is externally provided.
	Face Descriptors	Descriptors of Avatar's Face.
	Body Descriptors	Descriptors of Avatar's Body.
Produces	Output Speech	The output Speech Object .
	Audio Speech	The Audio of the Audio-Visual Scene Descriptors.
	Output Visual	The rendering of the animated avatar and the visual scene.

7.1.12.2 Reference Model

Figure 1 depicts the Reference Model of the Scene and Avatar Rendering (PAF-SAR) AIM.

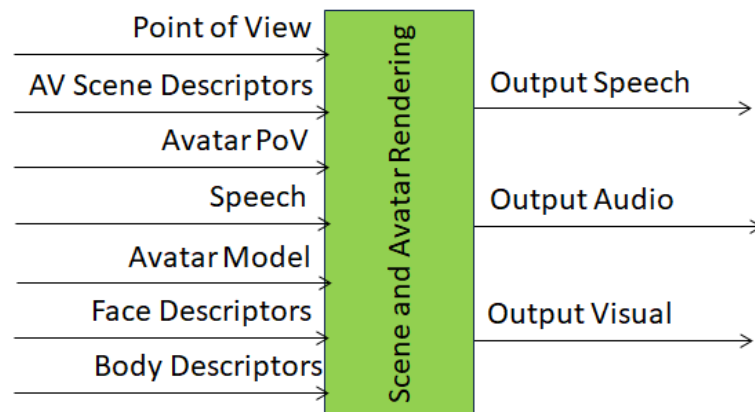


Figure 1– The Scene and Avatar Rendering (PAF-SAR) AIM AIM

7.1.12.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Scene and Avatar Rendering (PAF-SAR) AIM.

Table 1– I/O Data of the Scene and Avatar Rendering (PAF-SAR) AIM

Input	Description
-------	-------------

Point of View	Selected by viewer to see/hear the scene.
Audio-Visual Scene Descriptors	Describing the Audio-Visual Scene.
Avatar PoV	The Point of View of the Avatar in the AV Scene.
Input Speech	Speech alternative to Speech synthesised from Text Object.
Avatar Model	When a specific Avatar Model is externally provided.
Face Descriptors Object	Descriptors of Avatar's Face.
Body Descriptors Object	Descriptors of Avatar's Body.
Output	Description
Output Speech	The output Speech Object .
Output Speech	The Audio of the Audio-Visual Scene Descriptors.
Output Visual	The rendering of the animated avatar and the visual scene.

7.1.13 SubAIMs

No SubAIMs.

7.1.14 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/SceneAndAvatarRendering.json>

7.1.15 Profiles

No Profiles

7.1.16 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PSR AIM.

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

Table 2 – Conformance Testing Method for PAF-PSR AIM

Receives	Point of View	Shall validate against the Point of View Schema
	Audio-Visual Scene Descriptors	Shall validate against the AV Scene Descriptors Schema
	Avatar PoV	Shall validate against the Point of View Schema Schema
	Input Speech	Shall validate against the Speech Object Schema
	Avatar Model	Shall validate against the 3D Model Model Schema

	Face Descriptors Object	Shall validate against the Face Descriptors Schema
	Body Descriptors Object	Shall validate against the Body Descriptors Schema
Produces	Output Speech	Shall validate against the Speech Object Schema
	Output Speech	Shall validate against the Speech Object Schema
	Output Visual	Shall validate against the Visual Object Schema

7.1.17 Face Personal Status Extraction

7.1.17.1 Functions

The Face Personal Status Extraction (PAF-FPE) AIM receives an input Face Object or input Face Descriptors and extracts the Personal Status from the input data:

Receives	<i>Input Selector</i>	Signaling whether a Face Object or Face Descriptors are provided.
	<i>Face Object</i>	From which the Personal Status should be extracted.
	<i>Face Descriptors Object</i>	From which the Personal Status should be extracted (Externally computed).
Produces	<i>Face Personal Status</i>	The Personal Status of the Face Visual Object

7.1.17.2 Reference Model

Figure 1 specifies the Reference Model of the Face Personal Status Extraction (PAF-FPE) AIM.

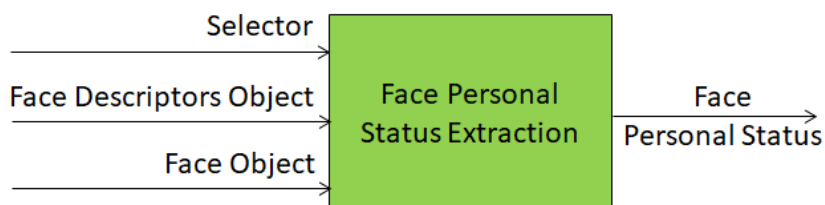


Figure 1 – Reference Architecture of the Face Personal Status Extraction (PAF-FPE) AIM

7.1.17.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Face Personal Status Extraction (PAF-FPE) AIM.

