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

MPAI Technical Report

MPAI Metaverse Model (MPAI-MMM)
Functionality Profiles

V1

WARNING

Use of the technologies described in this Technical Report 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 Report.

Readers are invited to review Annex 3 - Notices and Disclaimers.

Technical Report
MPAI Metaverse Model – Functionality Profiles
V1

1 Introduction

Metaverse is a widely used term that conveys a still nebulous notion encompassing new forms of communication expected to create new jobs, opportunities, and experiences with transformational impacts on virtually all sectors of human interaction. A "metaverse" can be considered as a **communication and interaction system supporting digital environments containing digital objects.** A simple example of this definition is an audioconference system where human participants are represented by audio objects that a server mixes and distributes to all participants.

In general, a metaverse instance is viewed as a more complex communication environment with several additional features, such as synchronous and persistent experiences and virtual reality features such as avatars that may or may not be controlled by humans and objects of the real world.

Communication implies an agreement – a standard – between communicating parties. MPAI defines Metaverse Interoperability as the ability of metaverse instance #1 to use data from and as intended by a metaverse instance #2.

At present, achieving the interoperability defined above is difficult because:
1. There is no common understanding of what a metaverse is or should be, in detail.
2. There is an abundance of existing and potential metaverse use cases.
3. Some independently designed metaverse implementations are very successful.
4. Some important technologies enabling more advanced and even unforeseen forms of the metaverse may be uncovered in the next several years.

The MPAI Metaverse Model (MPAI-MMM) project has been established to deal with this unusually challenging situation by providing Technical Reports and Technical Specifications that can be applied to as many kinds of metaverse instances as possible and enable varied metaverse implementations to interoperate. The **MPAI roadmap to metaverse interoperability** identifies 6 milestones for each of which MPAI plans on publishing a Technical Report or Specification.

The **first milestone** reached by the project is based on the idea of collecting the *functionalities* that potential metaverse users expect a metaverse instance to provide, rather than trying to define what the metaverse is. The first Technical Report [1] includes definitions, assumptions guiding the MPAI-MMM project, potential sources of functionalities, an organised list of commented functionalities, and an analysis of some of the main technology areas underpinning the development of the metaverse.

Potential metaverse users with different needs might require different technologies to support such needs. Therefore, an approach that tried to achieve the goal of making every M-Instance be able to interoperate with every other M-Instance would force implementers to take technologies on board that are of no use to them and potentially costly.

Reference [1] addresses this difficulty. It proposes that metaverse standardisation be based on the notion of Profiles and Levels[1] successfully adopted by digital media standardisation for three decades. A metaverse standard that includes Profiles and Levels would enable metaverse developers to use only the technologies they need that are offered by the profile that is most suitable to them.

The notion of profile can mitigate the impact of having many disparate metaverse users with diverse requirements. Unfortunately, the implementation of that notion is not currently possible because some key technologies are not yet available and at this time it is unclear which technologies, existing or otherwise, will eventually be adopted [2]. This Technical Report, published as **the second milestone**, copes with this situation by considering **Functionality Profiles**, i.e., profiles that are defined by the functionalities they offer, not by the technologies implementing them. Functionality Profiles are not meant to fully address the interoperability problem, but rather to allow a **technology-independent** definition of profiles based on the functional value they provide rather than on the "influence" of specific technologies.

The structure of this Technical Report is the following:

Chapter 2	Extends the metaverse-relevant definitions of [1].
Chapter 3	Develops a functional operation model of a metaverse instance based on <u>Sources</u> requesting <u>Destinations</u> to perform <u>Actions</u> on <u>Items</u> both containing <u>Data Types</u>.
Chapter 4	Specifies the payloads of the **Actions** that Sources request Destinations to perform on **Items** and of the responses provided by Destinations to such requests.
Chapter 6	Specifies the Metadata of the Items <u>without specifying the Formats of the Data</u>.
Chapter 7	Specifies the **Data Types** used by requests and responses.
Chapter 8	Verifies the methodology developed by this Technical Report by applying it to some relevant **Use Cases.**
Chapter 9	Provides a first set of **Functionality Profiles and Levels**.

This *Technical Report – MPAI Metaverse Model (MPAI-MMM) – Functionality Profiles* has been developed by the Requirements Standing Committee. MPAI may decide to develop new versions of this document.

MPAI plans on releasing more documents of the MPAI-MMM project when reaching a milestone. The plan for the *new* milestones is as follows:

3. **Architecture**: Functional blocks, metaverse API, and the Items exchanged by the blocks.
4. **Data Formats**: Functional requirements of Items exchanged between functional blocks.
5. **Technology landscape**: Table of Contents of the Metaverse Technology Specifications as envisaged in [2].
6. **MPAI Technologies**: Mapping of MPAI Technologies relevant to the metaverse to the said Table of Contents.

MPAI expects that, by reaching milestone #4 it will be possible to implement non-interoperable metaverse instances and by reaching milestone #5 it will be possible to implement metaverse instances that rely on mediated interoperability, e.g., by accessing a format conversion service. Milestone #5 will provide a list of all technologies whose specification is required to achieve direct

[1] A Profile is sets of one or more base standards and, if applicable, chosen classes, subsets, options, and parameters of those standards that are necessary for accomplishing a particular function. A Level is a subdivision of a Profile indicating the completeness of the user experience.

interoperability. Milestone #6 will document the MPAI contributions to the metaverse technology specification.

2 Definitions

This Technical Report continues the convention adopted in [1]. Terms beginning with a capital letter have the meaning defined in Table 1 that updates and extends the terms defined in [1]. Terms beginning with a small letter have the meaning commonly defined for the context in which they are used. For instance, Table 1 defines *User* and *Object* but does not define *human* and *object*.

A dash "-" preceding a Term in Table 1 means the following:

1. If the font is normal, the Term in the table without a dash and preceding the one with a dash should be placed before that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Decentralised followed by one of the words Application, Autonomous Organisation, Finance, System, and User Identifier, or definitions belonging to the same class, e.g., Action and Items.
2. If the font is *italic*, the Term in the table without a dash and preceding the one with a dash should be placed after that Term. The notation is used to concentrate in one place all the Terms that are composed of, e.g., the word Interface preceded by one of the words Brain-Computer, Haptic, Speech, and Visual.

Table 1 – General Terms and Definitions

Terms	Definitions
Account	An Item that uniquely references a human who has Registered.
Action	An operation affecting an Item.
- Authenticate	The Action of requesting confirmation that an Entity MM-Embedded at an M-Location is what it claims to be.
- Author	The Action of calling a Service to obtain an Entity with associated OutRights to Act on the Entity.
- Change	The Action of requesting that a Service modify the Rights of a User whose Persona is Embedded at an M-Location and provide OutRights, e.g., to further Change the Rights.
- Discover	The Action of requesting that a Service provide a DiscoveryOut Item containing the IDs of the Items relevant to the DiscoverIn Item with the OutRights to Act on the DiscoveryOut Item.
- Hide	The Action of requesting that a Service make the ID of an Item unavailable and provide OutRights, e.g., to make the ID available again.
- Identify	The Action of requesting that a Service produce an Item from MM-Sent or UM-Sent Data and Metadata or update an Item with the OutRights to Act on the Item.
- UM-Import	The Action of a User requesting that a Service read Data & Metadata stored at an Address.
- Inform	The Action of requesting that a Service provide an InformOut Item containing information about an InItem, such as the Metadata of the InItem, with the OutRights to Act on the InformOut Item.

- Interpret	The Action of requesting that a Service provide an InterpretOut Item containing interpretation of an InItem, such as translation or extraction of Personal Status, with the OutRights to Act on the InterpretOut Item.
- MM-Add	The Action of requesting that a Service add an Entity at an M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Added Entity.
- MM-Animate	The Action of requesting that a Service change the features of an Entity MM-Embedded at an M-Location with a Stream and provide the OutRights to Act on the MM-Added Entity.
- MM-Capture	The Action of requesting that a Service MM-Send selected Entities MM-Embedded at an M-Location to a User.
- MM-Disable	The Action of requesting that a Service stop MM-Enabling selected Entities Embedded at an M-Location and provide OutRights to Act on the MM-Disabled Entities.
- MM-Embed	The Composite Action of requesting that a Service MM-Add and MM-Enable an Entity either located at a Service or at an M-Location at a destination M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Embedded Entity.
- MM-Enable	The Action of enabling a User to MM-Capture an Entity MM-Added at an M-Location.
- MM-Send	The Action of requesting that a Process forward to another Process: 1. An Item with OutRights to Act on the Item, or 2. Data/Metadata.
- MU-Actuate	The Action of requesting that a Device present at a U-Location as Media with a Spatial Attitude an Entity MM-Embedded at an M-Location.
- MU-Export	The Action of requesting that a Process store an Item at an Address.
- MU-Render	The Composite Action of requesting that: 1. A Service MM-Send selected Entities Embedded at an M-Location to a Device. 2. The Device MU-Actuate at a U-Location with a Spatial Attitude the Entity received.
- Post	The Action of requesting that a Marketplace include an Asset.
- Register	The Action of a human requesting that an M-Instance/Environment grant their Users the Rights to perform Actions in the M-Instance/Environment.
- Track	The Composite Action of requesting: 1. A Service to MM-Add a Persona at an M-Location with a Spatial Attitude. 2. A Service to UM-Animate the Persona MM-Added at an M-Location. 3. A Service to MU-Render specified Entities at the M-Location to a U-Location.
- Transact	The Action of a User1 requesting that a Service: 1. Assign Rights on an Asset to User2. 2. Cause: 2.1. Wallet1 of User1 to be increased by Value1. 2.2. Wallet2 of User2 to be decreased by Value2. 2.3. Wallet3 of the Service enabling/facilitating the Transaction to be increased by Value3 (optionally).
- UM-Animate	The Composite Action of requesting:

	1. To UM-Capture an animation stream extracted from a scene at a U-Location. 2. To UM-Send the animation stream and Metadata to a User. 3. To Identify the Animation Stream. 4. To MM-Animate the Model MM-Embedded at the M-Location using the Animation Stream.
- UM-Capture	The Action of requesting that a Device acquire Media from a scene at a U-Location.
- UM Import	The Action of a User requesting that a Service make available Data & Metadata, or an Item stored at an Address.
- UM-Render	The Composite Action of requesting: 1. A Device to UM-Capture a scene at U-Location. 2. The Device to MM-Send Data and Device-provided Metadata to a User. 3. A Service to Identify an Entity from UM-Sent Data and Metadata. 4. A Service to MM-Embed the Entity at an M-Location with a Spatial Attitude.
Avatar	A rendered Digital Human.
Blockchain	A shared immutable ledger stored on a peer-to-peer network of computers.
Common Metaverse Specifications	(CMS) The collection of standards specifying the technologies and recognised Technology Profiles enabling Metaverse Interoperability including.
Connected Autonomous Vehicle	(CAV) A vehicle able to autonomously reach a U-Location by using its own sensing and processing capabilities to generate an M-Instance, sharing the M-Instance with other CAVs and issuing actuation commands.
Data	Information represented in digital form.
- Format	The syntax and semantics of Data.
- Type	Data characterised by its specific Format.
Data Type	Data used in Actions and Items.
- Address	A URL.
- Amount	A number expressing a Value in a Currency.
- Cognitive State	The representation of a User's Personal Status that reflects the way they understand the Environment, such as “Confused”, “Dubious”, “Convinced”.
- Coordinates	A set of numbers representing a Position in a Metaverse Environment using a coordinate system.
- Currency	A medium of exchange enabling Transactions in a Metaverse Instance.
- Emotion	The representation of a User's Personal Status that results from their interaction with an Environment, such as “Angry”, “Sad”, “Determined”.
- Orientation	The set of the 3 roll, pitch, yaw angles indicating the rotation around the principal axis (x) of an Object, its y axis having an angle of 90° counter-clockwise (right-to-left) with the x axis and its z axis (pointing up toward the viewer viewing from above).
- Personal Status	The representation of the information internal to a User characterising their behaviour.
- Point	A point in an M-Environment identified by the set of local Coordinates.
- Point of View	The Spatial Attitude of a Digital Human watching the Environment.
- Position	The coordinates of an Object with respect to a set of coordinates in a Metaverse Environment.

- Social Attitude	The representation of a User's Personal Status related to the way the User intends to position vis-à-vis a Metaverse Environment, e.g., "Respectful", "Confrontational", "Soothing".
- Spatial Attitude	The Position and Orientation of an Entity, and their velocities and accelerations.
- Time	A measure of time.
Decentralised	
- Application	(dApp) A Process that runs on a decentralised computing system.
- Autonomous Organisation	(DAO) An organisation without centralised leadership, where the main governing rules are typically encoded by means of a Smart Contract.
- Finance	(DeFi) A financial technology based on a secure infrastructure of distributed ledgers like those used by crypto currencies.
- System	A set of dApps enabling a group of Users to make decisions without a centralised entity.
Device	Equipment enabling: - A U-Environment to interact with an M-Instance and/or - An M-Instance to interact with a U-Environment.
Duty	A moral or legal obligation to act or behave.
Entitlement	The state of a User having certain Rights in an M-Instance.
Functionality	An attribute of an M-Instance expected to be enabled by a Common Metaverse Specifications Tool.
Human	
- *Digital*	Either a Digitised or a Virtual Human.
- *Digitised*	The digital representation of a human.
- *Virtual*	A computer-created Object that has a human appearance when rendered but is not a Digitised Human.
Governance	The action or manner of directing and controlling actors of the Metaverse Ecosystem.
Information and Communication Technologies	(ICT) Technologies that enable the processing and distribution of information via the network.
Interface	A communication pathway enabling systems to interact.
- *Brain-Computer*	(BCI) A communication pathway that allows a human to interact with an M-Instance by sensing and processing the electrical activity of the brain.
- *Haptic*	A communication pathway that allows a human to interact with an M-Instance through bodily movements and sensations.
- *Speech*	A communication pathway that allows a human to interact with an M-Instance using spoken language.
- *Visual*	A communication pathway that allows a human to interact with an M-Instance through bodily movements and visual messages.
Item	Data and Metadata recognised and identified by an M-Instance.
- Account	An Item that uniquely references a human who has Registered. A User may have more than one Account with one or more Services.
- Activity Data	An Item containing the record of the Actions of a User.
- Asset	An Item Embedded at an M-Location or Posted to a Service that may be the object of a Transaction.
- DiscoverIn	An Item containing a description of the Items to be Discovered.
- DiscoverOut	An Item containing the description of the Items Discovered.

-	Entity	Any of the following Items that can be MU-Rendered: Object, Model, Scene, Event, and Experience.
-	Event	An Entity corresponding to an M-Location, its Entities and their Animations starting from Start Time until End Time.
-	Experience	An Entity comprising an Event as MM-Captured by a User and the User Interactions with the Entities of the Event.
-	Identifier	An Item that uniquely references an Item. The Item can have more than one Identifier.
-	InformIn	An Item containing a description of the Item on which information is requested.
-	InformOut	An Item containing the description of the Item object of an InformIn.
-	Interaction	An Item containing the list of Actions made by a User on the Entities at an M-Locations and the corresponding Times.
-	InterpretIn	An Item containing a description of the Item to be Interpreted.
-	InterpretOut	An Item containing the description of the Item object of an InterpretIn.
-	Ledger	An Item containing a list of Transactions involving Assets.
-	Map	An Item containing a structure establishing a correspondence between U-Locations with M-Locations.
-	M-Environment	An identifiable portion of an M-Instance covered by an Account.
-	Message	An Item containing application-specific Data MM-Sent by a Source to a Destination.
-	M-Instance	A Metaverse implementation.
-	M-Location	An identifiable delimited portion of an M-Environment.
-	Model	An Object representing an object with its features ready to be UM-Animated by a Stream or MM-Animated.
-	Object	An Entity representing an object. Currently, the following types of Objects are supported: Audio, Visual, and Haptic.
-	Persona	A Model of a human ready.
-	Personal Profile	An Item containing the Data about the human represented by User.
-	Provenance	The Ledger associated with a specific Asset.
-	Request-Action	An Item containing the request to a Service to perform an Action. This definition applies to all Actions of this Technical Report.
-	Response-Action	An Item containing the response of a Service to a Request-Action. This definition applies to all Actions of this Technical Report.
-	Rights	An Item expressing the ability of a User to perform an Action on an Item until a Time.
-	Rules	An Item expressing the terms and conditions under which a User operates in an M-Instance/Environment.
-	Scene	A possibly hierarchical Composition of Objects each having a Spatial Attitude.
-	Social Graph	A representation of a User's network of connections with Items, Processes, and Services.
-	Stream	An Item made by a continuous flow of Data.
-	Transaction	Item representing the changed state of the Accounts and the Rights of a seller User and a buyer User on an Asset and optionally of the Service facilitating/enabling the Transaction: 1. The Value moving into the Wallet of User 1 (seller).

	2. The Value moved from the Wallet of User2 (buyer). 3. The Value moved into the Wallet of User 3 (service) - optional. 4. The Time the Values were moved. 5. The Rights to Act owned by User1 before Time. The Rights to Act owned by User2 after Time.
- U-Environment	A portion of the Universe.
- U-Location	An identifiable delimited portion of a U-Environment.
- User Data	An Item containing Activity Data, Personae, Social Graph, and User Profile of a User.
- Value	An Amount and the Currency with which the Amount is expressed.
- Wallet	A container of Currency units. In general, a Wallet is implemented outside of the Environment.
Level	A subdivision of a Profile indicating the completeness of the user experience provided by the Profile.
Metadata	An attribute of Data, e.g., of a User, an Environment, an Object, or a Service.
Metaverse	A set of services and devices implementing a subset of the following functions: 1) sense data from real environments, 2) process sensed data 3) create one or more digital environments populated by digitised and/or virtual objects, and 4) process the objects to 5) produce effects on real and/or virtual environments that 6) are consistent with the goal set to the metaverse, and achieved 7) with the capabilities of 8) and within the rules set to the metaverse.
- Action	An operation affecting an Item.
- Actuator	A component of a Device able to MU-Render an Entity and environment Data to a U-Environment.
- Asset	An Item Embedded at an M-Location or Posted to a Service that may be the object of a Transaction.
- Ecosystem	The ensemble of entities and rules ensuring that Metaverse Instances operate in the interest of Metaverse Stakeholders.
- Enabling Service Layer	The set of Services such as payment, security, identity, privacy, etc. that enable operation of an M-Instance.
- Entity	Any of the following Items that can be MU-Rendered: Scene, Object, Model, Event, and Experience.
- Environment	(M-Environment) A portion of an M-Instance run by a Registration system.
- Experience Layer	The set of functions, such as Devices, that generate Experiences.
- Functionality	The ability of an M-Instance to perform Actions that further the goals set by the Metaverse Manager for the M-Instance.
- Industry	The collection of players that support the design, development, deployment, operation, and content and service provisioning to Metaverse Instances.
- Instance	(M-Instance) An implementation providing all or a subset of the Metaverse Functionalities.
- Interoperability	The ability of M-Instance #1 to use data from and as intended by M-Instance #2. Interoperability can be Direct or Mediated by a conversion service.
- Infrastructure Layer	The set of functions such as network, transport, storage, and (cloud, edge) processing that enable an M-Instance to operate.
- Item	Metaverse-specific Data that may include Metadata that may include Rights.

- Level	A subdivision of a Profile that indicates the degree of completeness of the user experience provided by that Level.
- Location	(M-Location) An identifiable delimited portion of the Metaverse.
- Manager	The entity overseeing the operation of an M-Instance.
- Operator	The entity overseeing the operation of an M-Environment.
- Partner	A User participating in activities of a Metaverse Operator (i.e., a business customer of an Operator)
- Platform Layer	The set of Services, such as content creation, content discovery, and content access functions that enable an M-Instance to operate.
- Process	The instance of a program being executed.
- Profile	A recognised subset of the Functionalities (Functionality Profile) or Technologies (Technology Profile) specified by the Common Metaverse Specifications.
- Sensor	A Device able to UM-Capture a scene and other environment information as Data.
- Specifications	(CMS) A collection of standards specifying the technologies enabling Metaverse Interoperability.
- Stakeholder	An entity performing a function aimed at achieving a goal in an M-Instance.
- State	The set of values and stored data of an M-Instance at a given time.
- Tool	A Technology or group of Technologies enabling an M-Instance to provide a Functionality.
- Technology	A structured application of scientific and/or technical methods that supports a Functionality.
- User	One or a set of Processes representing a human.
Object	
- *Audio*	The digital representation of an object or a computer-generated Object that can be rendered to and perceived by a human ear.
- *Autonomous*	A Virtual Object animated by a Process giving it the ability to act (e.g., move, speak, respond, execute) with a degree of autonomy.
- *Composite*	An Object that includes more than one Object Type.
- *Digital*	A Digitised or a Virtual Object.
- *Digitised*	The digital representation of an object.
- *Haptic*	An Object with the haptic features of an object able to be rendered to provide haptic sensations in a human.
- *Human*	An Object representing a human.
- *Speech*	The digital representation of a sound emitted by the vocal tract of a human or generated by a computer with similar audio characteristics.
- Type	One of Audio, Visual, Haptic, Olfaction, and Gustation.
- *Virtual*	A computer-generated Object that is not a Digitised Object.
- *Visual*	The digital representation of an object captured by an electromagnetic or high-frequency audio signal or computer-generated that can be rendered to and perceived by a human eye.
Oracle	A Service providing information from a U-Environment to a Blockchain.
Privacy	The Rights of a User to keep their Personal Data secret.
Process	
- App	An application-specific Program executed on a Device.
- Device	A Process able to: 1. UM-Capture Data from a U-Location

	2. UM-Send Data and Metadata to a User. and/or 1. MM-Send an Entity from an M-Location to the Device. 2. MU-Render an Entity at a U-Location.
- Service	A Process that can be called to provide Functionalities.
- User	A Process representing a UM-Captured human as a Persona that are either UM-Animated by a Stream or MM-Animated by an autonomous agent.
Profile	Set of base standards and/or their subsets.
- *Functionality*	The grouping of Functionalities offered by a Metaverse Profile.
- *Technology*	The grouping of Technologies offered by a Metaverse Profile.
Registration	The provisioning by a human of a subset of User Data to an M-Instance/Environment to obtain an Account.
Representation	Data that represent an entity of a U-Environment in an M-Instance.
Sense of	
- Agency	The subjective awareness of being able to decide, execute, and control one's own actions in an M-Environment.
- Embodiment	The engagement of senses to form a complete Environment Experience.
- Presence	The feeling of being in an M-Environment with other Digital Humans for real.
Smart Contract	A Program stored on a Blockchain that runs when activated by an external entity, e.g., a User or another Smart Contract.
Token	
- *Fungible*	A representation of an Asset that is interchangeable with other Assets of the same type.
- *Non-Fungible*	(NFT) A unique digital identifier of an Asset that: - Cannot be copied (i.e., a copy is known to be a copy), substituted, or subdivided. - Is recorded in a digital ledger. - Is used to certify Object authenticity and ownership.
Trust-less system	A system allowing a User to make reliable Transactions without trusting or knowing the parties the User makes Transactions with.
Universe	The physical world.
- Environment	(U-Environment) A delimited portion of the Universe.
- Location	(U-Location) An identifiable delimited portion of the Universe.
Use Case	An example of how an application domain can be supported by an M-Instance/Environment.
User Keys	The pair of public and private keys where the public key is used to encrypt, and the private key is used to both encrypt and decrypt Data.
User Identifier	
- *Decentralised*	An Identifier that enables the verifiable association with a User without requiring a centralised registry.
- *Self-Sovereign*	A Decentralised Identifier derived from the User's Public Key owned and managed directly by the User based on the knowledge of their own Private Key, e.g., stored in the Crypto Wallet enabled by the Blockchain underpinning the Metaverse Instance.
Wallet	

- *Crypto*	Software or hardware holding the Public and Private Keys of a User to enable them to make Transactions by accessing their Account on a Blockchain.

