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

|  |  |
| --- | --- |
| **Public document** | |
| N244 | 2021/05/12 |
| Source | Communication (Ethics) |
| Title | The governance of the MPAI ecosystem |
| Target | MPAI Members |

[1 Introduction 1](#_Toc71739831)

[2 The MPAI approach to AI systems 1](#_Toc71739832)

[3 The MPAI ecosystem 2](#_Toc71739833)

[4 The MPAI ecosystem’s infrastructure 3](#_Toc71739834)

[5 Performance 4](#_Toc71739835)

[6 Performance Testing and Identifiers 4](#_Toc71739836)

[7 Scope of Performance Testing 5](#_Toc71739837)

[Annex 1 6](#_Toc71739838)

[Annex 2 8](#_Toc71739839)

# Introduction

At its 4th General Assembly on 20 January 2021, MPAI approved the MPAI Manifesto (N130). The Manifesto identifies the main features of MPAI’s mission built on the principles laid down in the MPAI Statutes (N80).

The last paragraph of the Manifesto reads:

*Finally, although it is a technical body, MPAI is aware of the revolutionary impact AI will have on the future of human society. MPAI pledges to address ethical questions raised by its technical work with the involvement of high-profile external thinkers. The initial significant step is to enable the understanding of the inner working of complex AI systems.*

The paragraph stresses the fact that an Artificial Intelligence (AI) system – a system that uses AI to achieve goals that often could not be achieved or could only be achieved poorly with traditional technologies – can be tested for performance before it can be accepted by a user. This is important because, unlike other data processing-based standards, the performance of AI systems often depends on how the system has been trained. Ethical requirements indicate a range of non-technical requirem­ents that an AI system shall satisfy beyond those of technical performance.

Ethical is an attribute that many researchers are trying to attach to AI systems. This is an area that may or may not provide practical results if an AI system is taken as a black box.

MPAI’s objective is to achieve a usable solution by making assumptions about the internal structure of an AI system with levels of guarantee about the “ethical performance” of an AI system that implements an MPAI standard. In the following, such a system is called Implementation.

This is a Working Draft of a document that is proposed to be an MPAI standard.

Chapter 2 describes the MPAI approach and its basic elements. Chapter 3 defines the MPAI ecosystem and its components. Chapter 4 introduces Identifiers. Chapter 5 defines Performance. Chapter 6 specifies how and by whom Performance will be Tested. Chapter 7 acknowledges the current state of development of the subject being investigated. Annex 1 formalises the terminology used in this document and Annex 2 outlines a possible structure of Identifiers.

# The MPAI approach to AI systems

MPAI standards target components and systems enabled by data coding technologies, especially, but not necessarily, using AI. MPAI subdivides an Implementation of an MPAI-specified Use Case into functional components called AI Modules (AIM). AI systems implementing a Use Case and AIMs standardised by an MPAI standard are both called Implementations.

MPAI assumes Implementations use Artificial Intelligence (AI) or Machine Learning (ML) or traditional Data Processing (DP) or a combination of these. The implementation technologies can be hardware or software or mixed hardware and software.

An AI system implementing a Use Case is an aggregation of interconnec­ted AIMs executed inside an AI Frame­work (AIF). MPAI is developing and plans on releasing such an AI Framework (MPAI-AIF) standard in July 2021.

The 2 basic elements of MPAI standardisation are represented in *Figure 1* and *Figure 2*.

|  |  |
| --- | --- |
| Diagram  Description automatically generated | Diagram  Description automatically generated |
| *Figure 1 – The MPAI AI Module (AIM)* | *Figure 2 – The MPAI AI Framework (AIF)* |

*Figure 1* shows a video coming from a camera shooting a human face. The function of this AIM (green block) is to detect the emotion on the face and the meaning of the sentence the human is uttering. The AIM can be implemented with a neural network or with DP technologies. In the latter case, the AIM accesses a knowledge base external to the AIM.

MPAI standards typically include several Use Cases. A Use Case normatively defines

1. The input/output data to the AI system
2. The function of the Use Case
3. The topology and the interconnections of the AIMs
4. The functions of the AIMs
5. The input/output data of the AIMs.

An implementation of a generic Use Case is depicted *Figure 2*. The input data enter the Execution area of the AIF where the work­flow is executed under the supervision of Management and Control. AIMs communicate via the AIF’s Commun­ication and Storage infrastructure and may access static or slowly changing data sources (e.g., those of *Figure 1*). The result of the execution of the workflow is provided as output data.

# The MPAI ecosystem

The MPAI ecosystem is constituted by Implementers developing and Users utilising Implementations, and Implementations.

The 3 terms are defined as follows:

1. Implementers offer Implementations
2. Users utilise Implementations
3. Implementations are components, devices, applications and services realised according to one MPAI standard.

**Implementers** (professional market) make components (i.e., AIMs) and offer them to other Implem­enters who make devices or applications or services using acquired and internally-devel­oped AIMs.