Table 1 – I/O Data of the Face Personal Status Extraction (PAF-FPE) AIM

Input	Description
Input Selector	Signals whether a Face Object or its Face Descriptors should be used to extracts Face Personal Status .
Face Visual Object	Face Object from which the Personal Status should be extracted..

Face Descriptors Object	Descriptors from which the Personal Status should be extracted..
Output	Description
Face Personal Status	The computed Face Personal Status.

7.1.17.4 SubAIMs

A Face Personal Status Extraction (PAF-FPE) AIM instance may be implemented as a Composite AIM as specified in Figure 2.

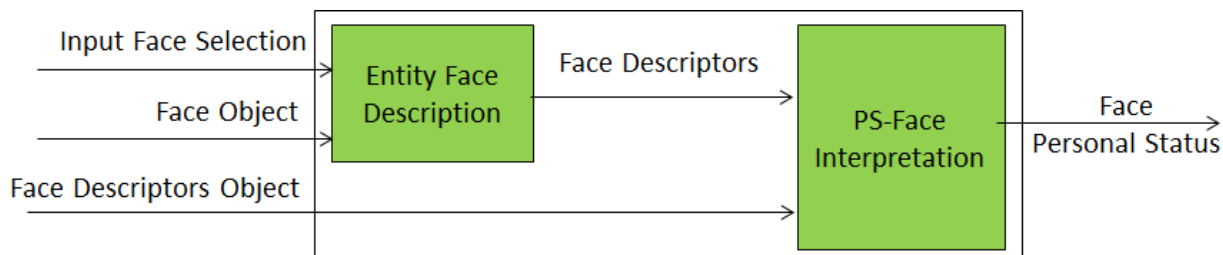


Figure 2 - Reference Model of Face Personal Status Extraction (PAF-FPE) Composite AIM

The AIMs composing the Face Personal Status Extraction (PAF-FPE) Composite AIM are:

AIM	AIMs	Names	JSON
PAF-FPE		Face Personal Status Extraction	X
	PAF-EFD	Entity Face Description	X
	PAF-PFI	PS-Face Interpretation	X

7.1.17.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/FacePersonalStatusExtraction.json>

7.1.17.6 Profiles

No Profiles.

7.1.17.7 Conformance Testing

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

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

Table 2 – Conformance Testing Method for PAF-FPE AIM

Receives	Input Selector	Shall validate against Selector Schema.
----------	--------------------------------	---

	Visual Object (Face)	Shall validate against Face Object Schema. Face Data shall conform with Visual Qualifier.
	Face Descriptors Object	Shall validate against Face Descriptors Schema.
Produces	Face Personal Status	Shall validate against Face Personal Status Schema.

Table 3 provides an example of MMC-FPE AIM conformance testing. This specification provides the methods and the datasets to test the conformance of a PAF-FPE instance without input Face Descriptors.

Table 3 – An example MMC-FPE AIM conformance testing

Input Data	Data Type	Input Conformance Testing Data
Face Object	AVC	All input Video files to be drawn from Video files .
Output Data	Data Type	Data Format
Emotion (Face)	JSON	All Emotion JSON Files shall validate against Emotion JSON Schema.

emotion_Name and emotion_SetName must be present in the output JSON file of Emotion. The value of either of the two may be null.

7.1.18 Service Participant Authentication

7.1.18.1 Functions

The Service Participant Authentication (PAF-SPA) AIM receives the Identifier of a Participant in a Service Session, the Face and/or speech segment of the Service Participant and produces an Identifier of a Service Subscriber:

Receives	Participant ID	ID of a Participant in a Service Session.
	Face Visual Object	The Face of the Service Participant.
	Speech Object	A speech segment of the Service Participant.
Uses	Face and Speech Object	To find the correspondence between the Participant ID of a Participant requesting access to Service and the Subscriber ID of the Service Provider.
Produces	Subscriber ID	The Subscriber identified by the PAS-SPA AIM.

7.1.18.2 Reference Model

Figure 1 specifies the Reference Model of Service Participant Authentication (PAF-SPA).

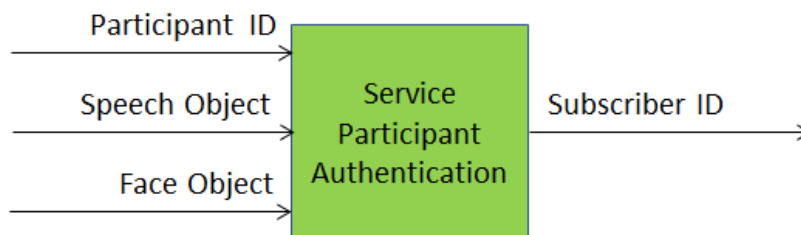


Figure 1 – The Service Participant Authentication (PAF-SPA) AIM

7.1.18.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Service Participant Authentication (PAF-SPA) AIM.