3 A functional operation model

3.1 Introduction

This Chapter illustrates the operation of Metaverse Instances or Environments by means of a walkthrough. The following should be noted:

1. The walkthrough defines and illustrates all the terms used in this Technical Report as they are introduced.
2. To the extent possible, the definition of a **Term** (indicated in bold with a capital first letter) is provided when it is introduced.
3. If a definition is slow in coming up because of the complexity of the walkthrough and the number of Terms introduced, the reader may rely on the common meaning of the term or access the definition in one of Table 1.
4. The walkthrough uses verbs called **Actions**, i.e., operations affecting an **Item**, i.e., Data and Metadata identified and recognised by a Metaverse Instance (**M-Instance**) using **Data Types**.
5. If a noun is defined, the corresponding verb may be introduced without engaging in an additional definition and vice-versa.
6. An Action may start from a Metaverse Location called **M-Location**, i.e., an identified delimited portion of a Metaverse Environment (**M-Environment**), i.e., a delimited portion of an M-Instance:
    6.1. From an M-Location and affect an M-Location in the same or different M-Instance.
    6.2. From an M-Location and affect a **U-Location**, i.e., an identifiable delimited portion of a **U-Environment**, a delimited portion of the **Universe**, i.e., the real world.
    6.3. From a U-Location and affect an M-Location.

    Accordingly, some Actions will be prefixed by **MM-**, **MU-**, or **UM-**.

3.2 M-Instances

Figure 1 depicts some of the main elements at the basis of this Technical Report.

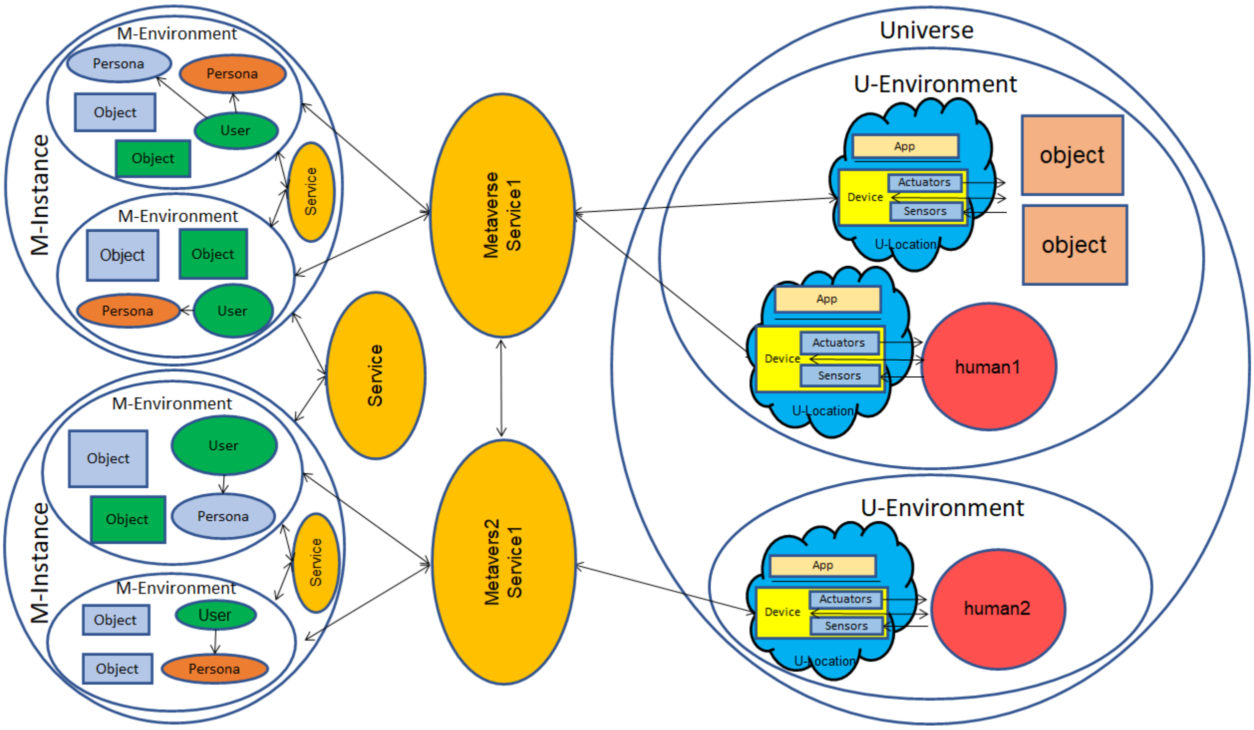

Figure 1 – An example of Metaverse Scenarios addressed by this Technical Report

The humans and objects in the U-Environments on the right-hand side of Figure 1 may be connected to one or more Metaverse Services through Devices that **UM-Capture** scenes with their **Sensors** and **MU-Render Entities**, i.e., Item that can be perceived, though their **Actuators**. Metaverse Services, implemented using centralised or decentralised architectures, generate the M-Instances on the left-hand side of Figure 1. Light blue Users or Objects indicates that they are generated by the M-Instance. Green Users or Objects indicates that they are digital twins of humans or objects. Green User are rendered either as **Personae**, i.e., **Models** representing humans that are **UM-Animated** by the movements of a human or be **MM-Animated** by an autonomous agent.

Direct Interoperability of two M-Instances is achieved when M-Instance #1 can use the Data from and as intended by M-Instance #2. **Mediated Interoperability** is achieved when one or both M-Instances request a Conversion Service to translate data from M-Instance #2 and vice-versa. Mediated Interoperability may not achieve the same result of Direct Interoperability if the data syntax and semantics of M-Instance #1 does not match M-Instance #2's or if the delay created by the Conversion Service negatively affects the user experience.

Therefore, an M-Instance is populated by Users and Objects potentially having a Device-enabled relationship with a U-Environment or else synthetically generated by an M-Instance. The Functionalities provided by an M-Instance enable its Users to achieve 1) their goals, 2) within the constraints of the Metaverse Service and Device capabilities, and 3) respecting the Rules under which Users operate in the M-Instance.

3.3 Registration

A human wishing to have its User(s) participate in an M-Instance/M-Environment may be asked to **Register**:

1. The human may be requested to provide a subset of their **User Data** that may include:

1.1. **Personal Profile**, i.e., Data about the human represented by the User.
1.2. Persona(e).
1.3. One or more **Wallet**, i.e., a container of Currency units.
1.4. **Activity Data**, i.e., the record of the Actions of a User.
1.5. **Social Graph**, i.e., the network of connections of a User with Items, M-Locations, U-Locations, and Services.

2. **Account** is an Item with the following characteristics:
   2.1. It unequivocally associates a Registered human with the subset of the Items they provide.
   2.2. A human may have more than one Account in one or more M-Instances/M-Environments.
   2.3. A User exists after a human Registers with an M-Instance/Environment.
   2.4. A User has certain Rights to Act in the M-Instance/M-Environment which is associated with the Account.
   2.5. An M-Instance/Environment may allow a human to have more than one User per Account.
   2.6. Different Users of an Account may have different Rights.

Note that some User Data <u>may be kept private</u> and that the laws of the jurisdiction under which the M-Instance/M-Environment operates may prescribe that it may not request certain User Data or handle it with special care.

The **Rules** of an M-Instance/M-Environment express:

1. The terms and conditions under which a User exists in an M-Instance/M-Environment and operates either there or in another M-Instance/M-Environment.
2. The obligations undertaken by the Registering human represented by the User.

A human Registered with an M-Instance/M-Environment may be able to join another M-Instance/M-Environment if they are Interoperable and the Rules enable that Functionality. Rules may prevent a human Registered with an M-Instance/M-Environment from interacting with another M-Instance/M-Environment.

Data entering an M-Instance, e.g., through the Action of **UM-Import**ing (e.g., from a device external to the M-Instance) may include Metadata and the Rights granted to a Process to perform Actions on the Data.

Rights is an Item expressing the ability to perform an Action on an Item.

- Rights include the User, the Actions, and the Items the User can perform Actions on.
- The Rules of some M-Instances/M-Environments may forfeit Rights enforcement on some Actions performed on some Items by some Users.

Identifier is an Item uniquely associated to a particular Item. An Item may be Identified by more than one Identifier. The following hierarchical structure allows for Identification of an Entity at an M-Location, in a M-Environment of an M-Instance:

[M-InstanceID] [M-EnvironmentID] [M-LocationID] [ItemID].

3.4 Processes

A **Process** performs Actions on Items, and Data and Metadata inside an M-Instance to the extent allowed by the Rights held by the Process in the M-Instance and/or in other M-Instances to the extent allowed by the Rights the Process holds in those other M-Instances. Actions are defined in Table 1 and specified in Chapter 4.

MPAI-MMM assumes that Processes are executed in M-Instances. A Process #1 interacts with a Process #2 by using a request-response protocol where:

1. Process #1 **MM-Sends** a **Request-Action** Item to Process #2 requesting:
   1.1. To perform an Action on one or more Item, provided as InItems that include InRights to perform Actions on the InItems. The InItems are located at a Service or at M- or U- Locations provided as InLocations.
   1.2. To provide the requested OutItems that include OutRights to perform Actions on the OutItems. The OutItems are located at a Service or at M- or U- Locations provided as OutLocations.
2. Process #2
   2.1. Executes Request if there are sufficient InRights.
   2.2. Responds with a **Response-Action** Item:
      2.2.1. Success: OutItem is provided.
      2.2.2. Error: e.g., no Rights, no such IDs, etc.

This Technical Report identifies four types of Process:

1. **Device** is a Process having either or both the capabilities to:
   1.1. **UM-Capture**, i.e., to acquire a scene at a U-Location as Media.
   1.2. MM-Send Data and Metadata from the Device to a User who then:
      1.2.1. **Identifi**es, i.e., gives an Identifier to and produces an Item.
      1.2.2. **MM-Embed**s, i.e., places an Entity at an M-Location with a Spatial Attitude, i.e., with a **Position** and **Orientation** and enables the MU-Rendering of the Entity.
2. **App** is an application-specific Program executed on a Device.
3. **User** is a Process representing a human who has an Account with an M-Instance/Environment. A User can be:
   3.1. **UM-Render**ed as an Object representing a human.
   3.2. MM-Embedded at an M-Location with a Spatial Attitude as a Persona
      3.2.1. UM-Animated by a Stream UM-Rendered from a U-Location.
      3.2.2. MM-Animated by an autonomous agent.
4. **Service** is a Process offering functionalities necessary for the proper functioning of an M-Instance/Environment, e.g., Discovery of an Entity or Process.

It should be noted that an M-Instance can be implemented to only execute a subset of the Actions on a subset of the Items defined by this Technical Report. An M-Instance can also implement more Functionalities requiring more Actions on more Items, either proprietary or belonging to future versions of this Technical Report. This case has obviously an impact on Interoperability.

3.5 Actions

A Process can request that another Process perform Actions on Items or Data & Metadata as follows:

1 UM-Import Data and Metadata.
2 Identify, e.g., by the following steps:
2.1 A Device MM-Sends Data and Metadata to a Service.
2.2 A Service MM-Sends the ID of the Entity that includes the MM-Sent Data and Metadata.

2.2.1 Data can be an Animation Stream coming from a human via a Device. The Device adds Metadata, e.g., Device ID and Rights to Act on the animation **Stream**, depending on the rights exercised or acquired by the Device when UM-Capturing the human.

2.2.2 The Metadata provided by the Device can be MM-Sent to a Service to Identify an Item or MM-Sent to the Service to modify the Metadata of an existing Item.

Note that the Identify Action is required to make Data and Metadata Actionable in an M-Instance/Environment.

3 **Hide** i.e., make the Identifier of an Item unavailable.

4 **Change**, i.e., modify the Item's Rights.

5 **MU-Export**, i.e., store as Item or Data at an Address.

6 **Authenticate**, i.e., the User requests a Service to provide evidence that an Entity is what it states it is.

7 **Post** to a Marketplace an **Asset**, i.e., an Item that can be Transacted.

8 **Transact** an Asset.

A User can request that a Service perform the Action to:

1. **Discover** Items by adding to the **Request-Discover** Item the **DiscoverIn** Item providing information about the Items and/or Processes to be searched. The User receives a **Response-Discover Item** with a **DiscoverOut** Item providing information about the Items found.
2. **Inform** about an Item by adding to the **Request-Inform** Item an **InformIn** Item providing information about the Item for which information is sought. The User receives a **Response-Inform** Item with a **InformOut** Item providing information about the requested Item.
3. **Interpret** Item by adding to the **Request-Interpret** Item an **InterpretIn** Item providing information about the Item for which interpretation is requested. The User receives a **Response-Interpret** Item with a **InterpretOut** Item providing the requested Interpretation.

The currently defined Actions performed on Entities are:

1. **Author**, i.e., a User calls an Authoring Tool Service with an accompanying request to obtain Rights to Act on the Authored Entity.
2. **MM-Add**, i.e., a User requests that an Entity be added to an M-Location with a Spatial Attitude. The original [M-InstanceID] [M-EnvironmentID] [EntityID] Identifier is changed to [M-InstanceID] [M-EnvironmentID] [M-LocationID] [EntityID].
3. **MM-Enable**, i.e., a Process can request that an Entity MM-Added at an M-Location be MU-Rendered.
4. **MM-Enable**, i.e., the User requests that a Process be allowed to MM-Capture an Entity that is MM-Added at an M-Location.
5. **MM-Capture**, i.e., the User requests that an Entity MM-Embedded at an M-Location be MM-Sent to a Process.
6. MM-Embed, i.e., the Composite Action of MM-Adding and MM-Enabling in one stroke.
7. UM-Animate, i.e., a User requests that a Process change the features of an Entity with a Stream obtained through the following process:
   7.1. UM-Capture an animation stream extracted from a scene at a U-Location.
   7.2. UM-Send the animation stream and Metadata to a User.
   7.3. Identify the Stream to make the Stream Entity usable in the M-Instance.
8. **MM-Disable**, i.e., a User requests that the MM-Enabling of the Entity be stopped.

A Device can:

1. **UM-Render** a scene at a U-Location to an M-Location, i.e.:
   1.1. UM-Capture, i.e., acquire a scene as Media from at a U-Location.

   1.2. MM-Send Data and Device-provided Metadata.
   1.3. Identify an Entity from Data and Metadata.
   1.4. MM-Embed the Entity at an M-Location with a Spatial Attitude.
2. **MU-Render** an Entity Embedded at an M-Location to a U-Location, i.e.:
   2.1. MM-Send, i.e., stream to a Device an Entity that is MM-Embedded at an M-Location.
   2.2. **MU-Actuate**, i.e., present the Entity as Media with a Spatial Attitude to a U-Location.

The Composite Action **Track** enables a User to request Services to:
1. MM-Embed a Persona at an M-Location with a Spatial Attitude.
2. UM-Animate the Persona MM-Embedded at an M-Location.
3. MU-Render selected Entities at the M-Location to a U-Location with a Spatial Attitude.

The full list of Actions is provided below organised by the type of Item the Action is executed on.

1. Actions on Entities:
   1.1. Authenticate
   1.2. Author
   1.3. Identify
   1.4. Inform
   1.5. Interpret
   1.6. MM-Add
   1.7. MM-Animate
   1.8. MM-Capture
   1.9. MM-Embed
   1.10. MM-Disable
   1.11. MM-Enable
   1.12. MM-Send
   1.13. MU-Actuate
   1.14. MU-Render
   1.15. Track
2. Actions on Assets:
   2.1. Post
   2.2. Transact
3. Actions on Data from the Universe:
   3.1. UM-Animate
   3.2. UM-Capture
   3.3. UM-Import
   3.4. UM-Render
4. Generic Actions on Items:
   4.1. Change
   4.2. Destroy
   4.3. Discover
   4.4. MU-Export
   4.5. Register

3.6 Items

An Item can belong to one of five categories:
1. Items characterised by the fact that they can be MM-Captured by a User.
2. Items that can cause an Entity to change its features.
3. Items that have space and time attributes.
4. Items that are finance related, called Assets.
5. Items that are generic data structures.

Items already defined above will not be defined again below.

Entity: is the <u>first type</u> of Item characterised by the fact that it can be MM-Captured by a User. This Technical Report identifies the following types of Entity:
1. **Event**: the set of Entities that are MM-Embedded at an M-Location from Start Time until End Time.
2. **Experience**: An Event as a User MM-Captured it and the User's Interactions with the Entities belonging to the Entity that spawned the Event.

3. **Object**: An Entity that is either virtually created or is the representation of an object including its features. This Technical Report currently considers the following Object types: Audio, Visual, and Haptic.
4. Model: An Object that can be UM-Animated by a Stream or a MM-Animated by a Process.
   4.1. Speech Model: An Object Model whose Object type is Audio, specifically Speech.
   4.2. Avatar Model: An Object Model where the Object type is Visual.
   4.3. Haptic Model: An Object Model where the Object type is Haptic.
   4.4. Persona: An Object Model that may include an Avatar Model, a Speech Model, and a Haptic Model. The Persona can be perceived visually and/or audibly and/or haptically. Note that a User may appear simultaneously, for instance as:
      4.4.1. The same or a different Persona UM-Animated by the same Stream at different M-Locations.
      4.4.2. The same or a different Persona where one Persona is UM-Animated by a real-time Stream and the other is UM-Animated by a recorded Stream.
      4.4.3. The same or a different Persona, one UM-Animated by a Stream and the other UM-Animated by an autonomous Process.
5. **Scene**: a dynamic composition of Objects described by Time and Spatial Attitudes.

The <u>second type</u> of Item can cause an Entity to change its perceptible features, i.e.:
1. **Interaction**: The Action made by a User on an Entity at a specific Time.
2. Stream: A continuous flow of:
   2.1. Data from a Device to a User, or
   2.2. Data from an Entity at an M-Location to a Device.

The <u>third type</u> of Item has space and time attributes:
1. M-Instance: an implementation of metaverse specifications identified by [M-InstanceID].
2. M-Environment: A portion of an M-Instance identified by [M-InstanceID] [M-EnvironmentID].
3. M-Location: An identifiable delimited portion of an M-Environment identified by [M-InstanceID] [M-EnvironmentID] [M-LocationID].
4. U-Environment: An identifiable portion of the Universe.
5. U-Location: An identifiable delimited portion of a U-Environment.
6. **Map**: An Item containing information connecting U-Locations, M-Locations, and optionally Metadata.

The <u>fourth type</u> of Item is Finance-related:
1. Asset: An Item that may be the object of a Transaction and is Embedded at an M-Location or Posted to a Service.
2. **Ledger**: the list of Transactions executed on Assets.
3. **Provenance**: the list of Transactions executed on an Asset starting from the first and including the last.
4. **Transaction**: Item representing the changed state of the Accounts and the Rights of a seller User and a buyer User on an Asset and optionally of the Service facilitating/enabling the Transaction.
5. **Value**: An Amount expressed in a Currency.
6. Wallet: A container of Currency units.

The <u>fifth type</u> of Item is non-perceptible, i.e.:

1. Account: An Item issued by an M-Instance/M-Environment that uniquely identifies a human who is Registered with the M-Instance/M-Environment.
2. Activity Data.
3. All Request-Action Item.
4. All Response-Action Item.
5. **Message**: Application-specific Data or Item Sent by a Source to a Destination.
6. Personal Profile.
7. Social Graph.
8. Rights.
9. Rules.

