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

MPAI Technical Specification

Context-based Audio Enhancement
MPAI-CAE – Use Cases

V2.4

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:
Context-based Audio Enhancement (MPAI-CAE) – Use Cases V2.4

Contents

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. <u>Publish</u> upfront clear Intellectual Property Rights licensing frameworks.
2. <u>Adhere</u> to a rigorous standard development process.
3. <u>Be friendly</u> 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. <u>Be attractive</u> to different industries, end users, and regulators.
5. <u>Address</u> 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.

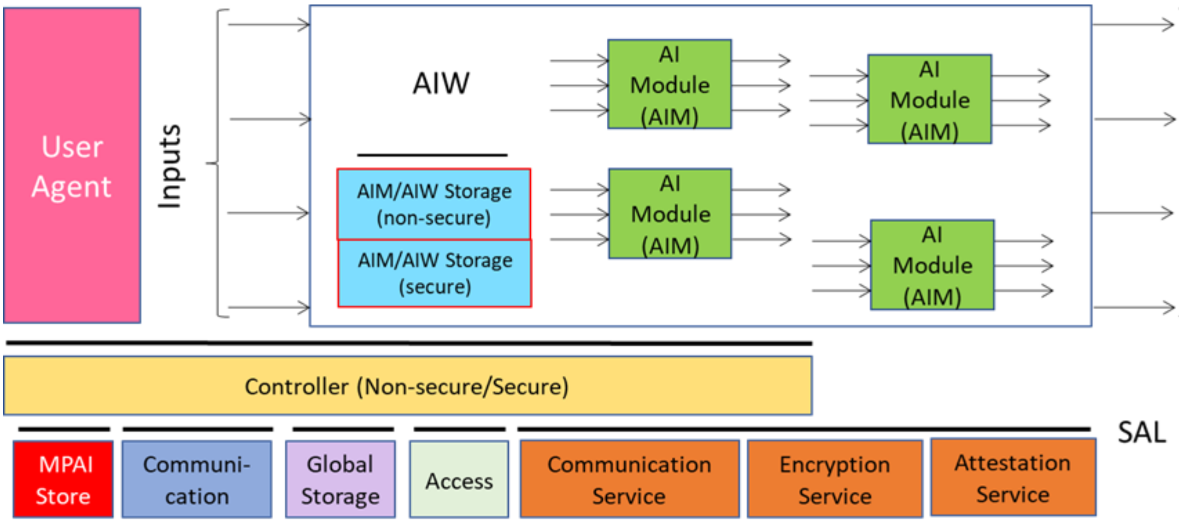


Figure 1 – The AI Framework (MPAI-AIF) V2 Reference Model

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.

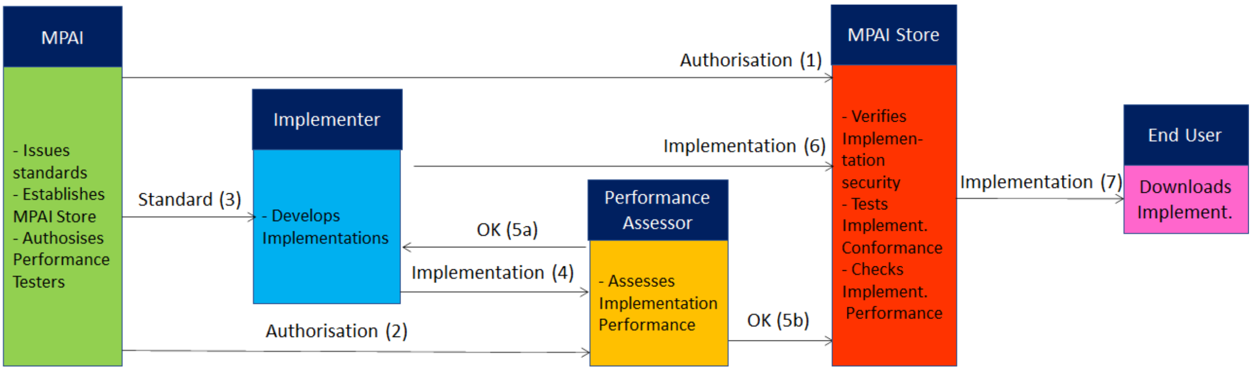


Figure 2 – The MPAI Ecosystem

2 Introduction (Informative)

(Informative)

Technical Specification: Context-based Audio Enhancement (MPAI-CAE) - Use Cases (CAE-USC) V2.4 collects Use Cases that improve the user experience of audio application using technologies that act on the input audio content using context information. The coverage of MPAI-CAE Use Cases includes entertainment, communication, teleconferencing, gaming, post-production, restoration etc. in a variety of contexts such as in the home, in the car, on-the-go, in the studio etc.

The Use Cases are implemented as AI Workflows, AI Modules, and Data Types according to Technical Specification: AI Framework (MPAI-AIF) V2.2.

The currently specified use cases are *Audio Recording Preservation (ARP), Emotion Enhanced Speech (EES), Enhanced Audioconference Experience (EAE), and Speech Restoration System (SSR)*.

In this Introduction and in the following Chapters, Capitalised Terms are defined in Table 1 if they are specific to this Technical Specification or online if they are shared with other MPAI Technical Specifications.

3 Scope

Technical Specification: Context-based Audio Enhancement (MPAI-CAE) – Use Cases (CAE-USC) V2.4 specifies AI Workflows, AI Modules, and Data Types that support four use cases: *Audio Recording Preservation (ARP), Emotion Enhanced Speech (EES), Enhanced Audioconference Experience (EAE), and Speech Restoration System (SSR)*.

Each Use Case normatively defines:

1. The Functions of the AIW and of the AIMs.
2. The Connections between and among the AIMs.

3. The Semantics and the Formats of the input and output data of the AIWs and its AIMs.

The word *normatively* implies that an Implementation claiming Conformance:

1. If an *AIW*, shall:
    1. Have the AIW Function specified in the relevant Use Case.
    2. Have all its AIMs and Connections conforming with the Reference Model of the AIW implementing the Use Case.
    3. Use the Data Types specified by the relevant web page.
2. If an *AIM*, shall:
    1. Have the AIM Function specified by the relevant web page.
    2. Use Data Types specified by the relevant web page.

Users of this Technical Specification should note that:

1. Implementers may use the Reference Software Implementations at the specified conditions.
2. The Conformance Testing specification can be used to test the Conformity of an Implementation to this Technical Specification.
3. Performance Assessors can assess the level of Performance of an Implementation based on the Performance Assessment specification of this Technical Specification.
4. Technical Specification: Governance of the MPAI Ecosystem (MPAI-GME) V2.0 specifies the operation of the MPAI Ecosystem.

This version of the MPAI-CAE Technical Specification has been developed by the CAE-DC Development Committee. Future Versions may revise and/or extend the Scope of this Technical Specification.

4 Definitions

Capitalised Terms used in this standard have the meaning defined in *Table 1*. All MPAI-defined Terms are accessible *online*.

Table 1 – Table of terms and definitions

Term	Definition
Access Copy Files	Set of files providing the information stored in an audio tape recording, including Restored Audio Files, suitable for audio information access, but not for long-term preservation.
Audio Block	A set of consecutive Audio samples.
Audio Channel	A sequence of Audio Blocks.
Audio Data	Digital representation of an analogue audio signal sampled at a frequency between 8-192 kHz with a number of bits/sample between 8 and 64.
Audio File	An Audio Object having a File Transport.
Audio Object	Audio Data and optional metadata regarding Sub-Types, Formats and Attributes of the Audio Data.
Audio Scene Geometry	A Data Type describing the spatial arrangement of the Audio Objects and Sub-Scenes of a Scene.
Audio Segment	An Audio Block with Start Time and an End Time Labels corresponding to the time of the first and last sample of the Audio Segment, respectively.
Audio-Visual File	An Audio-Visual Object having a File Transport.
Capstan	The capstan is a rotating spindle used to move recording tape through the mechanism of a tape recorder.

Damaged List	A list of strings of Texts corresponding to the Damaged Segments (if any) requiring replacement with synthetic segments.
Damaged Section	An Audio Segment which is damaged in its entirety and is contained in a Damaged Segment.
Damaged Segment	An Audio Segment containing only speech (and not containing music or other sounds) which is either damaged in its entirety or contains one or more Damaged Sections specified in the Damaged List.
Degree	Strength of a feature, specifically, with respect to Emotion, “High,” “Medium,” or “Low.”
Editing List	The description of the speed, equalisation and reading backwards corrections occurred during the restoration process.
Emotion	A Data Type representing the internal status of a human or avatar resulting from their interaction with the context or subsets of it, such as “Angry”, and “Sad”.
Emotionless Speech	An Audio File containing speech without music and other sounds, and in which little or no identifiable emotion is perceptible by native listeners.
Irregularity	An event of interest to preservation in *Table 26* and *Table 27*
Irregularity File	A JSON file containing information about Irregularities of the Audio Recording Preservation inputs.
Irregularity Image	An image corresponding to an Irregularity.
JSON	JavaScript object notation [18].
Microphone Array Geometry	Description of the position of each microphone comprising the microphone array and specific characteristics such as microphone type, look directions, and the array type.
Model Utterance	An Audio Segment used as a model or demonstration of the Emotion to be added to Emotionless Speech in order to produce Speech with Emotion.
Multichannel Audio	A data structure containing at least 2 time-aligned interleaved Audio Channels.
Multichannel Audio Stream	A data structure containing Audio Objects packaged with Audio Scene Geometry.
Neural Network Speech Model	A Neural Network Model trained on Speech Segments for Modelling and used to synthesise replacements for the entire Damaged Segment or Damaged Sections within it.
Passthrough AIM	An AIM with the same input and output data of an AIM without executing the Function of that AIM. E.g., a Noise Cancellation AIM that does not cancel the noise.
Preservation Audio File	The input Audio File resulting from the digitisation of an audio open-reel tape to be preserved and, in case, restored.
Preservation Audio-Visual File	The input Audio-Visual File produced by a camera pointed to the playback head of the magnetic tape recorder and the synchronised Audio resulting from the tape digitisation process.
Preservation Image	A Video frame extracted from Preservation Audio-Visual File.

Preservation Master Files	Set of files providing the information stored in an audio tape recording without any restoration. As soon as the original analogue recordings is no more accessible, it becomes the new item for long-term preservation.
Restored Audio Files	Set of Audio Files derived from the Preservation Audio File, where potential speed, equalisation or reading backwards errors that occurred in the digitisation process have been corrected.
Restored Speech Segment	An Audio Segment in which the entire segment has been replaced by a synthetic speech segment, or in which each Damaged Segment has been replaced by a synthetic speech segment.
Speech Features	Descriptor representing a variety of information elements incorporated in a Speech Segment, e.g., personal identity, Personal Status, additional factors such as vocal tension, creakiness, whispery quality, etc.
Speech Segments for Modelling	A set of Audio Files containing speech segments used to train the Neural Network Speech Model.
Speech With Emotion File	An Audio File containing speech with emotional features.
Spherical Coordinate System	A coordinate system where the position of a point is specified by three numbers: the radial distance of that point from a fixed origin, its polar angle measured from a fixed zenith direction, and the azimuthal angle of its orthogonal projection on a reference plane.
Spherical Grid Resolution	The maximum spherical angle between any two neighbouring sampled points on a sphere.
Text List	List of texts to be converted into speech by the Speech Synthesis for Restoration AIM.
Time Code	Number of ms from 1970-01-01T00:00:00.000 according to [8].
Time Label	A measure of time from a context-dependent zero time expressed as HH:mm:ss.SSS.
Transform Audio	An Audio Object whose data are represented in the Frequency Domain.
Enhanced Transform Audio	Transform Audio whose samples are Enhanced Transform Audio samples.
Useful Signal	Digital signal resulting from the A/D conversion of the analogue signal recorded in an audio tape.

5 References

5.1 Normative References

This standard normatively references the following technical specifications, both from MPAI and other standard organisations:

1. MPAI; Technical Specification: AI Framework (MPAI-AIF) V2.2.
2. MPAI; Technical Specification: Human and Machine Communication (MPAI-HMC) V2.1.
3. MPAI; Technical Specification: Multimodal Conversation (MPAI-MMC) V2.4.
4. MPAI; Technical Specification: Object and Scene Description (MPAI-OSD) V1.4.
5. MPAI; Technical Specification: Portable Avatar Format (MPAI-PAF) V1.5.
6. MPAI; Technical Specification: AI Module Profiles (MPAI-PRF) V1.0.

7. MPAI; Technical Specification: Data Types, Formats, and Attributes (MPAI-TFA) V1.4.
8. IETF; A Universally Unique IDentifier (UUID) URN Namespace; RFC 4122; July 2005.
9. IETF; Date and Time on the Internet: Time Stamps; RFC 3339; July 2002.
10. Universal Coded Character Set (UCS): ISO/IEC 10646; December 2020.
11. ITU-R BS.2088-1 (10/2019) - Long-form file format for the international exchange of audio programme materials with metadata.
12. ISO/IEC 14496-10; Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding.
13. ISO/IEC 23008-2; Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 2: High Efficiency Video Coding.
14. ISO/IEC 23094-1; Information technology – General video coding – Part 1: Essential Video Coding.
15. ISO/IEC 14496-12; Information technology – Coding of audio-visual objects – Part 12: ISO base media file format.
16. ZIP format, https://pkware.cachefly.net/webdocs/casestudies/APPNOTE.TXT.
17. Neural Network Exchange Format; https://www.khronos.org/registry/NNEF/specs/1.0/nnef-1.0.4.pdf; Khronos.
18. Open Neural Network Exchange (ONNX) format; https://www.ONNX.ai.
19. The JavaScript Object Notation (JSON) Data Interchange Format; https://datatracker.ietf.org/doc/html/rfc8259; IETF rfc8259; December 2017.
20. BS EN 60094-1:1994, BS 6288-1: 1994, IEC 94-1:1981 - Magnetic tape sound recording and reproducing systems - Part 1: Specification for general conditions and requirements.
21. K. Bradley, IASA TC-04 Guidelines in the Production and Preservation of Digital Audio Objects: standards, recommended practices, and strategies., 2nd ed. International Association of Sound and Audiovisual Archives, (2009): 2014.
22. ITU-R BS.2088-1: Long-form file format for the international exchange of audio programme materials with metadata.
23. ITU-T T-81: Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines.

5.2 Informative References

The references provided here are for information purpose.

24. MPAI; The MPAI Statutes.
25. MPAI; Patent Policy.
26. MPAI; Technical Specification: Governance of the MPAI Ecosystem (MPAI-GME) V2.0.
27. Framework Licence: Context-based Audio Enhancement Technical Specification (MPAI-CAE)
28. MPAI; Technical Specification: MPAI Metaverse Model (MPAI-MMM) – Technologies V2.0.
29. Ekman, Paul (1999), "Basic Emotions", in Dalgleish, T; Power, M (eds.), Handbook of Cognition and Emotion (PDF), Sussex, UK: John Wiley & Sons.
30. B. Rafaely, Fundamentals of spherical array processing, Springer, 2018.

5.3 Published papers

1. Marina Bosi, Sergio Canazza, Niccolò Pretto, Alessandro Russo, Matteo Spanio; From Tape to Code: An International AI-Based Standard for Audio Cultural Heritage Preservation - *Don't Play That Song for me* (If it's Not Preserved With ARP!).

6 AI Workflows

6.1 Technical Specification

Technical Specification: Context-based Audio Enhancement (MPAI-CAE) – Use Cases (CAE-USC) V2.4 assumes that Workflow implementations will be based on *Technical Specification: AI Framework (MPAI-AIF) V2.2* specifying an AI Framework (AIF) where AI Workflows (AIW) composed of interconnected AI Modules (AIM) are executed.

Table 1 provides the full list of AIWs specified by CAE-USC V2.4 with links to the pages dedicated to AI Workflows. Each of these includes Function; Reference Model; Input/Output Data; Functions of AIMs; Input/Output Data of AIMs; AIW, AIMs, and JSON metadata; Reference Software; Conformance Testing; and Performance Assessment.

All AI-Workflows specified by CAE-USC V2.3 (i.e., the preceding version) are superseded by those specified by CAE-USC V2.4. AI-Workflows specified by CAE-USC V2.3 may still be used if their version is explicitly indicated.

Table 1 - AIWs of CAE-USC V2.4

Acronym	Name	JSON	Acronym	Name	JSON
CAE-ARP	Audio Recording Preservation	X	CAE-EAE	Enhanced Audioconference Experience	X
CAE-EES	Emotion-Enhanced Speech	X	CAE-SRS	Speech Restoration System	X

6.1.1 Audio Recording Preservation

6.1.1.1 Functions

Preservation of audio assets recorded on analogue media is an important activity for a variety of application domains, in particular cultural heritage. Preservation goes beyond mere A/D conversion. For instance, the magnetic tape of an open reel may hold important information: it can carry annotations (by the composer or by the technicians) or it can include multiple splices and/or display several types of Irregularities (e.g., corruptions of the carrier, tape of different colour or chemical composition). This information shall be preserved for a correct playback. Nevertheless, some errors can occur during the digitisation as well as the digitisation could be partial because of the corruption of the carrier. These errors shall be restored to make the content listenable. The ARP Use Case concerns the creation of a digital copy of the digitized audio of open reel magnetic tapes for long-term preservation and of an access copy (restored, if necessary) for correct playback of the digitized recording.

In this Audio Recording Preservation Use Case, two files are fed into a preservation system:

The following is not required:

1. Alignment of the start and end times of the two files. However, the maximum tolerated misalignment is 10s.
2. Presence of signal at the start and the end of the two files.
3. Alignment of the Useful Signal on both files.
4. The same time base for both files. However, the time difference between the same samples in two files shall not be more than 30ms for a 1-hour audio tape.

The output of the restoration process is composed by:

1. Preservation Master Files.
2. Access Copy Files.

6.1.1.2 *Reference Model and its operation*

Figure 1 depicts the Audio Recording Preservation Reference Model.

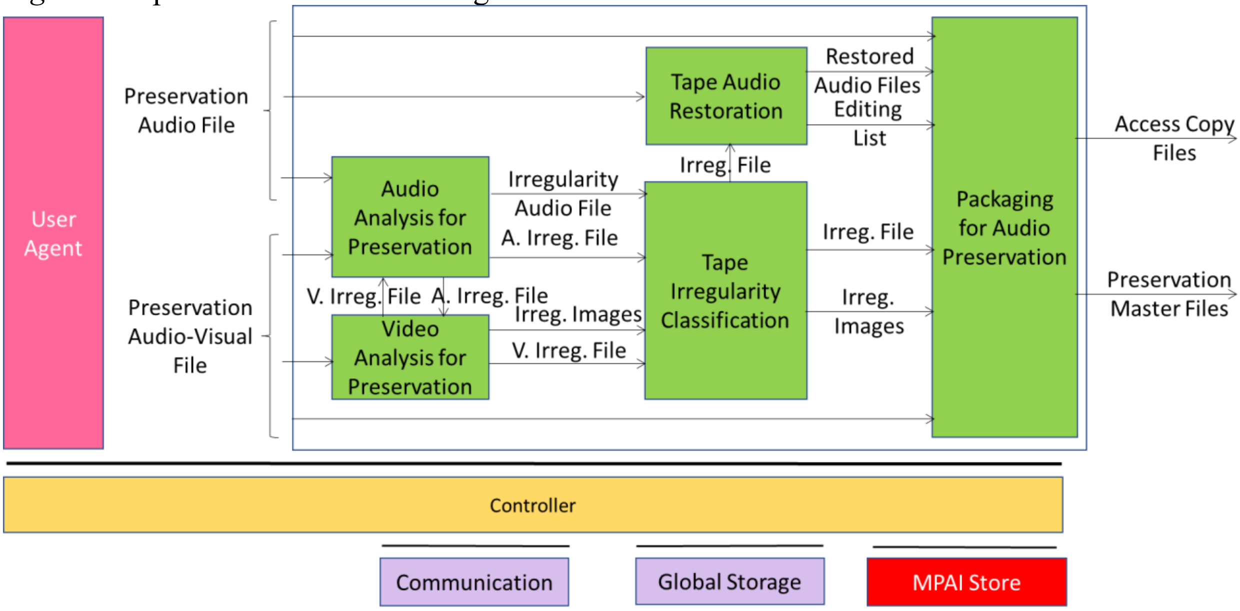


Figure 1 – Audio Recording Preservation Reference Model

The sequence of operations of the Audio Recording Preservation unfolds as follows:

1. The analogue audio signal from the open-reel tape recorder is digitised as Preservation Audio File.
2. Preservation Audio-Visual File is the combination of:
    1. The video camera pointed at the playback head of the open-reel tape recorder.
    2. The analogue audio signal digitised with the same video clock.
3. Audio Analysis for Preservation:
    1. Detects Irregularities.
    2. Assigns IDs to them that are unique to the analysed open-reel tape.
    3. Receives an Irregularity File from the Video Analysis for Recording
    4. Extracts the Audio Files corresponding to each Irregularity received or detected.
    5. Sends the Audio Files and the Irregularity File related to all Irregularities to the Tape Irregularity Classification.
4. Video Analysis for Preservation:
    1. Detects Irregularities.
    2. Assigns IDs to them that are unique to the analysed open-reel tape.
    3. Receives an Irregularity File from the Audio Analysis for Recording and the offset between Preservation Audio File and the Preservation Audio-Visual File.
    4. Extracts the Irregularity Images corresponding to each Irregularity received or detected.
    5. Sends the Irregularity Images and the Irregularity File related to all Irregularities to the Tape Irregularity Classification.
5. Tape Irregularity Classification:
    1. Receives an Irregularity File with the corresponding Images and Audio Files.
    2. Classifies and selects the ones considered relevant.
    3. If the Irregularity was detected by the Video Analysis for Recording, the selected Irregularity File and the corresponding Irregularity Images are sent to the Packaging for Audio Recording.
6. The Tape Audio Restoration uses the Irregularity File to identify and restore portions of the Preservation Audio File.

7. The Packaging for Audio Preservation collects the Preservation Audio File, Restored Audio Files, the Editing List, the Irregularity File and corresponding Irregularity Images if detected by the Video Analyser, and the Preservation Audio-Visual File and then it produces the Preservation Master Files and Access Copy Files.

6.1.1.3 I/O data of AI Workflow

Table 1 gives the input and output data of Audio Recording Preservation.

Table 1 – I/O data of Audio Recording Preservation

Input	Comments
Preservation Audio File	A Preservation Audio File obtained by digitising the analogue tape audio recording composed of music, soundscape or speech read from a magnetic tape.
Preservation Audio-Visual File	A Preservation Audio-Visual File produced by a camera pointed to the playback head of the magnetic tape recorder.
Output data	**Comments**
Preservation Master Files	Set of files providing the information stored in an audio tape recording without any restoration. As soon as the original analogue recordings is no more accessible, it becomes the new item for long-term preservation.
Access Copy Files	Set of Audio Files derived from the Preservation Audio File, where potential speed, equalisation or reading backwards errors that occurred in the digitisation process have been corrected.

6.1.1.4 Functions of AI Modules

The AIMs required by this Use Case are specified in *Table 2*.

Table 2 – Functions of AI Modules of Audio Recording Preservation

AIM	Function
Audio Analysis for Preservation	1. At the start, it calculates the offset between Preservation Audio and the Audio of the Preservation Audio-Visual File. 2. Sends Audio Irregularity File to and receives Video Irregularity Files from Video Analysis for Preservation. 3. Extracts the Audio Files corresponding to the Irregularities identified in both Irregularity Files. 4. Sends the Irregularity merged from the Audio and Video Irregularity Files to Tape Irregularity Classification with the corresponding Audio Files.
Video Analysis for Preservation	1. Detects and enters the Video Irregularities of the Preservation Audio-Visual File in the Video Irregularity File. 2. Sends Video Irregularity File to and receives Audio Irregularity Files from Audio Analysis for Preservation. 3. Extracts the Images corresponding to the Irregularities of both Irregularity Files. 4. Sends the Irregularity merged from the Audio and Video Irregularity Files to Tape Irregularity Classification with the corresponding Video Files.

Tape Irregularity Classification	1. Receives Irregularity File (Audio) and Audio Files from Audio Analysis for Preservation. 2. Receives Irregularity File (Video) and Irregularity Images from Video Analysis for Preservation. 3. Classifies and selects the relevant Irregularities of the Preservation Audio-Visual File and Preservation Audio File. 4. Sends the Irregularity File related to the selected Irregularities to Tape Audio Restoration. 5. Sends the Irregularity Files related to the selected Irregularities and the corresponding Irregularity Images to Packaging for Audio Recording.
Tape Audio Restoration	1. Detects and corrects speed, equalisation and reading backwards errors in Preservation Audio File. 2. Sends Restored Audio Files and Editing List to Packaging for Audio Preservation
Packaging for Audio Preservation	Produces Preservation Master Files and Access Copy Files.

6.1.1.5 I/O Data of AI Modules

Table 3 – CAE-ARP AIMs and their data

AIM	Input Data	Output Data
Audio Analysis for Preservation	Preservation Audio File Preservation Audio-Visual File Irregularity File	Audio Files Audio Irregularity File
Video Analysis for Preservation	Preservation Audio-Visual File Audio Irregularity File	Video Irregularity File Irregularity Images
Tape Irregularity Classification	Irregularity Audio Files Audio Irregularity File Irregularity Images Video Irregularity File	Irregularity File Irregularity Images
Tape Audio Restoration	Irregularity File Preservation Audio File	Editing List Restored Audio Files
Packaging for Audio Preservation	Preservation Audio File Restored Audio Files Editing List Irregularity File Irregularity Images Preservation Audio-Visual File	Access Copy Files Preservation Master Files

6.1.1.6 AIW, AIMs, and JSON Metadata

Table 4 - Acronyms and URs of JSON Metadata

AIW	AIMs	Name	JSON
CAE-ARP		Audio Recording Preservation	File

	CAE-AAP	Audio Analysis for Preservation	File
	CAE-VAP	Video Analysis for Preservation	File
	CAE-TIC	Tape Irregularity Classification	File
	CAE-TAR	Tape Audio Restoration	File
	CAE-PAP	Packaging for Audio Preservation	File

6.1.1.7 Reference Software

The CAE-ARP Reference Software can be downloaded from MPAI Git.