Table 1 – Service Participant Authentication (PAF-SPA) AIM

Input	Description
Participant Identifier	ID of a Participant in a session of a Service. From an upstream AIM or another AIW.
Speech Object	Speech segment of Participant.
Face Visual Object	Face of Participant.
Output	Description
Subscriber Identifier	ID of Service Subscriber

7.1.18.4 SubAIMs

No SubAIMs.

7.1.18.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/ServiceParticipantAuthentication.json>

7.1.18.6 Profiles

No Profiles.

7.1.18.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-SPA AIM.

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

Table 2 – Conformance Testing Method for PAF-SPA AIM

Receives	Participant Identifier	Shall be string.
	Speech Object	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Face Visual Object	Shall validate against Visual Object Schema. Speech Data shall conform with Visual Qualifier.
Produces	Subscriber Identifier	Shall validate against Instance ID Schema..

7.1.18.8 Performance Assessment

Performance Assessment of a PAF-SPA AIM Implementation shall be performed using the subsets of two datasets of:

1. Speech segments all in the same language for each segment of which the Identity of the Speaker is provided with reference to a Taxonomy.
2. Faces for each face of which the Identity of the face is provided with reference to a Taxonomy

for which the identity of a speech segment's speaker is univocally connected to the identity of a face and vice-versa.

The Performance Assessment Report of a PAF-SPA AIM Implementation shall include:

1. The Identifier of the PAF-SPA AIM.
2. The identifier of the speech segment dataset.
3. The identifier of the face dataset.
4. The Taxonomy of speaker identifiers.
5. The Taxonomy of face identifiers.
6. The Accuracy of the Service Participant Authentication AIM defined as the number of correctly recognised pairs of speech segments and faces of the same Speaker and Face to the total number of submitted Speech and Face pairs..

7.1.19 Gesture Personal Status Extraction

7.1.19.1 Functions

The Gesture Personal Status Extraction (PAF-GPE) AIM receives an input Body Object or input Body Descriptors and extracts the Personal Status from the input data:

Receives	<i>Input Selector</i>	Signaling whether a Gesture Object or Gesture Descriptors are provided.
	<i>Body Visual Object</i>	From which the Personal Status should be extracted.
	<i>Gesture Descriptors Object</i>	From which the Personal Status should be extracted (Externally computed).
Produces	<i>Gesture Personal Status.</i>	The Gesture Personal Status of the Body Image Object.

7.1.19.2 Reference Model

Figure 1 specifies the Reference Model of the Gesture Personal Status Extraction (PAF-GPE) AIM.

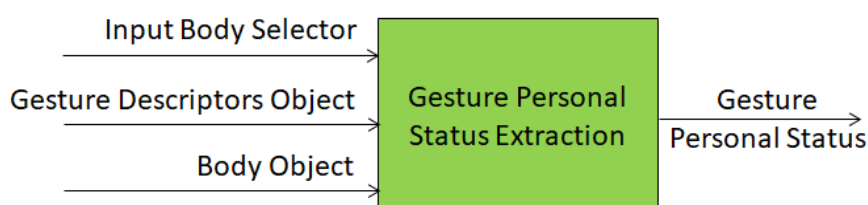


Figure 1 – Reference Architecture of the Gesture Personal Status Extraction (PAF-GPE) AIM

7.1.19.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Gesture Personal Status Extraction (PAF-GPE) AIM.

Table 1 – I/O Data of the Gesture Personal Status Extraction (PAF-GPE) AIM

Input	Description
Input Selector	Signals whether a Body Object or its Gesture Descriptors should be used to extracts Gesture Personal Status.

Body Visual Object	Gesture Object from which the Personal Status should be extracted.
Gesture Descriptors Object	Descriptors from which the Personal Status should be extracted..
Output	Description
Gesture Personal Status	The computed Gesture Personal Status.

7.1.19.4 SubAIMs

A Gesture Personal Status Extraction (PAF-GPE) AIM instance may be implemented as a Composite AIM as specified in Figure 2.

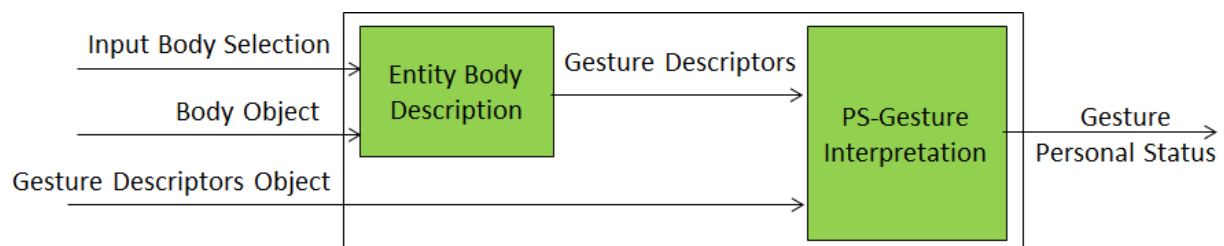


Figure 2 - Reference Model of Gesture Personal Status Extraction (PAF-GPE) Composite AIM