3.7 Data Types

Actions and Items may use several Data Types. Some Data Types may relate to a Metaverse Instance or the Universe; a U-/M- prefix may be added as needed. Data Types are currently defined as follow for the specific case on M-Instances/Environments/Locations:

1. **Address**: A URL.
2. **Amount**: A number expressing a Value in a Currency.
3. **Coordinates**: A set of numbers representing a Position in an M-Environment using a coordinate system.
4. **Currency**: A medium of exchange enabling Transactions in an M-Instance.
5. **Personal Status**: the representation of the information internal to a User characterising their behaviour.
   5.1. **Cognitive State**: the representation of a User's Personal Status that reflects the way it understands the environment, such as "Confused", "Dubious", "Convinced".
   5.2. **Emotion**: the representation of a User's Personal Status that results from its interaction with an environment, such as "Angry", "Sad", "Determined".
   5.3. **Social Attitude**: the representation of a User's Personal Status related to the way the User intends to position vis-à-vis an environment, e.g., "Respectful", "Confrontational", "Soothing".
6. **Point**: A point in an M-Environment identified by the set of local Coordinates.
7. **Spatial Attitude**: The Position and Orientation of an Entity, and their velocities and accelerations.
   7.1. **Position**: the coordinates of an Object with respect to a set of coordinates in an M-Environment.
   7.2. **Orientation**: The set of the 3 roll, pitch, yaw angles indicating the rotation around the principal axis (x) of an Object, its y axis having an angle of 90˚ counterclockwise (right-to-left) with the x axis and its z axis (pointing up toward the viewer viewing from above).
   7.3. **Point of View**: The Spatial Attitude of a Persona Perceiving the M-Environment.
8. **Time**: the measure of time.

4 Processes

A Request-Action contains an appropriate subset of the Data of Table 2. In the following, the logical ∨ symbol is used to indicate that either or both the elements at the left and the right of ∨ may be used, and the logical ∧ symbol is used to indicate that both elements at the left and the right of ∧ shall be used.

Table 2 – The elements of the Request-Action Item

Time	
Source	ProcessID
Destination	ProcessID
Action	One of the Actions
InItem	ActionIn ∨ ActionInID ∨ Item ∨ ItemID ∨ Data & Metadata ∨ Spatial Attitude
InLocations	M-LocationID ∨ U-LocationID ∨ ProcessID
OutLocations	M-LocationID ∨ U-LocationID ∨ ProcessID
OutRights	Rights ∨ RightsID

ActionIn is an Item specifically required by some Actions. For instance, DiscoverIn describes the type of Items a User is requesting a Discover Service to find with the Rights to Act on the DiscoverIn Item granted to the Discover Service.

Table 3 provides the semantics of the components of a Request-Action.

Table 3 - Semantics of the elements of the Request-Action Item

Time	The Time the Request-Action is issued.
Source	The Process making the request.
Destination	The Process receiving the request.
Action	The Action that the Source requests that the Destination perform on the InItems.
InItems	Items or Data & Metadata provided as input to the requested Action.
InLocations	The Locations of the InItems.
OutLocations	The requested Locations of the OutItems.
OutRights	The Rights requested by the Source on the OutItems.

Note1: A Request need not contain all the elements above.
Note2: The InRights and the OutRights are contained in the InItem and OutItem, respectively.

The Response-Action Item to a Request-Action Item is expressed by Table 4:

Table 4 -The Response-Action Item

Success	OutItem	OutItem ∨ OutItemID ∨ ActionOut ∨ ActionOutID
Error	FaultyReq.	Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	Unsupported	Item not supported
	Mismatch	Item type mismatch
	User Data	Faulty
	Wallet	Insufficient Value
	Clash	Entity clashes with another Entity
	M-Location	Out of range
	U-Location	Out of range
	Address	Incorrect

ActionOut is an Item required by some Actions. For instance, DiscoverOut is an Item that contains the results provided by a Discover Service with the Rights to Act on the DiscoverOut Item granted to the requesting Process.

The semantics of the elements of the Response are provided by Table 5.

Table 5 - Semantics of the Response-Action Item

Success	The execution of the requested Action was successful.	
	OutItem	The Items requested in the Request-Action.
Error	The execution of the Request failed.	
	FaultyReq	The content of the Request is Faulty.
	IDs	Some IDs in the Request are incorrect.
	Rights	Some Rights are missing or incomplete.
	Unsupported	An Item is not supported.
	Mismatch	Mismatch between Items.
	User Data	Faulty User Data.
	Wallet	The Wallet is not sufficient for the required Transaction.
	Clash	An Entity clashes with another Entity.
	M-Location	There is no such M-Location.
	U-Location	There is no such U-Location.
	Address	There is no such Address.

InItems and OutRights of a Request-Action and OutItems of a Response-Action may be expressed either as Items and Rights or as IDs.

In the following Subsections, the types of Processes are described.

4.1 App

Purpose	An application-specific Program executed on a Device.	
Data	TBD	
Metadata	AppID	The ID of the App.
	UserID	The ID of the User having Rights to Act on the App.
	InRightsID	The ID of the User’s Rights to Act granted to the App.
	OutRightsID	The ID of the Rights a User may acquire on the App.
	DescrMdata	Any description of the App.

4.2 Device

Purpose	A Process able to: 1. UM-Capture Data from a U-Location 2. UM-Send Data and Metadata to a User. and/or 3. MM-Send an Entity from an M-Location to the Device. 4. MU-Render an Entity at a U-Location.	
Data	TBD	
Metadata	DeviceID	The ID of the Device.

	UserID	The ID of the User having Rights to Act on the Device.
	InRightsID	The ID of the User's Rights to Act granted to the Device.
	OutRightsID	The ID of the Rights a User may acquire on the Device.
	DescrMdata	Any description of Device.

4.3 Service

Purpose	A Process that can be Called to provide Functionalities.	
Data	TBD	
Metadata	ServiceID	The ID of the Service.
	UserID	The ID of the User having Rights to Act on the Service.
	InRightsID	The ID of the User's Rights to Act granted to the Service.
	OutRightsID	The ID of the Rights to Act on the Service a User may acquire.
	DescrMdata	Any description of the Rights.

4.4 User

Purpose	A Process representing an MM-Captured human as a Persona that is either UM-Animated by a Stream or MM-Animated by an autonomous agent.	
Data	TBD	
Metadata	UserID	ID of User.
	RightsID	ID of Rights held by User
	AccountIDs	IDs of Accounts held by User.
	WalletIDs	IDs of Wallets held by User.
	UserDataID	ID of User Data.
	DescrMdata	Any description of the User.

5 Actions

5.1 General

The MPAI Metaverse Model assumes that a Source Process (User, Device, or Service) MM-Send a Request to a Destination Process (User, Device, or Service) to execute Actions on InItems placed at In-Locations. The Destination will execute the Request if the Source has the necessary InRights, place the OutItems at M-Locations, and grant to the Source OutRights to Act on the OutItems.

This version of this Technical Report preserves the different names of User, Device, and Service, instead of using the generic name "Process" to facilitate understanding of the nature of the different requests. However, it should be borne in mind that Sources and Destinations will all be implemented as Processes with appropriate Metadata.

This Chapter specifies the Actions that are supported by at least one Functionality Profile. An Action is called *Basic* when the request involves only one Action and *Composite* when it involves a plurality of Actions and potentially Processes.

Table 6 provides the elements of all Actions considered in this document. Note that some Actions are Composite, i.e., made of Basic Actions.

Table 6 - Table of the elements of the Action requests

Legend: D=Device, Mdata=Metadata, MLoc=M-Location, P=Process, S=Service, SA=Spatial Attitude, U=User, ULoc=U-Location, - = Absent.
The symbol ∧ in a list means that all the elements of the list should be included. The symbol - in a list means that the elements of the list should be considered separately.

	Source	**Destination**	**Action**	**InItem**	**InLocation**	**OutItem**	**OutLocation**	**OutRights**
Change	U	S	Change	Entity	MLoc	Entity	MLoc∨ -	Rights
Destroy	U	S	Destroy	Item	S	-	-	-
Discover	U	S	Discover	DiscoverIn	U	DiscoverOut	U	Rights
MU-Export	U	S	MU-Export	Item	S	Item	Address	Rights
Identify	U-D	S	Identify	Data & Mdata	Address-D	Item	S	Rights
UM-Import	U	D	Read	Data & Mdata	Address	Data & Mdata	D	Mdata
Inform	U	S	Inform	InformIn	MLoc	InformOut	U	Rights
Interpret	U	S	Interpret	InterpretIn	MLoc	InterpretOut	U	Rights
MM-Add	U	S	MM-Add	Entity ∧ SA	S	Entity	MLoc	Rights
MM-Animate	U	S	MM-Animate	Entity	MLoc	Entity	MLoc	Rights
MM-Capture	U	S	MM-Capture	Entity	MLoc	Entity	U	Rights
MM-Embed	U	S	MM-Add ∧ MM-Enable	Entity ∧ SA	S	Entity	MLoc	Rights
MM-Enable	U	S	MM-Enable	Entity	MLoc	Entity	MLoc	Rights
MM-Disable	U	S	MM-Disable	Entity	MLoc	-	-	-
MM-Send	P	P	Send	Item	P	Item	P	Rights Mdata
MU-Render	U	D	MM-Send	Entity	MLoc	Entity	D	Rights
	D	Act.	MU-Actuate	Data	D	media	ULoc	Mdata
MU-Actuate	U	D	MU-Actuate	Entity	D	Media	ULoc	Mdata
MU-Render	U	S	MM-Send	Entity	MLoc	Entity	D	Rights
	U	D	MU-Actuate	Entity	D	Media	ULoc	Mdata
Post	U	S	Post	Asset	S, Address	Asset	S	Rights
Register	human	S	Register	User Data	human ∨Address	Account	S	Rights
Track	U	S	MM-Embed	Persona ∧ SA	MLoc	Persona	MLoc	Rights
	U	D	UM-Capture	scene	ULoc	animation stream	D	Mdata
	D	U	UM-Send	Data & Mdata	D	Data & Mdata	U	Mdata
	U	S	Identify	Data & Mdata	S	Stream	S	Rights

	U	S	MM-Animate	Persona	MLoc	Entity	MLoc	Rights
	U	D	MU-Render	Persona	U	Media	ULoc	Mdata
Transact	U	S	Transact	Value	-	User	-	Rights
UM-Animate	U	S	UM-Animate	Entity ∧ Stream ∧ SA	MLoc	Entity	MLoc	Rights
UM-Capture	U-S	D	UM-Capture	scene	ULoc	Data	D	Rights
	D	U	UM-Send	Data & Mdata	D	Data & Mdata	U	Mdata
UM-Send	U	S	Identify	Data & Mdata	U	Entity	S	Rights
	U	S	MM-Embed	Entity ∧ SA	MLoc	Entity	MLoc	Rights
UM-Send	U	D	UM-Send	Data & Mdata	D	Data & Mdata	U	Mdata

Table 7 - Table of the elements present in a Response-Action

	OutItem	FaultyReq	Wrong IDs	Missing Rights	Unsupported	Wallet error	Clash	M-Location	U-Location	Address	Mismatch	UserData
Authenticate	x	x	x	x				x				
Author	x					x						
Change	x	x	x	x				x				
Identify	x	x	x	x		x						
Discover	x	x	x	x		x						
Hide	x	x	x	x								
Inform	x	x	x	x				x				
Interpret	x	x	x	x		x		x				
MM-Add	x	x	x	x			x	x				
MM-Animate	x	x	x		x			x			x	
MM-Capture	x	x	x	x				x				
MM-Disable	x	x	x	x				x				
MM-Embed	x	x	x	x			x	x				
MM-Enable	x	x	x	x				x				
MM-Send	x	x	x	x								
MU-Actuate	x	x	x	x				x	x			
MU-Render	x	x	x	x				x	x			
Post	x	x	x	x								
Read	x	x	x	x						x		
Register	x					x						x
Track	x	x	x	x					x			
Transact	x					x						
UM-Animate	x	x	x		x			x	x		x	
UM-Capture	x	x	x	x				x	x			

UM-Render	x	x	x	x			x	x	x			
UM-Send	x	x		x			x	x	x			
MU-Export	x									x		

This Technical Report does not specify the protocol carrying the Request-Action and the Response-Action Items.

5.2 Authenticate

<table>
<tr><td>Definition</td><td>The Action of requesting that a Service confirm that an Entity MM-Embedded at an M-Location is what it claims to be. An object at a U-Location can be Authenticated by UM-Rendering and then Authenticating it.</td></tr>
<tr><td>Request-Action</td><td><table>
<tr><td>Time</td><td></td></tr>
<tr><td>Source</td><td>UserID</td></tr>
<tr><td>Destination</td><td>ServiceID</td></tr>
<tr><td>Action</td><td>Authenticate</td></tr>
<tr><td>InItems</td><td>Entity ∨ EntityID</td></tr>
<tr><td>InLocation</td><td>M-LocationID ∨ UserID</td></tr>
<tr><td>OutLocation</td><td>UserID</td></tr>
<tr><td>OutRights</td><td>Rights ∨ RightsID</td></tr>
</table></td></tr>
<tr><td>Response-Action</td><td><table>
<tr><td>Success</td><td>OutItem</td><td>Rights ∨ RightsID</td></tr>
<tr><td>Error</td><td>FaultyReq</td><td>The Request is Faulty</td></tr>
<tr><td></td><td>IDs</td><td>Incorrect</td></tr>
<tr><td></td><td>Rights</td><td>Missing or incomplete</td></tr>
<tr><td></td><td>M-Location</td><td>Out of range</td></tr>
<tr><td></td><td>U-Location</td><td>Out of range</td></tr>
</table></td></tr>
</table>

5.3 Author

<table>
<tr><td>Definition</td><td>The Action of Calling a Service to obtain an Entity with associated OutRights to Act on the Entity.</td></tr>
<tr><td>Request-Action</td><td><table>
<tr><td>Time</td><td></td></tr>
<tr><td>Source</td><td>UserID</td></tr>
<tr><td>Destination</td><td>ServiceID</td></tr>
<tr><td>Action</td><td>Author</td></tr>
<tr><td>InItems</td><td>Item ∨ ItemID ∨ Data & Metadata</td></tr>
<tr><td>InLocation</td><td>UserID ∨ Address</td></tr>
<tr><td>OutItem</td><td>Entity ∨ EntityID</td></tr>
<tr><td>OutLocation</td><td>UserID ∨ ServiceID</td></tr>
<tr><td>OutRights</td><td>RightsID</td></tr>
</table></td></tr>
<tr><td>Response-Action</td><td><table>
<tr><td>Success</td><td>OutItem</td><td>Entity ∨ EntityID</td></tr>
<tr><td>Error</td><td>FaultyReq</td><td>The Request is Faulty</td></tr>
<tr><td></td><td>IDs</td><td>Incorrect</td></tr>
<tr><td></td><td>Rights</td><td>Missing or incomplete</td></tr>
<tr><td></td><td>Wallet error</td><td>Insufficient Value</td></tr>
</table></td></tr>
</table>

5.4 Change

Definition	The Action of requesting that a Service modify the Rights of a User whose Persona is Embedded at an M-Location and provide OutRights, e.g., to further Change the Rights.
Request-Action	

Time	
Source	UserID
Destination	ServiceID
Action	*Change*
InItems	UserID ∨ (Rights ∧ RightsID)
InLocation	M-LocationID
OutLocation	UserID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	Rights ∨ RightsID
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	M-Location	Out of range

5.5 Discover

Definition	The Action of requesting that a Service provide a DiscoveryOut Item containing the IDs of the Items relevant to the DiscoverIn Item with the OutRights to Act on the DiscoverOut Item.
Request-Action	

Time	
Source	UserID
Destination	ServiceID
Action	*Discover*
InItem	DiscoveryIn ∨ DiscoveryInID
InLocation	UserID ∨ ServiceID
OutLocation	UserID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	DiscoveryOut ∨ DiscoveryOutID
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete

5.6 Hide

Definition	The Action of requesting that a Service make the ID of an Item unavailable and provide OutRights, e.g., to make the ID available again.
Request-Action	

Time	
Source	UserID
Destination	ServiceID
Requested Action	*Hide*
InItem	Item ∨ ItemID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	Rights ∨ RightsID
Error	FaultyReq	The Request is Faulty

		IDs	Incorrect
		Rights	Missing or incomplete

5.7 Identify

Definition	The Action of requesting that a Service produce an Item from Data & Metadata or update an Item with the OutRights to Act on the Item.

Request-Action	Time	
	Source	Process ID
	Destination	ServiceID
	Action	*Identify*
	InItems	Data & Metadata ∨ InItems ∧ InItemID ∧ InRights
	InLocation	UserID
	OutLocation	ServiceID

Response-Action	Success	OutItem	ItemID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.8 Inform

Definition	The Action of requesting that a Service provide an InformOut Item containing information about an InItem, such as the Metadata of the InItem, with the OutRights to Act on the InformOut Item.

Request-Action	Time	
	Source	UserID
	Destination	ServiceID
	Action	*Inform*
	InItem	InformIn ∨ InformInID
	InLocation	M-LocationID
	OutLocation	UserID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	InformOut ∨ InformOutID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.9 Interpret

Definition	The Action of requesting that a Service provide an InterpretOut Item containing interpretation of an InItem, such as translation or extraction of Personal Status, with the OutRights to Act on the InterpretOut Item.

Request-Action	Time	
	Source	UserID
	Destination	ServiceID
	Action	*Interpret*
	InItem	InterpretIn ∨ InterpretInID
	OutLocation	UserID
	OutRights	Rights ∨ RightsID

Re-sponse-Action	Success	OutItem	InterpretOut ∨ InterpretOutID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.10 MM-Add

Definition	The Action of requesting that a Service add an Entity at an M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Added Entity.

Request-Action	Time	
	Source	UserID
	Destination	ServiceID
	Action	*MM-Add*
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude
	InLocation	ServiceID
	OutLocation	M-LocationID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	Entity ∨ EntityID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Clash	Entity clashes with another Entity
		M-Location	Out of range

5.11 MM-Animate

Definition	The Action of requesting that a Service change the features of a Model MM-Embedded at an M-Location with a Stream and provide the OutRights to Act on the MM-Added Entity.

Request-Action	Time	
	Source	UserID
	Destination	ServiceID
	Action	*MM-Animate*
	InItem	(Model ∨ ModelID) ∧ Spatial Attitude ∧ (Stream ∨ StreamID)
	InLocation	M-LocationID
	OutLocation	M-LocationID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItem	Rights ∨ RightsID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Item mismatch	Entity Data Type and Animation Stream Data Type.

5.12 MM-Capture

Definition	The Action of requesting that a Service MM-Send selected Entities MM-Embedded at an M-Location to a User.

Request-Action	Time		
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Capture*	
	InItem	List of EntityIDs	
	InLocation	M-LocationID	
	OutLocation	UserID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	List of EntityIDs
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.13 MM-Disable

Definition	The Action of requesting that a Service stop MM-Enabling selected Entities Embedded at an M-Location and provide OutRights to Act on the MM-Disabled Entities.		
Request-Action	Time		
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Disable*	
	InItem	List of EntityIDs	
	InLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Rights ∨ RightsID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range

5.14 MM-Embed

Definition	The Composite Action of requesting that a Service MM-Add and MM-Enable an Entity either located at a Service or at an M-Location at a destination M-Location with a Spatial Attitude and provide OutRights to Act on the MM-Embedded Entity.	
Request-Action	Time	
	Source	UserID
	Destination	ServiceID
	Action	*MM-Add*
	InItem	Entity ∨ EntityID
	InLocation	ServiceID ∨ M-LocationID
	OutLocation	M-LocationID
	OutRights	Rights ∨ RightsID
	Source	UserID
	Destination	ServiceID
	Action	*MM-Enable*
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude

	InLocation	M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Re-sponse-Action	Success	OutItem	Rights ∨ RightsID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Clash	Entity clashes with another Entity
		M-Location	Out of range

5.15 MM-Enable

Defini-tion	The Action of requesting that a Service accept requests to MM-Send selected Entities MM-Added at an M-Location or to MM-Embed those selected Entities at a destination MM-Location and provide OutRights to act on the M-Entities.		
Request-Action	Time		
	Source	UserID	
	Destination	ServiceID	
	Action	*MM-Enable*	
	InItem	Entity ∨ EntityID	
	InLocation	M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Re-sponse-Action	Success	OutItem	Rights ∨ RightsID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		M-Location	Out of range

5.16 MM-Send

Definition	The Action of requesting that a Service forward to a Process an Item with OutRights to Act on the Item, or Data/Metadata.		
Request-Action	Time		
	Source	ProcessID	
	Destination	ServiceID	
	Action	*Send*	
	InItem	Item ∨ ItemID ∨ Data & Metadata	
	InLocation	ProcessID ∨ M-Location	
	OutLocation	ProcessID ∨ M-Location	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	Item ∨ ItemID ∨ Data & Metadata
	Error	Request	Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.17 MU-Actuate

Definition	The Action of requesting that a Device present at a U-Location as Media with a Spatial Attitude an Entity MM-Embedded at an M-Location.
Request-Action	Time: Source: UserID Destination: DeviceID Action: *MU-Actuate* InItem: Data & Metadata ∧ Spatial Attitude InLocation: DeviceID OutLocation: U-LocationID OutRights: Metadata
Response-Action	Success – OutItem – Media ∧ Spatial Attitude Error – FaultyReq – The Request is Faulty IDs – Incorrect Rights – Missing or incomplete U-Location – Out of range

Request-Action:

Time	
Source	UserID
Destination	DeviceID
Action	*MU-Actuate*
InItem	Data & Metadata ∧ Spatial Attitude
InLocation	DeviceID
OutLocation	U-LocationID
OutRights	Metadata

Response-Action:

Success	OutItem	Media ∧ Spatial Attitude
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	U-Location	Out of range

5.18 MU-Export

Definition: The Action of requesting that a Process stores an Item containing the IDs of an Item at an Address.

Request-Action:

Time	
Source	ProcessID
Destination	ServiceID
Action	*MU-Export*
InItem	Item ∨ ItemID
InLocation	M-LocationID ∨ ProcessID
OutLocation	Address

Response-Action:

Success	OutItem	
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	Address	Incorrect

5.19 MU-Render

Definition: The Composite Action of requesting that:

1. A Service MM-Send selected Entities Embedded at an M-Location to a Device.
2. The Device MU-Actuate at a U-Location with a Spatial Attitude the Entity received.