6.1.1.8 Conformance Testing

Receives	Preservation Audio File	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Preservation Audio-Visual File	Shall validate against the Audio-Visual Object schema. The Qualifier shall validate against the Audio-Visual Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio-Visual Object Qualifier schema.
Produces	Preservation Master Files	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Access Copy Files	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

6.1.2 Emotion-Enhanced Speech

6.1.2.1 Functions

Speech carries information not only about its lexical content, but also about several other aspects including age, gender, identity, and **emotional state of the speaker**. Speech synthesis is evolving towards support of these aspects. In many use cases, emotional force can usefully be added to speech which by default would be neutral or emotionless, possibly with grades of a particular emotion. For instance, in a human-machine dialogue, messages conveyed by the machine can be more effective if they carry emotions appropriately related to the emotions detected in the human speaker.

Emotion-Enhanced Speech (EES):

1. Enables a user to indicate a model utterance or an Emotion to obtain an emotionally charged version of a given utterance.
2. Converts an individual emotionless speech segment to a segment that has a specified emotion. Both input and output speech segments are contained in files. The desired emotion is expressed either as a tag belonging to a standard list of emotions or derived by

extracting features from a model utterance. EES produces an output speech segment with emotion.

CAE-EES implementations can be used to create virtual agents communicating as naturally as possible, and thus improve the quality of human-machine interaction by bringing it closer to human-human interchange.

6.1.2.2 *Reference Model*

The Emotion-Enhanced Speech Reference Model depicted in Figure 1 supports two Modes or pathways enabling addition of emotional charge to an emotionless or neutral input utterance (Emotion-less Speech).

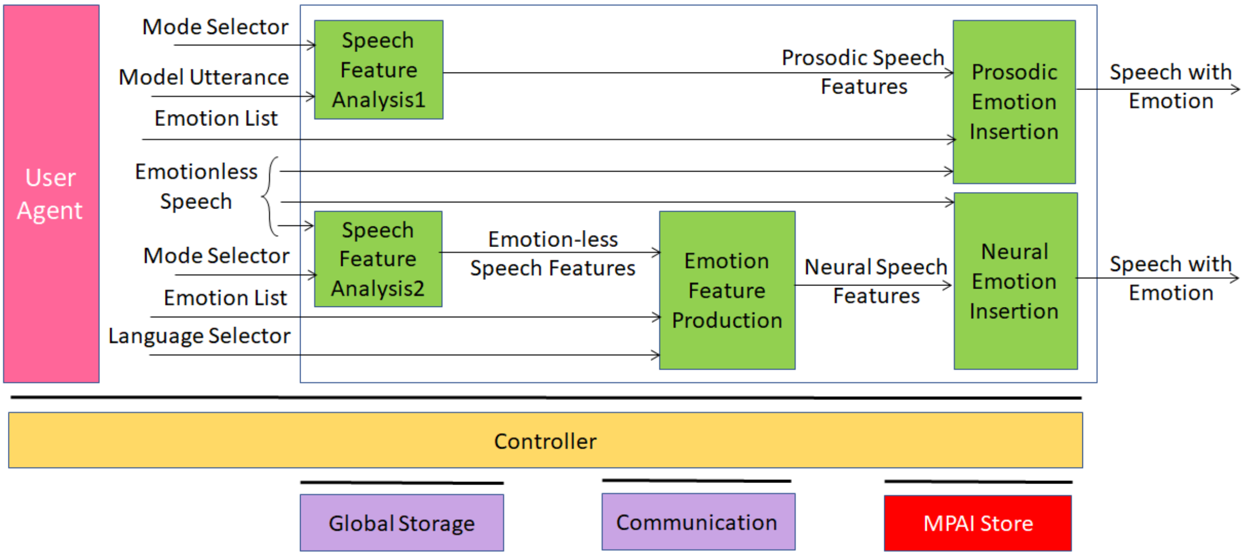


Figure 1 - Emotion-Enhanced Speech Reference Model

6.1.2.3 *I/O data of AI Workflow*

Table 1 gives the input and output data of Emotion-Enhanced Speech.

Table 1 – I/O data of Emotion-Enhanced Speech

Input data	**Description**
Mode Selector	Selects pathway 1 operation.
Model Utterance	An Audio Segment used as a model or demonstration of the Emotion to be added to Emotionless Speech in order to produce Speech with Emotion.
Emotion List	A set of Emotion values.
Emotionless Speech	A File containing Speech without music and other sounds, and in which little or no identifiable emotion is perceptible by native listeners.
Mode Selector	Selects pathway 2 operation.
Emotion List	A set of Emotion values.
Language Selector	Selects the language of the Enhanced Speech.
Output data	**Description**

Speech with Emotion	A File containing Speech with emotional features.

6.1.2.4 Functions of AI Modules

The AI Modules perform the functions described in *Table 2*.

Table 2 – AI Modules of Emotion-Enhanced Speech

AIM	Function
Speech Feature Analysis 1	Extracts Neural Speech Features of a model emotional utterance and transfers them to the Prosodic Emotion Insertion AIM.
Speech Feature Analysis 2	Extracts Emotionless Speech Features of an emotionless input utterance, passing these to the Emotion Feature Production AIM.
Emotion Feature Production	Receives the Emotionless Speech Features produced by Speech Feature Analysis2 plus a list of Emotions to be added. (If the Degree of an Emotion is not specified, the Medium value is used.)
Prosodic Emotion Insertion	Integrates the (emotional) Prosodic Speech Features with those of the Emotionless Speech input, yielding and delivering an emotionally modified utterance.
Neural Emotion Insertion	Integrates the (emotional) Neural Speech Features with those of the Emotionless Speech input, yielding and delivering an emotionally modified utterance.

6.1.2.5 I/O Data of AI Modules

Table 3 – CAE-EES AIMs and their data

AIM	Input Data	Output Data
Speech Feature Analysis 1	Mode Selector Model Utterance	Prosodic Speech Features
Speech Feature Analysis 2	Mode Selector Emotionless Speech	Emotionless Speech Features
Emotion Feature Production	Emotionless Speech Features Language Selector Emotion List	Neural Speech Features
Prosodic Emotion Insertion	Emotionless Speech Prosodic Speech Features	Speech with Emotion
Neural Emotion Insertion	Emotionless Speech Neural Speech Features	Speech with Emotion

6.1.2.6 AIW, AIMs, and JSON Metadata

Table 4 – AIW, AIMs, and JSON Metadata

AIW	AIMs	Name	JSON

CAE-EES		Emotion Enhanced Speech	File
	CAE-SF1	Speech Feature Analysis 1	File
	CAE-SF2	Speech Feature Analysis 2	File
	CAE-EFP	Emotion Feature Production	File
	CAE-PEI	Prosodic Emotion Insertion	File
	CAE-NEI	Neural Emotion Insertion	File

6.1.2.7 *Conformance Testing*

Table 5 provides the Conformance Testing Method for CAE-EES AIW. Conformance Testing of the individual AIMs are given by the individual AIM Specification.

Table 5 – Conformance Testing Method for OSD-EES AIW

Receives	Mode Selector	Shall validate against the Selection Schema.
	Model Utterance	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.
	Emotion List	Shall validate against the Emotion Schema.
	Emotionless Speech	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.
	Language Selector	Shall validate against the Selection Schema.
Produces	Speech with Emotion	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.

6.1.3 Enhanced Audioconference Experience

6.1.3.1 *Functions*

Enhanced Audioconference Experience addresses the situation where one or more speakers are active in a noisy meeting room and are trying to communicate with one or more interlocutors using speech over a network. In this situation, the user experience is very often far from satisfactory due to multiple competing speakers, non-ideal acoustical properties of the physical spaces that the speakers occupy and/or the background noise. These can lead to a reduction in speech intelligibility resulting in participants not fully understanding what their interlocutors are saying, a situation of distraction that may possibly lead to "o-called" *audioconference fatigue*. When microphone arrays are used to capture the speakers, most of the described problems can be resolved by appropriate processing of the captured signals. The speech signals from multiple speakers can be separated from each other, the non-ideal acoustics of the space can be improved, and background noise can be substantially suppressed.

CAE-EAE is concerned with extracting:

1. The individual speakers' speech signals from the microphone array as well as reducing the background noise and the reverberation that reduce speech intelligibility.

2. The speakers' Spatial Attitudes with respect to the position of the microphone array to facilitate the spatial representation of the speech signals at the receiver side if necessary.

Spatial Attitudes are represented in the Audio Scene Geometry format and further processes packaged for efficient delivery. Possible compression of the extracted speech signals as well as their reconstruction/representation at the receiver side are specified by the Qualifiers of the Speech Objects.

The Enhanced Audio Experience AIW:

Receives	Audio Object	Composed of Audio Data and Qualifier. The latter describes the format of the Audio Data and Attributes such as Microphone Array Geometry and capturing device geometry.
	Audio Source Model	A polynomial description of a simple acoustic source parameterised in terms of its direction with respect to the capture point.
Produces	Audio Scene Descriptors	The description of the Audio Scene including individual Audio Objects and their Spatial Attitudes. Audio Objects include Data and Qualifiers.

6.1.3.2 *Reference Model*

Figure 1 depicts the Reference Model for the CAE-EAE.

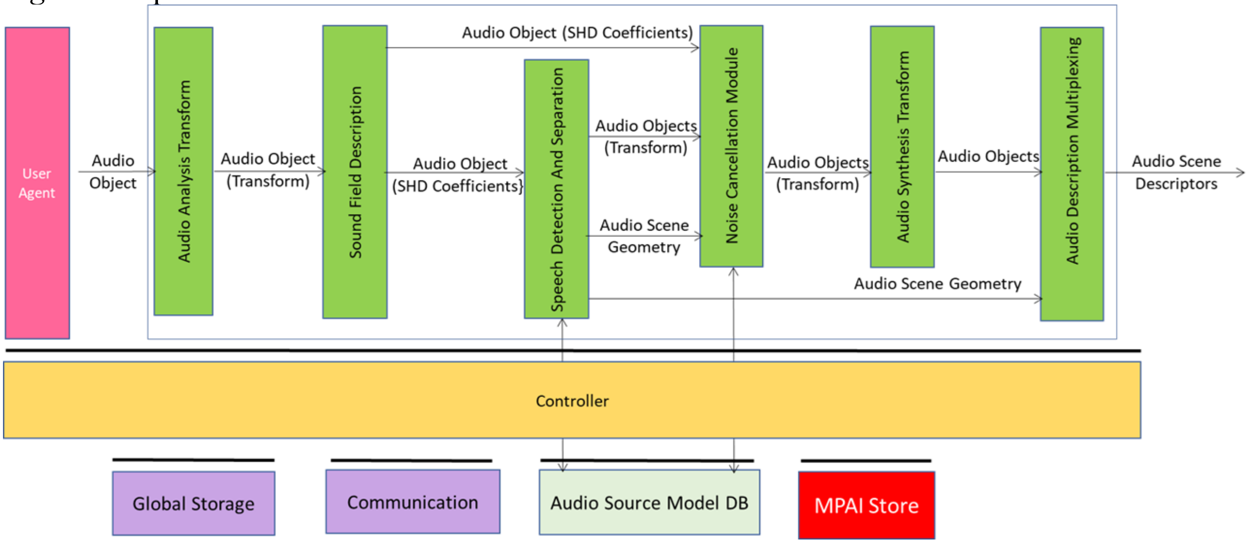


Figure 1 - Enhanced Audioconference Experience Reference Model

EAE receives An Audio Object which includes, Audio Data, Microphone Array Geometry and Device Geometry. Using this information, the system can detect the relative directions of the active speakers according to the microphone array and separate relevant audioconference speech sources from each other and from other spurious sounds. Since audio conferencing is a real-time application scenario, the use case operates on Audio Blocks.

The sequence of operations of the EAE use case is the following:

1. **Audio Analysis Transform** transforms the Multichannel Audio into frequency bands via a Fast Fourier Transform (FFT). The following operations are carried out in discrete frequency bands. When such a configuration is used a 50% overlap between subsequent audio blocks needs to be employed. The output is a data structure comprising complex valued audio samples in the frequency domain.
2. **Sound Field Description** converts the output from the Analysis Transform AIM into the spherical frequency domain [20]. If the microphone array used in capturing the scene is a spherical microphone array, Spherical Fourier Transform (SFT) can be used to obtain the Spherical Harmonic Decomposition (SHD) coefficients that represent the captured sound

field in the spatial frequency domain. For other types of arrays, more elaborate processing might be necessary. The output of this AIM is (M × (N+1)2) complex valued data frame comprising the SHD coefficients up to an order which depends on the number of individual microphones in the array.

3. **Speech Detection and Separation** receives the SHD coefficients of the sound field to detect directions of active sound sources and to separate them. Each separated source can either be a speech or a non-speech signal. Speech detection is carried out on an Audio Block basis by using on each separated source an appropriate voice activity detector (VAD) that is a part of this AIM. This AIM will output speech as an (M × S) Audio Block comprising transform domain speech signals and block-by-block the Audio Scene Geometry in JSON format comprising auxiliary information which contains a (M × 1) binary mask indicating the channels of the transform domain SHD coefficients that would be used by the Noise Cancellation AIM for denoising. Speech Detection and Separation AIM uses the Source Model KB which contains discrete-time and discrete-valued simple acoustic source models that are used in source separation.
4. **Noise Cancellation Module** eliminates background noise and reverberation which reduce the audio quality. If environmental conditions do not substantially add ambient noise to the desired speech, this AIM acts as a Passthrough AIM.
    1. It receives Transform Speech from Speech Detection and Separation AIM and Acoustic Scene Metadata which includes attributes pertaining to the Audio Block being processed for denoising, and SHD coefficients.
    2. It uses Source Model KB. The output of Noise Cancellation AIM is Denoised Transform Speech as an (M × S) complex-valued data structure which will in the next stage be processed through Synthesis Transform AIM to obtain Denoised Speech.
5. **Audio Synthesis Transform** receives Denoised Transform Speech and outputs Denoised Transform Speech (F × S) by applying the inverse of the analysis transform.
6. **Audio Descriptors Multiplexing**: produced the output Audio Scene Descriptors from Audio Objects and the Audio Scene Geometry.

6.1.3.3 I/O data of AI Workflow

Table 1 shows the input and output data for the Enhanced Audioconference Experience workflow.

Table 1 – I/O data of Enhanced Audioconference Experience

Inputs	Comments
Audio Object	Audio Data and optional metadata regarding Sub-Types, Formats and Attributes of the Audio Data.
Audio Source Model	A polynomial description of a simple acoustic source parameterised in terms of its direction with respect to the capture point.
Outputs	**Comments**
Audio Scene Descriptors	A Data Type including the Audio Objects of a scene, their sub-scenes, and their arrangement in the scene.

6.1.3.4 Functions of AI Modules

Table 2 gives the AIMs used by the Enhanced Audioconference Experience AIW.

Table 2 - AIMs of Enhanced Audioconference Experience

AIM	Function
Audio Analysis Transform	Represents the input Multichannel Audio in a new form amenable to further processing by the subsequent AIMs in the architecture.
Sound Field Description	Produces Spherical Harmonic Decomposition Coefficients of the Transformed Multichannel Audio.
Speech Detection and Separation	Separates speech and non-speech signals in the Spherical Harmonic Decomposition producing Transform Speech and Audio Scene Geometry.
Noise Cancellation Module	Removes noise and/or suppresses reverberation in the Transform Speech producing Enhanced Transform Audio.
Audio Synthesis Transform	Effects inverse transform of Enhanced Transform Audio producing Enhanced Audio Objects ready for packaging.
Audio Descriptors Multiplexing	Multiplexes Enhanced Audio Objects and the Audio Scene Geometry.

6.1.3.5 I/O Data of AI Modules

Table 3 specifies the I/O Data of CAE-EAE.

Table 3 – CAE-EAE AIMs and their data

AIM	Input Data	Output Data
Audio Analysis Transform	Audio Object	Audio Object (Transform)
Sound Field Description	Audio Object (Transform)	Audio Object (SHD Coefficients)
Speech Detection and Separation	Audio Object (SHD Coefficients) Audio Source Model	Audio Objects (Transform) Audio Scene Geometry
Noise Cancellation Module	Audio Object (SHD Coefficients) Audio Object (Transform) Audio Scene Geometry Audio Source Model	Enhanced Audio Objects (Transform)
Audio Synthesis Transform	Enhanced Audio Objects (Transform)	Enhanced Audio Objects
Audio Description Packaging	Enhanced Audio Objects Audio Scene Geometry	Audio Scene Descriptors

6.1.3.6 AIW, AIMs, and JSON Metadata

Table 4 provides links to the AI Modules and JSON Metadata.

Table 4 – AIW, AIMs, and JSON Metadata

AIW	AIMs	Names	JSON
CAE-EAE		Enhanced Audioconference Experience	File

	CAE-AAT	Audio Analysis Transform	File
	CAE-SFD	Sound Field Description	File
	CAE-SDS	Speech Detection and Separation	File
	CAE-NCM	Noise Cancellation Module	File
	CAE-AST	Audio Synthesis Transform	File
	CAE-ADP	Audio Descriptors Multiplexing	File

6.1.3.7 Reference Software

The CAE-EAE Reference Software can be downloaded from the MPAI Git.

6.1.3.8 Conformance Testing

Table 2 provides the Conformance Testing Method for CAE-EAE AIW. Conformance Testing of the individual AIMs are given by the individual AIM Specification.

Table 2 – Conformance Testing Method for OSD-EAE AIW

Receives	Audio Object	Shall validate against the Audio Object Schema. The Qualifier shall validate against the Audio Qualifier schema.
	Audio Source Model	Shall validate against the Audio Source Model Schema.
Produces	Audio Scene Descriptors	Shall validate against the Audio Scene Descriptors Schema. The Audio Objects shall validate against the Audio Object Schema. The Spatial Attitudes shall validate against the Spatial Attitudes Schemas. The Qualifier shall validate against the Audio Qualifier schema.

6.1.4 Speech Restoration System

6.1.4.1 Functions

Speech Restoration System restores a Damaged Segment of an Audio Segment containing only speech from a single speaker. The damage may affect the entire segment, or only part of it.

Restoration will not involve filtering or signal processing. Instead, *replacements* for the damaged vocal elements will be *synthesised* using a speech model. The latter is a component or set of components, normally including one or more neural networks, which accepts text and possibly other specifications, and delivers audible speech in a specified format – here, the speech of the required replacement or replacements. If the damage affects the entire segment, an entirely new segment is synthesized; if only parts are affected, corresponding segments will be synthesized individually to enable later integration into the undamaged parts of the Damaged Segment, with reference to appropriate Time Labels.

The speech segments necessary for the creation of the speech model can be flexibly resourced from undamaged parts of the input segment or from other recording sources that are consistent with the original segment's acoustic environment.

Restoration is carried out by synthesising replacements for the damaged vocal elements as follows:

The Speech Segments for Modelling – Audio Segments necessary for the creation of the Neural Network Speech Model – may be obtained from any undamaged parts of the input speech segment; however, other Audio Segments consistent with the original segment's sound environment can also be used.

6.1.4.2 Reference Model

Figure 1 gives the Reference Model of the Speech Restoration System.

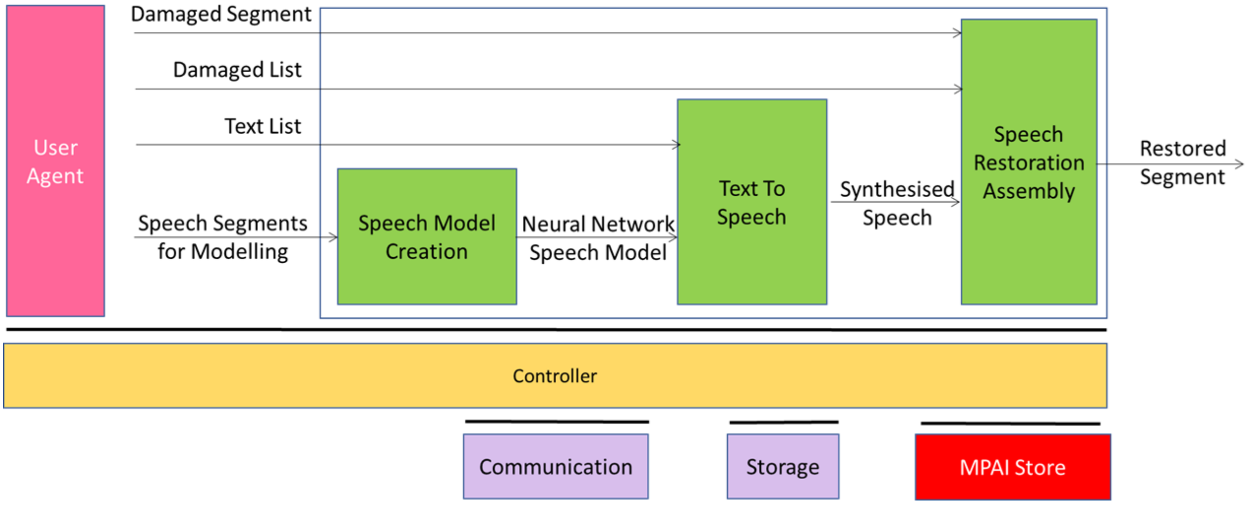


Figure 1 - Speech Restoration System (SRS) Reference Model

In the SRS use case, the entire Damaged Segment can be replaced by a synthesised segment, or parts within it can be synthesized to enable integration of the replaced segments.

6.1.4.3 I/O Data of AI Workflow

Table 1 gives the input and output data of Speech Restoration System.

Table 1 – I/O data of Audio Recording Preservation

Input Data	**Definition**
Speech Segments for Modelling	A set of Audio Files containing speech segments used to train the Neural Network Speech Model.
Text List	List of texts to be converted into speech by the Speech Synthesis for Restoration AIM.
Damaged List	A list of strings of Texts corresponding to the Damaged Segments (if any) requiring replacement with synthetic segments.
Damaged Segment	An Audio Segment containing only speech (and not containing music or other sounds) which is either damaged in its entirety or contains one or more Damaged Sections specified in the Damaged List.
Output Data	**Definition**

Restored Speech Segment	An Audio Segment in which the entire segment has been replaced by a synthetic speech segment, or in which each Damaged Segment has been replaced by a synthetic speech segment.

6.1.4.4 Functions of AI Modules

The AIMs required by the Speech Restoration System Use Case are described in *Table 2*.

Table 2 - AI Modules of Speech Recording System

AIM	Function
Speech Model Creation	1. Receives in separate files the Audio Segments for Modelling, adequate for model creation. 2. Creates the current Neural Network Speech Model. 3. Sends that Neural Network Speech Model to the Speech Synthesis for Restoration.
Text To Speech	1. Receives 1.1. The current Neural Network Speech Model. 1.2. Damaged List as a data structure: 1.2.1. Containing one element if Damaged Segment is damaged throughout or 1.2.2. Representing a list in which each element specifies via Time Labels the start and end of a damaged section within Damaged Segment. 2. Synthesizes each Damaged Section in Damaged List. 3. Sends the newly synthesised segments to the Speech Restoration Assembly as an ordered list.
Speech Restoration Assembly	1. Receives the Damaged Segment. 2. Receives the ordered list of synthetic segments. 3. Receives Damaged List Time Labels, indicating where the synthesized segments should be inserted in left-to-right order. In case Damaged Segment as a whole was damaged, the list contains one entry. 4. Assembles the final version of the Restored Segment.

6.1.4.5 I/O Data of AI Modules

Table 3 – CAE-SRS AIMs and their I/O Data

AIM	Input Data	Output Data
Speech Model Creation	Speech Segments for Modelling	Neural Network Speech Model
Text To Speech	Text List Neural Network Speech Model	Synthesised Speech
Speech Restoration Assembly	Damaged Segments Damaged List	Restored Segment

6.1.4.6 AIW, AIMs and JSON Metadata

Table 4 – AIMs and JSON Metadata

AIW	AIMs	Names	JSON
CAE-SRS		Speech Restoration System	File
	CAE-SMC	Speech Model Creation	File
	MMC-TTS	Text To Speech	File
	CAE-SRA	Speech Restoration Assembly	File

6.1.4.7 Conformance Testing

Table 5 provides the Conformance Testing Method for CAE-SRS AIW. Conformance Testing of the individual AIMs are given by the individual AIM Specification.

Table 5 – Conformance Testing Method for CAE-SRS AIW

Receives	Speech Segments for Modelling	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.
	Text List	Shall validate against the TextObject Schema. The Qualifier shall validate against the TextQualifier schema.
	Damaged List	Shall validate against the Damaged List Schema. Shall validate against the Time Schema. The Time Qualifier shall validate against the Time Qualifier schema.
	Damaged Segment	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.
Produces	Restored Speech Segment	Shall validate against the Speech Object Schema. The Qualifier shall validate against the Speech Qualifier schema.

6.2 Reference Software

As a rule, MPAI provides Reference Software implementing the AIWs released with the following disclaimers:

1. The CAE-USC V2.4 Reference Software Implementation, if in source code, is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to provide a working Implementation of CAE-USC V2.4, 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 copyright holders. Users shall verify that they have the right to use any third-party software required by this Reference Software.

Note that at this stage the CAE-USC V2.4 specifies Reference Software only for some AIWs.

6.3 Conformance Testing

An implementation of an AIW conforms with CAE-USC V2.4 if it accepts as input _and_ produces as output Data and/or Data Objects (the combination of Data of a Data Type and its Qualifier) conforming with those specified by ACAE-USC V2.4.
The Conformance is expressed by one of the two statements

1. "Data conforms with the relevant (Non-MPAI) standard" – for Data.
2. "Data validates against the Data Type Schema" – for Data Object.

The latter statement implies that:

1. Any Sub-Type of the Data conforms with the relevant Sub-Type specification of the applicable Qualifier.
2. Any Content and Transport Format of the Data conform with the relevant Format specification of the applicable Qualifier.
3. Any Attribute of the Data
    1. Conforms with the relevant (Non-MPAI) standard – for Data, or
    2. Validates against the Data Type Schema – for Data Object.