**Users** (consumer market) utilise Implementations (devices, applications or services). Because of the peculiarity of AI systems, Users should have a level of guarantee of the performance of Implemen­tations.

An **Implementation** developed according to an MPAI Use Case, e.g., Conversation with Emotion, runs a workflow that implements interconnected AIMs. The standard containing the Use Case, i.e., Multimodal Communication, specifies the input and output data of all AIMs of the Use Case, the topology of the AIMs and their connections.

The table below introduces the 3 key elements – AIF, Use Case and AIMs – that enable the creation of a governable ecosystem.

|  |  |
| --- | --- |
| AIF | is specified by MPAI with its AIMs and the topology. The execution of a workflow should be protected, e.g., the AIF should not   1. maliciously peek into the data that are exchanged by the AIMs, 2. have other external connections with malicious purposes, such as stealing data.   The set of allowed connections is specified by the standard.  If the workflow is implemented according to the standard, the AIMs must be connected according to the topology specified in the standard and they cannot have external data sinks that are not in the standard. |
| Use Case | is specified by MPAI. The User should be able to know that an Implementation s/he is utilising implements an identified Use Case that is being run in the AIF. |
| AIMs | are specified by MPAI. A User should be able to know that the AIMs used to implem­ent the Use Case are Implementations of the AIMs specified by the Use Case that is being run. |

It might be useful to consider the following additional element.

|  |  |
| --- | --- |
| AI tool | is a tool, e.g., a neural network, that is used by an AIM. A User need not be granted visibility to the inside of the neural networks used by the AIMs. However, the User might wish to know what version of the neural network is currently being used in an AIM and contributes to producing the output data of the AIF. |

# The MPAI ecosystem’s infrastructure

The typical infrastructure enabling the ecosystem described in the preceding section is based on identifiers. The MPAI infrastructure enabling the MPAI ecosystem is based on Identifiers:

1. The AIF made by Implementer F has an Identifier from which it is possible to know:
   1. the identity of the Implementer
   2. the version of the MPAI-AIF standard and the Profile
   3. the version of the Implementation of the AIF
2. The Use Case is identified by
   1. the MPAI standard
   2. the Profile of the standard
   3. the version of the standard
3. The AIM made by Implementer M has an Identifier from which it is possible to know:
   1. the identity of the Implementer
   2. the version of the MPAI standard and the Profile
   3. the version of the Implementation
4. Given the assumptions, an Implementer of an AIM may decide to assign the identifier of an AI tool, on condition that it is unique to the AI tool.

Note that Users do NOT require Identification.

The AIF implementing the Use Case needs identifiers, e.g., in the case of a distributed system. Such private, identifiers are internal to the AIF and need not be exposed outside of the AIF.

The public Identifiers help recognise an entity as part of a system. They do not intend to be a constraint, rather, a freely chosen tool that allows an Implementation to be recognised as a member of the MPAI ecosystem. The following examples illustrate the case:

1. An Implementer can use an AIF or another environment to implement a Use Case. In the latter case the AIF is not part of the MPAI ecosystem.
2. An Implementer can implement Use Cases that may or may not be MPAI-defined. In the latter case, even if the AIF is part of the ecosystem, and some standard AIMs may be used, the Implementation is not part of the MPAI ecosystem.
3. An Implementer can use standard AIMs or non standard AIMs or AIM-unrelated technologies. In the second and third case, the Implementation is not part of the MPAI ecosystem.
4. An Implementer can decide to attach Identifiers or identifiers or no identifiers at all to the components and to the AI system. In the second and third case, the implementation is not part of the MPAI ecosystem.
5. An Implementer can assign its own internal identifiers to the components that implement the Use Case in a workflow.

MPAI will take measures to prevent implementers from attaching Identifiers to entities that do not belong to the MPAI ecosystem.

The bottom line is that Implementers have two choices. The first choice is to implement AI systems using the MPAI spec­ified technologies without or with private identifiers. The second choice is to make their implementations part of the MPAI ecosystem by attaching Identifiers that give levels of guarantee to Users of the Implementations.

It should be mentioned that IPR holders may decide to develop Framework Licences and issue licences that are conditional on implementations being Implementations.

# Performance

Implementations should have the attribute of Performance. MPAI uses the word Performance to indicate:

1. *Reliability*: the Implementation performs as specified by the standard, profile and version the Implementation refers to, e.g., within the application scope, stated limitations, and for the period of time specified by the Implementer.
2. *Robustness*: the ability of the Implementation to cope with data outside of the stated application scope with an estimated degree of confidence.
3. *Fairness*: the training set and/or network is open to testing for bias and unanticipated results so that the extent of applicability of the system can be assessed.
4. *Replicability*: the Performance of an Implementation as Tested by an entity can be replicated, within an agreed level, by another entity.

The four definitions are meant to apply to data outside of the training set.