Request-Action:

Time	
Source	UserID
Destination	ServiceID
Action	*MM-Send*
InItem	List of Entities
InLocation	M-LocationID
OutLocation	DeviceID

	Source: UserID; Destination: DeviceID; Action: *MU-Actuate*; InItem: Data & Metadata ∧ Spatial Attitude; InLocation: DeviceID; OutLocation: U-LocationID; OutRights: Metadata

Source	UserID	
Destination	DeviceID	
Action	*MU-Actuate*	
InItem	Data & Metadata ∧ Spatial Attitude	
InLocation	DeviceID	
OutLocation	U-LocationID	
OutRights	Metadata	

Response-Action

Success	Action result	Media ∧ Spatial Attitude
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	M-Location	Out of range

5.20 Post

Definition	The Action of requesting that a Marketplace include an Asset to its repertory of Assets.

Request-Action

Time	
Source	UserID
Destination	ServiceID
Action	*Post*
InItem	Asset ∨ AssetID
InLocation	UserID ∨ ServiceID
OutLocation	ServiceID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	Wallet	Insufficient Value

5.21 Register

Definition	The Action of a human requesting that a Service (M-Instance/Environment) grant their Users the Rights to perform Actions in the M-Instance/Environment.

Request-Action

Time	
Source	humanID
Destination	ServiceID
Requested Action	*Register*
InItem	UserData ∨ UserDataID
InLocation	Address
OutLocation	ServiceID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	Account ∨ AccountID
Error	FaultyReq	The Request is Faulty
	Wallet	Insufficient Value

5.22 Track

Definition	The Composite Action of requesting: 1. Service to MM-Embed a Model at an M-Location with a Spatial Attitude. 2. Service to MU-Animate the Model MM-Embedded at an M-Location. 3. Service to MU-Render specified Entities at the M-Location to a U-Location.
Request-Action	

Time	
Source	UserID
Destination	ServiceID
Action	*MM-Embed*
InItem	(Model ∨ ModelID) ∧ Spatial Attitude
InLocation	ServiceID
OutLocation	M-LocationID
OutRights	Rights ∨ RightsID
Source	UserID
Destination	DeviceID
Action	*UM-Animate*
InItem	(Persona ∨ PersonaID) ∧ Stream
InLocation	M-LocationID
OutLocation	M-LocationID
OutRights	Rights ∨ RightsID
Source	UserID
Destination	ServiceID
Action	*MU-Render*
InItem	Data & Metadata
InLocation	M-LocationID
OutLocation	U-LocationID
OutRights	Rights ∨ RightsID

Response-Action

Success	OutItem	Media ∧ Spatial Attitude
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	M-LocationID	Out of range
	U-LocationID	Out of range

5.23 Transact

Definition	The Action of a $User_1$ requesting that a Service: 1. Assign Rights on an Asset to $User_2$. 2. Cause: 2.1. $Wallet_1$ of $User_1$ to be increased by $Value_1$. 2.2. $Wallet_2$ of $User_2$ to be decreased by $Value_2$. 2.3. $Wallet_3$ of the Service enabling/facilitating the Transaction to be increased by $Value_3$ (optionally).
Request-Action	

Time	
Source	UserID
Destination	ServiceID
Action	*Transact*

	InItem	Transaction ∨ TransactionID
	InLocation	M-LocationID ∨ ServiceID
	OutLocation	UserID ∨ ServiceID
	OutRights	Rights ∨ RightsID

Response-Action	Success	OutItems	AssetID ∧ $WalletID_1$ ∧ $WalletID_2$ ∧ $WalletID_3$
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		Wallet	$Wallet_2$ has insufficient Value

5.24 UM-Animate

Definition	The Composite Action of requesting to: 1. UM-Capture an animation stream extracted from a scene at a U-Location. 2. UM-Send the animation stream and Metadata to a User. 3. Identify the Animation Stream. 4. MM-Animate the Model MM-Embedded at the M-Location using the Animation Stream.

Request-Action	Time	
	Source	UserID
	Destination	DeviceID
	Action	*UM-Capture*
	InItem	scene
	InLocation	U-LocationID
	OutLocation	DeviceID
	OutRights	Metadata
	Source	DeviceID
	Destination	UserID
	Action	*MM-Send*
	InItem	Data & Metadata
	InLocation	DeviceID
	OutLocation	UserID
	OutRights	Metadata
	Source	UserID
	Destination	ServiceID
	Action	*Identify*
	InItem	Data (stream) & Metadata
	InLocation	UserID
	OutLocation	UserID
	OutRights	Rights ∨ RightsID
	Source	UserID
	Destination	ServiceID
	Action	*MM-Embed*
	InItem	(Entity ∨ EntityID) ∧ Spatial Attitude
	InLocation	UserID
	OutLocation	M-Location
	OutRights	Rights ∨ RightsID
	Source	UserID

	Destination	ServiceID	
	Action	*MM-Animate*	
	InItem	(ModelID ∨ StreamID) ∧ Stream	
	InLocation	M-LocationID	
	OutLocation	M-LocationID	
	OutRights	Rights ∨ RightsID	
Response-Action	Success	OutItem	EntityID
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range
		M-Location	Out of range

5.25 UM-Capture

Definition	The Action of requesting that a Device acquire Media from a scene at a U-Location.		
Request-Action	Time		
	Source	UserID	
	Destination	DeviceID	
	Action	*UM-Capture*	
	InItem	scene	
	InLocation	U-LocationID	
	OutLocation	DeviceID	
Response-Action	Success	OutItem	Data
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range

5.26 UM-Import

Definition	The Action of a User requesting that a Service make available Data & Metadata, or an Item stored at an Address.		
Request-Action	Time		
	Source	UserID	
	Destination	ServiceID	
	Requested Action	*Read*	
	InItem	Data & Metadata ∨ Item	
	InLocation	Address	
	OutLocation	UserID	
Response-Action	Success	OutItem	Data & Metadata
	Error	FaultyReq	The Request is Faulty
		IDs	Incorrect
		Rights	Missing or incomplete

5.27 UM-Render

Definition	The Composite Action of requesting:

	1. A Device to UM-Capture a scene at U-Location. 2. The Device to MM-Send Data and Device-provided Metadata to a User. 3. A Service to Identify an Entity from UM-Sent Data and Metadata. 4. A Service to MM-Embed the Entity at an M-Location with a Spatial Attitude.
Request-Action	see table below
Response-Action	see table below

Request-Action:

Time	
Action	*UM-Capture*
InItem	scene
InLocation	U-LocationID
OutLocation	DeviceID
Source	DeviceID
Destination	UserID
Action	*MM-Send*
InItem	Data & Metadata
InLocation	DeviceID
OutLocation	UserID
OutRights	Metadata
Source	UserID
Destination	ServiceID
Action	*Identify*
InItem	Data (stream) & Metadata
InLocation	UserID
OutLocation	UserID
OutRights	Rights ∨ RightsID
Source	UserID
Destination	ServiceID
Action	*MM-Embed*
InItem	(Entity ∨ EntityID) ∧ Spatial Attitude
InLocation	ServiceID
OutLocation	M-LocationID
OutRights	Rights ∨ RightsID

Response-Action:

Success	OutItem	Entity ∨ EntityID
Error	FaultyReq	The Request is Faulty
	IDs	Incorrect
	Rights	Missing or incomplete
	Clash	Entity clashes with another Entity
	M-Location	Out of range
	U-Location	Out of range

5.28 UM-Send

Definition	The Action of a Device forwarding Data & Metadata to a Process.
Request-Action	see table below

Request-Action:

Time	
Source	DeviceID
Destination	ProcessID
Action	*UM-Send*
InItem	Data & Metadata
InLocation	DeviceID

	OutLocation	ProcessID	
	OutRights	Metadata	
Response-Action	Success	OutItem	Data & Metadata
	Error	Request	Faulty
		IDs	Incorrect
		Rights	Missing or incomplete
		U-Location	Out of range

6 Items

6.1 General

This chapter specifies the Items that are subject to an Action in at least one Functionality Profile. Items are specified using the following format:

Table 8 - Item Format

Purpose	A functional description of the Item.	
Data	In general, the Item Data Format(s) is(are) not provided. Actions and Entities related to the Item are provided where available. Initial Functional Requirements are provided where possible.	
Acted on Metadata	ItemID	ID of the Item.
	UserID	ID of the User who holds Rights on the Item with ItemID.
	WalletID	ID of the Wallet held by User with UserID
	InRightsID	ID of the Rights the User with UserID has on the Item with ItemID.
	OutRightsID	ID of the Rights a User may acquire on the Item with ItemID.
	AuthorID	ID of the User who Authored the Item with ItemID.
	AuthorToolID	ID of the Service who provided the AuthorTool.
	ParentItemID	ID of the Item that spawned the Item.
	ServiceID	ID of the Service that is Called.
	ServiceWalletID	ID of the Wallet of a Service.
	InItemID	ID of the Item to be Acted on.
	TargetUserID	ID of the User to be affected by the Action.
	TargetWalletID	ID of the Wallet of the User to be affected by the Action.
	UserDataID	ID of a User Data.
	PersonaID	ID of a User’s Persona.
	PersonalDataID	ID of a User’s Personal Data.
	ActivityDataID	ID of a User’s Activity Data.
	DescrMdata	Any additional descriptive Metadata of the Item.

Table 9 provides a view of the Metadata Elements of each Items. For reason of space, ItemIDs and Descriptive Metadata are not reported.

Table 9 – Elements of Item Metadata

	User/human	InRights	InItem	OutRights	OutItem	Service	ParentItem	Author	AuthoringTool	Target User	User Data	Persona	Personal Data	ActivityData	Social Graph	Wallet	Target Wallet	Service Wallet
Account	x	x				x												
ActionIn	x	x		x		x												
ActionOut	x	x		x		x												
Activity Data	x	x		x		x												
Asset	x	x		x		x										x		
DiscoverIn	x	x				x										x		
DiscoverOut	x			x		x												
Event	x	x		x			x											
Experience	x	x		x			x											
Identifier	x	x		x		x	x	x	x	x	x	x	x	x	x	x	x	x
InformIn	x	x				x												
InformOut	x			x		x												
Interaction	x	x		x														
InterpretIn	x	x				x												
InterpretOut	x			x		x												
Map	x	x		x				x	x									
M-Environment	x	x		x														
Message	x					x												
M-Instance	x	x		x														
M-Location	x	x		x														
Model	x	x		x				x	x									
Object	x	x		x				x	x									
Personal Profile	x	x		x														
Provenance	x	x		x														
Request-Action	x	x	x	x		x												
Response-Action	x	x		x	x	x												
Rights	x	x		x														
Rules	x	x						x	x									
Scene	x	x		x				x	x									
Service	x	x		x														
Social Graph	x	x		x														
Stream	x	x		x														
Transaction	x	x		x		x				x						x	x	x
U-Location	x	x		x														
User Data	x	x		x														
Value	x	x																
Wallet	x	x																

6.2 Account

Elements	Definitions	
Purpose	An Item issued by an M-Instance/M-Environment that uniquely identifies a human who is Registered with the M-Instance/M-Environment.	
Data	TBD	
Metadata	AccountID	The ID of the Account.
	humanID	The ID of the Account holder.
	InRightsID	The ID of the human's Rights in the M-Instance/M-Environment.

	ServiceID	The ID of the Service the Account refers to.
	DescrMdata	Any description of the Account.

6.3 Activity Data

Purpose	An Item containing the record of the Actions of a User.	
Data	TBD	
Metadata	ActivityDataID	The ID of the Activity Data.
	UserID	The ID of the User "having Rights to Act on the Activity Data".
	InRightsID	The ID of the User's Rights to Act on the Activity Data.
	OutRightsID	The ID of the Rights a User may acquire on the Activity Data.
	ServiceID	The ID of the Service the Account refers to.
	Descriptive Metadata	Any description of the Activity Data.

6.4 Asset

Purpose	An Item Embedded at an M-Location or Posted to a Service that may be the object of a Transaction.	
Data	The Data of an Asset conform to the Format of the Item that has spawned it	
Metadata	AssetID	The ID of the Asset.
	UserID	The ID of the User "having Rights to Act on the Asset".
	InRightsID	The ID of the User's Rights to Act on the Asset.
	OutRightsID	The ID of the Rights a User may acquire on the Asset.
	DescrMdata	Any description of the Asset.

6.5 DiscoverIn

Purpose	An Item containing a description of the Items to be Discovered.	
Data	TBD	
Metadata	DiscoverInID	The ID of the DiscoverIn Item.
	UserID	The ID of the User generating the DiscoverIn Item.
	ServiceID	The ID of the Service providing Discovery Services.
	RightsID	The ID of the Rights "to Act on the DiscoverIn Item" granted to the Discovery Service.
	DescrMdata	Any description of the DiscoverIn Item.

6.6 DiscoverOut

Purpose	An Item containing the description of the Items Discovered.	
Data	TBD	
Metadata	DiscoverOutID	The ID of the Request-Discover Item.
	ServiceID	The ID of the Service providing the DiscoverOut Item.
	UserID	The ID of the User having requested the DiscoverOut Item.
	RightsID	The ID of the Rights "to Act on the DiscoverOut Item" granted to UserID.
	DescrMdata	Any description of the DiscoverOut Item.

6.7 Event

<table>
<tr><td>Purpose</td><td>The set of Entities that are MM-Embedded at an M-Location from Start Time until End Time.</td></tr>
<tr><td>Data</td><td><table>
<tr><td>M-Location</td><td></td></tr>
<tr><td>StartTime</td><td>The start Time of the Event.</td></tr>
<tr><td>EndTime</td><td>The end Time of the Event.</td></tr>
</table></td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>EventID</td><td>The ID of the Event.</td></tr>
<tr><td>UserID</td><td>The ID of the User “having Rights to Act on the Event”.</td></tr>
<tr><td>InRightsID</td><td>The ID of the Rights “to Act on the Event”.</td></tr>
<tr><td>OutRightsID</td><td>The ID of the Rights a User may acquire on the Event.</td></tr>
<tr><td>ParentItemID</td><td>The ID of the Entity “from which the Event is derived”.</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Event.</td></tr>
</table></td></tr>
</table>

6.8 Experience

<table>
<tr><td>Purpose</td><td>An Event as a User MM-Captured it and the User’s Interactions with the Entities belonging to the Entity that spawned the Event.</td></tr>
<tr><td>Data</td><td><table>
<tr><td>Time$_1$</td><td>Entity$_1$</td><td>Interaction$_1$</td><td>M-Location$_1$</td></tr>
<tr><td>Time$_2$</td><td>Entity$_2$</td><td>Interaction$_2$</td><td>M-Location$_2$</td></tr>
<tr><td>Time$_n$</td><td>Entity$_n$</td><td>Interaction$_n$</td><td>M-Location$_n$</td></tr>
</table></td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>ExperienceID</td><td>The ID of the Experience.</td></tr>
<tr><td>UserID</td><td>The ID of the User “having Rights to the Experience”.</td></tr>
<tr><td>InRightsID</td><td>The ID of the Rights “to Act on the Experience”.</td></tr>
<tr><td>OutRightsID</td><td>The ID of the Rights a User may acquire on the Experience.</td></tr>
<tr><td>ParentEntityID</td><td>The ID of the Event spawning the Experience.</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Experience.</td></tr>
</table></td></tr>
</table>

6.9 Identifier

<table>
<tr><td>Purpose</td><td>An Item that uniquely references an Item. An Item can have more than one Identifier.</td></tr>
<tr><td>Data</td><td>[M-InstanceID] [M-EnvironmentID] [M-Location] [ItemID].
Any of the elements preceding [ItemID] can be absent.
The Format of an Identifiers depends on the Technology it is based on, e.g.:
1. Multi-factor.
2. Device Biometrics (iris, fingertips, voice, face, gestures, hand motions, body motions, etc.).
3. Behavioural Biometrics.
4. Cryptographic Security Keys.
5. Certificate-based authentication.
6. Hardware Security Keys.
7. Device Identity Technologies.
8. Decentralised Identifiers (DIDs)
9. Self-Sovereign Identifiers (SSIs)</td></tr>
<tr><td>Metadata</td><td>No metadata</td></tr>
</table>

6.10 InformIn

Purpose	An Item containing a description of the Item on which information is requested.
Data	TBD

Metadata	InformInID	The ID of the InformIn Item.
	UserID	The ID of the User generating the InformIn Item.
	ServiceID	The ID of the Service providing Inform Services.
	RightsID	The ID of the Rights “to Act on the InformIn Item” granted to the Inform Service.
	DescrMdata	Any description of the InformIn Item.

6.11 InformOut

Purpose	An Item containing the description of the Item.	
Data	TBD	
Metadata	InformOutID	The ID of the InformOut Item.
	ServiceID	The ID of the Service providing the InformOut Item.
	UserID	The ID of the User having requested the InformIn Item.
	RightsID	The ID of the Rights “to Act on the InformOut Item” granted to UserID.
	DescrMdata	Any description of the InformOut Item.

6.12 Interaction

Purpose	An Item containing the list of Actions made by a User on the Entities MM-Embedded at an M-Locations and the corresponding Times.			
Data	Action_1	Entity_1	M-Location_1	Time_1
	Action_2	Entity_2	M-Location_2	Time_2
	Action_n	Entity_n	M-Location_n	Time_n
	Current Actions are Authenticate, Author, Discover, MU-Export, Hide, Identify, Interpret, MM-Add, MM-Capture, MM-Disable, MM-Embed, MM-Enable, MM-Send, MU-Render, Post, UM-Render, UM-Send, UM-Animate, UM-Sense, Transact.			
Metadata	InteractionID	The ID of the Interaction.		
	UserID	The ID of the User “having Rights to Act on the Interaction”.		
	InRightsID	The ID of the Rights “to Act on the Interaction”.		
	OutRights	The ID of the Rights a User may acquire on the Interaction.		
	EntityID	The ID of Entity “User Interacted with”.		
	DescrMdata	Any description of the Interaction.		

6.13 InterpretIn

Purpose	An Item containing a description of the Item to be Interpreted.	
Data	TBD	
Metadata	InterpretInID	The ID of the InterpretIn Item.
	UserID	The ID of the User generating the InterpretIn Item.
	ServiceID	The ID of the Service providing Interpret Services.
	RightsID	The ID of the Rights “to Act on the InterpretIn Item” granted to the Interpret Service.
	DescrMdata	Any description of the InterpretIn Item.

6.14 InterpretOut

Purpose	An Item containing the description of the Item.
Data	TBD

Metadata	InterpretOutID	The ID of the InterpretOut Item.
	ServiceID	The ID of the Service providing the InterpretOut Item.
	UserID	The ID of the User having requested the InterpretIn Item.
	RightsID	The ID of the Rights "to Act on the InterpretOut Item" granted to UserID.
	DescrMdata	Any description of the InterpretOut Item.

6.15 Ledger

Purpose	An Item containing a list of Transactions involving Assets.	
Data	TBD	
Metadata	LedgerID	The ID of the Ledger.
	UserID	The ID of the User who "has Rights on the Ledger".
	InRightsID	The ID of the Rights "to Act on the Ledger".
	OutRightsID	The ID of the Rights "to Act on the Ledger" a User may acquire.
	DescrMdata	Any descriptive Metadata.

6.16 Map

Purpose	An Item containing a structure establishing a correspondence between U-Locations with M-Locations.	
Data	TBD.	
Metadata	MapID	The ID of the Map.
	UserID	The ID of the User "having Rights to Act on the Map".
	InRightsID	The ID of the User Rights "to Act on the Map".
	OutRightsID	The ID of the Rights of a User may acquire on the Map.
	AuthorID	The ID of the User "who Authored the Map".
	AuthoringToolID	The ID of the Service "who provided the Authoring Tool".
	DescrMdata	Any description of the Map.

6.17 M-Environment

Purpose	A portion of an M-Instance covered by an Account	
Data	TBD	
Metadata	M-Envi-ronmentID	The ID of the M-Environment.
	UserID	The ID of the User "having Rights to Act on the M-Environment".
	InRightsID	The ID of the Rights "to Act on the M-Environment".
	Out-RightsID	The ID of the Rights a User may acquire on the M-Location.
	De-scrMdata	Any description of the M-Environment, e.g., about Persistence and Accessibility (Public/Private).

6.18 Message

Purpose	An Item containing application-specific Data MM-Sent by Source to Destination.	
Data	Source Destination Message content	
Metadata	MessageID	The ID of the Map.

	Descriptive Metadata	Any description of the Message.

6.19 M-Instance

Purpose	A Metaverse implementation.	
Data	TBD	
Metadata	M-InstanceID	The ID of the M-Instance.
	UserID	The ID of the User “having Rights to Act on the M-Instance”.
	InRightsID	The ID of the Rights “to Act on the M-Instance”.
	OutRightsID	The ID of the Rights a User may acquire on the M-Instance.
	DescrMdata	Any description of the M-Instance.

6.20 M-Location

Purpose	An identifiable delimited portion of an M-Environment.	
Data	TBD	
Metadata	M-LocationID	The ID of the M-Location Item.
	UserID	The ID of the User “having Rights to Act on M-Location”.
	InRightsID	The ID of the Rights “to Act on the M-Location.
	OutRightsID	The ID of the Rights of a User may acquire on the M-Location.
	DescrMdata	Any description of the M-Location.