The method to Test the Conformance of an instance of Data or Data Object is specified in the Data Types chapter.
Note that at this stage the CAE-USC V2.4 specifies Conformance Testing only for some AIWs.

6.4 Performance Assessment

Performance is an umbrella term used to describe a variety of attributes – some specific of the application domain the Implementation intends to address. Therefore, Performance Assessment Specifications provide methods and procedures to measure how well an AIW performs its function. Performance of an Implementation includes methods and procedures for all or a subset of the following characteristics:

1. Quality– for example, how satisfactory are the responses provided by an Answer to Multimodal Question
2. Robustness– for example, how robust is the operation of an implementation with respect to duration of operation, load scaling, etc.
3. Extensibility– for example, the degree of confidence a user can have in an Implementation when it deals with data outside of its stated application scope.
4. Bias – for example, how dependent on specific features of the training data is the inference, as in Company Performance Prediction when the accuracy of the prediction may widely change based on the size or the geographic position of a Company.
5. Security – for example, the machine driven by an AI System does not create risks for its users, e.g., physical and cyber.
6. Legality– for example, an AIW instance complies with a regulation, e.g., the European AI Act.

Note that at this stage the CAE-USC V2.4 specifies Performance Assessment only for some AIWs.

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: Context-based Audio Enhancement (MPAI-CAE) - Use Cases (CAE-***

USC) V2.4. The AI Modules specified by CAE-USC V2.4 supersede those specified by previous versions. These may be used if their Version is explicitly signaled. AIMs in bold are Composite.

Table 1 - Specifications and JSON syntax of AIMs used by CAE-USC V2.4

Acronym	AIM Name	JSON	Acronym	AIM Name	JSON
CAE-AAP	Audio Analysis for Preservation	File	CAE-PAP	Packaging for Audio Preservation	File
CAE-AAT	Audio Analysis Transform	File	CAE-PEI	Prosodic Emotion Insertion	File
CAE-AMX	Audio Descriptors Multiplexing	File	CAE-SFD	Sound Field Description	File
CAE-AOI	Audio Object Identification	File	CAE-SDS	Speech Detection and Separation	File
CAE-ASD	Audio Scene Description	File	CAE-SF1	Speech Feature Analysis 1	File
CAE-ASE	Audio Separation and Enhancement	File	CAE-SF2	Speech Feature Analysis 2	File
CAE-ASL	Audio Source Localisation	File	CAE-SMC	Speech Model Creation	File
CAE-AST	Audio Synthesis Transform	File	CAE-SRA	Speech Restoration Assembly	File
CAE-EFP	Emotion Feature Production	File	CAE-TAR	Tape Audio Restoration	File
CAE-NEI	Neural Emotion Insertion	File	CAE-TIC	Tape Irregularity Classification	File
CAE-NCM	Noise Cancellation Module	File	CAE-VAP	Video Analysis for Preservation	File

7.1.1 Audio Analysis for Preservation

7.1.1.1 Function

1. At the start, it calculates the offset between Preservation Audio and the Audio of the Preservation Audio-Visual File.
2. Sends Audio Irregularity File to and receives Video Irregularity Files from Video Analysis for Preservation.
3. Extracts the Audio Files corresponding to the Irregularities identified in both Irregularity Files.
4. Sends the Irregularity merged from the Audio and Video Irregularity Files to Tape Irregularity Classification with the corresponding Audio Files.

7.1.1.2 Reference Model

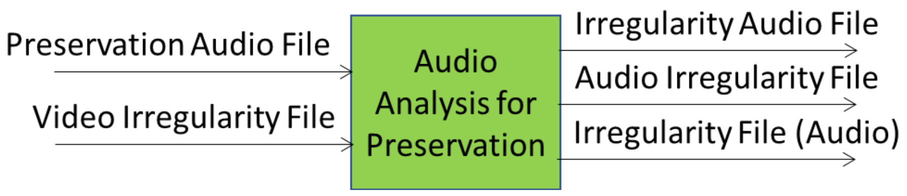


Figure 1 - Audio Analysis for Preservation AIM

7.1.1.3 Input/Output Data

Input data	**Semantics**
Preservation Audio File	The input Audio File resulting from the digitisation of an audio open-reel tape to be preserved and, in case, restored.
Preservation Audio-Visual File	The input Audio-Visual File resulting from the digitisation of an audio open-reel tape to be preserved and of the output of the video camera pointed at the reading head of the audio playback.
Video Irregularity File	A JSON file containing information about the Irregularities of the Preservation Audio-Visual File received from Video Analysis for Recording.
Output data	**Semantics**
Audio File	Audio Segments corresponding to Irregularities of the Preservation Audio File.
Audio Irregularity File	A JSON file containing information about Irregularities of the Preservation Audio File sent to Video Analysis for Recording.

7.1.1.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioAnalysisForPreservation.json

7.1.1.5 Reference Software

The CAE-AAP Reference Software can be downloaded from the MPAI Git.

7.1.1.6 Conformance Testing

Receives	Preservation Audio File	Shall be conforming Audio Object.
	Preservation Audio-Visual File	Shall be conforming Audio-Visual Object schema.
	Video Irregularity File	Shall be conforming Irregularity File schema.
Produces	Audio File	Shall be conforming Audio Object.
	Audio Irregularity File	Shall be conforming Irregularity File schema.

7.1.1.7 Performance Assessment

Table 23 gives the Audio Recording Preservation (ARP) Audio Analysis for Preservation Means and how they are used.

Table 23 – AIM Means and use of Audio Recording Preservation (ARP) Audio Analysis for Preservation .

Means	**Actions**

Performance Assessment Dataset	DS1: *n** Preservation Audio Files. DS2: *n* Preservation Audio-Visual Files related to DS1. DS3: *n* Irregularity Files related to DS2. DS4: *n* output Irregularity Files in the format of port IrregularityFileOutput_1 with all Irregularities correctly identified. DS5: *n* output Irregularity Files in the format of port IrregularityFileOutput_2 with the real offset and all Irregularities correctly identified and included from DS3. * A reasonable n for Assessment is 5<n<=10, since each file generates multiple irregularities to classify
Procedure	1. Feed Audio Analyser under Assessment with DS1, DS2 and DS3. 2. Compare the computed offsets with the ones contained in DS5. 3. Analyse the Irregularity Files resulting from port IrregularityFileOutput_1. 4. Analyse the Irregularity Files resulting from port IrregularityFileOutput_2.
Evaluation	1. Verify the conditions: a. The Irregularity Files are syntactically correct and performing to the JSON schema provided in CAE Technical Specification. b. All Irregularities from DS3 are included in the Irregularity Files coming from port IrregularityFileOutput_2. c. , where is the offset computed by the Audio Analyser under Assessment, is the real offset and FPS_{DS3} is the number of frames per second at which the DS3 video has been recorded. d. All output Audio Files are performing to RF64 file format [7]. e. For each of the *n* tuples of input records, the output Audio Files are extracted from the corresponding input Preservation Audio File at the Time Labels indicated in the Irregularity File coming from port IrregularityFileOutput_2. 2. By inspecting the Irregularity Files resulting from port IrregularityFileOutput_1, for each of the *n* tuples of input records, compute the values of Recall (*R*) and Precision (*P*). 3. Compute the average value of Recall ( ) and Precision ( ) measures obtained at point 2. 4. Accept the AIM under Assessment if: a. R'>0.9 b. P'> 0.9

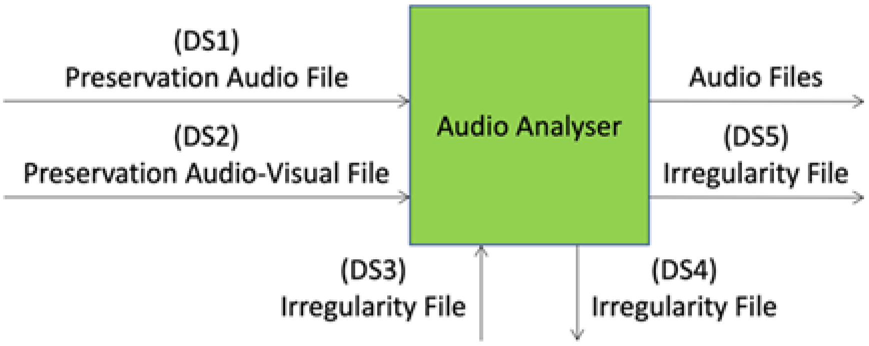


Figure 10 – Audio Analysis for Preservation.

After the Assessment, Performance Assessor shall fill out Table 24.

Table 24 – Performance Assessment form of Audio Recording Preservation (ARP) Audio Analysis for Preservation.

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form "CAE:ARP:1:0".
Name of AIM	Audio Analyser
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Assessment Dataset	Unique Dataset Identifier assigned by MPAI Store.
Assessment ID	Unique Assessment Identifier assigned by Performance Assessor.
Actual output	Actual output provided as a matrix of *n* rows containing and values. Tuple # 1 Measure 1 Measure 1 … … … *n* Measure *n* Measure *n*
Execution time*	Duration of Assessment execution.
Assessment comment*	-
Assessment Date	yyyy/mm/dd.

* *Optional field*

7.1.2 Audio Analysis Transform

7.1.2.1 Function

Audio Analysis Transform (CAE-AAT) receives n Audio Object (Multichannel Audio), into frequency bands via a Fast Fourier Transform (FFT), and produces an Audio Object In the Transform domain.

Receives	Audio Objec*t*	As Multichannel Audio
Transforms	Multichannel Audio	into frequency bands via a Fast Fourier Transform (FFT). The following operations are carried out in discrete frequency bands. When such a configuration is used, a 50% overlap between subsequent audio blocks needs to be employed. The output is a data structure comprising complex valued audio samples in the frequency domain.
Produces	Audio Object	In the Transform domain.

7.1.2.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Analysis Transform (CAE-AAT) AIM.

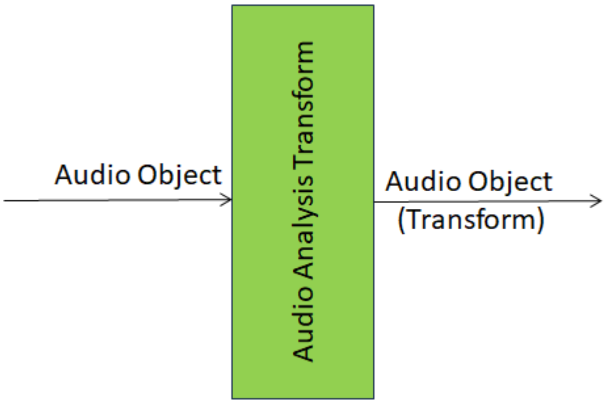


Figure 1 – Audio Analysis Transform (CAE-AAT) AIM

7.1.2.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Analysis Transform (CAE-AAT) AIM.

Table 1 – Audio Analysis Transform (CAE-AAT) AIM

Input	Description
Audio Object	Audio Object (with associated Microphone Array info)
Output	**Description**
Audio Object (Transform)	Audio Object in the Transform domain

7.1.2.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioAnalysisTransform.json

7.1.2.5 Reference Software

The Audio Analysis Transform Reference Software can be downloaded from the MPAI Git.

7.1.2.6 Conformance Testing

Table 2 – Conformance Testing Method for CAE-AAT AIM

Receives	Audio Object (Microphone Array)	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
Produces	Audio Object (Transform)	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.2.7 Performance Assessment

The following steps shall be followed when assessing the Performance of a CAE-AAT AIM instance.

1. Use the following datasets:
    1. DS1: *n* Test Audio Object files including Multichannel Audio as Interleaved Multichannel Audio format.
    2. DS2: *n* Expected Audio Object Output files including data in Transform Interleaved Multichannel Audio format.
2. Feed the AIM under test with the Test files (DS1).
3. Perform the following steps to analyse the Audio Object (Transform) with the Expected Audio Objects (DS2):
    1. Check the data format of the Audio Object (Transform) with the format of the given Expected Audio Objects.
    2. Calculate the peak-to-peak Amplitude (A) of each Audio block in the Expected Audio Objects.
    3. Calculate the RMSE of each Audio block by comparing the Audio Object (Transform) (*x*) with the Expected Audio Objects (*y*).
    4. Accept the AIM under test if, for each audio block, these two conditions are satisfied:
        1. Data format of the Audio Object (Transform) is the same as the format of the Expected Audio Object and
        2. RMSE < A* 0.1%.
4. The Performance Assessor will provide the following matrix containing a limited number of input records (*n*) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails.

Input data (DS1)	**Expected Output Data (DS2)**	**Data Format**	**RMSE**
Audio Object (Microphone Array) ID$_1$	Audio Object (Transform) ID$_1$	T/F	< _A*0.1%_
Audio Object (Microphone Array) ID$_2$	Audio Object (Transform) ID$_2$	T/F	< _A*0.1%_
Audio Object (Microphone Array) ID$_3$	Audio Object (Transform) ID$_3$	T/F	< _A*0.1%_
…	…	…	…
Audio Object (Microphone Array) ID$_n$	Audio Object (Transform) ID$_n$	T/F	< _A*0.1%_

5. Final evaluation: T/F Denoting with *i*, the record number in DS1 and DS2, the matrices reflect the results obtained with input records i with the corresponding outputs i.
6.

DS1	DS2	Audio Object (Transform) output value (from AIM under test)
DS1[*i*]	DS2[*i*]	Audio Object (Transform)[*i*]

Table 2 provides the Performance Assessment Method for the formats of the CAE-AAT AIM output.

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

7.1.3 Audio Descriptors Multiplexing

7.1.3.1 Functions

Audio Descriptor Multiplexing (CAE-AMX) multiplexes Enhanced Audio Objects and their Geometry into out Audio Scene Descriptors:

Receives	*Enhanced Audio Objects*	Audio Objects with reduced noise.
	Audio Scene Geometry	The spatial arrangement of Audio Objects.
Multiplexes	*Enhanced Audio Objects*	Enhanced-quality Audio Objects
	Audio Scene Geometry	Arrangement of Audio Objects
Produces	*Audio Scene Descriptors*	The Descriptors of the Audio Scene.

7.1.3.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Descriptor Multiplexing AIM.

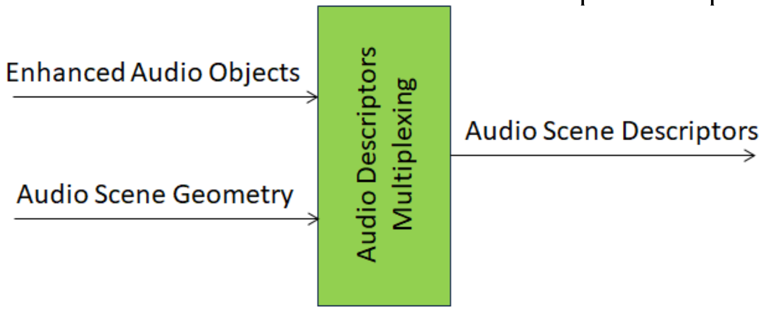


Figure 1 – The Audio Descriptor Multiplexing AIM

7.1.3.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Descriptor Multiplexing AIM.

Table 1 – I/O Data of Audio Descriptor Multiplexing

Input	**Description**
Enhanced Audio Object	Time-domain Audio Objects without noise.
Audio Scene Geometry	The Space-Time arrangement of Audio objects in an Audio Scene
Output	**Description**
Audio Scene Descriptors	The combination of Audio Scene Geometry and Audio Objects.

7.1.3.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioDescriptorsMultiplexing.json

7.1.3.5 Conformance Testing

Table 2 – Conformance Testing Method for CAE-AMX AIM

Receives	Enhanced Audio Objects	Shall validate against Audio Object schema. Audio Data shall conform with Audio Qualifier.
	Audio Scene Geometry	Shall validate against Audio Basic Scene Geometry schema.

Produces	Audio Scene Descriptors	Shall validate against Audio Basic Scene Descriptors schema.

7.1.3.6 Performance Assessment

The following steps shall be followed when assessing the Performance of a CAE-AMX AIM instance.

1. Use the following datasets:
    1. DS1: *n* Enhanced Audio Objects Test files.
    2. DS3: *n* Audio Scene Geometry of the Enhanced Audio Objects.
    3. DS4: *n* Expected Output Audio Scene Descriptors.
2. Feed the AIM under test with the Test files (DS1, DS3).
3. Analyse the Audio Scene Descriptors with the Expected Audio Scene Descriptors (DS4).
    1. Check the Audio Scene Descriptors with Expected given Audio Scene Descriptors.
    2. Calculate the peak-to-peak Amplitude (A) of each Audio block in the Expected Audio Scene Descriptors.
    3. Calculate the RMSE of each Audio block by comparing the output (*x*) with the Audio Scene Descriptors (*y*) Audio blocks.
    4. Accept the CAE-AMX AIM under test if, for each audio block, these the two conditions are satisfied:
        1. Data format of Audio Scene Descriptors is the same as the Expected Audio Scene Descriptors and
        2. RMSE < A * 0.1%
4. The Conformance Tester will provide the following matrix with a limited number of input records (*n*) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails.

Input data (DS1, DS3)	Expected Output Data (DS4)	Data Format	RMSE
Enhanced Audio Objects ID_1 Audio Scene Geometry ID_1	Audio Scene Descriptors ID_1	T/F	< A * 0.1%
Enhanced Audio Objects ID_2 Audio Scene Geometry ID_2	Audio Scene Descriptors ID_2	T/F	< A * 0.1%
Enhanced Audio Objects ID_3 Audio Scene Geometry ID_3	Audio Scene Descriptors ID_3	T/F	< A * 0.1%
…	…		…
Enhanced Audio Objects ID_n Audio Scene Geometry ID_n	Audio Scene Descriptors ID_n	T/F	< A * 0.1%

5. Final evaluation: T/F Denoting with *i*, the record number in DS1 and DS3, the matrices reflect the results obtained with input records i with the corresponding outputs i.

DS1	DS3	DS4	CAE-AMX output value (obtained through the AIM under test)
DS1[*i*]	DS3[i]	DS4[i]	Multiplexer[*i*]

Table 3 provides the Conformance Testing Method for the formats of the CAE-AMX AIM output.

Note: 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.

7.1.4 Audio Object Identification

7.1.4.1 Functions

Audio Object Identification (CAE-AOI) receives Audio Objects and their Scene Geometry and produces the Identities of the Audio Objects:

Receives	*Audio Scene Geometry*	The spatial arrangements of the Audio Objects.
	Audio Objects	The individual input Audio Objects
Identifies	The Audio Objects.	Provides Audio Object Identifiers
Produces	The *Audio Instance IDs*	The Instance Identifier of the Audio Objects.

7.1.4.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Object Identification AIM.

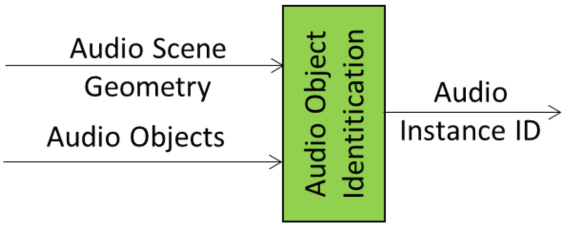


Figure 1 – Audio Object Identification AIM

The Audio Object Identification AIM shall be able to parse either an Audio-Visual Scene Geometry or its Audio Scene Geometry subset.

7.1.4.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Object Identification AIM.

Table 1 – I/O Data of the Audio Object Identification AIM

Input	**Description**
Audio Scene Geometry	The digital representation of the spatial arrangement of the Visual Objects of the Scene.
Audio Objects	The Audio Objects in the Audio Scene Geometry with an identifiable source target of identification.
Output	**Description**
Audio Instance Identifier	The Instance Identifier of the specific Audio Object.

7.1.4.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioObjectIdentification.json

7.1.4.5 Conformance Testing

Table 2 provides the Conformance Testing Method for the CAE-AOI AIM. Conformance Testing of the individual AIMs of the CAE-AOI 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 CAE-AOI AIM

Receives	Audio Scene Geometry	Shall validate against Audio Basic Scene Geometry schema.
	Audio Objects	Shall validate against Audio Object schema. Audio Data shall conform with Audio Qualifier.
Produces	Audio Instance Identifier	Shall validate against Instance ID schema.

7.1.5 Audio Scene Description

7.1.5.1 Functions

Audio Scene Description (CAE-ASD):

Receives	Space-Time	Audio Scene's Space-Time.
	Audio Object	Input Audio Object.
	Audio Scene Descriptors	Possible Audio Scene to be augmented.
Computes	Audio Objects	Inside the input Audio Object.
	Audio Scene Geometry	Possible Audio Scene to be augmented.
Adds	Audio Scene Descriptors	To make the new Audio Scene.
Produces	Audio Scene Descriptors (output)	The augmented Audio Scene.

Unlike CAE-BAS that can only describe an Audio Scene composed of Audio Objects, CAE-ASD can describe an Audio Scene in terms of Audio Objects *and* Audio Scenes. In their turn, Audio Scene can be composed of Objects and Scenes.

7.1.5.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Scene Description AIM.

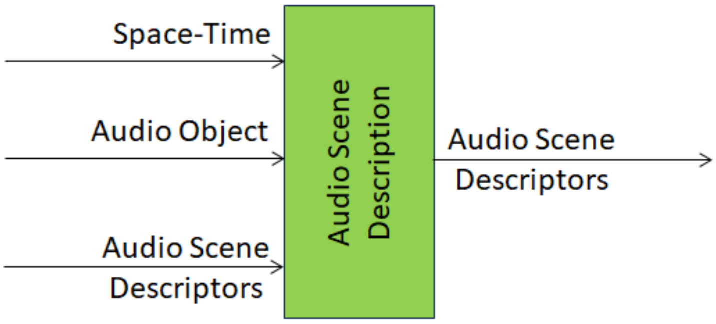


Figure 1 – The Audio Scene Description (CAE-ASD) AIM

7.1.5.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Scene Description AIM.

Table 1 – I/O Data of the Audio Scene Description (CAE-ASD) AIM

Input	**Description**
Space-Time	Audio Objects Space-Time information.
Audio Object	Multichannel Audio (with associated Microphone Array info)
Audio Scene Descriptors	Input Audio Scene Descriptors
Output	**Description**
Audio Scene Descriptors	Output Audio Scene Descriptors

7.1.5.4 SubAIMs

Audio Scene Description (CAE-ASD) is a Composite AIM with the structure depicted in Figure 2.

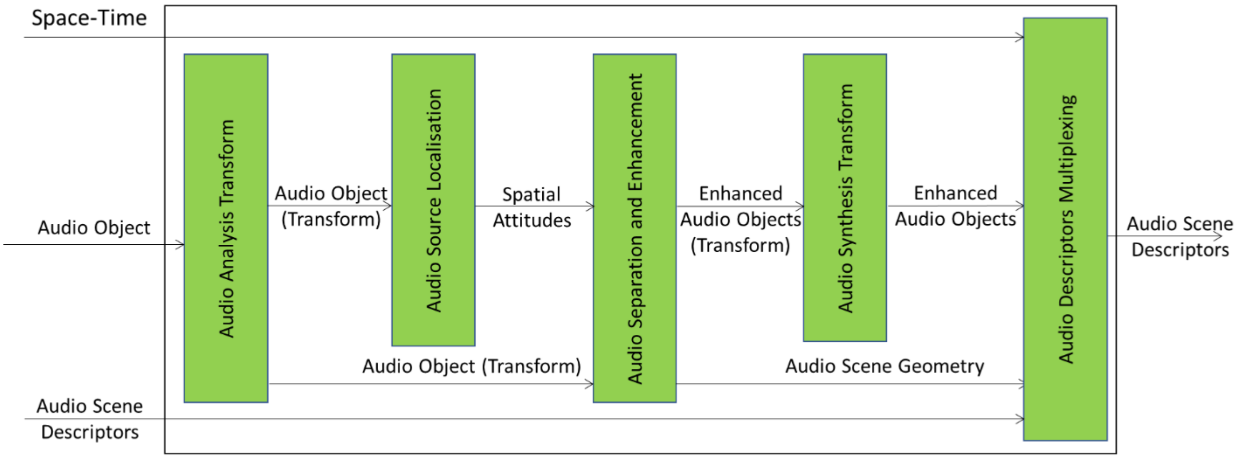


Figure 2 - The Audio Scene Description (CAE-ASD) Composite AIM

The specification of the CAE-ASD Basic AIMs are provided at the links of Table 2.

Table 2 – BASIC AIMs of Audio Scene Descriptors

AIW	AIMs	Names	JSON
CAE-ASD		Audio Scene Description	File
	CAE-AAT	Audio Analysis Transform	File
	CAE-ASL	Audio Source Localisation	File
	CAE-ASE	Audio Separation and Enhancement	File
	CAE-AST	Audio Synthesis Transform	File
	CAE-ADM	Audio Descriptors Multiplexing	File

7.1.5.5 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioSceneDescription.json

7.1.5.6 Conformance Testing

Table 2 provides the Conformance Testing Method for the CAE-ASD AIM. The Conformance Testing Method for the individual Basic AIMs of the CAE-ASD Composite AIM is provided by the individual Basic AIMs.

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-ASD AIM

Receives	Space-Time	Shall validate against Space-Time Schema.
	Audio Object	Shall validate against Audio-Object Schema. Audio Data shall conform with Audio Data Qualifier.
	Audio Scene Descriptors	Shall validate against Audio Scene Descriptors Schema.
Produces	Audio Scene Descriptors	Shall validate against Audio Scene Descriptors Schema.

7.1.6 Audio Separation and Enhancement

7.1.6.1 Functions

Audio Separation and Enhancement (CAE-ASE) receives Audio Objects in the Transform domain with their Spatial Attitude and produces Enhanced Audio Objects with their Scene Geometry.

Receives	*Audio Objects*	in the Transform domain.
	Audio Spatial Attitudes	Spatial Attitudes of the input Audio Objects.
Separates	*Audio Objects*	by using their Spatial Attitudes.
Produces	*Enhanced Audio Object*	in the Transform domain.
	Audio Scene Geometry	The Geometry of Audio Objects in the Scene.

7.1.6.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Separation and Enhancement AIM.