# Performance Testing and Identifiers

Performance Testing is the assessment of the level of Performance of an Implementation. The level can be above or below a certain threshold. The result of Performance Testing needs not be yes/no, as Performance could very well have “levels”. Performance is specific to the application domain and defined in the context of the domain.

MPAI defines or approves the Means, i.e., tools, procedures, data sets, etc., to be used to Test the Performance of an Implementation.

Individual Implementers and Testing Laboratories, in the following called Testing Entities, can develop Performance Testing Means specific to AIMs and/or to complete Implementations of Use Cases. MPAI approves an entity as a permanent Testing Entity for a particular domain.

MPAI provides or indicates the Means that enable Testing Entities to Test the Performance of an AIM and of an implemented Use Case. However, MPAI does not Test the Performance of Implementations.

MPAI or its delegated Authorities assign name spaces to Testing Entities.

Testing Entities assign Identifiers after successfully Testing the Performance of Implementations using Testing Means.

Annex 2 presents initial ideas of the structure of Identifiers.

Individual Users are free to assess the Performance of an Implementation and post the result, e.g., to a reputation system.

# Scope of Performance Testing

In some cases MPAI will be able to specify Performance Testing Means for Implementations. However, MPAI cannot guarantee that a any standard AIM or Use Case can be Tested for Performance.

# Annex 1

**Terminology**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| AI Framework | The environment where AIM-based workflows are executed. |
| AI Module | The basic processing elements receiving processing-specific inputs and producing processing-specific outputs. |
| AI system | A system that uses AI to achieve goals that often could not be achieved or could only be poorly achieved with traditional technologies. |
| Applicability | The function of a Use Case or of an AIM as defined by the relevant standard |
| Component | An AIM or a tool (e.g., a neural network) used by the AIM |
| Ethical requirement | A non-technical requirement that an AI system shall satisfy beyond those of technical conformance. |
| Explainability | The ability to trace the output of an Implementation back to the inputs that have produced it. |
| Fairness | The attribute of an Implementation whose extent of applicability can be assessed by making the training set and/or network open to testing for bias and unanticipated results. |
| Identifier | A name that identifies any of the following   1. an Implementer 2. (a Component of) an Implementation 3. A standard, its profiles and its versions. |
| Implementation | An implementation of a Use case or an AIM whose Performance has been Tested to be above a level defined in the relevant Standard. |
| MPAI ecosystem | The ensemble of Implementers developing and Users utilising Implemen­tations, and Implementation. |
| Normativity | An applicable set of attributes of a technology or a set of technologies specified by an MPAI standard. |
| Performance | The attribute of an Implementation of being Reliable, Robust, Fair and Replicable. |
| Performance Testing | The assessment of the level of Performance of an Implementation. |
| Profile | A particular subset of the technologies that are used in a Use Case standard and, where applicable, the classes, other subsets, options and parameters relevant to that subset. |
| Registration Entity | An entity that assigns Identifiers. |
| Reliability | The attribute of an Implementation that performs as specified by the standard, profile and version the Implementation refers to, e.g., within the application scope, stated limitations, and for the period of time specified by the Implementer. |
| Replicability | The attribute of an Implementation whose Performance, as Tested by a Testing Entity, can be replicated, within an agreed level, by another Testing Entity. |
| Robustness | The attribute of an Implementation that copes with data outside of the stated application scope with an estimated degree of confidence. |
| Standard | A set of Use Cases belonging to an application domain normatively specified by MPAI along with the AIMs required to Implement the Use Cases. MPAI may develop other types of standards. |
| Testing Entity | An Implementer or a Testing Laboratory authorised by MPAI to Test the Performance of an Implementation in a given domain |
| Testing Laboratory | A laboratory accredited by MPAI to Test Implementations for Performance |
| Testing Means | Elements such as tools, procedures, data sets, etc., developed or approved by MPAI to be used when Testing the Performance of an Implementation. |
| Use Case | A particular instance of the application domain covered by a Standard identified as Normative. |
| Version | A revision or extension of a Standard. |

# Annex 2

**Initial ideas for the Identifier structure**

MPAI Identifiers could have the following structure

* Field to signify that the identifier is an Identifier, e.g., MPAI, followed by a field separator
* Field to signify the Testing Entity, e.g., a 16-bit sequential number assigned by MPAI to each Testing entity, followed by a field separator
* Field to signify Implementer, Implementation number and version, e.g., 16-bit sequential number assigned to each Implementer, followed by a 16-bit number managed by the Implementer, followed by a field separator
* Field to signify standard, profile and version, e.g., 16-bit sequential number assigned to each standard by MPAI, followed by a field separator
* Field used to check that the identifier is a valid Identifier

Workflow identifiers need not have an MPAI-defined structure.

MPAI could decide that implementers use the MPAI Identification infrastructure to identify non-MPAI conforming implementations by assigning portions of the name space to private use.