6.21 Model

Purpose	An Object representing an object with its features ready to be UM-Animated by a Stream or to be MM-Animated by an autonomous agent.	
Data	Objects represented are: 1. An inanimate Object (e.g., a table) 2. An autonomous Object (e.g., a robot) 3. An animal, possibly with high accuracy 4. A human, possibly with high accuracy	
Metadata	ModelID	The ID of the Object Model.
	UserID	The ID of the User “having Rights to Act on Object Model”.
	InRightsID	The ID of the Rights “to Act on the Object Model”.
	OutRightsID	The ID of the Rights a User may acquire on the Object Model.
	AuthorID	The ID of the User “who Authored the Object Model”.
	Author-ingToolID	The ID of the Service “who provided the Authoring tool”.
	DescrMdata	Any description of the Model.

6.22 Object

Purpose	An Entity representing an object. Currently, the following types of Objects are supported: Audio, Visual, and Haptic.
Data	1. Audio Data representation 1.1. Mono (e.g., speech) 1.2. Stereo 1.3. Multichannel 1.4. Microphone array 1.5. Spatial Audio

<table>
<tr><td></td><td>2. Image Data representation
3. Visual Data representation
    3.1. Mono
    3.2. Camera array
    3.3. Light field
    3.4. Holography
4. Haptic Data representation</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>Object ID</td><td>The ID of the Object Identified by ObjectID.</td></tr>
<tr><td>UserID</td><td>The ID of the User “having Rights to Act on the Object”.</td></tr>
<tr><td>InRightsID</td><td>The ID of the Rights “to Act on the Object”.</td></tr>
<tr><td>OutRightsID</td><td>The ID of the Rights a User may acquire on the Object.</td></tr>
<tr><td>AuthorID</td><td>The ID of the User “who Authored the Object”.</td></tr>
<tr><td>AuthoringToolID</td><td>The ID of the Service “who provided the Authoring tool”</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Object.</td></tr>
</table></td></tr>
</table>

6.23 Personal Profile

<table>
<tr><td>Purpose</td><td>An Item containing the Data about the human represented by User.</td></tr>
<tr><td>Data</td><td>1. First Name
2. Last Name
3. Address
4. Country
5. Age
6. Interests
7. Biometric data
8. …</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>PersonalDataID</td><td>The ID of the Personal Data.</td></tr>
<tr><td>UserID</td><td>The ID of the User “Having Rights on the Personal Data”.</td></tr>
<tr><td>InRightsID</td><td>The ID of the Rights “to Act on the Personal Data”.</td></tr>
<tr><td>OutRightsID</td><td>The ID of the Rights a User may acquire on the Personal Data.</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Personal Data.</td></tr>
</table></td></tr>
</table>

6.24 Provenance

<table>
<tr><td>Purpose</td><td>The list of all Transactions executed on an Asset starting from the first and including the last.</td></tr>
<tr><td>Data</td><td><table>
<tr><td>Transaction$_1$</td><td>Time$_1$</td></tr>
<tr><td>Transaction $_2$</td><td>Time$_2$</td></tr>
<tr><td>Transaction $_n$</td><td>Time$_n$</td></tr>
</table></td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>ProvenanceID</td><td>The ID of the Provenance.</td></tr>
<tr><td>UserID</td><td>The ID of the User who “has Rights on the Provenance”.</td></tr>
<tr><td>InRightsID</td><td>The ID of the Rights “to Act on the Provenance”.</td></tr>
<tr><td>OutRightsID</td><td>The ID of the Rights “to Act on the Provenance” a User may acquire.</td></tr>
<tr><td>AssetID</td><td>The ID of the Asset the Provenance refers to.</td></tr>
<tr><td>DescrMdata</td><td>Any descriptive Metadata.</td></tr>
</table></td></tr>
</table>

6.25 Rights

Purpose	An Item expressing the ability of a User to perform an Action on an Item until a Time.
Data	Expiration Time.
Metadata	

RightsID	The ID of the Rights.
UserID	The IDs of the User "having Rights".
ActionID	The ID of the Action "User may perform".
ItemID	The ID of the Item "User can perform Actions on".
OutRightsID	The ID of the Rights "to Act on the Item" a User may acquire.
DescrMdata	Any description of the Rights.

6.26 Rules

Purpose	An Item expressing the terms and conditions under which a User operates in an M-Instance/Environment.
Data	Rules establish the Rights of a User to the Items they Act on an M-Instance/M-Environment. The jurisdiction of the M-Instance/M-Environment may specify Rights that must be granted to a User.
Metadata	

RulesID	The ID of the Rules.
UserID	The ID of the User having Rights on the Rules.
InRightsID	The ID of the Rights "to Act on the Rules".
M-InstanceID	The ID of the M-Instance "where the Rules hold (if an M-Instance)".
M-EnvironmentID	The ID of the M-Environment "where the Rules hold (if an M-Environment)".
DescrMdata	Any descriptive Metadata.

6.27 Scene

Purpose	A possibly hierarchical Composition of Objects each having a Spatial Attitude.
Data	TBD
Metadata	

SceneID	The ID of the Scene Identified by SceneID
UserID	The ID of the User "having Rights to Act on the Scene".
InRightsID	The ID of the Rights "to Act on the Scene".
OutRightsID	The ID of the Rights "to Act on the Scene" a User may acquire.
AuthorID	The ID of the User "who created the Scene".
Auth.ToolID	The ID of the Service "who provided the Creation tool".
DescrMdata	Any description of the Scene.

6.28 Social Graph

Purpose	A representation of the network of connections of a User with Items, Processes, and Services.
Data	TBD
Metadata	

SocialGraphID	The ID of the Social Graph.
UserID	The ID of the User "having Rights on the Social Graph".
inRightsID	The ID of the Rights "to Act on the Social Graph".
OutRightsID	The ID of the Rights "to Act on the Social Graph" a User may acquire.

	DescrMdata	Any description of the Social Graph.

6.29 Stream

Purpose	An Item made by a continuous flow of Data.	
Data	TBD	
Metadata	StreamID	The ID of the Stream.
	UserID	The ID of the User “having Rights to Act on the Stream”.
	InRightsID	The ID of Rights “to Act on the Stream”.
	OutRightsID	The ID of the Rights “to Act on the Stream” a User may acquire.
	DescrMdata	Any description of the Stream.

6.30 Transaction

Purpose	Item representing the changed state of the Account and the Rights of one or more Users and optionally of the Service facilitating/enabling the Transaction of an Asset: 1. The Value moving into the Wallet of User 1 (seller). 2. The Value moved from the Wallet of User2 (buyer). 3. The Value moved into the Wallet of User 3 (service) - optional. 4. The Time the Values were moved. 5. The Rights to Act owned by User1 before Time. 6. The Rights to Act owned by User2 after Time.	
Data	Value1	The Value moved into the seller’s Wallet.
	Value2	The Value moving from the buyer’s Wallet.
	Value3	The Value moved into the Marketplace’s Wallet (optional).
	Time	The Time when the Transaction is carried out.
Metadata	TransactionID	The ID of the Transaction.
	AssetID	The ID of the Asset the Transaction refers to.
	UserID	The ID of User1 “who grants the Rights”.
	InRightsID	The ID of the Rights of User1.
	WalletID1	The ID of the Wallet of UserID1.
	TargetUserID	The ID of the User2 “who is granted the Rights”.
	OutRightsID	The ID of the Rights “granted to User2”.
	TargetWalletID	The ID of the Wallet of UserID2.
	ServiceID	The ID of the Marketplace.
	ServiceWalletID	The ID of the Wallet of the Marketplace.
	DescrMdata	Any description of the Transaction.

6.31 U-Location

Purpose	An identifiable delimited portion of a Universe Environment.	
Data	TBD	
Metadata	U-LocationID	The ID of the U-Location.
	UserID	The ID of the User “having Rights to Act on the U-Location”.
	RightsID	The ID of the Rights “to perform Actions on the U-Location”.
	OutRightsID	The ID of the Rights “to perform Actions on the U-Location” a User may acquire.
	DescrMdata	Any description of the U-Location.

6.32 User Data

Purpose	An Item containing Activity Data, Personae, Social Graph, and User Profile of a User.	
Data	TBD	
Metadata	UserDataID	ID of UserData.
	UserID	ID of User having Rights on UserData
	RightsID	ID of Rights held by User
	PersonaIDs	IDs of Personae held User.
	PersonalDataID	ID of Personal Data.
	ActivityDataID	ID of Activity Data
	SocialGraphID	ID of SocialGraph
	DescrMdata	Any description of the User.

6.33 Value

Purpose	An Amount and the Currency with which the Amount is expressed.	
Data	TBD	
Metadata	ValueID	The ID of the Value.
	UserID	The ID of the User who has used the Value for a Transaction.
	DescrMdata	Any description of the User.

6.34 Wallet

Purpose	A container of Currency units. In general, a Wallet is implemented outside of the Environment.	
Data	A list of Values with the Time of the last Transaction.	
Metadata	WalletID	The ID of the Wallet.
	UserID	The ID of the User “having Rights to the Wallet”.
	InRightsID	The ID of the Rights “User has on the Wallet”.
	DescrMdata	Any description of the User.

7 Data Types

Actions and Items may use several Data Types. Some Data Types may relate to a Metaverse Instance or the Universe; a U-/M- prefix may be added as needed.

7.1 Address

Purpose	The URL of a storage facility.
Data	TBD
Metadata	No Metadata.

7.2 Amount

Purpose	A number expressing a Value in a Currency.
Data	A decimal number.
Metadata	No Metadata.

7.3 Cognitive State

<table>
<tr><td>Purpose</td><td>The representation of a User’s Personal Status that reflects the way they understand the Environment, such as “Confused”, “Dubious”, “Convinced”.</td></tr>
<tr><td>Data</td><td>TBD</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>PersonaID</td><td>The ID of the Persona “the Cognitive State refers to”.</td></tr>
<tr><td>CognitiveStateID</td><td>The ID of the Cognitive State.</td></tr>
<tr><td>DescrMetadata</td><td>Any description of the Cognitive State.</td></tr>
</table></td></tr>
</table>

7.4 Coordinates

<table>
<tr><td>Purpose</td><td>A set of numbers representing a Position in a Metaverse Environment using a coordinate system.</td></tr>
<tr><td>Data</td><td>TBD</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>Coordinate System</td><td>The ID of the coordinate system.</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Cognitive State.</td></tr>
</table></td></tr>
</table>

7.5 Currency

Purpose	A medium of exchange enabling Transactions in a Metaverse Environment.
Data	CurrencyID.
Metadata	No Metadata.

7.6 Emotion

<table>
<tr><td>Purpose</td><td>The representation of a User’s Personal Status that results from their interaction with an Environment, such as “Angry”, “Sad”, “Determined”.</td></tr>
<tr><td>Data</td><td>TBD</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>PersonaID</td><td>The ID of the Persona “the Emotion refers to”.</td></tr>
<tr><td>EmotionID</td><td>The ID of the Emotion Identified by EmotionID.</td></tr>
<tr><td>DescrMdata</td><td>Any description of the Emotion.</td></tr>
</table></td></tr>
</table>

7.7 Orientation

Purpose	The set of the 3 roll, pitch, yaw angles indicating the rotation around the principal axis (x) of an Object, its y axis having an angle of 90˚ counterclockwise (right-to-left) with the x axis and its z axis (pointing up toward the viewer viewing from above).
Data	TBD
Metadata	Any descriptive Metadata.

7.8 Personal Status

<table>
<tr><td>Purpose</td><td>The representation of the information internal to a User characterising their behaviour.</td></tr>
<tr><td>Data</td><td>TBD</td></tr>
<tr><td>Metadata</td><td><table>
<tr><td>PersonalStatusID</td><td>The ID of the Personal Status Identified by PersonalStatusID.</td></tr>
<tr><td>PersonaID</td><td>The ID of the Persona “with PersonalStatusID”.</td></tr>
<tr><td>DescrMetadata</td><td>Any descriptive Metadata.</td></tr>
</table></td></tr>
</table>

7.9 Point of View

Purpose	The Spatial Attitude of a Persona watching an Environment.
Data	As in Spatial Attitude.
Metadata	PersonaID: The ID of the Persona "with PersonalStatusID".

7.10 Position

Purpose	The Coordinates of a point in a Metaverse Environment using a Coordinate system.
Data	TBD
Metadata	CoordinateSystemID: ID of the Coordinate System used to express the Position.

7.11 Social Attitude

Purpose	The representation of a User's Personal Status representing the way User intends to position vis-à-vis other Users, e.g., "Respectful", "Confrontational", "Soothing".
Data	TBD
Metadata	PersonaID: The ID of the Persona "the Social Attitude refers to". SocialAttitudeID: The ID of the Social Attitude. DescrMetadata: Any description of the Social Attitude.

7.12 Spatial Attitude

Purpose	The Position and Orientation of an Entity, and their velocities and accelerations.
Data	TBD
Metadata	Any descriptive Metadata.

7.13 Time

Purpose	The representation of the measure of time.
Data	TBD
Metadata	Any descriptive Metadata

8 Use Cases

8.1 Introduction

This Chapter collects Metaverse Use Cases to facilitate the development of Functionality Profiles. Use Cases are populated by Users that request to perform Actions on different types of Items. In a Use Case, Users are identified by one subscript and Items by the same subscript of the User performing an Action on the Item followed by a sequential number. For instance:

1. $User_i$ MM-Embeds $Persona_{i.j}$ at $M\text{-}Location_{i.k}$, etc.
2. $User_i$ MU-Renders $Entity_{i.j}$ at $U\text{-}Location_{i.k}$, etc.
3. $User_i$ MM-Sends $Object_{1.j}$ to $User_k$.

Note1 The following abbreviations will be used: A = Audio, AV = Audio-Visual, AVH = Audio-Visual-Haptic, SA=Spatial Attitude.

Note2 The Basic Actions of a Composite Action are not listed unless they are independently used by the Use Case.

8.2 Virtual Lecture

8.2.1 Description

A student attends a lecture held by a teacher in a classroom created by a school manager:

1. A School Manager
   1.1. Authors and embeds a virtual classroom.
   1.2. Pays the teacher.
2. A Teacher
   2.1. Embeds its persona from home at the classroom's desk.
   2.2. Embeds and animates a 3D model.
   2.3. Leaves the classroom.
3. A Student
   3.1. When "at home" pays to attend and make a copy of their lecture Experience.
   3.2. Then Embeds a persona in the classroom.
   3.3. Goes close to the teacher's desk to feel the 3D model with haptic gloves.
   3.4. Stores their lecture Experience.
   3.5. Leaves the classroom.

The detailed workflow is:

1. User$_1$ (Manager):
   1.1. Authors Entity$_{1.1}$.
   1.2. MM-Embeds Entity$_{1.1}$ at M-Location$_{1.1}$.
2. User$_3$ (Teacher):
   2.1. Tracks Persona$_{3.1}$ (AV) at M-Location$_{3.1}$ with SA (establishes a connection between the human and the Persona).
   2.2. MM-Embeds Persona$_{3.1}$ at M-Location$_{3.2}$ (desk in classroom).
   2.3. MM-Disables Persona$_{3.1}$ at Location$_{3.1}$.
   2.4. MM-Embeds Model$_{3.1}$ at M-Location$_{3.3}$ (close to M-Location$_{3.2}$).
   2.5. MM-Animates Model$_{3.1}$.
3. User$_2$ (Student):
   3.1. Tracks Persona$_{2.1}$ (AV) at M-Location$_{2.1}$ with SA.
   3.2. Transacts Value$_{2.1}$.
   3.3. MM-Embeds Persona$_{2.1}$ (AV) at M-Location$_{2.2}$ with SA.
   3.4. MM-Disables Persona$_{2.1}$ at Location$_{2.1}$.
4. User$_3$ (Teacher):
   4.1. MM-Embeds Model$_{3.1}$ (AVH) at M-Location$_{3.3}$ (close to M-Location$_{3.2}$).
   4.2. MM-Animates Model$_{3.1}$.
5. User$_2$ (Student)
   5.1. MM-Adds Persona$_{2.1}$ (AV) at M-Location$_{2.3}$ (close to the desk).
   5.2. MU-Captures Model$_{3.1}$ (at AVH).
   5.3. MU-Exports Experience$_{2.1}$ at Address$_{2.1}$.
6. User$_1$ (Manager):
   6.1. Transacts Value$_{1.1}$ to User$_3$ (Teacher).
7. User$_3$ (Teacher):
   7.1. MM-Disables Persona$_{3.1}$ from M-Location$_{3.2}$
   7.2. MM-Embeds Persona$_{3.1}$ at M-Location$_{3.1}$.
8. User$_2$ (Student)
   8.1. MM-Disables Persona$_{2.1}$ from M-Location$_{2.2}$
   8.2. MM-Embeds Persona$_{2.1}$ at M-Location$_{2.1}$.

8.2.2 Workflow and Actions

Table 10 – Virtual Lecture workflow and Actions.

$User_1$ (Manager)	Authors	$Entity_{1.1}$	(VClassroom).
	MM-Embeds	$Entity_{1.1}$	M-$Location_{1.1}$.
$User_3$ (Teacher)	Tracks	$Persona_{3.1}$ (AV)	M-$Location_{3.1}$ w/ SA.
	MM-Embeds	$Persona_{3.1}$	M-$Location_{3.2}$ (desk).
	MM-Disables	$Persona_{3.1}$	M-$Location_{3.1}$
	MM-Embeds	$Model_{3.1}$	M-$Location_{3.3}$ (close to desk).
	MM-Animates	$Model_{3.1}$.	
$User_2$ (Student)	Tracks	$Persona_{2.1}$ (AV)	M-$Location_{2.1}$ w/ SA.
	Transacts	$Value_{2.1}$	(Lecture & Experience)
	MM-Embeds	$Persona_{2.1}$ (AV)	$Location_{2.2}$ with SA.
	MM-Disables	$Persona_{2.1}$	M-$Location_{2.1}$.
$User_3$ (Teacher)	MM-Embeds	$Model_{3.1}$ (AVH)	M-$Location_{3.3}$ (close to desk).
	MM-Animates	$Model_{3.1}$.	
$User_2$ (Student)	MM-Adds	$Persona_{2.1}$ (AV)	M-$Location_{2.3}$ (close to desk).
	MU-Captures	$Model_{3.1}$ (AVH)	
	MU-Exports	$Experience_{2.1}$	$Address_{2.1}$
$User_1$ (Manager)	Transacts	$Value_{1.1}$	$User_3$ (Lecture fees).
$User_3$ (Teacher)	MM-Disables	$Persona_{3.1}$	M-$Location_{3.2}$
	MM-Embeds	$Persona_{3.1}$	M-$Location_{3.1}$
$User_2$ (Student)	MM-Disables	$Persona_{2.1}$	M-$Location_{2.2}$
	MM-Embeds	$Persona_{2.1}$	M-$Location_{2.1}$

8.2.3 Actions, Items, and Data Types

Table 11 – Virtual Lecture Actions, Items, and Data Types.

Actions	Items	Data Types
Authenticate	Service	Amount
Author	Entity	Coordinates
MM-Embed	M-Location	Currency
MM-Disable	U-Location	Spatial Attitude
MM-Animate	Value	Value
Register	User	Orientation
Track	Persona	Position
Transact	Experience	
MU-Export		

8.3 Virtual Meeting

8.3.1 Description

A participant attends a meeting in a room created by a meeting manager. The manager deploys a virtual secretary to produce a summary of the conversations, enriched by information about participants' personal statuses. The participant gets a translation of sentences uttered in languages other than their own and makes a presentation using a 3D model.

This is the workflow of the use case:

1. $User_1$ (Meeting Manager)
   1.1. MM-Embeds a virtual meeting room at an M-$Location_{1.1}$.
   1.2. MM-Embeds $Persona_{1.1}$ (a Virtual Secretary, a Process MM-Animating a Persona of $User_1$ at M-$Location_{1.2}$).
   1.3. MM-Animates the Virtual Secretary.
2. $User_2$ (a meeting participant):
   2.1. Tracks $Persona_{2.1}$ at $Location_{2.1}$ ("its home").
   2.2. MM-Embeds $Persona_{2.1}$ (AV) at M-$Location_{2.2}$ ("the meeting room". A Persona can be MM-Enabled as Audio only, Audio-Visual, or Audio-Visual-Haptic).
   2.3. MM-Disables $Persona_{2.1}$ (AV) from $Location_{2.1}$ (disappears from "home").
   2.4. Interprets (requests translation of speech) of $Persona_{3.1}$ of $User_3$ (2^{nd} meeting participant).
   2.5. MM-Embeds $Entity_{2.1}$ (a 3D model) at M-$Location_{2.3}$ ("place in the meeting room").
   2.6. MM-Animates $Entity_{2.1}$ (to make the presentation controlling the 3D model).
3. Virtual Secretary:
   3.1. Interprets $Persona_{3.1}$'s Personal Status (requests estimation of the internal state of $Persona_{3.1}$, possibly in addition to $Persona_{3.1}$'s Speech Objects translation).
   3.2. Produces a text Summary of $Persona_{3.1}$'s Speech Object (with added graphical signs expressing $Persona_{3.1}$'s Personal Status).
   3.3. MM-Embeds the Summary at an M-$Location_{1.3}$ ("meeting room" for participants to comment).

8.3.2 Workflow and Actions

Table 12 – Virtual Meeting workflow and actions.