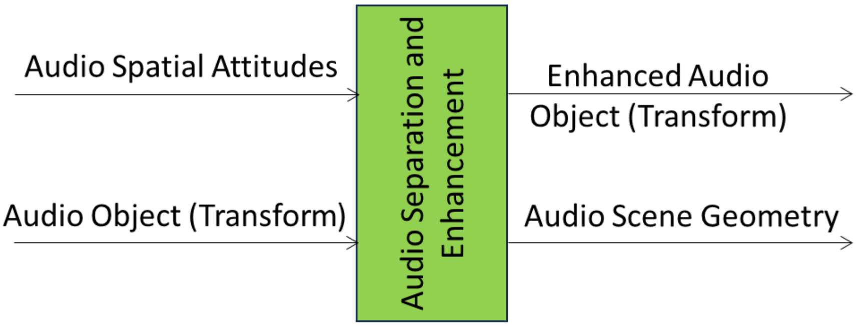


Figure 1 – Audio Separation and Enhancement AIM

7.1.6.3 Input/Output Data

Table 11 specifies the Input and Output Data of the Audio Separation and Enhancement AIM.

Table 1 – I/O Data of Audio Separation and Enhancement

Input	**Description**
Audio Object	The result of the application of the Fast Fourier Transform to the Multichannel Audio.
Audio Spatial Attitudes	The Spatial Attitudes of Audio Objects.
Output	**Description**
Enhanced Audio Object	Enhanced Multichannel Audio in the transform domain.
Audio Scene Geometry	The spatial arrangement of the Audio Objects.

7.1.6.4 JSON Metadata

https://schemas.mpai.community/CAE/V2.4/AIMs/AudioSeparationAndEnhancement.json

7.1.6.5 Conformance Testing

Table 2 provides the Conformance Testing Method for the formats of the CAE-ASE AIM output.
Note: 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-ASE AIM

Receives	Audio Object	Shall validate against Audio Object schema. Audio Data shall conform with Audio Qualifier.
	Audio Spatial Attitudes	Shall validate against Spatial Attitude schema.
Produces	Enhanced Audio Object	Shall validate against Audio Object schema. Audio Data shall conform with Audio Qualifier.
	Audio Scene Geometry	Shall validate against Audio Basic Scene Geometry schema.

7.1.6.6 Performance Assessment

The following steps shall be followed when assessing the Performance of a CAE-ASE AIM instance.

1. Use the following datasets:
    1. DS1: *n* Test files containing Audio Objects (Transform).
    2. DS4: *n* Test files containing the Spatial Attitudes of Audio Objects.
    3. DS2: *n* Expected Enhanced Audio Objects (Transform) Files.
    4. DS3: *n* Expected Audio Scene Geometries.
2. Feed the AIM under test with the Test Audio Objects (Transform) and Spatial Attitudes.
3. Analyse the Audio Scene Geometry and Enhanced Audio (Transform).
    1. Control the Audio Scene Geometry with the Expected Audio Scene Geometry:
        1. Count the number of Audio Objects in the Audio Scene Geometry.
        2. Calculate the angle difference (AD) in degrees between the Audio Objects (*u*) in the Audio Scene Geometry and the Audio Objects (*v*) in the Expected Audio Scene Geometry.
    2. Compare the number of Audio Blocks in the Expected Audio Objects with the number of Audio Blocks in the Audio Objects (Transform).
    3. Calculate Signal to Interference Ratio (SIR), Signal to Distortion Ratio (SDR), and Signal to Artefacts Ratio (SAR) between the Expected Audio Objects (Transform) and Output Audio Objects (Transform).
    4. Accept the CAE-ASE AIM under test if these four conditions are satisfied:
        1. The number of Audio Objects (Transform) in the Audio Scene Geometry is equal to the number of Audio Objects (Transform) in the Expected Audio Scene Geometry.
        2. The number of Audio Blocks in the Audio Objects (Transform) is equal to the number of Audio Blocks in the Expected Audio Objects (Transform).
        3. Compare each Audio Objects (Transform) in the Audio Scene Geometry with the Audio Objects (Transform) in the Expected Audio Scene Geometry.
            1. Each Audio Objects (Transform)’s AD between the Expected and Output is less than 5 degrees.
        4. Compare each Audio Objects (Transform) with the Audio Objects (Transform) in the Expected Audio Objects (Transform).
            1. If the room reverb time (T60) is greater than 0.5 seconds.
                1. Each object’s SIR between the Expected and Output is greater than or equal to 10 dB.
                2. Each object’s SDR between the Expected and Output is greater than or equal to 3 dB.
                3. Each object’s SAR between the Expected and Output is greater than or equal to 3 dB.
            2. If the room reverb time (T60) is less than 0.5 seconds.
                1. Each object’s SIR between the Expected and Output is greater than or equal to 15 dB.
                2. Each object’s SDR between the Expected and Output is greater than or equal to 6 dB.
4. The Performance Assessor shall provide the following matrix containing a limited number of input records (*n*) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails.

Input data (DS1, DS4)	**Expected Output Data (DS2, DS3)**	**Data Format**	**Audio Scene Geometry**	**Source Separation Metrics**
Spatial Attitude (ID$_1$) Audio Object (Transform) ID$_1$	Enhanced Audio Object (Transform) ID$_1$ Audio Scene Geometry ID$_1$	T/F	T/F	T/F
Spatial Attitude (ID$_2$) Audio Object (Transform) ID$_2$	Enhanced Audio Object (Transform) ID$_2$ Audio Scene Geometry ID$_2$	T/F	T/F	T/F
Spatial Attitude (ID$_3$) Audio Object (Transform) ID$_3$	Enhanced Audio Object (Transform ID$_3$ Audio Scene Geometry ID$_3$	T/F	T/F	T/F
…	…	…	…	…
Spatial Attitude (ID$_n$) Audio Object (Transform) ID$_n$	Enhanced Audio Object (Transform ID$_n$ Audio Scene Geometry ID$_n$	T/F	T/F	T/F

6. Final evaluation : T/F Denoting with *i*, the record number in DS1, DS2, and DS3, the matrices reflect the results obtained with input records i with the corresponding outputs i.

DS1	DS2	DS3	Sound Field Description output value (obtained through the AIM under test)
DS1[*i*]	DS2[*i*]	DS3[i]	SpeechDetectionandSeparation[*i*]

7.1.7 Audio Source Localisation

7.1.7.1 Functions

Audio Source Localisation (CAE-ASL) receives Audio Objects, detects the Audio Objects in the Audio Scene, and determines and produces as output their Spatial Attitudes:

Receives	*Audio Objects*	With associated Microphone Array information.
Detects	*Audio Objects*	In the Audio Scene.
Determines	*Spatial Attitudes*	Of Audio Objects.
Produces	*Spatial Attitudes*	Of input Audio Objects.

7.1.7.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Source Localisation (CAE-ASL) AIM.

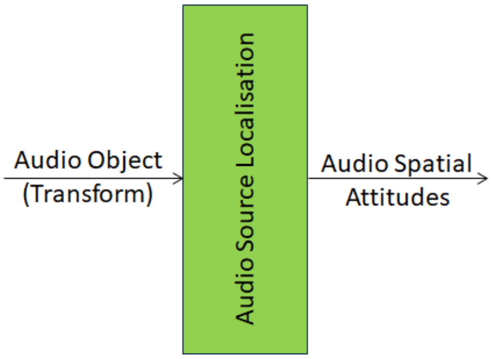


Figure 1 – Audio Source Localisation (CAE-ASL) AIM

7.1.7.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Source Localisation (CAE-ASL) AIM.

Table 1 – Audio Source Localisation (CAE-ASL) AIM

Input	**Description**
Audio Object	The result of the application of the Fast Fourier Transform to the Multichannel Audio (with associated Microphone Array info).
Output	**Description**
Audio Spatial Attitudes	The Orientations and Directions of Audio Objects.

7.1.7.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.3/AIMs/AudioSourceLocalisation.json

7.1.7.5 Conformance Testing

The following procedure shall be followed when testing the Conformance of a CAE-ASL AIM instance.

1. Use the following datasets:
    1. DS1: *n* Test files containing Audio Objects (Transform) .
    2. DS2: *n* Expected Spatial Attitudes.
2. Feed the AIM under test with the Test files.
3. Analyse the Spatial Attitudes produced by the CAE-ASL AIM instance.
4. Calculate the angle difference (AD) in degrees between the output Spatial Attitudes with the Expected Spatial Attitudes.

Input data (DS1)	**Expected Output Data (DS2)**	**Data Format**	**RMSE**
Audio Object (Microphone Array) ID_1	Spatial Attitude ID_1	T/F	< *A*0.1%*
Audio Object (Microphone Array) ID_2	Spatial Attitude (Transform) ID_2	T/F	< *A*0.1%*
Audio Object (Microphone Array) ID_3	Spatial Attitude (Transform) ID_3	T/F	< *A*0.1%*
…	…	…	…
Audio Object (Microphone Array) ID_n	Spatial Attitude (Transform) ID_n	T/F	< *A*0.1%*

5. Final evaluation: T/F Denoting with *i*, the record number in DS1 and DS2, the matrices reflect the results obtained with input records i with the corresponding outputs i.

DS1	DS2	Audio Object (Transform) output value (from AIM under test)
DS1[*i*]	DS2[*i*]	Audio Object (Transform)[*i*]

Table 2 provides the Conformance Testing Method for the formats of the CAE-ASL AIM output.

Note: 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-ASL AIM

Receives	Audio Object	Shall validate against Audio Object schema. Audio Data shall conform with Audio Qualifier.
Produces	Audio Spatial Attitudes	Shall validate against Spatial Attitude schema.

7.1.8 Audio Synthesis Transform

7.1.8.1 Functions

Audio Synthesis Transform (CAE-AST) receives an Enhanced Audio Object in the Transform domain, transforms the Audio Object back to the time domain and produces an Enhanced Audio Object with associated Microphone Array info:

Receives	*Enhanced Audio Object (Transform)*	with associated Microphone Array info.
Transforms	*Enhanced Audio Object (Transform)*	from the frequency domain to the time domain via an Inverse Fast Fourier Transform (IFFT).
Produces	*Enhanced Audio Object*	with associated Microphone Array info.

7.1.8.2 Reference Model

Figure 1 depicts the Reference Architecture of the Audio Synthesis Transform (CAE-AST) AIM.

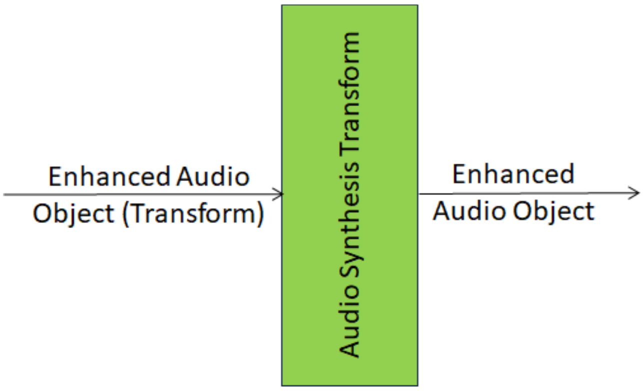


Figure 1 – The Audio Synthesis Transform (CAE-AST) AIM

7.1.8.3 Input/Output Data

Table 1 specifies the Input and Output Data of the Audio Synthesis Transform (CAE-AST) AIM.

Table 1 – I/O Data of Synthesis Transform (CAE-AST) AIM

Input	**Description**
Enhanced Audio Objects (time-frequency)	Audio Objects in the time-frequency domain.

Output	Description
Enhanced Audio Objects (time)	Audio Objects in the time domain.

7.1.8.4 *JSON Metadata*

https://schemas.mpai.community/CAE1/V2.4/AIMs/AudioSynthesisTransform.json

7.1.8.5 *Reference Software*

The Audio Synthesis Transform Reference Software can be downloaded from the MPAI Git.

7.1.8.6 *Conformance Testing*

Receives	Enhanced Audio Objects (time-frequency)	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
Produces	Enhanced Audio Objects (time)	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.8.7 *Performance Assessment*

Table 61 gives the *Enhanced Audioconference Experience (CAE-EAE) Synthesis Transform* Means and how they are used.

Table 61 – AIM Means and use of Enhanced Audioconference Experience (CAE-EAE) Synthesis Transform

Means	Actions
Performance Testing Dataset	DS1: *n* Test files including data in Denoised Transform Speech format. DS2: *n* Expected Output files including data in Denoised Speech format.
Procedure	1. Feed the AIM under test with the Test files (DS1). 2. Analyse the Denoised Speech with the Expected Output files (DS2).
Evaluation	1. Check the Denoised Speech data format with the given Expected Output files format. 2. Calculate the peak-to-peak Amplitude (A) of each Audio block in the Expected Output files. 3. Calculate the RMSE of each Audio block by comparing the output (x) with the Expected Output files (y) Audio blocks.

	4. Accept the AIM under test if, for each audio block, these the two conditions are satisfied: a. Data format of the Denoised Speech is the same with the Expected Output Files and b. RMSE < A* 0.1%

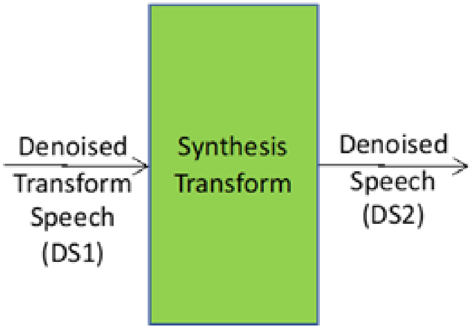


Figure 24 - Synthesis Transform Testing Flow

After the Tests, Performance Assessor shall fill out *Table 62Table 62*

Table 62 – Performance Assessment form of Enhanced Audioconference Experience (CAE-EAE) Synthesis Transform

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form "CAE:EAE:1:0".
Name of AIM	Synthesis Transform
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Testing Dataset	Unique Dataset Identifier assigned by MPAI Store.
Test ID	Unique Test Identifier assigned by Performance Assessor.
Actual output	The Performance Assessor will provide the following matrix with a limited number of input records (*n*) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails. Input data (DS1) Expected Output Data (DS2) Data Format RMSE

<table>
<tr><td></td><td>
<table>
<tr><td>Denoised Transform Speech ID$_1$</td><td>Denoised Speech ID$_1$</td><td>T/F</td><td>&lt; A* 0.1%</td></tr>
<tr><td>Denoised Transform Speech ID$_2$</td><td>Denoised Speech ID$_2$</td><td>T/F</td><td>&lt; A* 0.1%</td></tr>
<tr><td>Denoised Transform Speech ID$_3$</td><td>Denoised Speech ID$_3$</td><td>T/F</td><td>&lt; A* 0.1%</td></tr>
<tr><td>…</td><td>…</td><td></td><td>…</td></tr>
<tr><td>Denoised Transfom Speech ID$_n$</td><td>Denoised Speech ID$_n$</td><td>T/F</td><td>&lt; A* 0.1%</td></tr>
</table>
<p>Final evaluation: T/F</p>
<p>Denoting with <i>i</i>, the record number in DS1 and DS2, the matrices reflect the results obtained with input records i with the corresponding outputs i.</p>
<table>
<tr><td>DS1</td><td>DS2</td><td>Synthesis Transform output value (obtained through the AIM under test)</td></tr>
<tr><td>DS1[<i>i</i>]</td><td>DS2[<i>i</i>]</td><td>SynthesisTransform[<i>i</i>]</td></tr>
</table>
</td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of test execution.</td></tr>
<tr><td><b>Test comment*</b></td><td>Comments on test results and possible needed actions.</td></tr>
<tr><td><b>Test Date</b></td><td>yyyy/mm/dd.</td></tr>
</table>

** Optional field*

7.1.9 Emotion Feature Production

7.1.9.1 Function

Emotion Feature Production (MMC-EFP):

Receives	Emotionless Speech Features from Emotion Feature Production.
	Emotion List.
	Language Selector
Produces	Neural Speech Features

7.1.9.2 Reference Model

Figure 1 depicts the Emotion Feature Production (MMC-EFP) AIM:

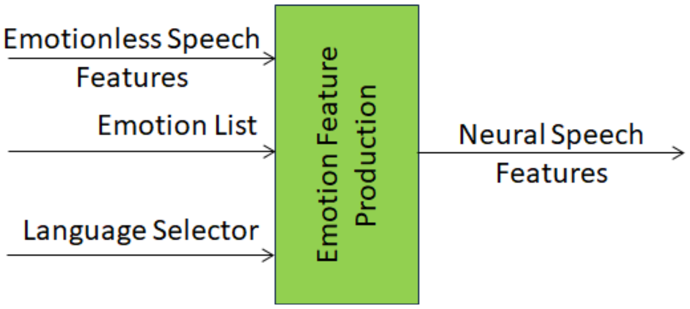


Figure 1 - Reference Model of Emotion Feature Production (MMC-EFP) AIM

7.1.9.3 Input/Output Data

Table 1 specifies the Input/Output Data of the Emotion Feature Production (MMC-EFP)

Table 1 - Input/Output Data of the Emotion Feature Production (MMC-EFP)

Input data	**Semantics**
Emotionless Speech Features	Speech Features of an Emotionless Speech utterance.
Emotion List	A List of Emotions to be added to Emotionless Speech,
Language Selector	The preferred language.
Output data	**Semantics**
Neural Speech Features	Speech Features to be added to the Emotionless Speech.

7.1.9.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/EmotionFeatureProduction.json

7.1.9.5 Reference Software

Reference Software not available.

7.1.9.6 Conformance Testing

Receives	Emotionless Speech Features	Shall validate against the Speech Features schema.
	Emotion List	Shall validate against the Emotion schema.
	Language Selector	Shall validate against the Selector schema.
Produces	Neural Speech Features	Shall validate against the Speech Features schema.

7.1.9.7 Performance Assessment

Table 8 gives the input/output data of the Emotion Feature Production AIM.

Table 8 – I/O Data of Emotion Enhanced Speech (EES) Emotion Feature Production

AIM	**Input Data**	**Output Data**
Emotion Feature Production	Emotionless Speech Speech Features1	Speech with Emotion

Table 9 gives the Emotion Enhanced Speech (EES) Emotion Feature Production Means and how they are used.

Table 9 – AIM Means and use of Emotion Enhanced Speech (EES) Emotion Feature Production (AIM2 in Figure 3)

Means	Actions
Conformance Testing Dataset	DS1: a dataset of at least $x > M$ Emotionless Speeches. DS2: a dataset of x Speech Features 1, each corresponding to a specific Emotionless Speech.
Procedure	For each of the x input pairs of DS1 and DS2: 1. Feed the Emotion Inserter 1 under test with an Emotionless Speech and its corresponding array of Speech Features 1. 2. Feed the reference Speech Feature Analyser 1 (ID: *S*) with the Speech with Emotion came as output from the Emotion Inserter 1 under test. 3. Verify that the number of features in Speech Features 1 array coming as output from the reference Speech Feature Analyser 1 equals the corresponding one in DS2. 4. For each feature of the output Speech Features 1 array, compute the *delta* (absolute difference) between: 1. the pitch property and the corresponding DS2 data in Hz. 2. the intensity property and the corresponding DS2 data in dB. 3. the duration property and the corresponding DS2 data in ms. 5. Compute the Average of: 1. The *deltas* of the pitch property. 2. The *deltas* of the intensity property. 3. The *deltas* of the duration property. Then, compute the Average for each of the three properties among the *n* Model Utterances.
Evaluation	1. Condition 3 shall be respected. 2. Given the three Averages computed at the end of the Procedure and denoting them with , where *p* represents one among the three properties (pitch, intensity and duration), if: $Res = \frac{A_{pitch}}{2} + \frac{A_{intensity}}{4} + \frac{A_{duration}}{4} < m$ the Emotion Inserter 1 module under test has passed the Conformance Test. 3. Otherwise, the submitter of Emotion Inserter 1 is given the opportunity to submit an implementation of Speech Feature Analyser 1. 4. The MPAI Store will test the combination of the two submitted AIMs. 5. If the quality of the output of the submitted combination of AIM1 and AIM2 is above threshold, Emotion Inserter 1 passes the Conformance Test as long as the corresponding Speech Feature Analyser 1 is made available to the MPAI Store. 6. Else, Emotion Inserter 1 does not pass the Conformance Test.

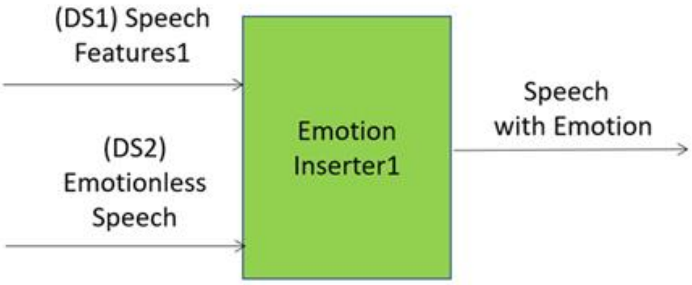


Figure 4 – EES Emotion Inserter1.

After the Tests, Conformance Tester shall fill out *Table 10*.

Table 10 – Conformance Testing form of Emotion Enhanced Speech (EES) Emotion Inserter1

<table>
<tr><td><b>Conformance Tester ID</b></td><td>Unique Conformance Tester Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EES:V:P”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Emotion Inserter1</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Conformance Testing Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Conformance Tester.</td></tr>
<tr><td><b>Actual output</b></td><td>Actual output provided as a matrix of n+1 rows containing all computed Average values:
<table>
<tr><th>#</th><th>Pitch</th><th>Intensity</th><th>Duration</th></tr>
<tr><td>1</td><td>$A_{pitch}[1]$</td><td>$A_{intensity}[1]$</td><td>$A_{duration}[1]$</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td>n</td><td>$A_{pitch}[n]$</td><td>$A_{intensity}[n]$</td><td>$A_{duration}[n]$</td></tr>
<tr><td>Averages</td><td>A_{pitch}</td><td>$A_{intensity}$</td><td>$A_{duration}$</td></tr>
</table>
Result:
Threshold: m
Final evaluation: Passed / Not passed</td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of test execution.</td></tr>
<tr><td><b>Test comment*</b></td><td></td></tr>
<tr><td><b>Test Date</b></td><td>yyyy/mm/dd.</td></tr>
</table>

** Optional field*

7.1.10 Neural Emotion Insertion

7.1.10.1 Functions

Neural Emotion Insertion (CAE-NEI)

Receives	Neural Speech Features from Emotion Feature Producer.
	Emotionless Speech.
Integrates	(Emotional) Neural Speech Features with those of the Emotionless Speech input.

Produces	Emotionally modified utterance Speech with Emotion.

7.1.10.2 Reference Model

Figure 1 depicts the Reference Model of Neural Emotion Insertion (CAE-NEI)

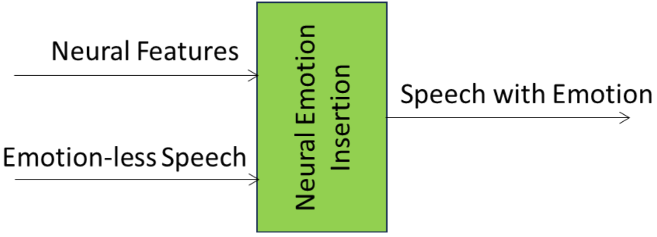


Figure 1 - Reference Model of Neural Emotion Insertion (CAE-NEI)

7.1.10.3 Input/Output Data

Table 1 provides the Input/Output Data of Neural Emotion Insertion (CAE-NEI)

Table 1 - Input/Output Data of Neural Emotion Insertion (CAE-NEI)

Input data	**Semantics**
Neural Speech Features	Speech Features of the Emotion Feature Producer.
Emotionless Speech	The speech without emotion to which emotion is added,
Output data	**Semantics**
Speech with Emotion	The Emotionless Speech to which emotion has been added

7.1.10.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/NeuralEmotionInsertion.json

7.1.10.5 Conformance Testing

Receives	Neural Speech Features	
	Emotionless Speech	Shall validate against the Speech Object schema. The Qualifier shall validate against the Speech Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Speech Object Qualifier schema.
Produces	Speech with Emotion	Shall validate against the Speech Object schema. The Qualifier shall validate against the Speech Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Speech Object Qualifier schema.

7.1.10.6 Performance Assessment

Table 18 gives the Emotion Enhanced Speech (EES) Neural Emotion Insertion Means and how they are used.

Table 18 – AIM Means and use of Emotion Enhanced Speech (EES) Neural Emotion Insertion

Means	**Actions**

Conformance Testing Dataset	DS1: a dataset of at least $y > N$ Emotionless Speech Segments. DS2: a dataset of y Emotion Lists. DS3: a dataset of one element, specifying the Language in question. DS4: a dataset of y Speech with Emotion Segments, where each is associated with specific elements of DS1, DS2, and DS3 used as input, and thus represents one correct output, given this input.
Procedure	Given a reference Speech Feature Analyser 2 (ID: *sfa2*), a reference Emotion Feature Producer (ID: *efp*) and an Emotion Inserter 2 module that we want to test, we measure the quality of Emotion Inserter 2 in relation to the reference modules as follows: 1. Connect the three modules. 2. Repeat many times: 1. Select an input set comprised of a DS1 (Emotionless Speech segment), a DS2 (an Emotion List), and a DS3 (a Language). 2. Feed that set to the system composed by the connected modules. 3. Measure the quality of the Speech with Emotion output generated by the system by comparing it with the corresponding "correct" result in DS4 as measured by PESQ [6]. 3. The quality of Emotion Inserter 2 is then the *average value* of the multiple quality measurements of 2c.
Evaluation	1. If the *average value* of the quality measurements is above a threshold above 2.0 as specified by PESQ, Emotion Inserter 2 has passed the Conformance Test. 2. If the quality is below threshold, the submitter of Emotion Inserter 2 is given the opportunity to submit an implementation of Speech Feature Analyser 2 and Emotion Feature Producer. 3. The MPAI Store will test the combination of the three submitted AIMs. 4. If the quality of the output of the submitted combination is above threshold, Emotion Inserter 2 passes the Conformance Test as long as the corresponding Speech Feature Analyser 2 and Emotion Feature Producer are made available to the MPAI Store. 5. Else, Emotion Inserter 2 doesn't pass the Conformance Test.