The AIMs composing the Gesture Personal Status Extraction (PAF-GPE) Composite AIM are:

AIM	AIMs	Names	JSON
PAF-GPE		Gesture Personal Status Extraction	X
	PAF-EGD	Entity Gesture Description	X
	PAF-PGI	PS-Gesture Interpretation	X

7.1.19.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/GesturePersonalStatusExtraction.json>

7.1.19.6 Profiles

No Profiles.

7.1.19.7 Conformance Testing

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

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

Table 2 – Conformance Testing Method for PAF-GPE AIM

Receives	Input Selector	Shall validate against Selector Schema.
	Visual Object (Body)	Shall validate against Visual Object Schema. Body Data shall conform with Visual Object.
	Gesture Descriptors Object	Shall validate against Body Descriptors Schema.
Produces	Gesture Personal Status	Shall validate against Body Personal Status Schema.

7.1.20 Visual Scene Creation

7.1.20.1 Functions

The Visual Scene Creation (PAF-VSC) AIM receives input Avatars with their Identifiers and input Spatial Attitudes and input Audio-Visual Scene Descriptors and produces output Scene Descriptors by adding the Avatars to the input Audio-Visual Scene Descriptors with the input Spatial Attitudes and providing the Spatial Attitudes of the Avatars' mouths:

Receives	Avatar ID	Avatar Identifiers.
	Audio-Visual Scene Descriptors	Descriptors of the full AV Scene.
	Avatars	Avatars in the Visual Scene
	Spatial Attitude	Spatial Attitudes of Avatars
Produces	Visual Scene Descriptors	where the Avatars have been added to input AV Scene Descriptors
	Spatial Attitude of Avatars' Mouths	The Mouths of the Avatars' Spatial Attitude

7.1.20.2 Reference Model

Figure 1 depicts the Reference Architecture of the Visual Scene Creation AIM ((PAF-VSC).

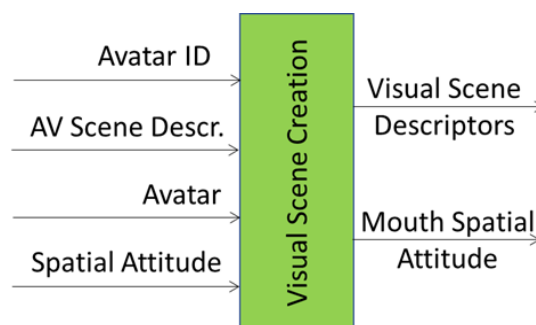


Figure 1– The Visual Scene Creation (PAF-VSC) AIM

7.1.20.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Visual Scene Creation (PAF-VSC) AIM.

Table 1– I/O Data of the Visual Scene Creation (PAF-VSC) AIM

Input	Description
Avatar IDs	Identifiers of Avatars being handled.
Audio-Visual Scene Descriptors	Descriptors of the Initial Visual Scene
Avatars	Avatars of Participants and VMS
Spatial Attitudes	Spatial Attitudes of Participants' and VMS' Avatars
Output	Description
Visual Scene Descriptors	Descriptors of the entire Visual Scene
Mouth Spatial Attitudes	The Spatial Attitudes of the Mouths of Participants' and VMS' Avatars

7.1.20.4 SubAIMs

No SubAIMs.

7.1.20.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/VisualSceneCreation.json>

7.1.20.6 Profiles

No Profiles

7.1.20.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-VSC AIM.

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

Table 2 – Conformance Testing Method for PAF-VSC AIM

Receives	Avatar ID	Shall be string or validate against Instance ID Schema.
	Audio-Visual Scene Descriptors	Shall validate against AV Scene Descriptors Schema.
	Avatars	Shall validate against Avatar Schema. Avatar Data shall conform with 3D Model Qualifier.
	Spatial Attitudes	Shall validate against Spatial Attitude Schema.
Produces	Visual Scene Descriptors	Shall validate against Visual Scene Descriptors Schema.
	Spatial Attitude of Avatars' Mouths	Shall validate against Spatial Attitude Schema.