Who	Does	What	Where/comment
$User_1$ (Manager)	MM-Embeds	$Entity_{1.1}$	(VMeeting room) M-$Location_{1.1}$
	MM-Embeds	$Persona_{1.1}$	(Virtual Secretary) M-$Location_{1.2}$
	MM-Animates	$Persona_{1.1}$	Operates Virtual Secretary.
$User_2$ (Participant)	Tracks	$Persona_{2.1}$ (AV)	M-$Location_{2.1}$ w/ SA
	MM-Embeds	$Persona_{2.1}$ (AV)	M-$Location_{2.2}$ w/ SA
	MM-Disables	$Persona_{2.1}$ (AV)	M-$Location_{2.1}$
$User_3$ (Participant)	Tracks	$Persona_{3.1}$ (AV)	M-$Location_{3.1}$ w/ SA
	MM-Embeds	$Persona_{3.1}$ (AV)	M-$Location_{3.2}$ w/ SA
	MM-Disables	$User_3$	M-$Location_{3.1}$
$User_2$ (Participant)	Interprets	$Persona_{3.1}$	(Requests translation)
	MM-Embeds	$Model_{2.1}$	(3D presentation) M-$Location_{2.2}$
	MM-Animates	$Model_{2.1}$	
Virtual Secretary	Interprets	$Persona_{2.1}$	(Personal Status)
	Produces	$Entity_{1.2}$	(Summary)
	MM-Embeds	$Entity_{1.2}$	M-$Location_{1.3}$ (VMeeting room)
	MM-Disables	$Persona_{1.1}$	M-$Location_{1.2}$
$User_2$ (Participant)	MU-Exports	$Event_{2.1}$	$Address_{2.1}$
	MM-Embeds	$Persona_{2.1}$ (AV)	M-$Location_{2.1}$ (back home)
	MM-Disables	$Persona_{2.1}$ (AV)	$Location_{2.2}$
$User_3$ (Participant)	MM-Embeds	$Persona_{3.1}$ (AV)	M-$Location_{2.1}$ (back home)

	MM-Disables	$Persona_{3.1}$ (AV)	$Location_{3.2}$ (back home)

8.3.3 Actions, Items, and Data Types

Table 13 – Virtual Meeting Actions, Items, and Data Types.

Actions	Items	Data Types
Authenticate	Entity	Coordinates
Call	Persona	Orientation
Interpret	Service	Position
MM-Animate	User	Spatial Attitude
MM-Capture		
MM-Disable		
MM-Embed		
MU-Export		
Register		
Track		

8.4 Hybrid working

8.4.1 Description

A company applies mixed in-presence and remote working policy.

1. Physical Workers attend Company physically.
2. All Workers
   2.1. Are Authenticated.
   2.2. Are present in the Virtual office.
   2.3. Communicate by Sharing AV messages (except R-worker to R-worker).
   2.4. Participate in Virtual meetings.

This is the detailed workflow:

1. $User_1$ (Manager):
   1.1. Authors $Entity_{1.1}$ (Virtual office).
   1.2. Embeds $Entity_{1.1}$ (Virtual office).
   1.3. Embeds $Entity_{1.2}$ at $Location_{1.1}$ (Office gateway)
   1.4. MM-Animates $Persona_{1.1}$ as “Virtual time clock”.
2. real worker:
   2.1. Comes to real office.
   2.2. Registers.
3. $User_2$ (R-worker):
   3.1. MM-Embeds $Persona_{2.1}$ at $M\text{-}Location_{2.1}$ (Virtual office).
4. $User_3$ (V-worker):
   4.1. Tracks $Persona_{3.1}$ at $M\text{-}Location_{3.1}$
   4.2. MM-Embeds $Persona_{3.1}$ at $Location_{3.2}$ (Virtual office).
5. User1 (“virtual time clock”) Authenticates:
   5.1. $User_2$ (R-worker)
   5.2. $User_3$ (V-worker).
6. $User_3$ (V-worker):
   6.1. MM-Disables $Persona_{3.1}$ at $Location_{3.1}$

6.2. MM-Sends $Object_{3.1}$ (A) to $User_2$ (R-worker).
6.3. MM-Embeds $Persona_{3.1}$ at $Location_{3.3}$ (close to R-worker).
6.4. MM-Disables $Persona_{3.1}$ at $Location_{3.2}$
6.5. MM-Embeds $Persona_{3.1}$ at $Location_{3.4}$ (Vmeeting room).
6.6. MM-Disables $Persona_{3.1}$ at $Location_{3.3}$

7. $User_2$ (R-worker)
   7.1. MM-Embeds $Model_{2.1}$ (VWiteboard) at $M\text{-}Location_{2.2}$ (Vmeeting room).
   7.2. MM-Animates VWhiteboard
   7.3. MM-Disables $Persona_{2.1}$ at $Location_{2.2}$ (Vmeeting room).
8. $User_3$ (V-worker):
   8.1. MM-Embeds $Persona_{3.1}$ at $Location_{3.1}$ (Home).
   8.2. MM-Disables $Persona_{3.1}$ at $Location_{3.1}$ (Vmeeting room).

8.4.2 Workflow and Actions

Table 14 – Hybrid Working workflow and actions.

Who	Does	What	Where/comment
$User_1$ (Manager)	Authors	$Entity_{1.1}$ (AV)	V-Office
	MM-Embeds	$Entity_{1.1}$	$M\text{-}Location_{1.1}$
	MM-Embed	$Persona_{1.1}$ (AV)	$M\text{-}Location_{1.2}$ (VTime clock)
	MM-Animates	$Persona_{1.1}$ (AV)	$M\text{-}Location_{1.2}$
$User_2$ (R-Worker)	Tracks	$Persona_{2.1}$ (AV)	$M\text{-}Location_{2.1}$ (VOffice)
	Registers		
	MM-Embeds	$Persona_{2.1}$	$M\text{-}Location1_{2.1}$ (VOffice)
$User_3$ (V-Worker)	Tracks	$Persona_{3.1}$ (AV)	$M\text{-}Location_{3.1}$ (home)
	MM-Embeds	$Persona_{3.1}$	$M\text{-}Location_{3.2}$ w/ SA (V-Desk)
$User_1$ (VTime clock)	Authenticates	$User_2$	
	Authenticates	$User_3$	
$User_3$ (V-Worker)	MM-Sends	$Objects_{3.1}$ (A)	$Persona_{2.1}$ (AV)
	MM-Embeds	$Persona_{3.1}$	$M\text{-}Location_{3.3}$ (talk “in person”)
	MM-Disables	$Persona_{3.1}$ (AV)	$M\text{-}Location_{3.2}$
	MM-Embeds	$Persona_{3.1}$	$M\text{-}Location_{3.4}$ (V-Meeting)
	MM-Disables	$Persona_{3.1}$ (AV)	$M\text{-}Location_{3.3}$
$User_2$ (R-Worker)	MM-Embeds	$Persona_{2.1}$	$M\text{-}Location_{3.4}$ (V-Meeting)
	MM-Disables	$Persona_{3.1}$ (AV)	$M\text{-}Location_{2.2}$
	MM-Embeds	$Entity_{2.1}$	(Whiteboard) $M\text{-}Location_{3.4}$
	MM-Animate	$Entity_{2.1}$	To operate Whiteboard
	MM-Embeds	$Persona_{2.1}$ (AV)	$M\text{-}Location_{2.1}$ (back home)
	MM-Disables	$Persona_{2.1}$	From $M\text{-}Location_{3.4}$
$User_3$ (V-Worker)	MM-Embeds	$Persona_{3.1}$ (AV)	$M\text{-}Location_{3.1}$ (back home)
	MM-Disables	$Persona_{3.1}$ (AV)	From $M\text{-}Location_{3.4}$

8.4.3 Actions, Items, and Data Types

Table 15 – Hybrid Working Actions, Items, and Data Types

Actions	Items	Data Types

Author	Entity	Coordinates
Call	M-Location	Orientation
MM-Disable	Object (A)	Position
MM-Embed	Persona (AV)	Spatial Attitude
MM-Send	Service	
Track	U-Location	
	User	

8.5 eSports Tournament

8.5.1 Description

A site manager makes a game landscape available to a game manager. The game manager deploys autonomous characters and places virtual camera and microphone in the landscape. They display the whole landscape onto a big screen and stream the whole landscape.

This is the detailed workflow:

1. $User_1$ (Site Manager)
    1.1. Authors $Entity_{1.1}$ (game landscape).
    1.2. MM-Embeds $Entity_{1.1}$ (game landscape) at $M\text{-}Location_{1.1}$.
2. $User_2$ (Game Manager)
    2.1. MM-Embeds $Personae_{2.i}$ with Spatial Attitude at $M\text{-}Locations_{2.i}$ (Autonomous characters).
    2.2. MM-Animates $Personae_{2.i}$.
    2.3. Calls $Service_{2.1}$ (virtual camera/microphone control).
    2.4. MU-Renders the Entities at $M\text{-}Location_{1.1}$ (landscape) to:
        2.4.1. $U\text{-}Location_{2.1}$ via $Device_{1.1}$ (screen).
        2.4.2. Various U-Locations (via streaming).
3. $User_3$ (Player) tracks $Persona_{3.1}$ (AV) at $Location_{3.1}$ with Spatial Attitude
4. Calls $Process_{3.1}$ to provide role-specific:
    4.1. Costumes (e.g., magician, warrior).
    4.2. Forms, physical features, and abilities (e.g., cast spells, shoot, fly, jump).

8.5.2 Workflow

Table 16 – eSports Tournament workflow and actions.

$User_1$ (Site Manager)	Authors	$Entity_{1.1}$	(Game landscape).
	MM-Embeds	$Entity_{1.1}$	(Landscape) at $M\text{-}Location_{1.1}$.
$User_2$ (Game Manager)	MM-Embeds	$Personae_{2.i}$ w/SA	$M\text{-}Locations_{2.i}$ (AA).
	MM-Animates	$Personae_{2.i}$.	$M\text{-}Locations_{2.i}$ (AA).
	Calls	$Service_{2.1}$	(Vcamera/microphone control).
	MU-Renders	Entities	(Landscape) to: $U\text{-}Location_{2.1}$ (via screen). U-Locations (via streaming).
$User_3$ (Player)	Tracks	$Persona_{3.1}$ w/ SA	$Location_{3.1}$
	Calls	$Process_{3.1}$	to provide role-specific: Costumes (magician, warrior). Forms, features, abilities.

8.5.3 Actions, Items, and Data Types

Table 17 – eSports Tournament Actions, Items, and Data Types.

Actions	Items	Data Types
Author	User	Spatial Attitude
Call	Persona (AV)	Coordinates
MM-Animate	Entity	Orientation
MM-Embed	Service	Position
Track	U-Location	
	M-Location	

8.6 Virtual performance

8.6.1 Description

Participant buys a ticket for an event with the right to stay close to the performance stage for 5 minutes. The event is organised by an organiser who has created a virtual auditorium and generates special effects by calling a service to collect participants' preferences and another service to extract the participants status. The participant utters a private speech to another participant.

1. $User_1$ (Organiser)
    1.1. Transacts $Value_{1.1}$ (buys a parcel at $M\text{-}Location_{1.1}$).
    1.2. Authors $Entity_{1.1}$ (virtual auditorium).
    1.3. MM-Embeds $Entity_{1.1}$ at $M\text{-}Location_{1.1}$.
    1.4. Calls $Service_{1.1}$ (to collect Users' Preferences).
2. $User_2$ (Performer)
    2.1. Tracks $Persona_{2.1}$ at $Location_{2.1}$.
    2.2. Embeds $Persona_{2.1}$ (AV) with Spatial Attitude at $MLocation_{2.2}$ (in virtual auditorium).
    2.3. MM-Disables $Persona_{2.1}$ from $Location_{2.1}$.
3. $User_3$ (Participant)
    3.1. Tracks $Persona_{3.1}$ at $M\text{-}Location_{3.1}$ (at home).
    3.2. Transacts $Value_{3.1}$ (buys ticket).
    3.3. Embeds $Persona_{3.1}$ (AV) with Spatial Attitude at $Location_{3.2}$ (in virtual auditorium).
    3.4. MM-Disables $Persona_{3.1}$ (AV) from $Location_{3.1}$.
    3.5. MM-Sends $Object_{3.1}$(A) to $Persona_{4.1}$ (Participant).
    3.6. Calls $Service_{1.1}$ (expresses preferences).
    3.7. MM-Adds $Persona_{3.1}$ at $Location_{3.2}$ (close to stage for 5 minutes).
4. $User_1$ (Organiser)
    4.1. MM-Disables $Persona_{3.1}$ from $Location_{3.2}$ (5 minutes passed).
    4.2. Calls $Service_{1.1}$ (Collects preferences).
    4.3. Interprets Participants Status (of all participants).
    4.4. MM-Embeds $Entities_{1.i}$ (SFX).
    4.5. Transacts $Value_{1.2}$ to $User_2$ (performance fees).
5. $User_2$ (Performer)
    5.1. MM-Embeds $Persona_{2.1}$ (AV) to $M\text{-}Location_{2.1}$.
    5.2. MM-Disables $Persona_{2.1}$ from $M\text{-}Location_{2.2}$.
6. $User_3$ (Participant)
    6.1. MM-Embeds $Persona_{3.1}$ (AV) to $M\text{-}Location_{3.1}$.

6.2. MM-Disables $Persona_{3.1}$ from $M\text{-}Location_{3.2}$.

8.6.2 Workflow and Actions

Table 18 – Virtual Event workflow and actions.

$User_1$ (Organiser)	Transacts	$Value_{1.1}$	(Parcel at $M\text{-}Location_{1.1}$).
	Authors	$Entity_{1.1}$	(Virtual auditorium).
	MM-Embeds	$Entity_{1.1}$	$M\text{-}Location_{1.1}$.
	Calls	$Service_{1.1}$	Collect Users' Preferences).
$User_2$ (Performer)	Tracks	$Persona_{2.1}$	$Location_{2.1}$.
	Embeds	$Persona_{2.1}$ (AV) w/ SA	$MLocation_{2.2}$ (Vauditorium).
	MM-Disables	$Persona_{2.1}$	at $Location_{2.1}$.
$User_3$ (Participant)	Tracks	$Persona_{3.1}$	$M\text{-}Location_{3.1}$ (at home).
	Transacts	Value	(Buys ticket).
	Embeds	$Persona_{3.1}$ (AV) w/ SA	$Location_{3.2}$ (in Vauditorium).
	MM-Disables	$Persona_{3.1}$ (AV)	$Location_{3.1}$.
	MM-Sends	$Object_{3.1}$ (A)	$Persona_{4.1}$ (Participant).
	Calls	$Service_{1.1}$	(Expresses preferences).
	MM-Adds	$Persona_{3.1}$	$Location_{3.2}$ (close to stage).
$User_1$ (Organiser)	MM-Disables	$Persona_{3.1}$	$Location_{3.2}$ (after 5').
	Calls	$Service_{1.1}$	(Collects preferences).
	Interprets	Participants $Status_{1.1}$.	
	MM-Embeds	$Entities_{1.i}$	(SFX).
	Transacts	$Value_{1.2}$	$User_2$ (performance fees).
$User_2$ (Performer)	MM-Embeds	$Persona_{2.1}$ (AV)	$M\text{-}Location_{2.1}$.
	MM-Disables	$Persona_{2.1}$	$M\text{-}Location_{2.2}$.
$User_3$ (Participant)	MM-Embeds	$Persona_{3.1}$ (AV)	$M\text{-}Location_{3.1}$.
	MM-Disables	$Persona_{3.1}$	$M\text{-}Location_{3.2}$.

8.6.3 Actions, Items, and Data Types

Table 19 – Virtual Event Actions, Items, and Data Types.

Actions	**Items**	**Data Types**
Author	Entity (AV)	Amount
Call	Object (A)	Cognitive State
Interpret	Persona (AV)	Coordinates
MM-Disable	M-Location	Currency
MM-Embed	Service	Emotion
MM-Send	U-Location	Orientation
Register	User	Personal Status
Track	Value	Position
Transact		Social Attitude
		Spatial Attitude
		Value

8.7 AR Tourist Guide

8.7.1 Description

In his Use Case $human_3$ engages the following humans:

1. $human_1$ to cause their $User_1$ to buy a virtual parcel and develop a virtual landscape suitable for a tourist application.
2. $human_2$ to cause their $User_2$ to develop scenes and autonomous agents for the different places of the landscape.
3. $human_4$ to create an app that alerts the holder of a smart phone where the app is installed.
4. $human_5$ holding a smart phone with the app to perceive Entities and talk to Personae MM-Embedded at M-Locations and MM-Animated by autonomous agents (AA).

This is the detailed workflow of the Use Case:

1. $User_1$
   1.1. Buys $M\text{-}Location_{1.1}$ (parcel) in an M-Environment.
   1.2. Authors $Entity_{1.1}$ (landscape suitable for a virtual path through n sub-M-Locations).
   1.3. Embeds $Entity_{1.1}$ (landscape) on $M\text{-}Location_{1.1}$ (parcel).
   1.4. Sells $Entity_{1.1}$ (landscape) and $M\text{-}Location_{1.1}$ (parcel) to a $User_2$.
2. $User_2$
   2.1. Authors $Entity_{2.1}$ to $Entity_{2.n}$ for the M-Locations.
   2.2. Embeds the Entities at $M\text{-}Location_{2.1}$ to $M\text{-}Location_{2.n}$.
   2.3. Sells the result to $User_3$.
3. $human_4$
   3.1. Develops
      3.1.1. Map recording the pairs $M\text{-}Location_{2.i}$ – $U\text{-}Location_{2.i}$
      3.1.2. App alerting a $human_5$ holding the Device with the App installed that a key U-Location has been reached.
   3.2. Sells Map and App to $human_3$.
4. $User_3$ MM-Embeds one or more autonomous Personae at $M\text{-}Location_{2.1}$ to $M\text{-}Location_{2.n}$.
5. $human_5$ gets close to key $U\text{-}Location_{5.1}$:
6. $App_{5.1}$ MM-Sends $Message_{5.1}$ to $Device_{5.1}$
7. $Device_{5.1}$
   7.1. MU-Renders $Entity_{5.1}$ MM-Embedded at $M\text{-}Location_{5.1}$ to key $U\text{-}Location_{5.1}$.
   7.2. MU-Rendered $Entity_{5.1}$.
   7.3. MU-Animated $Persona_{5.1}$.
   7.4. UM-Animated $Persona_{5.2}$.

8.7.2 Workflow

Table 20 – AR Tourist Guide workflow.

Who	**Does**	**What**	**Where/comment**
$User_1$	Transacts	$Value_{1.1}$	$M\text{-}Location_{1.1}$'s parcel.
	Authors	$Entity_{1.1}$	(Path of n M-Locations).
	Embeds	$Entity_{1.1}$	(Parcel).
	Transacts	$Value_{1.1}$	(Landscape & parcel to $User_2$).
$User_2$	Authors	$Entity_{2.1}$ to $Entity_{2.n}$	(For $M\text{-}Locations_{2.1\text{-}2.n}$).
	Embeds	$Entity_{2.1\text{-}2.n}$	$M\text{-}Location_{2.1\text{-}2.n}$

	$Transacts_{2.1}$	$Entity_{2.1-2.n}$	$User_{3}$.
$human_{4}$	develops	Map	M-Locations & U-Locations
	develops	App	
	sells	Map and App	To $human_{3}$.
$User_{3}$	MM-Embeds	Personae	M-$Location_{2.1-2.n}$.
	MM-Animates	Personae	M-$Location_{2.1-2.n}$.
$human_{5}$	comes to		At key U-$Location_{2.i}$.
$App_{5.1}$	MM-Sends	$Message_{5.1}$	$Device_{5.1}$
$Device_{5.1}$	MU-Renders	$Entity_{5.1}$ @M-$Location_{5.1}$	At key U-$Location_{5.1}$.
	MU-Renders	MM-Animated $Persona_{5.1}$	At key U-$Location_{5.1}$.
	MU-Renders	UM-Animated $Persona_{5.2}$	At key U-$Location_{5.1}$.

8.7.3 Actions, Items, and Data Types

Table 21 – AR Tourist Guide Actions, Items, and Data Types.

Actions	**Items**	**Data Types**
Author	App	Amount
Author	Device	Coordinates
MM-Animate	Entity	Currency
MM-Animate	Map	Orientation
MM-Embed	M-Location	Position
MM-Send	Persona	Spatial Attitude
MU-Export	Service	Value
MU-Render	U-Location	
Send	User	
Transact		
UM-Animate		

8.8 Virtual Dance

8.8.1 Description

This Use Cases envisages that:

1. A dance teacher:
   1.1. Is in its office.
   1.2. Places a virtual secretary in the dance school animated by an autonomous agent.
2. A student #1:
   2.1. Is at home.
   2.2. Shows up at the school.
   2.3. Greets the secretary.
3. The secretary
   3.1. Reciprocates the greeting.
   3.2. Sends a private vocal message to the teacher.
4. The teacher:
   4.1. Places its $persona_{2.1}$ (AVH) in the dance school.
   4.2. Dances with student #1.
5. A student #2:
   5.1. Is at home.

    5.2. Shows up at school.
6. The teacher:
    6.1. Places its $persona_{2.1}$ (AVH) close to student #2.
    6.2. Places its $persona_{2.3}$ (AVH) where $persona_{2.1}$ was before.
    6.3. Animates its $persona_{2.3}$ with autonomous agent to dance with student #1.
    6.4. Dances with student #2.