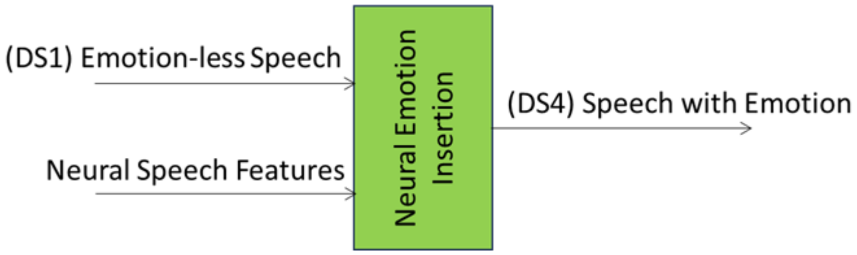


Figure 8 – Neural Emotion Inserter.

After the Tests, Conformance Tester shall fill out *Table 19*.

Table 19 – Conformance Testing form of Emotion Enhanced Speech (EES) Neural Emotion Insertion

Conformance Tester ID	Unique Conformance Tester Identifier assigned by MPAI

<table>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE-EES-V2.4”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Neural Emotion Insertion</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Conformance Testing Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Conformance Tester.</td></tr>
<tr><td><b>Actual output</b></td><td>The Conformance Tester will provide the following matrix related to the modules utilized for the tests. Denoting with <i>i</i> and <i>j</i>, and , the record number in DS1 and DS2 respectively, the matrices reflect the results obtained with a limited number of random multiple inputs and the corresponding outputs.
Example:
<table>
<tr><td>DS1</td><td>DS2</td><td>DS4</td><td>Emotion Inserter2 output value</td></tr>
<tr><td>DS1[<i>i</i>]</td><td>DS2[<i>j</i>]</td><td>DS4[<i>i</i>, <i>j</i>]</td><td>SpeechWithEmotion[<i>i</i>, <i>j</i>]</td></tr>
</table>
Language: DS3</td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of test execution.</td></tr>
<tr><td><b>Test comment*</b></td><td>In case step 1 of Conformance Testing fails, the Conformance Tester shall request the implementer to provide a Speech Feature Analyser2 and Emotion Feature Producer AIMs.
In case step 4 or 5 of Conformance Testing also fails, the Conformance Tester shall inform the implementer that the Emotion Inserter2 did not pass the CT.</td></tr>
<tr><td><b>Test Date</b></td><td>yyyy/mm/dd.</td></tr>
</table>

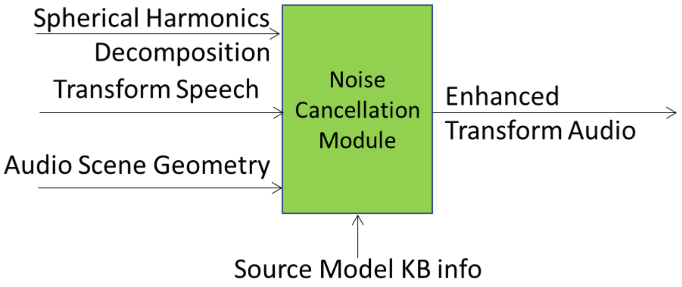
* *Optional field*

7.1.11 Noise Cancellation Module

7.1.11.1 Functions

1. Receives Spherical Harmonic Decomposition Coefficients, Transform Speech, Audio Scene Geometry, and Source Model KB info.
2. Eliminates background noise and reverberation which reduce the audio quality, acting as a Passthrough AIM if environmental conditions do not substantially add ambient noise to the desired speech.
3. Produces Denoised Transform Speech

7.1.11.2 *Reference Model*



7.1.11.3 *Input/Output Data*

Input data	Semantics
Spherical Harmonics Decomposition Coefficients	Result of the transformation of Transform Multichannel Audio into the spherical frequency domain.
Transform Audio	Audio Object in the Transform Domain.
Audio Scene Geometry	Spatial arrangement of Audio Objects.
Source Model KB info	Discrete-time and discrete-valued simple acoustic source models used in source separation.
Output data	**Semantics**
Enhanced Transform Speech	Transform Speech whose noise level has been reduced.

7.1.11.4 *JSON Metadata*

https://schemas.mpai.community/CAE1/V2.4/AIMs/NoiseCancellationModule.json

7.1.11.5 *Reference Software*

The Noise Cancellation Module can be downloaded from the MPAI Git.

7.1.11.6 *Conformance Testing*

Receives	Spherical Harmonics Decomposition Coefficients	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Transform Audio	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Audio Scene Geometry	Shall validate against Audio Scene Geometry schema.

	Source Model KB info	Discrete-time and discrete-valued simple acoustic source models used in source separation.
Produces	Enhanced Transform Speech	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.11.7 Performance Assessment

Table 58 gives the *Enhanced Audioconference Experience (CAE-EAE) Noise Cancellation* Means and how they are used.

Table 58 – AIM Means and use of Enhanced Audioconference Experience (CAE-EAE) Noise Cancellation

Means	Actions
Performance Assess Dataset	DS1: *n* Test files containing SHD. DS2: *n* Test files containing Transform Speech. DS3: *n* Test files containing Audio Scene Geometry. DS4: *n* Expected Denoised Transform Speech.
Procedure	1. Feed the AIM under test with the Test files (DS1, DS2, DS3). 2. Analyse the Denoised Transform Speech (DS4).
Evaluation	1. Compare the number of Audio Blocks in the Expected Denoised Transform Speech with the number of Audio Blocks in the Denoised Transform Speech Files. 2. Compute Perception Evaluation of Speech Quality (PESQ) between the Expected and Output Denoised Transform Speech Files [6]. 3. Accept the AIM under test if these two conditions are satisfied: a. The number of Audio Blocks in the Denoised Transform Speech is the same with the number of Audio Blocks in the Expected Denoised Transform Speech. b. Compare each Denoised Transform Speech with the Expected Denoised Transform Speech. c. If the room reverb time (T60) is greater than 0.5 seconds. i. Each object's PESQ between the Expected and Output is greater than P=2.0. d. If the room reverb time (T60) is smaller than 0.5 seconds. i. Each object's PESQ between the Expected and Output is greater than P=3.0.

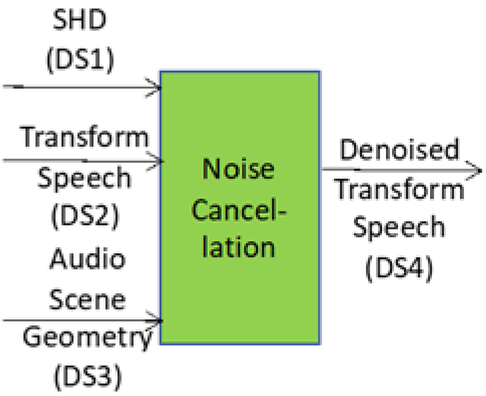


Figure 23 - Noise Cancellation Testing Flow

After the Tests, Performance Assessor shall fill out *Table 59.Table 59*

Table 59 – Performance Assessment form of Enhanced Audioconference Experience (CAE-EAE) Noise Cancellation

<table>
<tr><td><b>Performance Assessor ID</b></td><td>Unique Performance Assessor Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EAE:1:0”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Noise Cancellation</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Performance Assessment Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Performance Assessor.</td></tr>
<tr><td><b>Actual output</b></td><td>The Performance Assessor will provide the following matrix containing a limited number of input records (<i>n</i>) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails.
<table>
<tr><td>Input data (DS1, DS2, DS3)</td><td>Expected Output Data (DS4)</td><td>Data Format</td><td>PESQ Score</td></tr>
<tr><td>SHD ID$_1$
Transform Speech ID$_1$
Audio Scene Geometry ID$_1$</td><td>Denoised Transform Speech ID$_1$</td><td>T/F</td><td>&gt; P</td></tr>
<tr><td>SHD ID$_2$
Transform Speech ID$_2$</td><td>Denoised Transform Speech ID$_2$</td><td>T/F</td><td>&gt; P</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td>
<table>
<tr><td>Audio Scene
Geometry ID$_2$</td><td></td><td></td><td></td></tr>
<tr><td>SHD ID$_3$
Transform Speech ID$_3$
Audio Scene Geometry ID$_3$</td><td>Denoised Transform Speech ID$_3$</td><td>T/F</td><td>> P</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td>SHD ID$_n$
Transform Speech ID$_n$
Audio Scene Geometry ID$_n$</td><td>Denoised Transform Speech ID$_n$</td><td>T/F</td><td>> P</td></tr>
</table>
Final evaluation : T/F
Denoting with *i*, the record number in DS1, DS2, and DS3, the matrices reflect the results obtained with input records i with the corresponding outputs i.
<table>
<tr><td>DS1</td><td>DS2</td><td>DS3</td><td>DS4</td><td>Noise Cancellation output value (obtained through the AIM under test)</td></tr>
<tr><td>DS1[*i*]</td><td>DS2[*i*]</td><td>DS3[i]</td><td>DS4[i]</td><td>NoiseCancellation[*i*]</td></tr>
</table>
</td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of test execution.</td></tr>
<tr><td><b>Test comment*</b></td><td>Comments on test results and possible needed actions.</td></tr>
<tr><td><b>Test Date</b></td><td>yyyy/mm/dd.</td></tr>
</table>

** Optional field*

7.1.12 Packaging for Audio Preservation

7.1.12.1 Function

1. Receives Preservation Audio File, Restored Audio Files, Editing List, Irregularity File, Irregularity Images, and Preservation Audio-Visual Files.
2. Produces Preservation Master Files and Access Copy Files

7.1.12.2 Reference Model

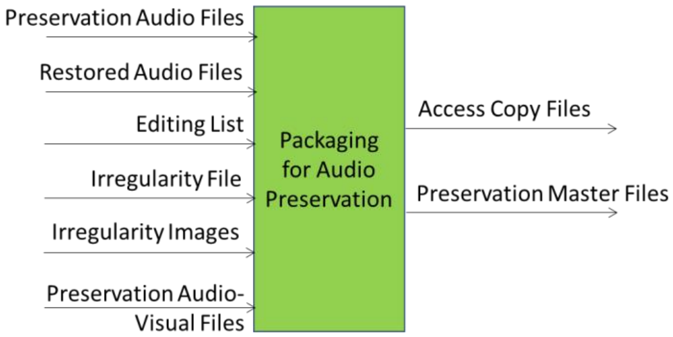


Figure 1 - Reference Model of Packaging for Audio Preservation

7.1.12.3 Input/Output Data

Input data	**Semantics**
Preservation Audio File	File obtained by digitising the analogue tape audio recording composed of music, soundscape or speech read from a magnetic tape.
Restored Audio Files	Preservation Audio File restored by Tape Audio Restoration.
Editing List	The list for editing the Preservation Audio File..
Irregularity File	Irregularity File produced by Tape Irregularity Classification.
Irregularity Images	Images corresponding to the Irregularities received or detected
Output data	**Semantics**
Editing List	The list for editing the Preservation Audio File.
Restored Audio Files	Audio Files obtained by restoring the Preservation Audio File per Editing List.

7.1.12.4

No SubAIMs.

7.1.12.5 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/PackagingForAudioPreservation.json

7.1.12.6 Reference Software

The CAE-PAP Reference Software can be downloaded from the MPAI Git.

7.1.12.7 Conformance Testing

Receives	Preservation Audio File	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Restored Audio Files	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Editing List	Shall validate against the Editing List schema.
	Irregularity File	Shall validate against the Irregularity File schema.
	Irregularity Images	Shall validate against the Visual Object schema. The Qualifier shall validate against the Visual Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Visual Object Qualifier schema.
Produces	Editing List	Shall validate against the Editing List schema.

	Restored Audio Files	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.12.8 Performance Assessment

Table 35 gives the Audio Recording Preservation (ARP) *Packager* Means and how they are used.

Table 35 – AIM Means and use of Audio Recording Preservation (ARP) Packager.

Means	**Actions**
Performance Assessment Dataset	DS1: *n* Preservation Audio Files. DS2: *n* Preservation Audio-Visual Files related to DS1. DS3: *n* Restored Audio Files arrays related to DS1 coming from Tape Audio Restoration. DS4: *n* Editing Lists related to DS3 coming from Tape Audio Restoration. DS5: *n* Irregularity Files related to DS1 coming from Tape Irregularity Classifier. DS6: *n* Irregularity Images related to DS5 coming from Tape Irregularity Classifier. DS7: *n* Access Copy Files. DS8: *n* Preservation Master Files.
Procedure	1. Feed Packager under Assessment with DS1, DS2, DS3, DS4, DS5 and DS6. 2. Compare the output Access Copy Files with DS7. 3. Compare the output Preservation Master Files with DS8.
Evaluation	For a given input tuple, verify that: 1. The output Access Copy Files contain the Restored Audio Files, the Editing List, the Irregularity File and the set of Irregularity Images in a .zip file and is therefore equal to DS7. 2. The output Preservation Master Files contain the Preservation Audio File, the Preservation Audio-Visual File with the audio of the Preservation Audio File, the Irregularity File, and the Irregularity Images, and is therefore equal to DS8. An error on any of the output arrays will make the Packager under test not conformant.

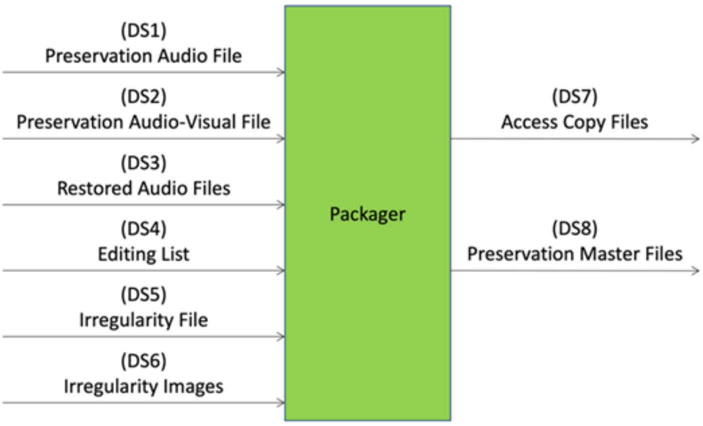


Figure 14 – Packager.

After the Assessment, Performance Assessor shall fill out Table 36.

Table 36 – Performance Assessment form of Audio Recording Preservation (ARP) Packager.

<table>
<tr><td><b>Performance Assessor ID</b></td><td>Unique Performance Assessor Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE-ARP-2.40”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Packager</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Performance Assessor Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Assessment ID</b></td><td>Unique Assessment Identifier assigned by Performance Assessor.</td></tr>
<tr><td><b>Actual output</b></td><td>Actual output provided as a matrix of <i>n</i> rows containing output assertions.
<table>
<tr><td>Output</td><td>Files</td><td>1</td><td>…</td><td><i>n</i></td></tr>
<tr><td rowspan="4">Access Copy Files</td><td>Restored Audio Files</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Editing List</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Irregularity File</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Irregularity Images</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td rowspan="4">Preservation Master Files</td><td>Preservation Audio File</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Preservation Audio-Visual File</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Irregularity File</td><td>T/F</td><td>…</td><td>T/F</td></tr>
<tr><td>Irregularity Images</td><td>T/F</td><td>…</td><td>T/F</td></tr>
</table>
Final assertion: T/F</td></tr>
</table>

Execution time*	Duration of Assessment execution.
Assessment comment*	Comments on Assessment results and possible needed actions.
Assessment Date	yyyy/mm/dd.

** Optional field*

7.1.13 Prosodic Emotion Insertion

7.1.13.1 Functions

Prosodic Emotion Insertion (CAE-PEI)

Receives	Prosodic Speech Features
	Emotionless Speech
Integrates	(Emotional) Prosodic Speech Features with those of the Emotionless Speech input.
Produces	Emotionally modified utterance Speech with Emotion

7.1.13.2 Reference Model

Figure 1 depicts the Prosodic Emotion Insertion (CAE-PEI) AIM

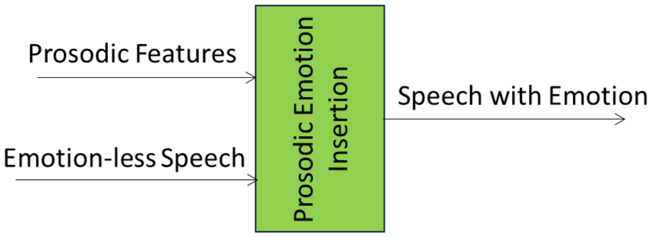


Figure 1 - Reference Model of Prosodic Emotion Insertion (CAE-PEI) AIM

7.1.13.3 Input/Output Data

Table 1 provides the Input/Output Data of the Prosodic Emotion Insertion (CAE-PEI) AIM

Table 1 - Input/Output Data of the Prosodic Emotion Insertion (CAE-PEI) AIM

Input data	**Semantics**
Prosodic Speech Features	Speech Features from Speech Feature Analyser 1.
Emotionless Speech	The speech without emotion to which Emotion is added.
Output data	**Semantics**
Speech with Emotion	The Emotionless Speech to which emotion has been added

7.1.13.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/ProsodicEmotionInsertion.json

7.1.13.5 Conformance Testing

Receives	Prosodic Speech Features	Shall validate against the Speech Features schema.
	Emotionless Speech	Shall validate against the Speech Object schema. The Qualifier shall validate against the Speech Qualifier schema.

		The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Speech Object Qualifier schema.
Produces	Speech with Emotion	Shall validate against the Speech Object schema.

7.1.13.6 Performance Assessment

Table 9 gives the Emotion Enhanced Speech (EES) Prosodic Emotion Insertion Means and how they are used.

Table 9 – AIM Means and use of Emotion Enhanced Speech (EES) Prosodic Emotion Insertion (AIM2 in Figure 3)

Means	Actions
Conformance Testing Dataset	DS1: a dataset of at least $x > M$ Emotionless Speeches. DS2: a dataset of x Speech Features 1, each corresponding to a specific Emotionless Speech.
Procedure	For each of the x input pairs of DS1 and DS2: 1. Feed the Emotion Inserter 1 under test with an Emotionless Speech and its corresponding array of Speech Features 1. 2. Feed the reference Speech Feature Analyser 1 (ID: *S*) with the Speech with Emotion came as output from the Emotion Inserter 1 under test. 3. Verify that the number of features in Speech Features 1 array coming as output from the reference Speech Feature Analyser 1 equals the corresponding one in DS2. 4. For each feature of the output Speech Features 1 array, compute the *delta* (absolute difference) between: 1. the pitch property and the corresponding DS2 data in Hz. 2. the intensity property and the corresponding DS2 data in dB. 3. the duration property and the corresponding DS2 data in ms. 5. Compute the Average of: 1. 1. The *deltas* of the pitch property. 2. The *deltas* of the intensity property. 3. The *deltas* of the duration property. Then, compute the Average for each of the three properties among the *n* Model Utterances.
Evaluation	1. Condition 3 shall be respected. 2. Given the three Averages computed at the end of the Procedure and denoting them with , where *p* represents one among the three properties (pitch, intensity and duration), if: $Res = \frac{A_{pitch}}{2} + \frac{A_{intensity}}{4} + \frac{A_{duration}}{4} < m$ the Neural Emotion Insertion module under test has passed the Conformance Test. 3. Otherwise, the submitter of Emotion Inserter 1 is given the opportunity to submit an implementation of Speech Feature Analyser 1. 4. The MPAI Store will test the combination of the two submitted AIMs. 5. If the quality of the output of the submitted combination of AIM1 and AIM2 is above threshold, Emotion Inserter 1 passes the Conformance Test as

	long as the corresponding Speech Feature Analyser 1 is made available to the MPAI Store. 6. Else, Emotion Inserter 1 does not pass the Conformance Test.

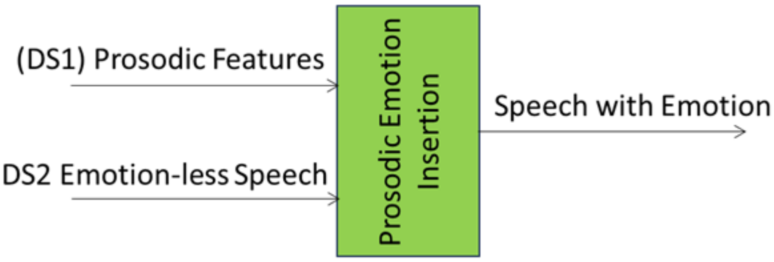


Figure 4 – EES Prosodic Emotion Insertion

After the Tests, Conformance Tester shall fill out *Table 10*.

Table 10 – Conformance Testing form of Emotion Enhanced Speech (EES) Prosodic Emotion Insertion

Conformance Tester ID	Unique Conformance Tester Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form "CAE:EES:V:P".
Name of AIM	Prosodic Emotion Insertion
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Conformance Testing Dataset	Unique Dataset Identifier assigned by MPAI Store.
Test ID	Unique Test Identifier assigned by Conformance Tester.
Actual output	Actual output provided as a matrix of *n*+1 rows containing all computed Average values: # Pitch Intensity Duration 1 … … … … *n* Averages Result: Threshold: *m* Final evaluation: Passed / Not passed
Execution time*	Duration of test execution.
Test comment*	
Test Date	yyyy/mm/dd.

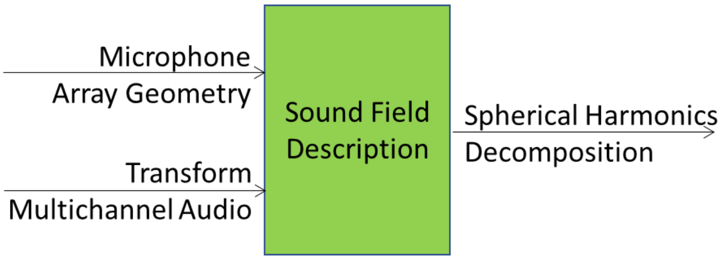
** Optional field*

7.1.14 Sound Field Description

7.1.14.1 Functions

Receives	Microphone Array Geometry	Geometry of the Microphone Array
	Transform Multichannel	Audio Object in the Transform domain
Produces	Spherical Harmonic Decomposition Coefficients	Result of the transformation into spherical frequency domain.

7.1.14.2 Reference Model


Microphone Array Geometry
Transform Multichannel Audio
Sound Field Description
Spherical Harmonics Decomposition

7.1.14.3 Input/Output Data

Input data	**Semantics**
Microphone Array Geometry	Geometry of the Microphone Array
Transform Multichannel Audio	Audio Object in the Transform domain
Output data	**Semantics**
Spherical Harmonics Decomposition Coefficients	Result of the transformation into spherical frequency domain.

7.1.14.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/SoundFieldDescription.json

7.1.14.5 Reference Software

The Sound Field Description Reference Software can be downloaded from the MPAI Git.

7.1.14.6 Conformance Testing

Receives	Microphone Array Geometry	Shall validate against the Microphone Array Geometry Schema.
	Transform Multichannel Audio	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

Produces	Spherical Harmonics Decomposition Coefficients	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.14.7 Performance Assessment

Table 52 Table 52 gives the *Enhanced Audioconference Experience (CAE-EAE) Sound Field Description* Means and how they are used.

Table 52 – AIM Means and use of Enhanced Audioconference Experience (CAE-EAE) Sound Field Description

Means	Actions
Performance Testing Dataset	DS1: *n* Test files containing real recordings or simulations structured in Transform Multichannel Audio format. DS2: *n* Microphone Array Geometry associated with the real recordings or simulations. DS3: *n* Expected Output files including data in SHD format.
Procedure	1. Feed the AIM under test with the Test files (DS1) and their associated Microphone Array Geometry (DS2). 2. Analyse the SHD with the Expected Output files (DS3).
Evaluation	1. Check the output SHD data format with the given Expected Output files format. 2. Calculate the peak-to-peak Amplitude (A) value of each Audio block in the Expected Output files. 3. Calculate the RMSE of each Audio block in SHD by comparing the output (*x*) with the Expected Output files (*y*). 4. Accept the AIM under test if, for each audio block, these two conditions are satisfied: a. Data format of the SHD is the same with the Expected Output Files and b. RMSE < A * 0.1%

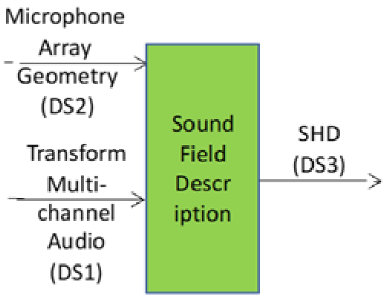

Microphone Array Geometry (DS2)
Transform Multi-channel Audio (DS1)
Sound Field Descr iption
SHD (DS3)

Figure 21 - Sound Field Description Testing Flow

After the Tests, Performance Assessor shall fill out *Table 53*.

Table 53 – Performance Testing form of Enhanced Audioconference Experience (CAE-EAE) Sound Field Description

<table>
<tr><td><b>Performance Assessor ID</b></td><td>Unique Performance Assessor Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EAE:1:0”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Sound Field Description</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Performance Testing Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Performance Assessor.</td></tr>
<tr><td><b>Actual output</b></td><td>The Performance Assessor will provide the following matrix containing a limited number of input records (<i>n</i>) with the corresponding outputs. If an input record fails, the Assessor would specify the reason why the test case fails.
<table>
<tr><td>Input data (DS1, DS2)</td><td>Expected Output Data (DS3)</td><td>Data Format</td><td>RMSE</td></tr>
<tr><td>Transform Multichannel Audio ID$_1$
Microphone Array Geometry ID$_1$</td><td>SHD ID$_1$</td><td>T/F</td><td>&lt; A * 0.1%</td></tr>
<tr><td>Transform Multichannel Audio ID$_2$
Microphone Array Geometry ID$_2$</td><td>SHD ID$_2$</td><td>T/F</td><td>&lt; A * 0.1%</td></tr>
<tr><td>Transform Multichannel Audio ID$_3$</td><td>SHD ID$_3$</td><td>T/F</td><td>&lt; A * 0.1%</td></tr>
</table></td></tr>
</table>

	Microphone Array Geometry ID$_3$ … … … … Transform Multichannel Audio ID$_n$ SHD ID$_n$ T/F < A * 0.1% Microphone Array Geometry ID$_n$ Final evaluation: T/F Denoting with *i*, the record number in DS1, DS2, and DS3, the matrices reflect the results obtained with input records i with the corresponding outputs i. DS1 DS2 DS3 Sound Field Description output value (obtained through the AIM under test) DS1[*i*] DS2[*i*] DS3[i] SoundFieldDescription[*i*]
Execution time*	Duration of test execution.
Test comment*	Comments on test results and possible needed actions.
Test Date	yyyy/mm/dd.