7.1.21 Portable Avatar Demultiplexing

7.1.21.1 Functions

The Portable Avatar Demultiplexing (PAF-PDX) AIM extracts the components of an input Portable Avatar - Portable Avatar ID, Avatar Space-Time, Avatar, Language Selector, Speech Object, Text Object, Speech Model, Personal Status, Audio-Visual Scene Descriptors, and Audio-Visual Scene Space Time:

Receives	Portable Avatar
Demultiplexes	Elements in Portable Avatar.
Produces	- Portable Avatar ID
	- Avatar Space-Time
	- Avatar
	- Language Selector
	- Speech Object
	- Text Object
	- Speech Model
	- Personal Status
	- Audio Visual Scene Descriptors
	- Audio Visual Scene Space Time

7.1.21.2 Reference Model

Figure 1 specifies the Reference Model of the Personal Avatar Demultiplexing (PAF-PDX) AIM.

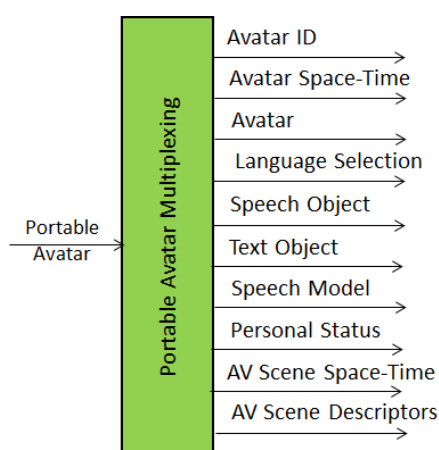


Figure 1– The Personal Avatar Demultiplexing (PAF-PDX) AIM

7.1.21.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Portable Avatar Demultiplexing (PAF-PDX) AIM.

Table 1 – Portable Avatar Demultiplexing (PAF-PDX) AIM

Input	Description
Portable Avatar	From an upstream AIM or another AIW.
Output	Description
AvatarID	Avatar ID.
Avatar Space-Time	Portable Avatar Time.
Avatar	Avatar in Portable Avatar.
Language Selector	Language of Avatar.
Speech Object	The Speech in the time when the PA is valid.
Text Object	The Time in the time when the PA is valid.
Speech Model	The NN Model used to synthesise text.
Avatar Personal Status	The Avatar's Personal Status.
AV Scene Descriptors	Descriptors of AV Scene.
AV Scene Space-Time	Space-Time info of AV Scene.

7.1.21.4 SubAIMs

No SubAIMs.

7.1.21.5 JSON Metadata

<https://schemas.mpai.community/PAF/V1.5/AIMs/PortableAvatarDemultiplexing.json>

7.1.21.6 Profiles

No Profiles.

7.1.21.7 Conformance Testing

Table 2 provides the Conformance Testing Method for PAF-PDX AIM.

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

Table 2 – Conformance Testing Method for PAF-PDX AIM

Receives	Portable Avatar	Shall validate against Portable Avatar Schema. Portable Avatar Data shall conform with respective Qualifiers.
Produces	Portable Avatar ID	Shall be string or validate against Instance ID Schema.

	Avatar Space-Time	Shall validate against Space-Time Schema.
	Avatar	Shall validate against Avatar Schema. Avatar Model Data shall conform with 3D Model Qualifier.
	Language Selector	Shall validate against “Language” Selector Schema.
	Text Object	Shall validate against Text Object Schema. Text Data shall conform with Text Qualifier.
	Speech Object	Shall validate against Speech Object Schema. Speech Data shall conform with Speech Qualifier.
	Speech Model	Shall validate against Machine Learning Model Schema. Speech Model Data shall conform with Machine Learning Model Qualifier.
	Personal Status	Shall validate against Personal Status Schema.
	Audio Visual Scene Descriptors	Shall validate against AV Scene Descriptors Schema.
	Audio Visual Scene Space-Time	Shall validate against Space-Time Schema.

7.2 Reference Software

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

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

Note that at this stage only part of the MPAI-PAF AIMS have a Reference Software Implementation.

7.3 Conformance Testing

An implementation of an AI Module conforms with MPAI-PAF V1.5 if it accepts as input and produces as output Data and/or Data Objects (combination of Data of a Data Type and its Qualifier) conforming with those specified by MPAI-PAF V1.5.

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

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

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

7.4 Performance Assessment

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

1. *Quality*: Performance Assessment measures how well an AIM performs its function, using a metric that depends on the nature of the function, e.g., how well a Face Identity Recognition (FIR) AIM identifies Faces.
2. *Bias*: Performance Assessment measures the preference given by an AIM to certain elements, using a metric that depends on a bias related to certain attributes of the AIM. For instance, a Face Identity Recognition (FIR) AIM tends to have a higher correct identification of Face having a particular skin colour.
3. *Legal* compliance: Performance Assessment measures how well an AIM performs its function, using a metric that assesses its accordance with a certain legal standard.
4. *Ethical* compliance: the Performance Assessment of an AIM can measure the compliance of an AIM to a target ethical standard.

The current MPAI-PAF V1.4 does not provide AIM Performance Assessment methods.

8 Data Types

8.1 Technical Specifications

This page gives the links to the specification of Data Types of **Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5**. All previously specified MPAI-PAF Data Types that are specified by V1.4 are superseded. Use of earlier versions is permitted if their version is explicitly signalled.