1. $User_{2}$ (dance teacher)
    1.1. Tracks $Persona_{2.1}$ at $M\text{-}Location_{2.1}$
    1.2. MM-Embeds $Persona_{2.2}$ (AV) (another of its Personae) at $M\text{-}Location_{2.2}$.
    1.3. MM-Animates $Persona_{2.2}$ (AV) (as virtual secretary to attend to students coming to learn dance).
2. $User_{1}$ (dance student #1):
    2.1. MM-Embeds its $Persona_{1.1}$ (AV) at $Location_{1.1}$ (its "home").
    2.2. MM-Embeds $Persona_{1.1}$ (AVH) at $Location_{1.2}$ close to virtual secretary).
    2.3. MM-Sends $Object_{1.1}$ (A) to $Persona_{2.2}$ (greets virtual secretary).
    2.4. MM-Disables $Persona_{1.1}$ from $Location_{1.1}$.
3. $User_{2}$ ($Persona_{2.2}$):
    3.1. MM-Sends $Object_{2.1}$ (A) (to dance student #1 to reciprocate greeting).
    3.2. MM-Send $Object_{2.2}$ (A) (to call teacher's $Persona_{2.1}$).
4. Dance teacher ($Persona_{2.1}$):
    4.1. MM-Embeds (AVH) $Persona_{2.1}$ at $Location_{2.3}$ (classroom).
    4.2. UM-Animates $Persona_{2.1}$ (dances with student #1).
5. While $Persona_{1.1}$ (student #1) and $Persona_{2.1}$ (teacher) dance, $User_{3}$ (dance student #2):
    5.1. MM-Embeds (AV) $Persona_{3.1}$ (its digital twin) at $Location_{3.1}$ (its "home").
    5.2. MM-Embeds (AVH) $Persona_{3.1}$ at $Location_{3.2}$ (close to secretary).
    5.3. MM-Disables $Persona_{3.1}$ from $Location_{3.1}$.
6. After a while, $User_{2}$ (dance teacher):
    6.1. MM-Embeds (AVH) $Persona_{2.1}$ at $Location_{2.4}$ (close to student #2's position).
    6.2. MM-Disables $Persona_{2.1}$ (from where it was dancing with student #1).
    6.3. MM-Embeds (AVH) $Persona_{2.3}$
    6.4. MM-Animates $Persona_{2.3}$ with autonomous agent (to dance with student #1).

8.8.2 Workflow

Table 22 – Virtual Dance workflow.

$User_{2}$ (teacher)	Tracks	$Persona_{2.1}$	$M\text{-}Location_{2.1}$
	MM-Embeds (AV)	$Persona_{2.2}$	$M\text{-}Location_{2.2}$.
	MM-Animates (AV)	$Persona_{2.2}$	(As VS for students).
$User_{1}$ ($student_{1}$)	MM-Embeds (AV)	$Persona_{1.1}$	$Location_{1.1}$ (its "home").
	MM-Embeds (AVH)	$Persona_{1.1}$	$Location_{1.2}$ (close to VS).
	MM-Sends	$Object_{1.1}$ (A)	$Persona_{2.2}$ (greets VS).
	MM-Disables	$Persona_{1.1}$	from $Location_{1.1}$.
$User_{2}$ ($Persona_{2.1}$)	MM-Sends	$Object_{2.1}$ (A)	(Responds to student #1).
	MM-Send	$Object_{2.2}$ (A)	(Calls teacher's $Persona_{2.1}$).
$User_{2}$ ($Persona_{2.2}$)	MM-Embeds (AVH)	$Persona_{2.1}$	$Location_{2.3}$ (classroom).
$User_{2}$ ($Persona_{2.2}$)	UM-Animates	$Persona_{2.1}$	(Dances with student #1).

$User_3$ ($student_2$)	MM-Embeds (AV)	$Persona_{3.1}$	$Location_{3.1}$ (its "home").
	MM-Embeds (AVH)	$Persona_{3.1}$	$Location_{3.2}$ (close to VS).
	MM-Disables	$Persona_{3.1}$	from $Location_{3.1}$.
$User_2$ (teacher)	MM-Embeds (AVH)	$Persona_{2.1}$	$Location_{2.4}$ (near $student_2$).
	MM-Disables	$Persona_{2.1}$	(From previous position).
	MM-Embeds (AVH)	$Persona_{2.3}$	
	MM-Animates	$Persona_{2.3}$	(w/ AA with student #1).

8.8.3 Actions, Items, and Data Types

Table 23 – Virtual Dance Actions, Items, and Data Types.

Actions	**Items**	**Data Types**
MM-Animate	M-Location	Amount
MM-Disable	Object (A)	Currency
MM-Embed	Persona (AV)	Orientation
MM-Send	Persona (AVH)	Position
Track	Service	Spatial Attitude
Transact	U-Location	
	Value	

8.9 Virtual Car Showroom

8.9.1 Description

This Use Cases envisages that:

1. A car dealer
   1.1. Is in its office.
   1.2. Embeds a persona animated by an autonomous agent in the car showroom (attendant).
2. A customer in its home
   2.1. Embeds its persona in the car showroom.
   2.2. Greets the showroom attendant.
3. The Showroom attendant
   3.1. Reciprocates the greeting.
   3.2. Privately calls the dealer.
4. The dealer
   4.1. Animates the attendant with the human car dealer.
   4.2. Converses with the customer.
   4.3. Embeds a 3D AVH model of a car.
5. The customer
   5.1. Interacts with the model (has a virtual drive).
   5.2. Buys the car.
   5.3. Returns home.

This is the detailed workflow of the Use Case:

1. $User_1$ (car dealer):
   1.1. Tracks $Persona_{1.1}$ at M-$Location_{1.1}$ ("office").
   1.2. MM-Embeds $Persona_{1.2}$ with Spatial $Attitude_{1.1}$ at M-$Location_{1.2}$ ("showroom").

1.3. MM-Animates Persona$_{1.2}$ (with an autonomous agent – showroom attendant).
2. User$_2$ (customer):
   2.1. Tracks Persona$_{2.1}$ at M-Location$_{2.1}$ (“home”).
   2.2. MM-Embeds Persona$_{2.1}$ at M-Location$_{2.1}$ (“in the showroom”).
   2.3. MM-Sends Object$_{1.1}$ (A) to Persona$_{1.2}$ (greets showroom attendant).
   2.4. MM-Disables Persona$_{2.1}$ at M-Location$_{2.1}$ (“home”).
3. User$_1$ (Persona$_{1.2}$):
   3.1. MM-Sends Object$_{1.1}$ (A) to Persona$_{2.1}$ (responds to greetings).
   3.2. MM-Sends Object$_{1.2}$ (A) to Persona$_{1.1}$ (“come attend customer”).
4. User$_1$ (Persona$_{1.1}$)
   4.1. MM-Embeds Persona$_{1.1}$ at M-Location$_{1.3}$ (“in the showroom”).
   4.2. MM-Sends Object$_{1.2}$ (A) to Persona$_{2.1}$ (engages in conversation).
   4.3. MM-Embeds Model$_{1.1}$ (AVH) at M-Location$_{1.4}$ (model car “in the showroom”).
   4.4. MM-Animates Model$_{1.1}$ (“animate model car”).
5. User$_2$ (customer)
   5.1. MM-Embeds Persona$_{2.1}$ at M-Location$_{2.3}$ (where the virtual car is located)
   5.2. UM-Animates Persona$_{2.1}$.
   5.3. Transacts Value$_{2.1}$ (buys car).
   5.4. MM-Disables Persona$_{2.1}$ at M-Location$_{1.3}$.
   5.5. MM-Embeds Persona$_{2.1}$ at M-Location$_{2.1}$ (“at home”).

8.9.2 Workflow

Table 24 – Virtual Car Showroom workflow.

User$_1$ (car dealer)	Tracks	Persona$_{1.1}$	M-Location$_{1.1}$ (“office”).
	MM-Embeds	Persona$_{1.2}$ w/ SA$_{1.1}$	M-Location$_{1.2}$ (“showroom”).
	MM-Animates	Persona$_{1.2}$ w/ AA	(Showroom attendant).
User$_2$ (customer)	Tracks	Persona$_{2.1}$	M-Location$_{2.1}$ (“home”).
	MM-Embeds	Persona$_{2.1}$	M-Location$_{2.1}$ (“showroom”).
	MM-Sends	Object$_{1.1}$ (A)	Persona$_{1.2}$ (greets attendant).
	MM-Disables	Persona$_{2.1}$	M-Location$_{2.1}$ (“home”).
User$_1$ (Persona$_{1.2}$)	MM-Sends	Object$_{1.1}$ (A)	Persona$_{2.1}$ (responds to greetings).
	MM-Sends	Object$_{1.2}$ (A)	Persona$_{1.1}$ (“attend customer”).
User$_1$ (Persona$_{1.1}$)	MM-Embeds	Persona$_{1.1}$	M-Location$_{1.3}$ (“showroom”).
	MM-Sends	Object$_{1.2}$ (A)	Persona$_{2.1}$ (converses).
	MM-Embeds	Model$_{1.1}$ (AVH)	M-Location$_{1.4}$ (“in showroom”).
	MM-Animates	Model$_{1.1}$ w/ AA	(“Animate model car”).
User$_2$ (customer)	MM-Embeds	Persona$_{2.1}$	M-Location$_{2.3}$ (in virtual car)
	UM-Animates	Persona$_{2.1}$	(Drives virtual car)
	Transacts	Value$_{2.1}$	(Buys car).
	MM-Disables	Persona$_{2.1}$	M-Location$_{1.3}$.
	MM-Embeds	Persona$_{2.1}$	M-Location$_{2.1}$ (“at home”).

8.9.3 Actions, Items, and Data Types

Table 25 – Virtual Car Showroom Actions, Items, and Data Types.

Actions	**Items**	**Data Types**

Call	Object (A)	Amount
Embeds	Persona	Currency
MM-Animate	Scene (AVH)	Orientation
MM-Disable	Value	Position
MM-Embed		Spatial Attitude
MM-Send		
Track		
Transacts		
UM-Animate		

8.10 Drive a Connected Autonomous Vehicle

8.10.1 Description

MPAI is developing a Technical Report that includes a reference model of a Connected Autonomous Vehicle (CAV) based on the subdivision of a CAV in 4 subsystem each implemented as a workflow of AI Modules executed in the standard MPAI-AIF framework [5]. A CAV has the capability to autonomously reach a U-Location by understanding human utterances, planning a Route, sensing the U-environment, building a representation of it, exchanging such representations with other CAVs and CAV-aware entities, making decisions about how to execute the Route, and acting on the Motion Actuation Subsystem to implement the decision (see Figure 2).

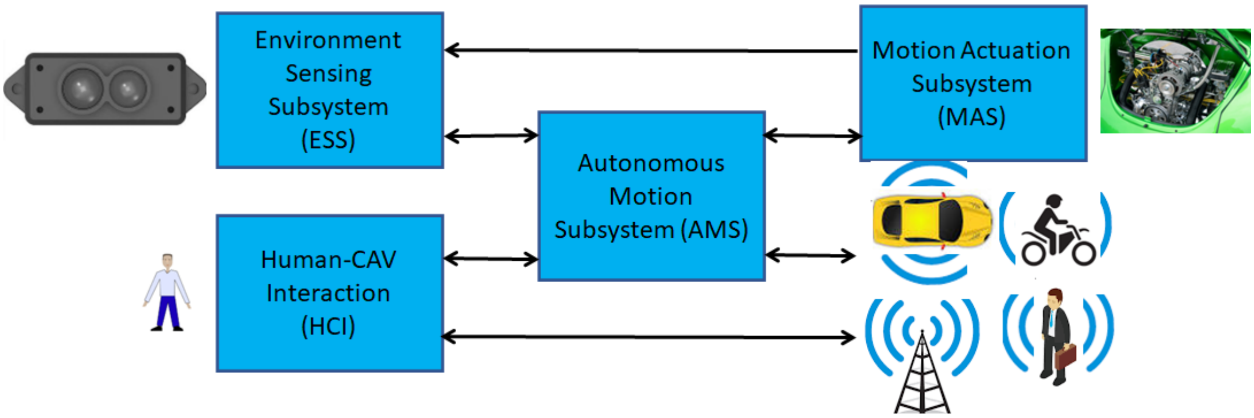

Figure 2 - MPAI-CAV Reference Model

This Use Case assumes that a human having rights to or owning a CAV *Registers* with the CAV by providing:

1. The requested subset of their Personal profile.
2. Two User Processes required to operate a CAV:
   2.1. $User_1$ to operate the Human-CAV Interaction Subsystem.
   2.2. $User_2$ to operate the Autonomous Motion Subsystem.
3. $User_1$'s Personae.
4. WalletIDs.

The Processes of a CAV generate a persistent M-Instance resulting from the integration of:

1. The Environment Representation generated by the Environment Sensing Subsystem by *UM-Captur*ing the U-Location being traversed by the CAV.
2. The Scenes *MM-Capture*d from the M-Locations of the M-Instances that are digital twins of the U-Locations being traversed by other CAVs that are close to the U-Location of the Ego CAV and improve the accuracy of the CAV's M-Locations.

3. Relevant Experiences of the Autonomous Motion Subsystem at the M-Location.

For simplicity, the Use Case assumes that there are only two CAVs: CAV_A and CAV_B. The convention of having Users identified by a sequential number is extended as follows: "a User of CAV_A is identified by A followed by a sequential number". Items affected by $User_{A.1}$ are identified as A.1 followed by a sequential number.

1. $User_{A.1}$
   1.1. Authenticates the human and gives it driving instructions.
   1.2. Requests a series of options satisfying the instructions.
2. $User_{A.2}$
   2.1. Gets information about CAV_A position.
   2.2. Gets options from Route Planner.
   2.3. Communicates options to $User_{A.1}$.
3. $User_{A.1}$
   3.1. Converts the options to an utterance.
   3.2. Communicates the utterance to the human.
4. human sends utterance with the option to $User_{A.1}$
5. $User_{A.1}$ converts utterance to a command.
6. $User_{A.2}$
   6.1. Authenticates its peer $User_{B.2}$
   6.2. Gets the Environment Representation from its ESS and $User_{A.2}$
   6.3. Sends a command to the Motion Actuation Subsystem.
   6.4. Requests permission to place $Persona_{A.1.1}$ in the virtual cabin of CAV_B.
   6.5. Receives permission and places $Persona_{A.1.1}$
   6.6. Watches the landscape through the window of CAV_B.

This is the detailed workflow:

1. $human_A$ Registers.
2. $User_{A.1}$
   2.1. Tracks $Persona_{A.1.1}$ at $M\text{-}Location_{A.1.1}$ (connects CAV_A's M-Instance with U-Location).
   2.2. Authenticates $Object_{A.1.1}$(AV) (recognises $human_A$).
   2.3. Interprets $Object_{A.1.1}$(A) ($human_A$'s request to drive).
   2.4. MM-Sends $HCI\text{-}AMSCommand_{A.1.1}$ to $User_{A.2}$.
3. $User_{A.2}$
   3.1. MM-Sends ESS's $Scene_{A.2.1}$ to RoutePlanner.
   3.2. MM-Sends $Route_{A.2.1}$ to $User_{A.1}$.
4. $User_{A.1}$
   4.1. MU-Renders $Object_{A.1.2}$ (A) (to $human_A$).
   4.2. UM-Renders $Object_{A.1.3}$ (A) ($human_A$'s Route selection).
   4.3. Interprets $Object_{A.1.3}$ (A) (understand Route).
   4.4. MM-Sends $HCI\text{-}AMSCommand_{A.1.2}$ to $User_{A.2}$.
5. $User_{A.2}$
   5.1. Authenticates $User_{B.2}$.
   5.2. MM-Sends
      5.2.1. ESS's $Scene_{A.2.2}$ to Environment Representation Fusion (ERF).
      5.2.2. $Scene_{A.2.3}$ at $M\text{-}Location_{A.2.1}$ (in CAV_B's M-Instance) to ERF.
      5.2.3. $Path_{A2.1}$ to Motion Planner.
      5.2.4. $Trajectory_{A.2.1}$ to Obstacle Avoider (does not request a new Trajectory).
      5.2.5. $Trajectory_{A.2.1}$ to Command Issuer.
      5.2.6. $AMS\text{-}MASCommand_{A.2.1}$ to Motion Actuation Subsystem.

    5.2.7. MAS-AMS Response$_{A.2.1}$.
6. User$_{A.1}$
   6.1. Authenticates User$_{A.2}$.
   6.2. MM-Sends Object$_{A.1.4}$ (A) (request to see CAV$_B$ virtual cabin) to User$_{B.1}$.
7. User$_{B.1}$ MM-Sends Object$_{B.1.1}$ (A) (request accepted) to User$_{A.1}$.
8. User$_{A.1}$ MM-Capture Scene$_{A.1.4}$ (AV) (request to Embed Persona$_{1.1}$ in CAV$_B$ virtual cabin) to User$_{B.1}$.

User$_{A.1}$ can now view and navigate the M-instance created by CAV$_A$ traversing the U-Environment (M-Location$_{A.1.2}$) and Device$_{A.1.}$ allows User$_{A.1}$ to be MM-Embedded, i.e., to become part of the M-Location$_{A.1.1}$ (User$_{B.1}$'s cabin).

8.10.2 Workflow

Table 26 – Drive a Connected Autonomous Vehicle workflow.

Who	Does	What	Where/(comment)
human$_A$	Registers		(With CAV$_A$).
User$_{A.1}$	Tracks	Persona$_{A.1.1}$	(M-Location$_{A.1.1}$ connects U-Location$_{A.1.1}$).
	Authenticates	Object$_{A.1.1}$(AV)	(Recognises human$_A$).
	Interprets	Object$_{A.1.1}$(A)	(human$_A$'s request to go).
	MM-Sends	HCI-AMSCmd$_{A.1.1}$	User$_{A.2}$.
User$_{A.2}$	MM-Sends	ESS's Scene$_{A.2.1}$	Route Planner.
	MM-Sends	AMS-HCIResp$_{A.2.1}$	Route$_{A.2.1}$ to User$_{A.1}$
User$_{A.1}$	MU-Renders	Object$_{A.1.2}$ (A)	(To human$_A$).
	UM-Renders	Object$_{A.1.3}$ (A)	(Route selection).
	Interprets	Object$_{A.1.3}$ (A)	(Understand Route).
	MM-Sends	HCI-AMSCmd$_{A.1.2}$	User$_{A.2}$
User$_{A.2}$	Authenticates	User$_{B.2}$	
	MM-Sends	ESS's Scene$_{A.2.2}$	(To Environment Representation Fusion).
		Scene$_{A.2.3}$	M-Location$_{A.2.1}$ (in CAV$_B$'s to ERF).
		Path$_{A2.1}$	Motion Planner.
		Trajectory$_{A.2.1}$	Obstacle Avoider.
		Trajectory$_{A.2.1}$	Command Issuer.
		AMS-MASCmd$_{A.2.1}$	MAS.
		MAS-AMS Resp$_{A.2.1}$.	From MAS.
User$_{A.1}$	Authenticates	User$_{A.2}$.	
	MM-Sends	Object$_{A.1.4}$ (A)	(Request to Embed to User$_{B.1}$)
User$_{B.1}$	MM-Sends	Object$_{B.1.1}$ (A)	(Request accepted to User$_{A.1)}$.
User$_{A.1}$	MM-Embeds	Persona$_{A.1.1}$	M-Location$_{A.1.2}$ (User$_{B.1}$'s cabin).
	MM-Captures	Scene$_{A.1.1}$	M-Location$_{A.1.3}$ (out of User$_{B.1}$'s cabin).

8.10.3 Actions, Items, and Data Types

Note: The MPAI-CAV specific Items are included.

Table 27 – Drive a Connected Autonomous Vehicle Actions, Items, and Data Types.

Action	Item	Data Types
Authenticate	AMS-MASCommand	Spatial Attitude

Interpret	AMS-HCIResponse	Coordinates
MM-Capture	Environment Representation	Orientation
MM-Capture	HCI-AMSCommand	Position
MM-Embed	MAS-AMS Response	
MM-Send	M-Location	
MM-Send	Object (A)	
MU-Render	Object (AV)	
Register	$\text{Path}_{\text{A2.1}}$	
Request	Persona	
Request	Route	
Track	Scene	
UM-Render	Trajectory	
	User	

9 Functionality Profiles

9.1 Profile elements

Table 1 lists the currently identified Actions, Items and Data Types. Actions in italic require more than one Action to be executed.

Table 28 – Metaverse Actions, Entities, and Data Types

Actions	Items	Data Types
Authenticate	Account	Address
Author	Activity Data	Amount
Change	App	Cognitive State
Discover	Asset	Coordinates
MU-Export	Device	Currency
Hide	DiscoverIn	Emotion
Identify	DiscoverOut	Personal Status
Inform	Event	Point
Interpret	Experience	Point of View
MM-Add	Identifier	Social Attitude
MM-Animate	InformIn	Spatial Attitude
MM-Capture	InformOut	Time
MM-Embed	InterpretIn	Orientation
MM-Enable	InterpretOut	Position
MM-Disable	Interaction	
MM-Send	Ledger	
MU-Render	Map	
MU-Actuate	M-Environment	
MM-Send	Message	
Post	M-Instance	
Read	M-Location	
Register	Model	
Track	Object	
Transact	Persona	

UM-Animate	Personal Profile	
UM-Capture	Process	
UM-Render	Provenance	
UM-Send	Rights	
	Rules	
	Scene	
	Service	
	Social Graph	
	Stream	
	Transaction	
	U-Environment	
	U-Location	
	User	
	User Data	
	Value	
	Wallet	

9.2 Profile structure

The structure of the Metaverse Functionality Profiles derived from the above includes hierarchical Profiles and independent Profiles. Profiles may have Levels. As depicted in Figure 3, the currently identified Profiles are Baseline, Management, Finance, and High. Details are provided in the next Sections. The currently identified Levels for Baseline, Management, and High Profiles are Audio only, Audio-Visual, and Audio-Visual-Haptic.

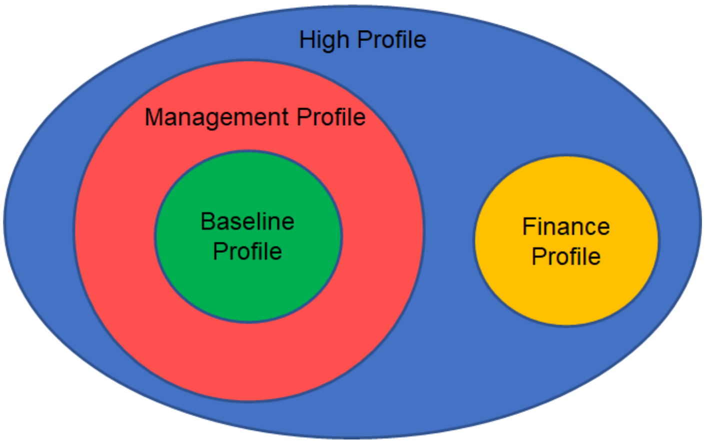

Figure 3 - The currently identified Functionality Profiles

The allocation of Actions and Items is made for The Baseline, Finance, and Management Profiles. The identified four Profiles serve well the needs conveyed by the identified Functionalities. As more of them will be added, the number of Profiles and potentially of Levels, is likely to increase.