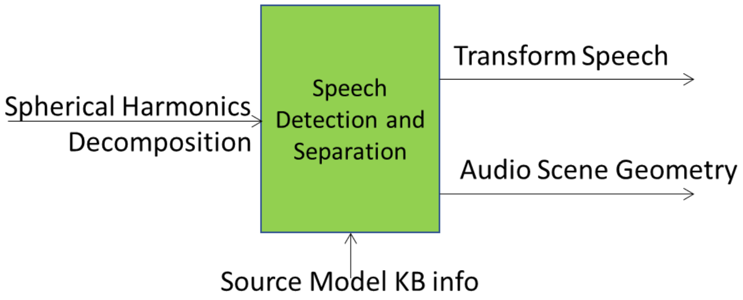
** Optional field*

7.1.15 Speech Detection and Separation

7.1.15.1 Functions

1. Receives the Spherical Harmonic Decomposition coefficients of the sound field
2. Detects and directions of active sound sources (either be a speech or a non-speech signal) and to separate them.
3. Produces the Transformed Speech and Audio Scene Geometry..

7.1.15.2 Reference Model


Spherical Harmonics Decomposition
Speech Detection and Separation
Transform Speech
Audio Scene Geometry
Source Model KB info

7.1.15.3 Input/Output Data

Input data	Semantics
Spherical Harmonics Decomposition Coefficients	Result of the transformation of Transform Multichannel Audio into the spherical frequency domain.

Source Model KB info	Discrete-time and discrete-valued simple acoustic source models used in source separation.
Output data	**Semantics**
Transform Audio	Audio in the Transform domain.
Audio Scene Geometry	Geometry of the Audio Scene.

7.1.15.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/SpeechDetectionAndSeparation.json

7.1.15.5 Reference Software

The Speech Detection and Separation Reference Software can be downloaded from the MPAI Git.

7.1.15.6 Conformance Testing

Receives	Spherical Harmonics Decomposition Coefficients	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Source Model KB info	
Produces	Transform Audio	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.15.7 Performance Assessment

Table 55 gives the *Enhanced Audioconference Experience (CAE-EAE) Speech Detection and Separation* Means and how they are used.

Table 55 – AIM Means and use of Enhanced Audioconference Experience (CAE-EAE) Speech Detection and Separation

Means	**Actions**
Performance Testing Dataset	The Performance Assessment Dataset is composed b: DS1: *n* Test files containing SHD. DS2: *n* Expected Transform Speech Files. DS3: *n* Expected Audio Scene Geometry.
Procedure	1. Feed the AIM under test with the Test files. 2. Analyse the Audio Scene Geometry. 3. Analyse Transform Speech Files.

Evaluation	1. Control the Audio Scene Geometry with the Expected Audio Scene Geometry: a. Count the number of objects in the Audio Scene Geometry. b. Calculate the angle difference (AD) in degrees between the objects ( ) in the Audio Scene Geometry and the objects ( ) in the Expected Audio Scene Geometry. 2. Compare the number of Audio Blocks in the Expected Transform Speech with the number of Audio Blocks in the Transform Speech Files. 3. Calculate Signal to Interference Ratio (SIR), Signal to Distortion Ratio (SDR), and Signal to Artefacts Ratio (SAR) between the Expected and Output Transform Speech Files [12]. 4. Accept the AIM under test if these four conditions are satisfied: a. The number of speech objects in the Audio Scene Geometry is equal to the number of speech objects in the Expected Audio Scene Geometry. b. The number of Audio Blocks in the Transform Speech is equal to the number of Audio Blocks in the Expected Transform Speech. c. Compare each Speech Object in the Audio Scene Geometry with the Speech Object in the Expected Audio Scene Geometry. i. Each object's AD between the Expected and Output is less than 5 degrees. d. Compare each Speech Object in the Transform Speech with the Speech Object in the Expected Transform Speech. i. If the room reverb time (T60) is greater than 0.5 seconds. 1. Each object's SIR between the Expected and Output is greater than or equal to 10 dB. 2. Each object's SDR between the Expected and Output is greater than or equal to 3 dB. 3. Each object's SAR between the Expected and Output is greater than or equal to 3 dB. ii. If the room reverb time (T60) is less than 0.5 seconds. 1. Each object's SIR between the Expected and Output is greater than or equal to 15 dB. 2. Each object's SDR between the Expected and Output is greater than or equal to 6 dB. 3. Each object's SAR between the Expected and Output is greater than or equal to 6 dB.

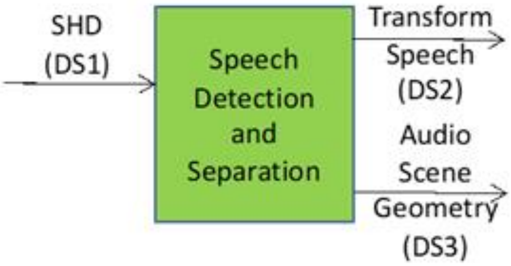


Figure 22 - Speech Detection and Separation Testing Flow

After the Tests, Performance Assessor shall fill out *Table 56.Table 56*

Table 56 – Performance Assessment form of Enhanced Audioconference Experience (CAE-EAE) Speech Detection and Separation

<table>
<tr><td><b>Performance Assessor ID</b></td><td>Unique Performance Assessor Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EAE:1:0”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Speech Detection and Separation</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Performance Assessment Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Performance Assessor.</td></tr>
<tr><td><b>Actual output</b></td><td>The Performance Assessor will provide the following matrix containing a limited number of input records (<i>n</i>) with the corresponding outputs. If an input record fails, the tester would specify the reason why the test case fails.
<table>
<tr><td>Input data (DS1)</td><td>Expected Output Data (DS2, DS3)</td><td>Data Format</td><td>Audio Scene Geometry</td><td>Source Separation Metrics</td></tr>
<tr><td>SHD ID$_1$</td><td>Transform Speech ID$_1$
Audio Scene Geometry ID$_1$</td><td>T/F</td><td>T/F</td><td>T/F</td></tr>
<tr><td>SHD ID$_2$</td><td>Transform Speech ID$_2$
Audio Scene Geometry ID$_2$</td><td>T/F</td><td>T/F</td><td>T/F</td></tr>
<tr><td>SHD ID$_3$</td><td>Transform Speech ID$_3$
Audio Scene Geometry ID$_3$</td><td>T/F</td><td>T/F</td><td>T/F</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td>SHD ID$_n$</td><td>Transform Speech ID$_n$
Audio Scene Geometry ID$_n$</td><td>T/F</td><td>T/F</td><td>T/F</td></tr>
</table>
Final evaluation : T/F
Denoting with <i>i</i>, the record number in DS1, DS2, and DS3, the matrices reflect the results obtained with input records i with the corresponding outputs i.
<table>
<tr><td>DS1</td><td>DS2</td><td>DS3</td><td>Sound Field Description output value (obtained through the AIM under test)</td></tr>
<tr><td>DS1[<i>i</i>]</td><td>DS2[<i>i</i>]</td><td>DS3[i]</td><td>SpeechDetectionandSeparation[<i>i</i>]</td></tr>
</table>
</td></tr>
</table>

Execution time*	Duration of test execution.
Test comment*	Comments on test results and possible needed actions.
Test Date	yyyy/mm/dd.

** Optional field*

7.1.16 Speech Feature Analysis 1

7.1.16.1 Functions

Speech Feature Analysis 1 (CAE-SF1):

Receives	Model Utterance containing emotion.
Extracts	Speech Features1 from the Model Utterance.
Produces	Prosodic Speech Features.

7.1.16.2 Reference Model

Figure 1 depicts the Speech Feature Analysis 1 (CAE-SF1) AIM:

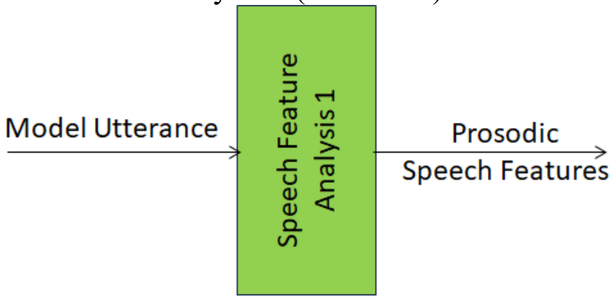


Figure 1 - Speech Feature Analysis 1 (CAE-SF1) AIM

7.1.16.3 Input/Output Data

Table 1 gives the Input/Output Data of the Speech Feature Analysis 1 (CAE-SF1) AIM.

Table 1 - Input/Output Data of the Speech Feature Analysis 1 (CAE-SF1) AIM

Input data	**Semantics**
Model Utterance	Utterance provided as a model.
Output data	**Semantics**
Prosodic Speech Features	A type of Speech Features (Descriptors).

7.1.16.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/SpeechFeatureAnalysis1.json

7.1.16.5 Reference Software

Reference Software not available.

7.1.16.6 Conformance Testing

Receives	Model Utterance	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

Produces	Prosodic Speech Features	Shall validate against the Speech Features Schema.

7.1.16.7 Performance Assessment

Table 6 gives the Emotion Enhanced Speech (EES) Speech Feature Analyser1 Means (verification procedures) and how they are used.

Table 6 – Means and use of Emotion Enhanced Speech (EES) Speech Feature Analyser1 AIM

Means	**Actions**
Conformance Testing Dataset	DS1: a dataset of at least *n* > *M* Model Utterances. DS2: a dataset of *n* Speech Features 1 arrays, where each is associated with a specific utterance of DS1 used as input, and thus represents one correct output, given this input.
Procedure	For each of the *n* Model Utterances in input: 1. Feed the Speech Feature Analyser (SFA) 1 under test with the current Model Utterance. 2. Verify that the number of features in output Speech Features 1 array equals the corresponding one in DS2. 3. For each feature of the output Speech Features 1 array, compute the *delta* (absolute difference) between: 1. the pitch property and the corresponding DS2 data in Hz. 2. the intensity property and the corresponding DS2 data in dB. 3. the duration property and the corresponding DS2 data in ms. 4. 4. Compute the Average of: 1. The *deltas* of the pitch property. 2. The *deltas* of the intensity property. 3. The *deltas* of the duration property. Then, compute the Average for each of the three properties among the *n* Model Utterances. Considering one of the three properties (pitch, intensity and duration) and denoting it as *p*, a mathematical representation of the computation for each property is: $$A_{p_i} = \frac{\sum_{k=1}^{m_i} \lvert SFA1_p(k) - DS2_p(k) \rvert}{m_i}$$ $$A_p = \frac{\sum_{i=1}^{n} A_{p_i}}{n}$$ Where: - For $1 \leq i \leq n$, A_{p_i} represents the Average of the *deltas* of the *i-th* Speech Features 1 array for property *p*. - m_i is the length of the *i-th* Speech Features 1 arrays. - For $1 \leq k \leq m_i$, $SFA1_p(k)$ and $DS2_p(k)$ denote the *k-th* value for property *p* of, respectively, the Speech Features 1 array coming from the Speech Feature Analyser 1 under test and the Speech Features 1 array contained in DS2. - A_p represents the final Average for property *p*.

Evaluation	1. Condition 2 shall be respected. 2. Given the three A_p Averages computed at the end of the Procedure, if: $$Res = \frac{A_{pitch}}{2} + \frac{A_{intensity}}{4} + \frac{A_{duration}}{4} < m$$ the Speech Feature Analyser 1 module under test has passed the Conformance Test. 3. Otherwise, Speech Feature Analyser 1 does not pass the Conformance Test.

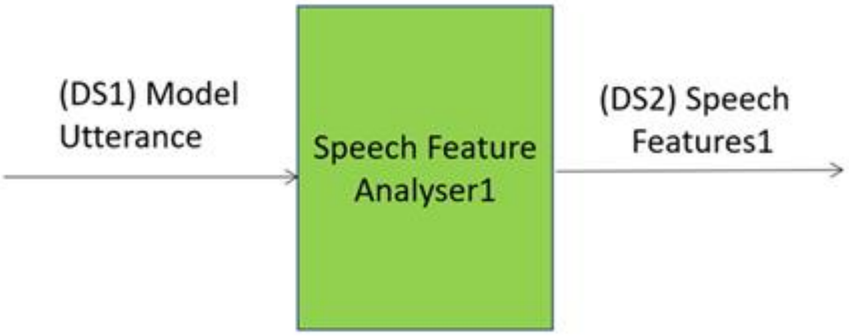


Figure 3 – EES Speech Feature Analyser1.

After the Tests, Conformance Tester shall fill out *Table 7*.

Table 7 – Conformance Testing form of Emotion Enhanced Speech (EES) Speech Feature Analyser1 (AIM1)

<table>
<tr><td><b>Conformance Tester ID</b></td><td>Unique Conformance Tester Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EES:1.2:0”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Speech Feature Analyser1</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by Conformance Tester.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Test Dataset</b></td><td>Unique Dataset Identifier assigned by Conformance Tester.</td></tr>
<tr><td><b>Test ID</b></td><td>Unique Test Identifier assigned by Conformance Tester.</td></tr>
<tr><td><b>Actual output</b></td><td>Actual output provided as a matrix of <i>n</i>+1 rows containing all computed Average values:
<table>
<tr><th>#</th><th>Pitch</th><th>Intensity</th><th>Duration</th></tr>
<tr><td>1</td><td>$A_{pitch}[1]$</td><td>$A_{intensity}[1]$</td><td>$A_{duration}[1]$</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td><i>n</i></td><td>$A_{pitch}[n]$</td><td>$A_{intensity}[n]$</td><td>$A_{duration}[n]$</td></tr>
<tr><td>Averages</td><td>A_{pitch}</td><td>$A_{intensity}$</td><td>$A_{duration}$</td></tr>
</table>
Result:
Threshold: <i>m</i>
Final evaluation: Passed / Not passed</td></tr>
</table>

Execution time*	Duration of test execution.
Test comment*	
Test Date	yyyy/mm/dd.

** Optional field*

7.1.17 Speech Feature Analysis 2

7.1.17.1 Function

Speech Feature Analysis 2 (CAE-SF2):

Receives	Emotionless Speech.
Extracts	Emotionless Speech Features from Emotionless Speech.
Produces	Prosodic Speech Features.

7.1.17.2 Reference Model

Figure 1 depicts the Speech Feature Analysis2 (CAE-SF2) AIM:

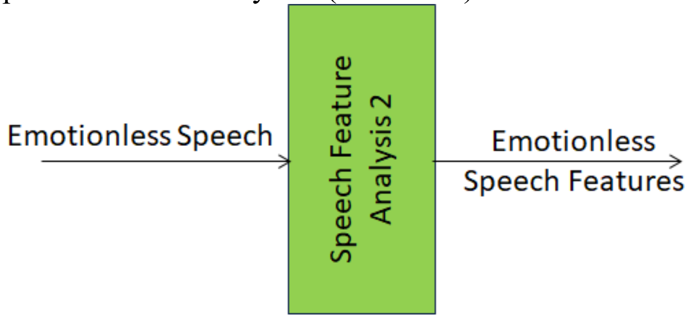


Figure 1 - Speech Feature Analysis 2 (CAE-SF2) AIM

7.1.17.3 Input/Output Data

Table 1 gives the Input/Output Data of the Speech Feature Analysis 2 (CAE-SF2) AIM.

Table 1 - Input/Output Data of the Speech Feature Analysis 2 (CAE-SF2) AIM

Input data	**Semantics**
Emotionless Speech	Utterance provided as a model.
Output data	**Semantics**
Emotionless Speech Features	Descriptors of the Soeech without Emotion..

7.1.17.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/SpeechFeatureAnalysis2.json

7.1.17.5 Conformance Testing

Receives	Emotionless Speech	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

Produces	Emotionless Speech Features	Shall validate against the Speech Features Schema.

7.1.17.6 Performance Assessment

Table 12 gives the Emotion Enhanced Speech (EES) Speech Feature Analysis 2 Means (verification procedures) and how they are used.

Table 12 – Means and use of Emotion Enhanced Speech (EES) Speech Feature Analysis2 AIM

Means	Actions
Conformance Testing Dataset	DS1: a dataset of at least $y > N$ Emotionless Speech Segments. DS2: a dataset of y Emotion Lists. DS3: a dataset of one element, specifying the Language in question. DS4: a dataset of y Speech with Emotion Segments, where each is associated with specific elements of DS1, DS2, and DS3 used as input, and thus represents one correct output, given this input.
Procedure	Given a reference Emotion Feature Producer (ID: *efp*), a reference Emotion Inserter 2 (ID: *ei2*) and a Speech Feature Analysis 2 module that we want to test, we measure the quality of Speech Feature Analysis 2 in relation to the reference modules as follows: 1. Connect the three modules. 2. Repeat many times: 1. Select an input set comprised of a DS1 (Emotionless Speech segment), a DS2 (an Emotion List), and a DS3 (a Language). 2. Feed that set to the system composed by the connected modules. 3. Measure the quality of the Speech with Emotion output generated by the system by comparing it with the corresponding “correct” result in DS4 as measured with PESQ [6]. 3. The quality of Speech Feature Analyser 2 is then the *average value* of the multiple quality measurements of 2c.
Evaluation	1. If the *average value* of the quality measurements is above a threshold greater than 2.0 as specified by PESQ, Speech Feature Analyser 2 has passed the Conformance Test. 2. If the quality is below threshold, the submitter of Speech Feature Analyser 2 is given the opportunity to submit an implementation of Emotion Feature Producer and Emotion Inserter 2. 3. The MPAI Store will test the combination of the three submitted AIMs. 4. If the quality of the output of the submitted combination is above threshold, Speech Feature Analyser 2 passes the Conformance Test as long as the corresponding Emotion Feature Producer and Emotion Inserter 2 are made available to the MPAI Store. 5. Else, Speech Feature Analysis 2 doesn’t pass the Conformance Test.

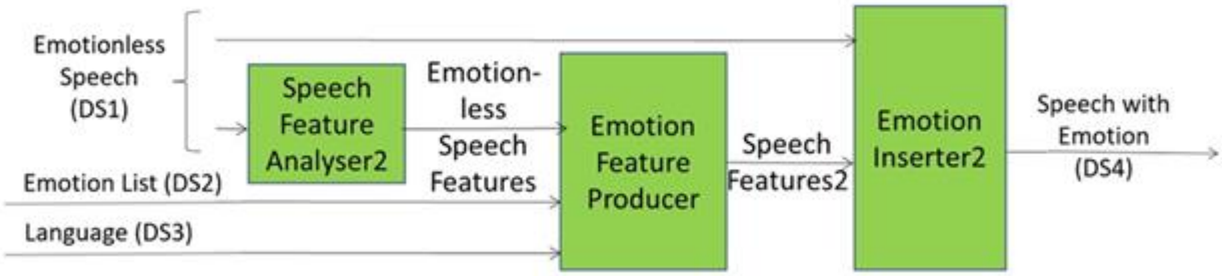


Figure 5 - EES path 2

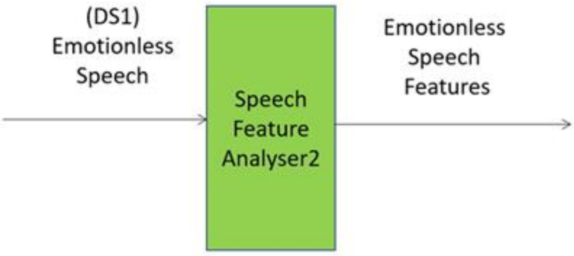


Figure 6 - EES Speech Feature Analyser2.

After the Tests, Conformance Tester shall fill out *Table 13*.

Table 13 – Conformance Testing form of Emotion Enhanced Speech (EES) Speech Feature Analysis2 AIM

Conformance Tester ID	Unique Conformance Tester Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE:EES:1:0”.
Name of AIM	Speech Feature Analyser2
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Conformance Testing Dataset	Unique Dataset Identifier assigned by MPAI Store.
Test ID	Unique Test Identifier assigned by Conformance Tester.
Actual output	The Conformance Tester will provide the following matrix related to the modules utilized for the tests. Denoting with *i* and *j*, $0 \le i < x$ and $0 \le j < y$, the record number in DS1 and DS2 respectively, the matrices reflect the results obtained with a limited number of random multiple inputs and the corresponding outputs. Example: DS1 DS2 DS4 Emotion Inserter2 output value DS1[*i*] DS2[*j*] DS4[*i*, *j*] SpeechWithEmotion[*i*, *j*] Language: DS3
Execution time*	Duration of test execution.

Test comment*	In case step 1 of Conformance Testing fails, the Conformance Tester shall request the implementer to provide an Emotion Feature Producer AIM (AIM2). In case step 4 or 5 of Conformance Testing also fails, the Conformance Tester shall inform the implementer that the Speech Feature Analyser2 (AIM1) did not pass the CT.
Test Date	yyyy/mm/dd.

** Optional field*

7.1.18 Speech Model Creation

7.1.18.1 Function

1. Receives Audio Segments for Modeling.
2. Produces Natural Language Speech Model.

7.1.18.2 Reference Model

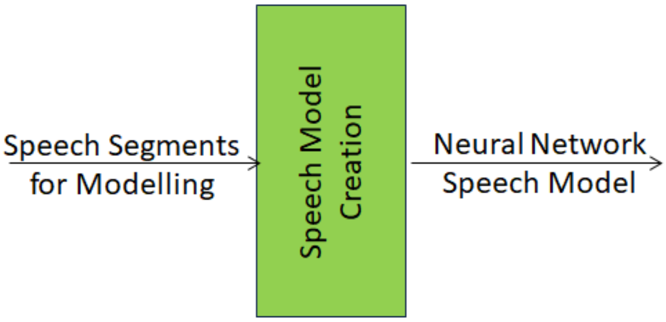


Figure 1 - Speech Model Creation (CAE-SMC) AIM

7.1.18.3 Input/Output Data

Input data	Semantics
Audio Segments for Modelling	A set of Audio Files containing speech segments used to train the Neural Network Speech Model.
Output data	**Semantics**
Neural Network Speech Model	A Neural Network Model trained on Speech Segments for Modelling and used to synthesise replacements for the entire Damaged Segment or Damaged Sections within it.

7.1.18.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/SpeechModelCreation.json

7.1.18.5 Conformance Assessment

Receives	Audio Segments for Modelling	Shall validate against the Audio Object schema. The Audio Segments for Modelling Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Audio

		Segments for Modelling Qualifiers shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
Produces	Neural Network Speech Model	Shall validate against ML Model schema. The Neural Network Speech Model Qualifier shall validate against the ML Model Qualifier Schema. The values of any Sub-Type, Format, and Attribute of the Neural Network Speech Model Qualifier shall correspond with the Sub-Type, Format, and Attributes of the ML Qualifier schema.

7.1.18.6 Performance Assessment

Table 39 gives the Speech Restoration System *(CAE-SRS) Speech Model Creation* Means and how they are used.

Table 39 - AIM Means and use of Speech Restoration System (CAE-SRS) Speech Model Creation

Means	Actions
Performance Assessment Dataset	DS1: a set of *n* Audio Segments, suitable input for creation of a Neural Network Speech Model for a specific speaker.
Procedure	1. Pass Audio Segments for Modelling to Speech Module Creation AIM as input, according to its declared standard procedure. 2. Provide resulting Neural Network Speech Model as input to the reference Speech Synthesiser AIM (ID: *ss*). 3. Synthesise all texts in canonical Text List.
Evaluation	1. Evaluate synthesis quality using Perception Evaluation of Speech Quality (PESQ). 2. If the score is above a threshold of 2.0, the Speech Model Creation AIM is judged adequate. 3. If the quality is below threshold, the submitter of Speech Model Creation is given the opportunity to submit an implementation of Speech Synthesiser. 4. The MPAI Store will test the combination of the two submitted AIMs. 5. If the quality of the output of the submitted combination is above threshold, Speech Model Creation passes the Performance Test as long as the corresponding Speech Synthesiser are made available to the MPAI Store. 6. Else, Speech Model Creation doesn’t pass the Performance Test.

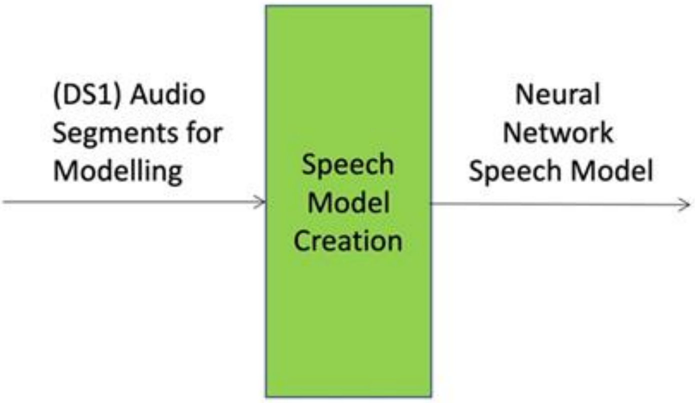

(DS1) Audio Segments for Modelling
Speech Model Creation
Neural Network Speech Model

Figure 16 - Speech Model Creation.

After the Tests, Performance Assessor shall fill out *Table 40.*

Table 40 - Performance Assessment form of Speech Restoration System (SRS) Speech Model Creation.

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form "CAE:SRS:1:0".
Name of AIM	Speech Model Creation
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Assessment Dataset	Unique Dataset Identifier assigned by MPAI Store.
Test ID	Unique Test Identifier assigned by Performance Assessor.
Actual output	Actual output provided as a matrix of *n* rows containing output assertions. For example: **#** **Input** **Final assertion** 1 DS1[1] T/F … … … *n* DS1[*n*] T/F Final evaluation: T/F Legend: - #: Speech Model Creation input dataset tuple number. - **DS1**: Audio Segments for Modelling - **Final assertion**: T if Neural Network Speech Model is well-formed, else F - **Final evaluation**: T if all **Final assertions** are T, else F
Execution time*	Duration of test execution.
Test comment*	Comments on test results and possible needed actions.
Test Date	yyyy/mm/dd.