Table 1 provides the Data Types specified by MPAI-PAF V1.5. MPAI-PAF AIWs and AIMs also utilise Data Types specified by other MPAI Technical Specifications.

Acronym	Name	JSON	Acronym	Name	JSON
PAF-AVT	Avatar	X	PAF-BDO	Body Descriptors Object	X
PAF-FDO	Face Descriptors Object	X	PAF-GSD	Gesture Descriptors Object	X
PAF-MCO	MoCap Object	X	PAF-PAV	Portable Avatar	X

8.1.1 Avatar

8.1.1.1 Definition

A Data Type that includes data characterising the Avatar, the Virtual Space in which the Avatar is located, and the Space-Time information of the Avatar in the Virtual Space

8.1.1.2 Functional Requirements

Avatar conveys the following information:

1. The ID of the Virtual Space (M-Instance).
2. The ID of the Avatar.
3. The Time and Spatial Attitude of the Avatar is in the M-Instance.
4. The set of Data characterising an Avatar:
 1. 3D Model
 2. Face Descriptors
 3. Body Descriptors

4. Process ID morphing the Model
5. ClothID

An Avatar Model of a human may:

1. Faithfully reproduce the visual appearance of the human.
2. Have their visual appearance altered, compared to that of the human.
3. Have an unrelated visual appearance.
4. Display a presumptive Personal Status in speech, face, and gesture.

8.1.1.3 Syntax

<https://schemas.mpai.community/PAF/V1.5/data/Avatar.json>

8.1.1.4 Semantics

Label	Description
Header	Avatar Header.
– Standard-Avatar	The characters “PAF-AVT-V”
– Version	Major version
– Dot-separator	The character "."
– Subversion	Minor version
MInstanceID	ID of Virtual Space the Avatar belongs to.
AvatarSpaceTime	The inherent Space-Time info of the Avatar.
AvatarID	Identifier of Avatar.
AvatarData	Set of Avatar-related Data.
- AvatarModel	Model of Avatar.
- BodyDescriptorsObject	Avatar Body Descriptors Object.
- FaceDescriptorsObject	Avatar Face Descriptors Object of Avatar.
- ProcessID	ID of process morphing Model.
- ClothID	ID of cloth work by Avatar.
DescrMetadata	Descriptive Metadata

8.1.1.5 Conformance Testing

A Data instance Conforms with Avatar (PAF-AVT) V1.5 if:

1. JSON Data validate against the Avatar’s JSON Schema.
2. All Data in the Avatar's Schema
 1. Have the specified type.
 2. Validate against their JSON Schemas.

8.1.2 Body Descriptors Object

8.1.2.1 Definition

Body Descriptors Object refers to

1. Body Descriptors Data representing a human or a humanoid.
2. Body Descriptors Data Qualifier specified by MPAl-TFA providing information on Sub-Types, Formats and Attributes of Body Descriptors Data.

8.1.2.2 Functional Requirements

Body Descriptors should enable the representation of the joints of a body.

8.1.2.3 Syntax

<https://schemas.mpai.community/PAF/V1.5/data/BodyDescriptorsObject.json>

8.1.2.4 Semantics

Label	Description
Header	Body Descriptors Object Header.
– Standard-BodyDescriptorsObject	The characters “OSD-BDO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	ID of the Virtual Space where the Body is located.
BodyDescriptorsObjectID	Identifier of Body Descriptors.
BodyDescriptorsSpaceTime	Space-Time information of Body in the Virtual Space
BodyDescriptorsData	Data of Body Descriptors
BodyDescriptorsQualifier	Qualifier of Body Descriptors
DescrMetadata	Descriptive Metadata

8.1.2.5 Conformance Testing

A Data instance Conforms with Body Descriptors Object (PAF-BDO) if

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

8.1.3 Face Descriptors Object

8.1.3.1 Definition

Face Descriptors Object refers to

1. Face Descriptors Data representing the features of the Face of an Entity.
2. Face Descriptors Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of Body Descriptors Data.

8.1.3.2 Functional Requirements

The features depend on the motion of the muscles of the Face of an Entity.

8.1.3.3 Syntax

<https://schemas.mpai.community/PAF/V1.5/data/FaceDescriptorsObject.json>

8.1.3.4 Semantics

Label	Description
Header	Face Descriptors Object Header.