9.3 Baseline Functionality Profile

The Baseline Functionality Profile enables a human equipped with a Device to allow their Users to perform the functions of Table 29. Basic forms of baseline lecture, meeting, and hang-out are supported but not Transactions and User management.

Table 29 - Actions and Items of the Baseline Functionality Profile

Functions	Action	Items
Call Service	Call	Service, Item
Read Items and Data	Read	Item
Create and Identify Item	Identify	Item, Identifier
Make Item inaccessible	Hide	Item
Author Entity	Author	Item
Place Entity at an M-Loc, no perception	MM-Add	Entity, M-Loc
Make MM-Added Entity perceptible	MM-Enable	Entity
Place an Entity at M-Loc, w/ perception	MM-Embed	Entity, M-Loc
Animate Model with an autonomous Process	MM-Animate	Model
Make available Object to a User	MM-Capture	Entity
Stop perception of Scene	MM-Disable	Entity, M-Loc
Render at U-Loc Entity at M-Loc	MU-Render	Entity, M-Loc, U-Loc
Make scene at U-Loc available to a Device	UM-Capture	U-Loc
Make Data of a Device available to a Process	UM-Send	Device, Process
Transfer Data between Processes	MM-Send	Message
Animate Model @M-Loc w/ Data from U-Loc	UM-Animate	Model, U-Loc
Place Entity @ M-Loc w/ Data from U-Loc	UM-Render	Entity, M-Loc, U-Loc
Send an Entity at an M-Loc to a Device	MM-Send	Entity, Device, M-Loc
Place & Animate Model at M-Loc, Render	Track	Model, Stream
Store Item	MU-Export	Item
App triggers perception of Entities		Map

Note that this Use Case implies: M-Instance, User and that Entity implies Scene, Object, Model, Persona.

Table 30 lists the Actions, Entities, and Data Types of the Baseline Functionality Profile.

Table 30 – Actions, Entities, and Data Types of the Baseline Functionality Profile

Actions	Author	Call	Identify	Destroy
	MU-Export	MM-Add	MM-Animate	MM-Capture
	MM-Embed	MM-Disable	MM-Enable	MM-Send
	MU-Render	Read	Track	UM-Animate
	UM-Capture	UM-Render	UM-Send	
Items	Identifier	Map	Message	M-Instance
	M-Location	Model	Object	Process
	Scene	Service	Stream	U-Location
	User			
Data Types	Address	Coordinates	Orientation	Position
	Spatial Attitude			

9.4 Finance Functionality Profile

The Financial Functionality Profile enables a User to Post Assets and make Transactions. As depicted in *Figure 5,* this Profile is independent of the Basic and Management Functionality Profiles.

It contains a subset of the Actions, Items, and Data Types of the Baseline Profile and adds Financia-related Actions, Items, and Data Types. Currently, this Profile does not have Levels.

The Finance Functionality Profile enables a human equipped with a Device to enable their Users to perform the functions of Table 31. The Table does not include Actions and Item already introduced in the Baseline Profile.

Table 31 - Actions and Items of the Finance Functionality Profile

Functions	Action	Items
Register	Register	M-Environment, Account, Activity Data Personal Profile, Rules, Social Graph
Submit Asset to marketplace	Post	Asset
Make a Transaction of an Asset	Transact	Asset, Ledger, Provenance, Rights, Transactions, Value, Wallet
Discover Assets	Discover	Request-Discover, Response-Discover
Get information on Asset, User	Inform	Request-Inform, Response-Inform
Change User Rights	Change	

Table 32 lists the Actions, Entities, and Data Types of the Finance Functionality Profile.

Table 32 – Actions, Entities, and Data Types of the Finance Profile

Actions	Change	Discover	Inform	Post
	Register	Transact		
Items	Account	Activity Data	Asset	Ledger
	M-Environment	Personal Profile	Provenance	Request-Discover
	Request-Inform	Response-Discover	Response-Inform	Rights
	Rules	Social Graph	Transaction	Value
	Wallet			
Data Types	Amount	Currency	Time	

9.5 Management Functionality Profile

The Management Functionality Profile supports all Actions, Items, and Data Types of the Baseline Profile and those enabling a controlled ecosystem where humans Register, Users are Authenticated, and advanced Services, such as Discover and Interpret, can be Called.

The Management Functionality Profile extends the capabilities of the Baseline Profile to perform the functions of Table 33. The Table does not include Actions and Item already introduced in the Baseline and Finance Functionality Profiles. As depicted in *Figure 5* the Management Functionality Profile is a superset of the Baseline Profile.

Table 33 - Actions and Items of the Baseline Functionality Profile

Functions	Action	Items
Register with an M-Environment	(Register)	M-Environment
Check that an Entity is what it says it is	Authenticate	Request-Authenticate, Response-Authenticate
Request interpretation of Item	Interpret	Request-Interpret,

		Response-Interpret
Save the Experience of an Event	(MU-Export)	Interaction, Experience, Event

Table 34 lists the Actions, Entities, and Data Types of the Management Functionality Profile that are not already included in the Baseline and Finance Profile.

Table 34 – Actions, Entities, and Data Types of Management Profile

Actions	Authenticate	Interpret		
Items	Event	Experience	Interaction	M-Environment
	Request-Authenticate	Response-Authenticate	Request-Interpret	Response-Interpret
Data Types	Cognitive State	Emotion	Personal Status	Social Attitude

9.6 High Functionality Profile

This Profile includes all other Profiles. The list of Actions, Entities, and Data Types required for this Profile is provided by Table 28.

10 Conclusions

This Technical Report provides the following foundational elements supporting the MPAI roadmap targeting Interoperability of Metaverse Instances/Environments. It is organised as follows:

1. Definitions
2. Metaverse walkthrough
3. Basic Metaverse elements:
   3.1. Actions
   3.2. Items
   3.3. Data Types
4. Use Cases
5. Functionality Profiles.

This Technical Report demonstrates the feasibility of the first two milestones of the MPAI roadmap to Metaverse Interoperability by identifying and specifying four Functionality Profiles supporting application-oriented Functionalities. New Functionalities Profiles will require more Metaverse Action and Items.

The next step is the development of Technical Specification – MPAI Metaverse Model (MPAI-MMM) – Metaverse Architecture.

11 References

1. MPAI; Technical Report – MPAI Metaverse Model – Functionalities (MPAI-MMM); January 2023; https://mpai.community/standards/mpai-mmm/mpai-metaverse-model/mmm-functionalities/
2. Matt White; Synthetic Reality: AI and the Metaverse; 2023 February 16; https://matthewdwhite.medium.com/synthetic-reality-ai-and-the-metaverse-5c2acf5a3fe6

3. ISO; ISO/IEC 23005 – Media context and control.
4. MPAI; Technical Specification: The Governance of the MPAI Ecosystem V1, 2021; https://mpai.community/standards/resources/#GME.
5. MPAI; Technical Specification: Artificial Intelligence Framework (MPAI-AIF) V1.1; https://mpai.community/standards/resources/#AIF. Also available as IEEE Standard 3301-2022.

Annex 1 - MPAI Basics

In recent years, Artificial Intelligence (AI) and related technologies have been introduced in a broad range of applications, have started affecting the life of millions of people and are expected to do so even more in the future. As digital media standards have positively influenced industry and billions of people, so AI-based data coding standards are expected to have a similar positive impact. Indeed, research has shown that data coding with AI-based technologies is generally *more efficient* than with existing technologies for, e.g., compression and feature-based description.

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

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

As a rule, MPAI standards include four documents: Technical Specification, Reference Software Specifications, Conformance Testing Specifications, and Performance Assessment Specifications. The last type of Specification includes standard operating procedures to enable users of MPAI Implementations to make informed decision about their applicability based on the notion of Performance, defined as a set of attributes characterising a reliable and trustworthy implementation.

In the following, If a Term begins with a small letter, it has the commonly used meaning and if with a capital letter, it has either the meaning defined in *Table 1* if it is specific to this Technical Report and in *Table 35* if it is common to all MPAI Standards.

In general, MPAI Application Standards are defined as aggregations – called AI Workflows (AIW) – of processing elements – called AI Modules (AIM) – executed in an AI Framework (AIF). MPAI defines Interoperability as the ability to replace an AIW or an AIM Implementation with a functionally equivalent Implementation.

MPAI also defines 3 Interoperability Levels of an AIF that executes an AIW. The AIW and its AIMs may have 3 Levels:
Level 1 – Implementer-specific and satisfying the MPAI-AIF Standard.
Level 2 – Specified by an MPAI Application Standard.
Level 3 – Specified by an MPAI Application Standard and certified by a Performance Assessor.

MPAI offers Users access to the promised benefits of AI with a guarantee of increased transparency, trust and reliability as the Interoperability Level of an Implementation moves from 1 to 3. Additional information on Interoperability Levels is provided in reference [4].

Figure 4 depicts the MPAI-AIF Reference Model under which Implementations of MPAI Application Standards and user-defined MPAI-AIF Conforming applications operate [5].

MPAI Application Standards normatively specify the Syntax and Semantics of the input and output data and the Function of the AIW and the AIMs, and the Connections between and among the AIMs of an AIW.

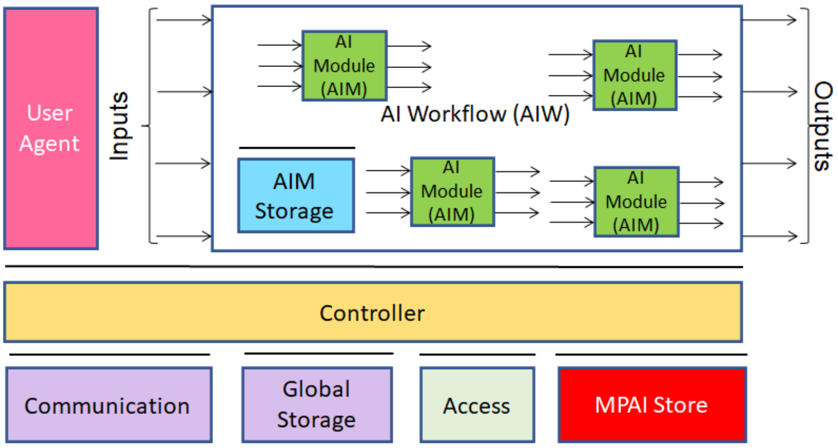

Figure 4 – The AI Framework (AIF) Reference Model

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

MPAI Standards are designed to enable a User to obtain, via standard protocols, an Implementation of an AIW and of the set of corresponding AIMs and execute it in an AIF Implementation. The MPAI Store in *Figure 4* is the entity from which Implementations are downloaded. MPAI Standards assume that the AIF, AIW, and AIM Implementations may have been developed by independent implementers. A necessary condition for this to be possible, is that any AIF, AIW, and AIM implementations be uniquely identified. MPAI has appointed an ImplementerID Registration Authority (IIDRA) to assign unique ImplementerIDs (IID) to Implementers.[2]

A necessary condition to make possible the operations described in the paragraph above is the existence of an ecosystem composed of Conformance Testers, Performance Assessors, the IIDRA and an instance of the MPAI Store. Reference [4] provides an example of such ecosystem.

[2] At the time of publication of this Technical Report, the MPAI Store was assigned as the IIDRA.

Annex 2 - MPAI-wide terms and definitions

The Terms used in this Technical Report whose first letter is capital and are not already included in Table 1 are defined in *Table 35*.

Table 35 – MPAI-wide Terms

Term	Definition
Access	Static or slowly changing data that are required by an application such as domain knowledge data, data models, etc.
AI Framework (AIF)	The environment where AIWs are executed.
AI Model (AIM)	A data processing element receiving AIM-specific Inputs and producing AIM-specific Outputs according to according to its Function. An AIM may be an aggregation of AIMs.
AI Workflow (AIW)	A structured aggregation of AIMs implementing a Use Case receiving AIW-specific inputs and producing AIW-specific outputs according to the AIW Function.
Application Standard	An MPAI Standard designed to enable a particular application domain.
Channel	A connection between an output port of an AIM and an input port of an AIM. The term “connection” is also used as synonymous.
Communication	The infrastructure that implements message passing between AIMs
Composite AIM	An AIM aggregating more than one AIM.
Component	One of the 7 AIF elements: Access, Communication, Controller, Internal Storage, Global Storage, Store, and User Agent
Conformance	The attribute of an Implementation of being a correct technical Implementation of a Technical Specification.
Conformance Tester	An entity Testing the Conformance of an Implementation.
Conformance Testing	The normative document specifying the Means to Test the Conformance of an Implementation.
Conformance Testing Means	Procedures, tools, data sets and/or data set characteristics to Test the Conformance of an Implementation.
Connection	A channel connecting an output port of an AIM and an input port of an AIM.
Controller	A Component that manages and controls the AIMs in the AIF, so that they execute in the correct order and at the time when they are needed
Data Format	The standard digital representation of data.
Data Semantics	The meaning of data.
Ecosystem	The ensemble of actors making it possible for a User to execute an application composed of an AIF, one or more AIWs, each with one or more AIMs potentially sourced from independent implementers.
Explainability	The ability to trace the output of an Implementation back to the inputs that have produced it.
Fairness	The attribute of an Implementation whose extent of applicability can be assessed by making the training set and/or network open to testing for bias and unanticipated results.

Function	The operations effected by an AIW or an AIM on input data.
Global Storage	A Component to store data shared by AIMs.
Internal Storage	A Component to store data of the individual AIMs.
Identifier	A name that uniquely identifies an Implementation.
Implementation	1. An embodiment of the MPAI-AIF Technical Specification, or 2. An AIW or AIM of a particular Level (1-2-3) conforming with a Use Case of an MPAI Application Standard.
Implementer	A legal entity implementing MPAI Technical Specifications.
ImplementerID (IID)	A unique name assigned by the ImplementerID Registration Authority to an Implementer.
ImplementerID Registration Authority (IIDRA)	The entity appointed by MPAI to assign ImplementerID's to Implementers.
Interoperability	The ability to functionally replace an AIM with another AIW having the same Interoperability Level
Interoperability Level	The attribute of an AIW and its AIMs to be executable in an AIF Implementation and to: 1. Be proprietary (Level 1) 2. Pass the Conformance Testing (Level 2) of an Application Standard 3. Pass the Performance Testing (Level 3) of an Application Standard.
Knowledge Base	Structured and/or unstructured information made accessible to AIMs via MPAI-specified interfaces
Message	A sequence of Records transported by Communication through Channels.
Normativity	The set of attributes of a technology or a set of technologies specified by the applicable parts of an MPAI standard.
Performance	The attribute of an Implementation of being Reliable, Robust, Fair and Replicable.
Performance Assessment	The normative document specifying the Means to Assess the Grade of Performance of an Implementation.
Performance Assessment Means	Procedures, tools, data sets and/or data set characteristics to Assess the Performance of an Implementation.
Performance Assessor	An entity Assessing the Performance of an Implementation.
Profile	A particular subset of the technologies used in MPAI-AIF or an AIW of an Application Standard and, where applicable, the classes, other subsets, options and parameters relevant to that subset.
Record	A data structure with a specified structure
Reference Model	The AIMs and theirs Connections in an AIW.
Reference Software	A technically correct software implementation of a Technical Specification containing source code, or source and compiled code.
Reliability	The attribute of an Implementation that performs as specified by the Application Standard, profile and version the Implementation refers to, e.g., within the application scope, stated limitations, and for the period of time specified by the Implementer.
Replicability	The attribute of an Implementation whose Performance, as Assessed by a Performance Assessor, can be replicated, within an agreed level, by another Performance Assessor.

Robustness	The attribute of an Implementation that copes with data outside of the stated application scope with an estimated degree of confidence.
Scope	The domain of applicability of an MPAI Application Standard
Service Provider	An entrepreneur who offers an Implementation as a service (e.g., a recommendation service) to Users.
Standard	The ensemble of Technical Specification, Reference Software, Conformance Testing and Performance Assessment of an MPAI application Standard.
Technical Specification	(Framework) the normative specification of the AIF. (Application) the normative specification of the set of AIWs belonging to an application domain along with the AIMs required to Implement the AIWs that includes: 1. The formats of the Input/Output data of the AIWs implementing the AIWs. 2. The Connections of the AIMs of the AIW. 3. The formats of the Input/Output data of the AIMs belonging to the AIW.
Testing Laboratory	A laboratory accredited to Assess the Grade of Performance of Implementations.
Time Base	The protocol specifying how Components can access timing information
Topology	The set of AIM Connections of an AIW.
Use Case	A particular instance of the Application domain target of an Application Standard.
User	A user of an Implementation.
User Agent	The Component interfacing the user with an AIF through the Controller
Version	A revision or extension of a Standard or of one of its elements.

Annex 3 - Notices and Disclaimers Concerning MPAI Standards (Informative)

The notices and legal disclaimers given below shall be borne in mind when downloading and using approved MPAI Standards.

In the following, "Standard" means the collection of four MPAI-approved and published documents: "Technical Specification", "Reference Software" and "Conformance Testing" and, where applicable, "Performance Testing".

Life cycle of MPAI Standards
MPAI Standards are developed in accordance with the MPAI Statutes. An MPAI Standard may only be developed when a Framework Licence has been adopted. MPAI Standards are developed by especially established MPAI Development Committees who operate on the basis of consensus, as specified in Annex 1 of the MPAI Statutes. While the MPAI General Assembly and the Board of Directors administer the process of the said Annex 1, MPAI does not independently evaluate, test, or verify the accuracy of any of the information or the suitability of any of the technology choices made in its Standards.

MPAI Standards may be modified at any time by corrigenda or new editions. A new edition, however, may not necessarily replace an existing MPAI standard. Visit the web page to determine the status of any given published MPAI Standard.

Comments on MPAI Standards are welcome from any interested parties, whether MPAI members or not. Comments shall mandatorily include the name and the version of the MPAI Standard and, if applicable, the specific page or line the comment applies to. Comments should be sent to the MPAI Secretariat. Comments will be reviewed by the appropriate committee for their technical relevance. However, MPAI does not provide interpretation, consulting information, or advice on MPAI Standards. Interested parties are invited to join MPAI so that they can attend the relevant Development Committees.

Coverage and Applicability of MPAI Standards
MPAI makes no warranties or representations of any kind concerning its Standards, and expressly disclaims all warranties, expressed or implied, concerning any of its Standards, including but not limited to the warranties of merchantability, fitness for a particular purpose, non-infringement etc. MPAI Standards are supplied "AS IS".

The existence of an MPAI Standard does not imply that there are no other ways to produce and distribute products and services in the scope of the Standard. Technical progress may render the technologies included in the MPAI Standard obsolete by the time the Standard is used, especially in a field as dynamic as AI. Therefore, those looking for standards in the Data Compression by Artificial Intelligence area should carefully assess the suitability of MPAI Standards for their needs.

IN NO EVENT SHALL MPAI BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO: THE NEED TO PROCURE SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND

Annex 4 - The Governance of the MPAI Ecosystem (Informative)

Level 1 Interoperability

With reference to *Figure 4* MPAI issues and maintains a standard – called MPAI-AIF – whose components are:

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

A Level 1 Implementation shall be an Implementation of the MPAI-AIF Technical Specification executing AIWs composed of AIMs able to call the MPAI-AIF APIs.

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

Level 2 Interoperability

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

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

Level 3 Interoperability

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

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

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

The MPAI ecosystem

The following *Figure 5* is a high-level description of the MPAI ecosystem operation applicable to fully conforming MPAI implementations:

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

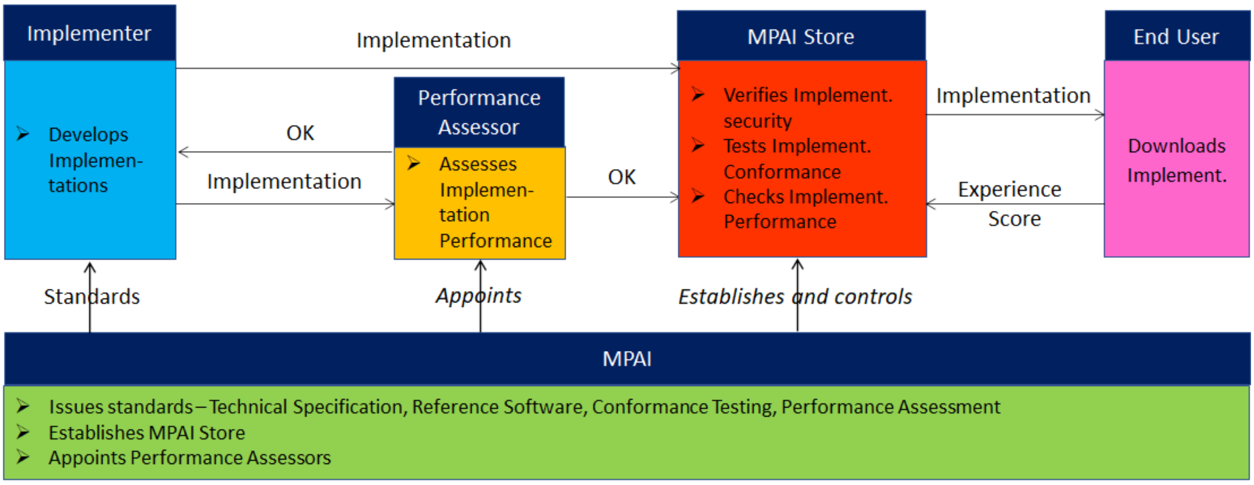

Figure 5 – The MPAI ecosystem operation