** Optional field*

7.1.19 Speech Restoration Assembly

7.1.19.1 Function

Receives	Damaged Segment	A Damaged Segment.
	Damaged List	The list of Damaged Segments.
	Synthesised Speech	To be used as replacement of Damaged Segment.

Produces	Restored Segment	The Restored Segment to be used used in lieu of Damaged Segments.

7.1.19.2 *Reference Model*

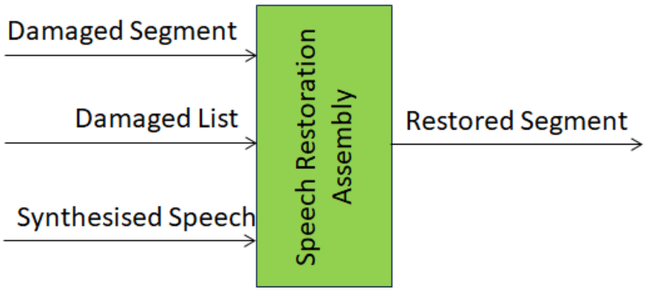


Figure 1 - Speech Restoration Assembly

7.1.19.3 *Input/Output Data*

Input data	**Semantics**
Damaged Segment	An Audio Segment containing only speech (and not containing music or other sounds) which is either damaged in its entirety or contains one or more Damaged Sections specified in the Damaged List.
Damaged List	List of Damaged Segments.
Synthesised Speech	Speech synthesised by Neural Network Speech Model.
Output data	**Semantics**
Restored Speech Segment	Speech synthesised by Speech Restoration Assembly

7.1.19.4 *JSON Metadata*

https://schemas.mpai.community/CAE1/V2.4/AIMs/SpeechRestorationAssembly.json

7.1.19.5 *Conformance Testing*

Receives	Damaged Segment	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Damaged Segment Qualifiers shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Damaged List	Shall validate against the Damaged List schema. The Time shall validate against the Time schema.
	Synthesised Speech	Shall validate against the Audio or Speech Object schema. The Qualifier shall validate against the Audio or Speech Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Synthesised Speech Qualifier shall correspond with the Sub-

		Type, Format, and Attributes of the Audio or Speech Object Qualifier schema.
Produces	Restored Speech Segment	Shall validate against the Audio or Speech Object schema. The Qualifier shall validate against the Audio or Speech Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Restored Speech Segment Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio or Speech Object Qualifier schema.

7.1.19.6 Performance Assessment

Table 45 gives the *Speech Restoration System (CAE-SRS) Speech Restoration Assembly* Means and how they are used.

Table 45 - AIM Means and use of Speech Restoration System (CAE-SRS) Speech Restoration Assembly

Means	Actions
Performance Assessment Dataset	DS1: a canonical set of *n* Damaged Segments DS2: a canonical set of *n* Damaged Lists DS3: a canonical set of *n* Synthesised Speeches.
Procedure	1. Pass DS1, DS2 and DS3 to Assembler, according to its declared standard Procedure. 2. Perform all specified assembly operations: Synthesised Speech results shall replace all bad sections of Damaged Segment as specified by Damaged List.
Evaluation	1. Restored Segment shall be evaluated for quality using Perception Evaluation of Speech Quality (PESQ). Restoration shall be seamless, so that listeners are unable to reliably identify locations of repaired sections. 2. If the scores exceed a declared threshold, Assembler is judged adequate. 3. Else, Assembler doesn’t pass the Performance Assessment.

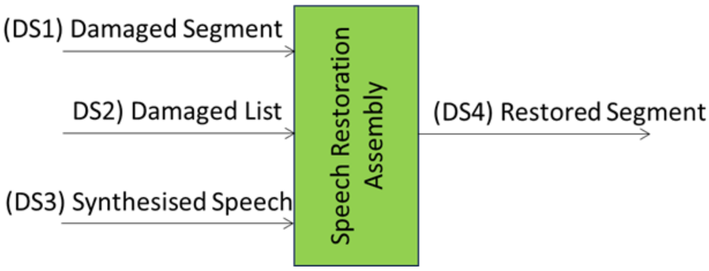


Figure 18 - Speech Restoration Assembly.

After the Assessment, Performance Assessor shall fill out *Table 46.*

Table 46 - Performance Assessment form of Speech Restoration System (SRS) Speech Restoration Assembly.

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form "CAE:SRS:V:P".
Name of AIM	Assembler
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Assessment Dataset	Unique Dataset Identifier assigned by MPAI Store.
AssessmentID	Unique Assessment Identifier assigned by Performance Assessor.
Actual output	Actual output provided as a matrix of *n* rows containing output assertions. For example: # Input tuple Final assertion 1 DS1[1], DS2[1], DS3[1] T/F … … … *n* DS1[*n*], DS2[*n*], DS3[*n*] T/F Final evaluation: T/F Legend: - #: Assembler input dataset tuple number.

	- **DS1**: Damaged Segment (within which damaged sub-segments may be listed) - **DS2**: Damaged List (of damaged sub-segments within current Damaged Segment) - **DS3**: Synthesised Speech (list of synthesised sub-segments corresponding to damaged sub-segments of DL) - **Final assertion**: T if Restored Segment is well-formed (single audio file without audible interruptions or gaps, produced without error messages or breaks), else F - **Final evaluation**: T if all **Final assertions** are T, else F
Execution time*	Duration of Assessment execution.
Assessment comment*	Comments on Assessment results and possible needed actions.
Assessment Date	yyyy/mm/dd.

** Optional field*

7.1.20 Tape Audio Restoration

7.1.20.1 Function

1. Detects and corrects speed, equalisation and reading backwards errors in Preservation Audio File.
2. Sends Restored Audio Files and Editing List to Packaging for Audio Recording.

7.1.20.2 Reference Model

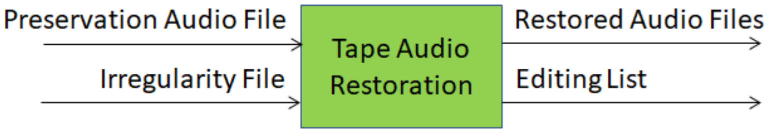


Figure 1 - Reference Model of Tape Audio Restoration

7.1.20.3 Input/Output Data

Input data	Semantics
Irregularity File	Irregularity File produced by Tape Irregularity Classification.
Preservation Audio File	The input Audio File containing the digitised version of an audio open-reel tape.
Output data	**Semantics**
Editing List	A JSON file whose syntax and semantics is specified by CAE-ARP.
Restored Audio Files	Audio Files obtained by restoring the Preservation Audio File per Editing List.

7.1.20.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/TapeAudioRestoration.json

7.1.20.5 Reference Software

The CAE-TAR Reference Software can be downloaded from the MPAI Git.

7.1.20.6 Conformance Testing

Receives	Irregularity File	Shall validate against the Irregularity File schema.
	Preservation Audio File	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
Produces	Editing List	Shall validate against the Editing List schema.
	Restored Audio Files	Shall validate against the Audio Object schema. The Qualifier shall validate against the AudioQualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.

7.1.20.7 Performance Assessment

Table 32 gives the Audio Recording Preservation (ARP) *Tape Audio Restoration* Means and how they are used.

Table 32 – AIM Means and use of Audio Recording Preservation (ARP) Tape Audio Restoration.

Means	**Actions**
Performance Assessment Dataset	DS1: *n* Preservation Audio Files created with (i) SB, (ii) SSV or (iii) ESV errors. These kinds of errors can occur singularly or superimposed one over another. DS2: *n* Irregularity Files related to DS1 coming from Tape Irregularity Classifier. DS3: *n* Restored Audio Files arrays containing the corrected files generated from DS1 with the information contained in DS2. DS4: *n* Editing Lists JSONs containing the edits made in DS3.
Procedure	1. Feed Tape Audio Restoration under Assessment with DS1 and DS2. 2. Compare the output Editing Lists with DS4. 3. Compare the samples of output Restored Audio Files with DS3 for SB and SSV correction evaluation. 4. Compare the samples and the spectral content of output Restored Audio Files with DS3 for ESV correction evaluation.
Evaluation	1. Verify the conditions: a. The Editing Lists are syntactically correct and conforming to the JSON schema provided in CAE Technical Specification. b. The output Editing Lists contain the same edits listed in DS4. c. All output Restored Audio Files are conforming to RF64 file format [7]. 2. Whenever SB or SSV corrections are required, each file of the output Restored Audio Files array: a. Shall be of the same duration of the corresponding file in DS3.

	b. Shall present the maximum value of the Cross-Correlation function for . Considering the Cross-Correlation definition in **Error! Reference source not found.**, is the Restored Audio File under evaluation and is the corresponding file in DS3. 3. Whenever an ESV correction is <u>required</u>, each file of the output Restored Audio Files shall have: a. Time domain samples with RMSE < *0.1*A*. Considering the RMSE definition in **Error! Reference source not found.**, is the Restored Audio File under evaluation and is the corresponding file in DS3. Where A is the Amplitude of the signal from DS3. b. for in [20, 20k] Hz, where is the PSD of the AIM (Tape Audio Restoration) under Assessment, is the PSD of the corresponding file in DS3 and is the PSD of the corresponding Preservation Audio File Audio Segment. 4. If, for any of the *n* input tuples, the above conditions are not satisfied, the Tape Audio Restoration module under Assessment does not pass the Performance Assessment.

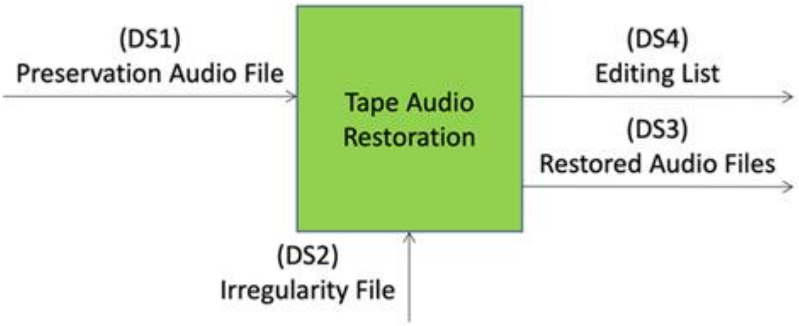


Figure 13 – Tape Audio Restoration.

After the Assessment, Performance Assessor shall fill out Table 33.

Table 33 – Performance Assessment form of Audio Recording Preservation (ARP) Tape Audio Restoration.

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE-ARP-V2.4”.
Name of AIM	Tape Audio Restoration
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Assessment Dataset	Unique Dataset Identifier assigned by MPAI Store.
Assessment ID	Unique Assessment Identifier assigned by Performance Assessor.

<table>
<tr><td><b>Actual output</b></td><td>Actual output provided as a matrix of <i>n</i> rows containing output assertions.
For example:
<table>
<tr><td>#</td><td>Irr</td><td>Editing List errors</td><td>Duration</td><td>Xcorr</td><td>RMSE</td><td>PSD</td><td>Final assertion</td></tr>
<tr><td>1</td><td>SB, ESV, SSV</td><td>0, 1, 2…</td><td>T/F</td><td>T/F</td><td>T/F</td><td>T/F</td><td>T/F</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td><i>n</i></td><td>SB, SSV</td><td>0, 1, 2…</td><td>T/F</td><td>T/F</td><td>-</td><td>-</td><td>T/F</td></tr>
</table>
Final evaluation: T/F
Legend:
- #: CT dataset tuple number.
- <b>Irr</b>: Irregularity types present on the Preservation Audio File
- <b>Editing List errors</b>: number of edits incorrectly performed. It has negative impact if different from 0
- <b>Duration</b>: flag to check the Restored Audio Files duration
- <b>Xcorr</b>: flag to check the cross-correlation measures
- <b>RMSE</b>: flag to check the RMSE measures (only for ESV)
- <b>PSD</b>: flag to check the PSD measures (only for ESV)
- <b>Final assertion</b>: AND operation between previous results</td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of Assessment execution.</td></tr>
<tr><td><b>Assessment comment*</b></td><td>Comments on Assessment results and possible needed actions.</td></tr>
<tr><td><b>Assessment Date</b></td><td>yyyy/mm/dd.</td></tr>
</table>

** Optional field*

7.1.21 Tape Irregularity Classification

7.1.21.1 Function

1. Receives Irregularity File (Audio) and Audio Files from Audio Analysis for Recording.
2. Receives Irregularity File (Video) and Irregularity Images from Video Analysis for Recording.
3. Classifies and selects the relevant Irregularities of the Preservation Audio-Visual File and Preservation Audio File.
4. Sends the Irregularity File related to the selected Irregularities to Tape Audio Restoration.
5. Sends the Irregularity Files related to the selected Irregularities and the corresponding Irregularity Images to Packaging for Audio Recording.

7.1.21.2 Reference Model

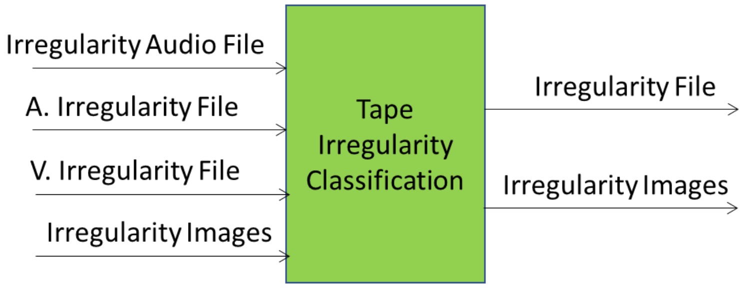


Figure 1 - Reference Model of Tape Irregularity Classification

7.1.21.3 Input/Output Data

Input data	Semantics
Irregularity Audio File	Audio Segments corresponding to Irregularities of the Preservation Audio File.
Audio Irregularity File	A file containing all Irregularities detected by Audio Analysis for Recording and Audio Analysis for Recording.
Video Irregularity File	A file containing all Irregularities detected by Video Analysis for Recording and Audio Analysis for Recording.
Irregularity Images	Images corresponding to Irregularities of the Preservation Video File.
Output data	**Semantics**
Irregularity File	Irregularity File produced by Tape Irregularity Classifier sent to Tape Audio Restoration.
Irregularity Images	Irregularity Images produced by Video Analysis for Recording.

7.1.21.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/data/TapeIrregularityClassification.json

7.1.21.5 Reference Software

The CAE-TICReference Software can be downloaded from the MPAI Git.

7.1.21.6 Conformance Testing

Receives	Irregularity Audio File	Shall validate against the Audio Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio-Visual Object Qualifier schema.
	Audio Irregularity File	Shall validate against the Irregularity File schema.
	Video Irregularity File	Shall validate against the Irregularity File schema.
	Irregularity Images	Shall validate against the Visual Object schema. The Qualifier shall validate against the Visual Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Visual Object Qualifier schema.
Produces	Irregularity File	Shall validate against the Irregularity File schema.
	Irregularity Images	Shall validate against the Visual Object schema. The Qualifier shall validate against the Visual Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Visual Object Qualifier schema.

7.1.21.7 Performance Assessment

Table 29 gives the Audio Recording Preservation (ARP) *Tape Irregularity Classifier* Means and how they are used.

Table 29 – AIM Means and use of Audio Recording Preservation (ARP) Tape Irregularity Classifier.

Means	Actions
Performance Assessment Dataset	DS1: *n* Irregularity Files from Audio Analyser. DS2: *n* Audio Files related to DS1. DS3: *n* Irregularity Files from Video Analyser. DS4: *n* Irregularity Images related to DS3. DS5: *n* output Irregularity Files in the format of port IrregularityFileOutput_1, containing correctly classified Irregularities. DS6: *n* output Irregularity Files in the format of port IrregularityFileOutput_2, containing correctly classified Irregularities.
Procedure	1. Feed Tape Irregularity Classifier under Assessment with DS1, DS2, DS3 and DS4. 2. Analyse the Irregularity Files resulting from port IrregularityFileOutput_1. 3. Analyse the Irregularity Files resulting from port IrregularityFileOutput_2.
Evaluation	1. Verify the conditions: a. The Irregularity Files are syntactically correct and conforming to the JSON schema provided in CAE Technical Specification. b. The Irregularity Files resulting from port IrregularityFileOutput_1 contain only Irregularities of interest for the Tape Audio Restoration (i.e., Irregularities with IrregularityType SSV, ESV or SB). c. All output Irregularity Images are conforming to the JPEG standard [8]. d. For each of the *n* tuples of input records, the output Irregularity Images are equal to the input Irregularity Images corresponding to the Time Labels indicated in the Irregularity Files coming from port IrregularityFileOutput_2. 2. By inspecting the Irregularity Files resulting from port IrregularityFileOutput_1, for each of the *n* tuples of input records, compute the values of Recall (*R*) and Precision (*P*) for each of the 13 labels of IrregularityType defined in Tables 17 and 18 of [3]. 3. For each label *l* of IrregularityType, compute the average value of Recall ( ) and Precision ( ) measures obtained at point 2. 4. Compute the *average value* of Recall ( ) and Precision ( ) measures obtained at point 3. 5. Accept the AIM under Assessment if: a. R'>0.9 b. P'>0.9

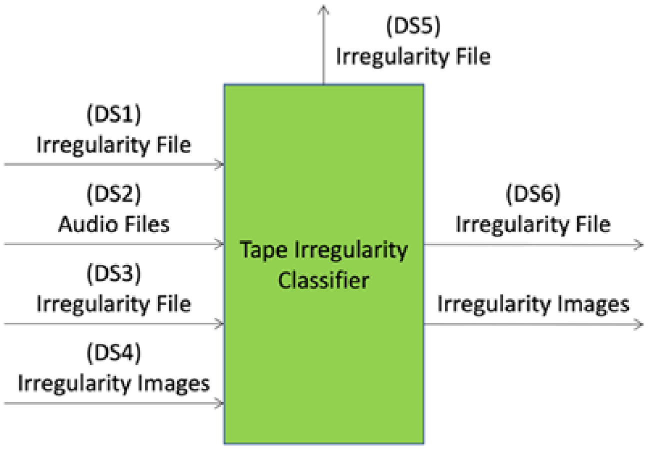


Figure 12 – Tape Irregularity Classifier.

After the Assessment, Perormance Assessor shall fill out Table 30.

Table 30 – Performance Assessment form of Audio Recording Preservation (ARP) Tape Irregularity Classifier.

<table>
<tr><td><b>Performance Assessor ID</b></td><td>Unique Performance Assessor Identifier assigned by MPAI</td></tr>
<tr><td><b>Standard, Use Case ID and Version</b></td><td>Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE-ARP-V2.4”.</td></tr>
<tr><td><b>Name of AIM</b></td><td>Tape Irregularity Classifier</td></tr>
<tr><td><b>Implementer ID</b></td><td>Unique Implementer Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>AIM Implementation Version</b></td><td>Unique Implementation Identifier assigned by Implementer.</td></tr>
<tr><td><b>Neural Network Version*</b></td><td>Unique Neural Network Identifier assigned by Implementer.</td></tr>
<tr><td><b>Identifier of Performance Assessment Dataset</b></td><td>Unique Dataset Identifier assigned by MPAI Store.</td></tr>
<tr><td><b>Assessment ID</b></td><td>Unique Assessment Identifier assigned by Performance Assessor.</td></tr>
<tr><td><b>Actual output</b></td><td>Actual output provided as a matrix of <i>n</i> rows containing and values.
<table>
<tr><td>Tuple #</td><td>Label</td><td>R</td><td>P</td></tr>
<tr><td>1</td><td>SP</td><td>Measure 1</td><td>Measure 1</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td>1</td><td>SB</td><td>Measure</td><td>Measure</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td><i>n</i></td><td>SP</td><td>Measure</td><td>Measure</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td></tr>
<tr><td><i>n</i></td><td>SB</td><td>Measure</td><td>Measure</td></tr>
</table></td></tr>
<tr><td><b>Execution time*</b></td><td>Duration of Assessment execution.</td></tr>
<tr><td><b>Assessment comment*</b></td><td>-</td></tr>
</table>

Assessment Date	yyyy/mm/dd.

* *Optional field*

7.1.22 Video Analysis for Preservation

7.1.22.1 Function

1. Detects and enters the Video Irregularities of the Preservation Audio-Visual File in the Video Irregularity File.
2. Sends Video Irregularity File to and receives Audio Irregularity Files from Audio Analysis for Preservation.
3. Extracts the Images corresponding to the Irregularities of both Irregularity Files.
4. Sends the Irregularity merged from the Audio and Video Irregularity Files to Tape Irregularity Classification with the corresponding Video Files.

7.1.22.2 Reference Model

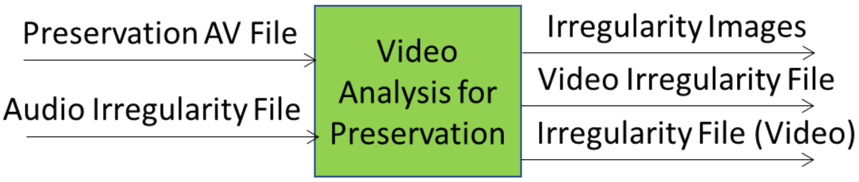


Figure 1 – Video Analysis for Preservation AIM

7.1.22.3 Input/Output Data

Input data	**Semantics**
Preservation Audio-Visual File	The input Audio File resulting from the digitisation of an audio open-reel tape to be preserved and, in case, restored.
Audio Irregularity File	A JSON file containing information about the Irregularities of the Preservation Audio File received from Audio Analysis for Preservation.
Video Irregularity File	A JSON file containing information about the Irregularities of the Preservation Audio-Visual File received from Video Analysis for Preservation.
Output data	**Semantics**
Video Irregularity File	A JSON file containing information about the Irregularities of the Preservation Audio-Visual File received from Video Analysis for Preservation.
Irregularity Images	Images corresponding to the Irregularities received or detected

7.1.22.4 JSON Metadata

https://schemas.mpai.community/CAE1/V2.4/AIMs/VideoAnalysisForPreservation.json

7.1.22.5 Reference Software

The CAE-VAP Reference Software can be downloaded from the MPAI Git.

7.1.22.6 Conformance Testing

Receives	Preservation Audio-Visual File	Shall validate against the Audio-Visual Object schema. The Qualifier shall validate against the Audio-Visual Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio-Visual Object Qualifier schema.
	Audio Irregularity File	Shall validate against the Visual Object schema. The Qualifier shall validate against the Audio Qualifier schema. The values of any Sub-Type, Format, and Attribute of the Qualifier shall correspond with the Sub-Type, Format, and Attributes of the Audio Object Qualifier schema.
	Video Irregularity File	Shall validate against the Irregularity File schema.
Produces	Video Irregularity File	Shall validate against the Irregularity File schema.
	Irregularity Images	Shall validate against the Irregularity File schema.

7.1.22.7 Performance Assessment

Table 26 gives the Audio Recording Preservation (ARP) *Video Analyser* Means and how they are used.

Table 26 – AIM Means and use of Audio Recording Preservation (ARP) Video Analyser.

Means	Actions
Performance Assessment Dataset	DS1: *n* Preservation Audio-Visual Files. DS2: *n* Irregularity Files related to the Preservation Audio File related to DS1. DS3: *n* output Irregularity Files in the format of port IrregularityFileOutput_1 with all Irregularities correctly identified. DS4: *n* output Irregularity Files in the format of port IrregularityFileOutput_2 with all Irregularities correctly identified and included from DS3.
Procedure	1. Feed Video Analyser under Assessment with DS1 and DS2. 2. Analyse the Irregularity Files resulting from port IrregularityFileOutput_1. 3. Analyse the Irregularity Files resulting from port IrregularityFileOutput_2.
Evaluation	1. Verify the conditions: a. The Irregularity Files are syntactically correct and conforming to the JSON schema provided in CAE Technical Specification.

	b. All Irregularities from DS2 are included in the Irregularity Files resulting from port IrregularityFileOutput_2. c. All output Irregularity Images are conforming to the JPEG standard [8]. d. For each of the *n* tuples of input records, the output Irregularity Images are extracted from the corresponding input Preservation Audio-Visual File at the Time Labels indicated in the Irregularity Files coming from port IrregularityFileOutput_2. 2. By inspecting the Irregularity Files resulting from port IrregularityFileOutput_1, for each of the *n* tuples of input records, compute the values of Recall (*R*) and Precision (*P*). 3. Compute the average value of Recall ( ) and Precision ( ) measures obtained at point 2. 4. Accept the AIM under Assessment if: a. R'>0.9 b. P'>0.9

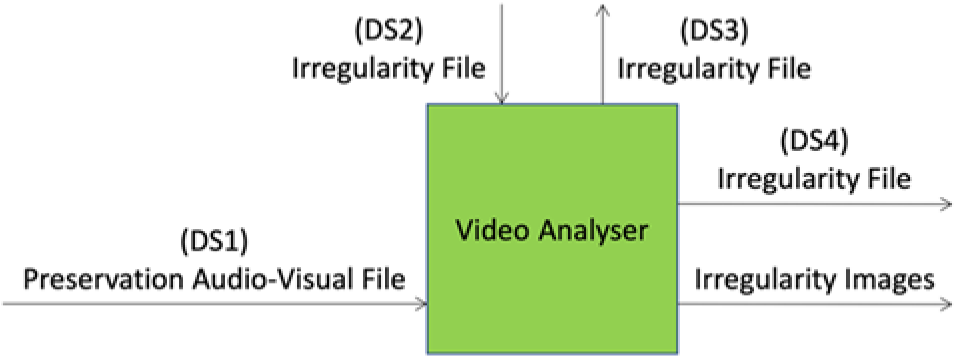


Figure 11 – Video Analyser.

After the Assessment, Performance Assessor shall fill out Table 27.

Table 27 – Performance Assessment form of Audio Recording Preservation (ARP) Video Analyser.