– Standard-FaceDescriptorsObject	The characters “OSD-FDO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	ID of the Virtual Space where the Face is located.
FaceDescriptorsObjectID	Identifier of Face Descriptors.
FaceDescriptorsSpaceTime	Space-Time information of Face in the Virtual Space
FaceDescriptorsData	Data of Face Descriptors
FaceDescriptorsQualifier	Qualifier of Face Descriptors
DescrMetadata	Descriptive Metadata

8.1.3.5 Conformance Testing

A Data instance Conforms with Face Descriptors Object (PAF-FDO) if

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

8.1.3.6 Face Action Units (informative)

The Face Actions Units of the Facial Action Coding System (FACS) were originally developed by Carl-Herman Hjortsjö, adopted by Paul Ekman and Wallace V. Friesen (1978) and updated by [Ekman, Friesen, and Joseph C. Hager](#) (2002). Each Action Unit is represented by an Action Unit ID. The provide a set of Face Descriptors.

AU	Description	Facial muscle generating the Action
1	Inner Brow Raiser	Frontalis, pars medialis
2	Outer Brow Raiser	Frontalis, pars lateralis
4	Brow Lowerer	Corrugator supercilii, Depressor supercilii
5	Upper Lid Raiser	Levator palpebrae superioris
6	Cheek Raiser	Orbicularis oculi, pars orbitalis
7	Lid Tightener	Orbicularis oculi, pars palpebralis
9	Nose Wrinkler	Levator labii superioris alaeque nasi
10	Upper Lip Raiser	Levator labii superioris
11	Nasolabial Deepener	Zygomaticus minor
12	Lip Corner Puller	Zygomaticus major
13	Cheek Puffer	Levator anguli oris (a.k.a. Caninus)
14	Dimpler	Buccinator
15	Lip Corner Depressor	Depressor anguli oris (a.k.a. Triangularis)
16	Lower Lip Depressor	Depressor labii inferioris
17	Chin Raiser	Mentalis
18	Lip Pucker	Incisivii labii superioris and Incisivii labii inferioris
20	Lip stretcher	Risorius with platysma

22	Lip Funneler	Orbicularis oris
23	Lip Tightener	Orbicularis oris
24	Lip Pressor	Orbicularis oris
25	Lips part	Depressor labii inferioris or relaxation of Mentalis, or Orbicularis oris
26	Jaw Drop	Masseter, relaxed Temporalis and internal Pterygoid
27	Mouth Stretch	Pterygoids, Digastric
28	Lip Suck	Orbicularis oris
41	Lid droop	Relaxation of Levator palpebrae superioris
42	Slit	Orbicularis oculi
43	Eyes Closed	Relaxation of Levator palpebrae superioris; Orbicularis oculi, pars palpebralis
44	Squint	Orbicularis oculi, pars palpebralis
45	Blink	Relaxation of Levator palpebrae superioris; Orbicularis oculi, pars palpebralis
46	Wink	Relaxation of Levator palpebrae superioris; Orbicularis oculi, pars palpebralis
61	Eyes turn left	Lateral rectus, medial rectus
62	Eyes turn right	Lateral rectus, medial rectus
63	Eyes up	Superior rectus, Inferior oblique
64	Eyes down	Inferior rectus, Superior oblique

The eye motion Action Units (Eyes turn left/right and Eyes up/down) have an additional parameter representing the degrees of the motion of a line perpendicular to the eye with respect to the 0° of the resting position when the eye looks horizontally and perpendicular to the body. The value of the angle is expressed with 1 Byte where 0= 0° degrees for the resting position (0°) and 255 for 90°.

8.1.3.7 Mapping of Face Action Units to Personal Status (Informative)

MPAI has defined a set of Cognitive States, Emotions, and Social Attitudes included in [Personal Status](#). The Table below offers an informative mapping of some elements of Personal Status to Action Units (from [1](#)).

Personal Status	Cognitive State	Emotion	Prototypical (and variant AUs)
Happy		17	12, 25 [6 (51%)]
Sad		32	4, 15 [1 (60%), 6 (50%), 11 (26%), 17 (67%)]
Fearful		13	1, 4, 20, 25 [2 (57%), 5 (63%), 26 (33%)]
Angry		2	4, 7, 24 [10 (26%), 17 (52%), 23 (29%)]
Surprised	18		1, 2, 25, 26 [5 (66%)]
Disgusted		11	9, 10, 17 [4 (31%), 24 (26%)]

This Table was obtained through a series of experiments with human subjects. AUs used by a subset of the subjects are shown in brackets with the percentage of the subjects using this less common AU in parentheses.

[1] [Compound facial expressions of emotion | PNAS](#)

8.1.4 Gesture Descriptors Object

8.1.4.1 Definition

Body Descriptors Object refers to

1. Body Descriptors Data representing a human or a humanoid.
2. Body Descriptors Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of Body Descriptors Data.

8.1.4.2 Functional Requirements

Body Descriptors should enable the representation of the joints of a body.