Performance Assessor ID	Unique Performance Assessor Identifier assigned by MPAI
Standard, Use Case ID and Version	Standard ID and Use Case ID, Version and Profile of the standard in the form “CAE-ARP-V2.4”.
Name of AIM	Video Analyser
Implementer ID	Unique Implementer Identifier assigned by MPAI Store.
AIM Implementation Version	Unique Implementation Identifier assigned by Implementer.
Neural Network Version*	Unique Neural Network Identifier assigned by Implementer.
Identifier of Performance Assessment Dataset	Unique Dataset Identifier assigned by MPAI Store.
Assessment ID	Unique Assessment Identifier assigned by Performance Assessor.
Actual output	Actual output provided as a matrix of *n* rows containing and values. Tuple # R P

	1 … *n*	Measure 1 … Measure *n*	Measure 1 … Measure *n*
Execution time*	Duration of Assessment execution.		
Assessment comment*	-		
Assessment Date	yyyy/mm/dd.		

* *Optional field*

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 CAE-USC V2.4 Reference Software Implementation, if in source code, is released with the BSD-3-Clause licence.
2. The purpose of this Reference Software is to provide a working Implementation of CAE-USC V2.4, 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 copyright holders. Users shall verify that they have the right to use any third-party software required by this Reference Software.

Note that at this stage CAE-USC V2.4 specifies Reference Software only for some AIMs.

7.3 Conformance Testing

An implementation of an AI Module conforms with CAE-USC V2.4 if it accepts as input _and_ produces as output Data and/or Data Objects (the combination of Data of a Data Type and its Qualifier) conforming with those specified by CAE-USC V2.4 .

The Conformance is expressed by one of the two statements

1. "Data conforms with the relevant (Non-MPAI) standard" – for Data.
2. "Data validates against the Data Type Schema" – for Data Object.

The latter statement implies that:

1. Any Sub-Type of the Data conforms with the relevant Sub-Type specification of the applicable Qualifier.
2. Any Content and Transport Format of the Data conform with the relevant Format specification of the applicable Qualifier.
3. Any Attribute of the Data
    1. Conforms with the relevant (Non-MPAI) standard – for Data, or
    2. Validates against the Data Type Schema – for Data Object.

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

Note that at this stage the CAE-USC V2.4 specifies Conformance Testing only for some AIMs.

7.4 Performance Assessment

Performance is an umbrella term used to describe a variety of attributes – some specific of the application domain the Implementation intends to address. Therefore, Performance Assessment Specifications provide methods and procedures to measure how well an AIW or an AIM performs its function. Performance of an Implementation includes methods and procedures for all or a subset of the following characteristics:

1. Quality – for instance, how well a Face Identity Recognition AIM recognises faces, how precise or error-free are the changes in a Visual Scene detected by a Visual Change Detection AIM, or how satisfactory are the responses provided by an Answer to Multimodal Question AIW.
2. Robustness – for instance, how robust is the operation of an implementation with respect to duration of operation, load scaling, etc.
3. Extensibility – for instance, the degree of confidence a user can have in an Implementation when it deals with data outside of its stated application scope.
4. Bias: – for instance, how dependent on specific features of the training data is the inference, as in Company Performance Prediction when the accuracy of the prediction may widely change based on the size or the geographic position of a Company; or face recognition in Television Media Analysis.
5. Legality – for instance, in which jurisdictions the use of an AIM or an AIW complies with a regulation, e.g., the European AI Act.
6. Ethics: may indicate the conformity of an AIM or AIW to a target ethical standard.

Note that at this stage the CAE-USC V2.4 specifies Performance Assessment only for some AIMs.

8 Data Types

8.1 Technical Specifications

Table 1 provides the full list of AI Modules (AIM) specified by CAE-USC V2.4 with links to the pages dedicated to Data Types. Each of these includes Definition, Functional Requirements, Syntax, Semantics, Conformance Testing, and Performance Assessment.
All AIMs specified by CAE-USC V2.3 are superseded by those specified by CAE-USC V2.4. AIMs specified by CAE-USC V2.4 may still be used if their version is explicitly indicated.

Table 1 - Data Types specified by CAE-USC V2.4

Acronym	Name	JSON	Acronym	Name	JSON
CAE-DLS	Damaged List	File	CAE-ELS	Editing List	File
CAE-IRR	Irregularity File	File	CAE-MAG	Microphone Array Geometry	File
CAE-PLN	Audio Source Model	File			

8.1.1 Damaged List

8.1.1.1 Definition

List of Text strings whose elements represent the content of the Damaged Segments (if any) requiring replacement with synthetic speech segments.

8.1.1.2 Functional Requirements

A Damaged List may be empty.

8.1.1.3 Syntax

https://schemas.mpai.community/CAE1/V2.4/data/DamagedList.json

8.1.1.4 Semantics

Label	Description
Header	Damaged List Header
- Standard-DamagedList	The characters “CAE-DLS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
DamagedListData[]	Data associated to Damaged List.
- DamagedSectionTime	Time of the start and end of the Damaged Segment.
DescrMetadata	Descriptive Metadata.

8.1.1.5 Conformance Testing

A Data instance Conforms with CAE-USC Damaged List (CAE-DLS) if:

1. Its JSON Object validates against its JSON Schema.
2. Any included JSON Object validates against its JSON Schema.
3. All Data in the JSON Object:
   1. Have the specified Data Types.
   2. Conform with the Qualifiers signaled in their JSON Schemas.

8.1.2 Editing List

8.1.2.1 Definition

The description of the speed, equalisation and reading backwards corrections occurred during the restoration process.

8.1.2.2 Syntax

https://schemas.mpai.community/CAE1/V2.4/data/EditingList.json

8.1.2.3 Semantics

Label	Description
Header	Editing List Header
- Standard-EditingList	The characters “CAE-ELS-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
EditingListData	Set of data related to Editing List.
OriginalSpeedStandard	Speed standard applied to the tape recorder during the digitisation of an open-reel tape. It can be one of the following values: 0.9375, 1.875, 3.75, 7.5, 15, 30. These values are in inch per seconds (ips). This field is optional.

OriginalEqualisationStandard	Equalisation standard applied to the tape recorder during the digitisation of an open-reel tape. It can be one of the following values: “IEC”, “IEC1”, “IEC2”. The notation refers to documents [16,17]. The association with OriginalSpeedStandard shall be compliant to the values indicated in [16,17]. This field is optional.
OriginalSamplingFrequency	UUID [3] that identifies a Restoration.
Restorations	List of restorations objects. Each object shall have at least the following fields: RestorationID, RestoredAudioFileURI, PreservationAudioFileStart, PreservationAudioFileEnd, AppliedSamplingFrequency, ReadingBackwards.
RestorationID	UUID [7] that identifies a Restoration.
PreservationAudioFileStart	Time Label indicating the instant of the Preservation Audio File when the restoration starts.
PreservationAudioFileEnd	Time Label indicating the instant of the Preservation Audio File when the restoration ends.
RestoredAudioFileURI	URI of a Restored Audio File.
ReadingBackwords	Boolean value indicating if the audio signal direction has been inverted during the restoration process.
AppliedSpeedStandard	Speed standard applied during the restoration process. It can be one of the following values: 0.9375, 1.875, 3.75, 7.5, 15, 30. These values are in inch per seconds (ips). This field is optional.
AppliedSamplingFrequency	Specifies the sampling frequency of the Restored Audio File. This field is mandatory.
AppliedEqualisationStandard	Equalisation standard applied during the restoration process. It can be one of the following values: “IEC”, “IEC1”, “IEC2”. The notation refers to documents [16,17]. The association with AppliedSpeedStandard shall be compliant to the values indicated in [16,17].

8.1.2.4 Conformance Testing

A Data instance Conforms with CAE-USC Editing List (CAE-ELS) if:

1. Its JSON Object validates against the CAE-ELS JSON Schema.
2. Any JSON Object included validates against it JSON Schema.
3. All Data in the JSON Objects:
    1. Have the specified Data Types.
    2. Conform with the Qualifiers signaled in their JSON Schemas.

8.1.3 Irregularity File

8.1.3.1 Definition

A file containing information about Irregularities of the Preservation Audio File and Audio-Visual Preservation File.

8.1.3.2 Syntax

https://schemas.mpai.community/CAE1/V2.4/data/IrregularityFile.json

8.1.3.3 Semantics

Label	Description
Header	Irregularity File Header
- Standard-IrregularityFile	The characters "CAE-IRF-V"
- Version	Major version – 1 or 2 characters
- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
IrregularityFileData	Data associated to Damaged List.
Offset	Integer value indicating the time offset (in milliseconds) between Preservation Audio File and Preservation Audio-Visual File. The time reference is the Preservation Audio File.
Irregularities	Array of Irregularities. Each Irregularity shall have at least an IrregularityID, TimeLabel and TimeReference.
IrregularityID	*UUID* [7] that identifies an Irregularity.
Source	"a": if the Irregularity is detected by the Audio Analyser. "v": if the Irregularity is detected by the Video Analyser. "b": if the Irregularity is detected by both Audio Analyser and Video Analyser.
TimeLabel	Time Label indicating the timing of an Irregularity. The time reference is the Preservation Audio File.
AudioFileURI	*URI* of the Audio File related to an Irregularity. It is only used in the message between Audio Analyser and Tape Irregularity Classifier.
IrregularityType	Class of an Irregularity (see values in following Tables).
IrregularityProperties	Optional object containing additional specifications about the current Irregularity.
ReadingSpeedStandard	Speed standard applied during the digitisation phase. It can be one of the following values: 0.9375, 1.875, 3.75, 7.5, 15, 30. These values are in inch per seconds (ips). This field is optional.
ReadingEqualisationStandard	Equalisation standard applied during the digitisation phase. It can be one of the following values: "IEC", "IEC1", "IEC2". The notation refers to documents [14,15]. The association with ReadingSpeedStandard shall be compliant to the values indicated in [14,15]. This field is optional.
WritingSpeedStandard	Speed standard applied during the recording phase. It can be one of the following values: 0.9375, 1.875, 3.75, 7.5, 15, 30. These values are in inch per seconds (ips). This field is optional.
WritingEqualisationStandard	Equalisation standard applied during the recording phase. It can be one of the following values: "IEC", "IEC1", "IEC2". The notation refers to documents [14,15].

	The association with WritingSpeedStandard shall be compliant to the values indicated in [14,15]. This field is optional.
ImageURI	*URI* of the Image related to an Irregularity. It is only used in the messages between Audio Analyser, Tape Irregularity Classifier, and Packager.

Table 1 – Extended list of Irregularities that can be detected by the Video Analyser

Code	*Name*	*Definition*
sp	**Splice**	Splice of magnetic tape to magnetic tape, or leader tape to magnetic tape (or vice versa).
b	**Brands on tape**	Most of the brands consist of the full name of the tape manufacturer, logo, or tape model codes. The brand changes in size, shape, and colour, depending on the tape used.
sot	**Start of tape**	It refers to what happens when the tape playback starts, at which point it is neither under tension nor in contact with the capstan and pinch roller. The distinguishing visual characteristic of this class is the tape coming in tension and in contact with the capstan and pinch roller. This happens at the beginning of the Preservation Audio-Visual File.
eot	**Ends of tape**	It refers to what happens when the tape reaches its end of playback, at which point it is neither under tension nor in contact with the capstan and pinch roller. The distinguishing visual characteristic of this class is the tape coming free or completely detached from the capstan. This happens at the end of the Preservation Audio-Visual File.
da	**Damaged tape**	It groups all kinds of damages on the surface of the tape and alterations of the tape shape. This class includes:
di	**Dirt**	Tape contamination and dirt: presence of mould, powder, crystals, other biological contaminations, or similar sullying.
m	**Marks**	Marks, signs or words written on the back of the tape (i.e., the nonmagnetic side) or on the adhesive tape of splices.
s	**Shadows**	The class contains frames in which shadows or reflections are temporarily cast on the tape by external objects in motion.
wf	**Wow and flutter**	Pitch variation due to the recording or playback equipment. If this effect is due to recording equipment it is detectable only on the Preservation Audio File and not on the Preservation Audio-Visual File.

Table 2 – List of Irregularities that can be detected only on the Preservation Audio File

Code	*Name*	*Definition*
pps	**Play, pause and stop**	Sound audio effects derived by play, pause or stop buttons during the recording. In a single tape several recordings from different sources can be recorded. This kind of irregularities cannot be identified in the digital video.

Code	*Name*	*Definition*
ssv	**Speed standard variation**	Instant when the recording has a variation of the speed (and, in case, of the equalization) standard.
esv	**Equalization standard variation**	Instant when the recording has a variation of the equalization standard without a change of the speed.
sb	**Signal backward**	Instant when a recording start playback audio signal backwards. This could happen in case of incorrect signal recording or digitization.

8.1.3.4 Conformance Testing

A Data instance Conforms with CAE-USC Irregularity File (CAE-IRF) if:

1. Its JSON Object validates against its JSON Schema.
2. Any included JSON Object validates against its JSON Schema.
3. All Data in the JSON Object:
    1. Have the specified Data Types.
    2. Conform with the Qualifiers signaled in their JSON Schemas.

8.1.4 Microphone Array Geometry

8.1.4.1 Definition

A Data Type describing the features of the microphone array and the individual microphones.

8.1.4.2 Functional Requirements

The Microphone Array Geometry Data Type includes:

1. The ID of the M-Instance.
2. The Microphone Array Geometry ID.
3. The Space-Time of the Microphone Array.
4. The Attributes of the Microphone Array, including:
    1. Array Type
    2. Array Scat
    3. URI of Array Filter coefficients for microphone array equalisation.
    4. Sampling features
    5. Block Size
    6. Number of Microphones
    7. Attributes of each microphone

8.1.4.3 Syntax

https://schemas.mpai.community/CAE1/V2.4/data/MicrophoneArrayGeometry.json

8.1.4.4 Semantics

Label	Description
Header	Microphone Array Geometry Header
- Standard - MicrophoneArrayGeometry	The "CAE-MAG-V" string
- Version	Major version – 1 or 2 characters

- Dot-separator	The character "."
- Subversion	Minor version – 1 or 2 characters
MicrophoneArrayGeometryID	Identifier of the Microphone Array Geometry.
MicrophoneArrayAttributes	Microphone feature data.
- MicrophoneArrayType	Type of microphone array arrangement.
- ArrayScat	Type of microphone array: 0:Rigid, 1:Open.
- CalibrationArrayFilterURI	URI of a local/remote file containing specific calibration filter.
- EqualisationArrayFilterURI	URI of a local/remote file containing specific equalisation filter matrix.
- SamplingParameters	Sampling frequency and sample precision.
- BlockSize	Minimum BlockSize: 256.
- NumberOfMicrophones	Number of Microphones in the Microphone Array.
MicrophoneAttributes[]	A list containing Microphone attributes.
- MicrophoneID	ID of the individual Microphone.
- ChannelCount	The number of Audio channels.
- MicrophonePointOfView	Position and Orientation of Microphone.
- MicrophoneDirectivity	The directivity pattern of the specific microphone.
DescrMetadata	Descriptive Metadata.

8.1.4.5 Conformance Testing

A Data instance Conforms with Microphone Array Geometry (CAE-MAG) if:

1. Its JSON Object validates against its JSON Schema.
2. Any JSON Object included validates against its JSON Schema.
3. All Data in the JSON Object:
    1. Have the specified Data Types.
    2. Conform with the Qualifiers signaled in their JSON Schemas.

8.1.5 Audio Source Model

8.1.5.1 Definition

A polynomial description of a simple acoustic source parameterised in terms of its direction with respect to the capture point.

8.1.5.2 Syntax

https://schemas.mpai.community/CAE1/V2.3/data/Polynomial.json

8.1.5.3 Semantics

Label	**Description**
Header	Polynomial Header

- Standard-Polynomial	The characters “CAE-PLN-V”
- Version	Major version – 1 or 2 characters
- Dot-separator	The character “.”
- Subversion	Minor version – 1 or 2 characters
PolynomialID	ID of this Polynomial instance.
PolynomialVariable	One of AngleSine, AngleCosine
Polynomial[]	Data associated to Polynomial.
- Coefficient	A Coefficient.
- Exponent	The exponent of the the Polynomial Variable multiplied by the coefficient.
DescrMetadata	Descriptive Metadata.

8.1.5.4 Conformance Testing

A Data instance Conforms with CAE-USC V2.4 Polynomial (CAE-PLN) if the JSON Data

1. Have the specified types.
2. Validate against the Polynomial JSON Schema.

8.2 Conformance testing

The Conformance a Data instance conforms with CAE-USC V2.4 is expressed by one of the two statements:

1. "Data conforms with the relevant (Non-MPAI) standard" – for Data.
2. "Data validates against the Data Type Schema" – for Data Object.

The latter statement implies that:
A Data instance Conforms with CAE-USC specified Data Type if:

1. Its JSON Object validates against its JSON Schema.
2. Any included JSON Object validates against its JSON Schema.
3. All Data in the JSON Object:
    1. Have the specified Data Types.
    2. Conform with the Qualifiers signaled in their JSON Schemas. For example, if the data claims to be UNICODE, it should conform with what the Text Qualifier (MPAI-TFA V1.4) defines as UNICODE.

Note that at this stage the CAE-USC V2.4 specifies Conformance Testing only for some Data Types.

8.3 Performance Assessment

Performance is an umbrella term used to describe a variety of attributes – some specific of the application domain served by a specific Data Type. Therefore, Performance Assessment Specifications provide methods and procedures to measure how well a Data instance represents an original Data entity. Performance of an Implementation includes methods and procedures for all or a subset of the following characteristics:

1. Quality– for example, how well a Scene Descriptors instance represent a scene.
2. Bias: – for example, how dependent on specific features of the training data is the inference represented by the Data instance.
3. Legality– for example, whether the Data instance was produced in a jurisdiction at a time by an AIM that complies with the relevant a regulation, e.g., the European AI Act.

4. Ethics – for example, the data instance complies to a target ethical standard.

Note that at this stage the CAE-USC V2.4 specifies Performance Assessment only of some Data Types.

9 Informative Examples

(Informative)

9.1 Audio Scene Geometry

An example of Audio Scene Geometry.

```
{
 "BlockIndex": 1,
 "BlockStart": 1631536788000,
 "BlockEnd": 1631536788063,
 "SpeechCount": 2,
 "SpeechList": [
        {
            "SpeechID": "09859d16-3c73-4bb0-9c74-91b451e34925",
            "ChannelID": 1,
            "AzimuthDirection": 90.0,
            "ElevationDirection": 30.0,
            "Distance": 2.0,
            "DistanceFlag": false
        },
        {
            "SpeechID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
            "ChannelID": 2,
            "AzimuthDirection": 180.0,
            "ElevationDirection": 30.0,
            "Distance": 1.27,
            "DistanceFlag": false
        }
 ],
 "SourceDetectionMask": [0,1]
}
```

9.2 Damaged List

An example of a damaged list JSON file:

```
{
    "DamagedSections": [
  {
    "SegmentStart": "00:00:01.351",
     "SegmentEnd": "00:01:55.654",
  },
   {
    "SegmentStart": "00:01:55.654",
    "SegmentEnd": "00:02:35.168",
   }
    ]
}
```

9.3 Editing List

Example of a complete Editing List with two elements: the first related to reading backwards error, whereas the second to speed and equalisation errors.

```
{
   "OriginalSpeedStandard": 15,
   "OriginalEqualisationStandard": "IEC1",
   "OriginalSampleFrequency": 96000,
   "Restorations": [{
     "RestorationID": "09859d16-3c73-4bb0-9c74-91b451e34925",
     "PreservationAudioFileStart": "00:00:00.000",
     "PreservationAudioFileEnd": "00:00:05.125",
     "RestoredAudioFileURI": "http://www.place_to_be_defined.com/restored_1",
     "ReadingBackwords": true,
     "AppliedSpeedStandard": 15,
     "AppliedSampleFrequency": 96000,
     "OriginalEqualisationStandard": "IEC1"
   },
   {
     "RestorationID": "3cdc2973-e95e-4125-acb7-121ad89067ef ",
     "PreservationAudioFileStart": "00:00:05.125",
     "PreservationAudioFileEnd": "00:00:15.230",
     "RestoredAudioFileURI": "http://www.place_to_be_defined.com/restored_2",
     "ReadingBackwords": false,
     "AppliedSpeedStandard": 7.5,
     "AppliedSampleFrequency": 48000,
     "OriginalEqualisationStandard": "IEC2"
   }]
}
```

9.4 Irregularity File

An example of Irregularity File from Audio Analyser to Video Analyser is:

```
{
   "Offset": 150,
   "Irregularities": [{
      "IrregularityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
      "Source": "a",
      "TimeLabel": "00:02:45.040"
   },{
      "IrregularityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
      "Source": "a",
      "TimeLabel": "00:04:89.020"
   }]
}
```

An example of Irregularity File from Video Analyser to Audio Analyser is:

```
{
   "Irregularities": [{
      "IrregularityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
      "Source": "v",
      "TimeLabel": "00:02:45.040"
```

```
  },{
    "IrregularityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
    "Source": "v",
    "TimeLabel": "00:04:89.020"
  }]
}
```

An example of Irregularity File from Audio Analyser to Tape Irregularity Classifier is:

```
{
  "Offset": 150,
  "Irregularities": [{
    "IrregularityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
    "Source": "a",
    "TimeLabel": "00:02:45.040",
    "AudioSegmentURI": "http://www.place_to_be_defined.com/audio_segment_1",
    "IrregularityType": "ssv",
    "IrregularityProperties: {
      "ReadingSpeedStandard": 15,
      "ReadingEqualisationStandard": "IEC1",
      "WritingSpeedStandard": 7.5,
      "WritingEqualisationStandard": "IEC2"
    }
  },{
    "IrregularityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
    "Source": "v",
    "TimeLabel": "00:04:89.020",
    "AudioSegmentURI": "http://www.place_to_be_defined.com/audio_segment_2"
  }]
}
```

An example of Irregularity File from Video Analyser to Tape Irregularity Classifier is:

```
{
  "Offset": 150,
  "Irregularities": [{
    "IrregularityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
    "Source": "a",
    "TimeLabel": "00:02:45.040",
    "ImageURI": "http://www.place_to_be_defined.com/image_1"
  },{
    "IrregularityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
    "Source": "v",
    "TimeLabel": "00:04:89.020",
    "ImageURI": "http://www.place_to_be_defined.com/image_2"
  }]
}
```

An example of Irregularity File from Tape Irregularity Classifier to Tape Audio Restoration is:

```
{
  "Irregularities": [{
    "IrregulatityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
    "Source": "a",
    "TimeLabel": "00:02:45.040",
    "IrregularityType": "ssv",
    "IrregularityProperties: {
        "ReadingSpeedStandard": 15,
        "ReadingEqualisationStandard": "IEC1",
        "WritingSpeedStandard": 7.5,
        "WritingEqualisationStandard": "IEC2"
    }
  },{
    "IrregulatityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
    "Source": "a",
    "TimeLabel": "00:04:89.020",
    "IrregularityType": "esv",
    "IrregularityProperties: {
        "ReadingSpeedStandard": 7.5,
        "ReadingEqualisationStandard": "IEC2",
        "WritingSpeedStandard": 7.5,
        "WritingEqualisationStandard": "IEC1"
    }
  }]
}
```

An example of Irregularity File from Tape Irregularity Classifier to Packager is:

```
{
    "Offset": 150,
    "Irregularities": [{
        "IrregulatityID": "09859d16-3c73-4bb0-9c74-91b451e34925",
        "Source": "v",
        "TimeLabel": "00:02:45.040",
        "IrregularityType": "sot",
        "ImageURI": "http://www.place_to_be_defined.com/image_1"
    },{
        "IrregulatityID": "3cdc2973-e95e-4125-acb7-121ad89067ef",
        "Source": "b",
        "TimeLabel": "00:04:89.020",
        "IrregularityType": "sp",
        "ImageURI": "http://www.place_to_be_defined.com/image_2"
    }]
}
```

9.5 Microphone Array Geometry

```
{
  "MicrophoneArrayType": 0,
  "MicrophoneArrayScat": 0,
  "MicrophoneArrayFilterURI": "https://mpai.community/standards/mpai-cae/",
  "SamplingRate": 4,
  "SampleType": 0,
  "BlockSize": 3,
  "NumberofMicrophones": 4,
  "MicrophoneList": [
          {
                "xCoord": 1.0,
                "yCoord": 2.0,
                "zCoord": 3.0,
                "directivity": 0,
                "micxLookCoord": 70.2,
                "micyLookCoord": 75.5,
                "miczLookCoord": 87.3
            },
            {
                "xCoord": 5.3,
                "yCoord": 5.6,
                "zCoord": 74.3,
                "directivity": 1,
                "micxLookCoord": 67.9,
                "micyLookCoord": 75.2,
                "miczLookCoord": 90.0
            },
            {
                "xCoord": 34.2,
                "yCoord": 65.2,
                "zCoord": 56.9,
                "directivity": 2,
                "micxLookCoord": 56.8,
                "micyLookCoord": 87.9,
                "miczLookCoord": 78.3
            },
            {
                "xCoord": 34.9,
                "yCoord": 29.7,
                "zCoord": 89.8,
                "directivity": 3,
                "micxLookCoord": 56.9,
                "micyLookCoord": 65.4,
                "miczLookCoord": 72.9
            }
    ],
    "MicrophoneArrayLookCoord": [{
      "xLookCoord": 56.0,
      "yLookCoord": 90.0,
      "zLookCoord": 86.3
    }]
}
```

9.6 Prosodic Speech Features

```
{
   "intonations": [{
      "pitch": 300,
      "intensity": 88.7,
      "duration”:100.0
   },{
      "pitch": 180,
      "intensity": 85.2,
    "duration”:98.0
  },{
    "pitch": 280,
    "intensity": 92.5,
    "duration”:92.0
  },{
    "pitch": 230,
    "intensity": 81.9,
    "duration”:98.0
  },{
    "pitch": 150,
    "intensity": 78.3,
    "duration”:98.0
  }],
  "unit": "phoneme"
}
```

9.7 Neural Speech Features

```
[
  1.456,
  5.1289,
  0.12,
  12345.54378,
  12389943.2837,
  58.29
]
```