8.1.4.3 Syntax

<https://schemas.mpai.community/PAF/V1.5/data/BodyDescriptorsObject.json>

8.1.4.4 Semantics

Label	Description
Header	Body Descriptors Object Header.
– Standard-BodyDescriptorsObject	The characters “OSD-BDO-V”
– Version	Major version – 1 or 2 characters
– Dot-separator	The character “.”
– Subversion	Minor version – 1 or 2 characters
M-InstanceID	ID of the Virtual Space where the Body is located.
BodyDescriptorsObjectID	Identifier of Body Descriptors.
BodyDescriptorsSpaceTime	Space-Time information of Body in the Virtual Space
BodyDescriptorsData	Data of Body Descriptors
BodyDescriptorsQualifier	Qualifier of Body Descriptors
DescrMetadata	Descriptive Metadata

8.1.4.5 Conformance Testing

A Data instance Conforms with Body Descriptors Object (PAF-BDO) if

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

8.1.4.6 Definition

MoCap Object refers to

1. MoCap Descriptors Data representing the movement of a human.
2. MoCap Descriptors Data Qualifier specified by MPAI-TFA providing information on Sub-Types, Formats and Attributes of Body Descriptors Data.

8.1.4.7 Functional Requirements

MoCap Object represents the motion of the joints of a body.

8.1.4.8 Syntax

<https://schemas.mpai.community/PAF/V1.5/MoCapObject.json>

8.1.4.9 Semantics

Label	Description
Header	The Header of the MoCap Object.
– Standard-MoCapObject	The characters “PAF-MCO-V”
– Version	Major version
– Dot-separator	The character "."
– Subversion	Minor version
MInstanceID	The ID of the M-Instance.
MoCapObjectID	Identifier of the MoCap Object.
MoCapQualifier	Set of Data related to MoCap.
DescrMetadata	Descriptive Metadata

8.1.5 Portable Avatar

8.1.5.1 Definition

A Data Type that includes:

1. A set of avatar-related Data: M-Instance ID, Avatar ID, Avatar Space-Time, Avatar, Language Selector, Text, Speech Object, Personal Status, and
2. Descriptors of the Audio-Visual Scene where the Avatar is embedded and its Space-Time information.

8.1.5.2 Functional Requirements

Portable Avatar provides the following set of Data characterising a speaking avatar in a virtual space (M-Instance):

1. The ID of the virtual space (M-Instance) where the Portable Avatar is to be placed.
2. The space and time information of the “environment” to be placed in the M-Instance.
3. The Audio-Visual Scene representing the “environment”.
4. The space and time information of the Avatar in the scene.
5. The Avatar represented as a 3D Model, its Face Descriptors and Body Descriptors.
6. The Language Preference of the Avatar.
7. The Text Object the Avatar is associated with, or which will be converted into a Speech Object.
8. The Speech Model used to synthesise the Text Object.
9. The Speech Object alternative to the Text Object that the Avatar utters.
10. The Personal Status of the Avatar.

8.1.5.3 Syntax

<https://schemas.mpai.community/PAF/V1.5/data/PortableAvatar.json>

8.1.5.4 Semantics

Label	Description
Header	The Header of the Portable Avatar Data.
– Standard-PortableAvatar	The characters “PAF-PAV-V”
– Version	Major version
– Dot-separator	The character "."
– Subversion	Minor version
MInstanceID	The ID of the M-Instance.
PortableAvatarID	Identifier of the Portable Avatar.
PortableAvatarData	Set of Data related to Avatar.
- AudioVisualSceneSpaceTime	Space and Time info of AV Scene in M-Instance.
- AudioVisualSceneDescriptors	AV Scene Descriptors.
- AvatarSpaceTime	Space-Time of Avatar instance in AV Scene.
- Avatar	Avatar's Model and Face and Body Descriptors.
- LanguageSelector	Avatar's Language Preference.
- TextObject	Text associated with Avatar.
- SpeechObject	Set of Data related to Speech Object.
- SpeechModel	Neural Network Model for Speech Synthesis.
- PersonalStatus	Personal Status of Avatar.
DescrMetadata	Descriptive Metadata

8.1.5.5 Conformance Testing

A Data instance Conforms with Portable Avatar (PAF-PAV) V1.5 if:

1. JSON Data validate against the Portable Avatar 's JSON Schema.
2. All Data in the Portable Avatar 's JSON Schema
 1. Have the specified type.
 2. Validate against their JSON Schemas.
 3. Conform with their Data Qualifiers if present.

8.2 Conformance testing

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

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

8.3 Performance Assessment

Performance Assessment provides methods of assessing the performance of an Data instance. Performance may have various connotations, such as:

1. *Quality*: Performance Assessment measures the quality of the Data instance using a metric that depends on the nature of the Data, e.g., the accuracy of reproduction of face emotions on a synthetic face.
2. *Bias*: Performance Assessment measures the disparity of treatment applied to the Data instance using a metric that depends on a bias related to certain attributes of the Data instance.
3. Legal compliance: Performance Assessment uses an appropriate metric to measure how well the Data instance complies with with a certain legal standard.