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

**Public document**

|  |  |
| --- | --- |
| **N1454** | 2023/10/25 |
| **Source** | Requirements (XRV) |
| **Title** | High-level MPAI-XRV – Live Theatrical Stage Performance requirements |
| **Target** | MPAI Community |

**Scope**

MPAI has issued a [Call for Technologies](https://mpai.community/standards/mpai-xrv/) seeking proposals for MPAI-XRV – Live Theatrical Stage Performance, an MPAI standard project seeking to define interfaces of AI Modules (AIM) facilitating live multisensory immersive stage performances. By running AI Workflows (AIW) composed of AIMs, it will be possible to obtain a more direct, precise yet spontaneous show implementation and control of multiple complex systems to achieve the show director’s vision.

**Setting**

An implementation of MPAI-XRV – Live Theatrical Stage Performance includes:

1. A physical stage.
2. Lighting, projections (e.g., dome or immersive display, holograms, AR goggles), and spatialised audio and Special Effects (FX) including touch, smell, pyro, motion seats etc.
3. Audience (standing or seated) in the real and virtual venue and external audiences via interactive streaming.
4. Interactive interfaces to allow audience participation (e.g., voting, branching, real-virtual action generation).
5. Performers on stage, platforms around domes or moving through the audience (immersive theatres).
6. Capture of biometric data from audience and/or performers from wearables, sensors embedded in the seat, remote sensing (e.g., audio, video, lidar), and from VR headsets.
7. Show operator(s) to allow manual augmentation and oversight of an AI that has been trained by show operator activity.
8. Virtual Environment (metaverse) that mirrors selected elements of the Real Environment. For example, performers on the stage are mirrored by digital twins in the metaverse, using motion capture (MoCap), green screen or volumetrically captured 3D images.
9. Real Environment can also mirror selected elements of the Metaverse similar to in-camera visual effects/virtual production techniques. Elements of the Metaverse such as avatars, landscape, sky, objects can be represented in the Real Environment through immersive displays, video mapped stages and set pieces, AR overlays, lighting, and FX.
10. The physical stage and set pieces blend seamlessly into the virtual 3D backdrop projected onto the immersive display such that the spectators perceive as a single immersive environment.
11. The Script or cue list describes the show events, guiding and synchronising the actions of all AI Modules (AIM) as the show evolves from cue to cue and scene to scene. In addition to performing the show, the AIMs might spontaneously innovate show variations, amplify the actions of performers or respond to commands from operators by modifying the Real or Virtual Environment within scripted guidelines.

**Requirements**

Respondents to [Call for Technologies](https://mpai.community/standards/mpai-xrv/) should consider the system requirements depicted in the Figure and in the following text. Note that terms in **bold underlined** refer to AIMs and those in **bold** refer to data types.



**Participants Description** capture the mood, engagement, and choices of participants (audience) and produce **Participant Descriptors** with the following features:

1. Use a known (i.e., standard) format to enable processing by subsequent AIMs.
2. Express Participants’:
	1. Visual behaviour (hand waving, standing, etc.)
	2. Audio reaction (clapping, laughing, booing, etc.)
	3. Choice (voting, motion controller, text, etc.)

**Performance Description** extracts information from the stage regarding position and orientation of performers and immersive objects and produce the **Scene Descriptors** with the following features:

1. Use existing or new formats independent of the capturing technology, e.g.,
	1. For audio: Multichannel Audio, Spatial Audio, Ambisonics, etc.
	2. For visual: audio and raw video, volumetric, MoCap, etc.
2. Allow for the following features:
	1. Accurate description of spatial and AV components.
	2. Individual accessibility and processability of objects.
	3. Unique association of objects with their digital representations.
	4. Association of the audio and visual components of audio-visual objects.

**Participants Status** interprets the Participants Descriptors and provides the **Participants Status** data type expressed either by:

1. A Format supporting the semantics of a set of statuses over time in terms of:
	1. Sentiment (e.g., measurement of spatial position-based audience reaction).
	2. Expression of choice (e.g., voting, physical movement of audience).
	3. Emergent behaviour (e.g., pattern emerging from coordinated movement).
2. A Language describing the Participant Status both at a time and as a trend.

**Performance Status** interprets the Scene Descriptors and indicates the current **Cue point** determined by a real or virtual phrase, gesture, dance motion, prop status etc. according to the Script.

**Operator Command Interpreter** interprets data and commands such as Audio/DJ/VJ, Show Control, Lighting/FX and generates **Interpreted Operator Controls**, a data type independent of the specific input format and includes:

1. Show control consoles (e.g., rigging, elevator stages, prop motions, and pyro).
2. Audio control consoles (e.g., controls audio mixing and effects).
3. DJ/VJ control consoles (e.g., real-time AV playback and effects).

All consoles may include sliders, buttons, knobs, gesture/haptic interfaces, joystick, touch pads.

Interpreted Controls may be Script-dependent.

**Script** includes written Show Script including character dialog, song lyrics, stage action, etc. plus a Master Cue Sheet with a corresponding technical description of all experiential elements including sound, lighting, follow spots, set movements, cue number, and show time of cue.

The cue sheet advances to the next cue point based on quantifiable or clearly defined actions such as spoken word, gesture, etc.

Various formats for show scripts and cue sheets exist.

**Script** is a data type with the following features:

1. It is machine readable and actionable.
2. It has an extensible set of clearly defined events/criteria for triggering the cue.
3. Uses a language for expressing Action Descriptors, which define the experiential elements associated with each cue.

**Action Generation**

Action Generation uses Participants Status, Scene Descriptors, Cue Points and Interpreted Operator Controls to produce **Action Descriptors**, a data type that may be either existing or known (e.g., text prompts) or new with the following features:

1. Ability to describe the Actions necessary to create the complete experience – in both the Real and Virtual Environments – in accordance with the Script.
2. Ability to express all aspects of the experience including the performers’ and objects’ position, orientation, gesture, costume, etc.
3. Independence of the specifications of a particular venue.

**Virtual Environment Experience Generation** processes Action Descriptors and produces commands that are actionable in the Virtual Environment. It produces the following data:

1. **Virtual Environment Descriptors** A variety of controls for 3D geometry, shading, lighting, materials, cameras, physics, etc. as required to affect the Virtual Environment, e.g., OpenUSD. The actual format used is dependent upon the current Virtual Environment Venue Specification.
2. **Audio-Visual (A/V)** Data and commands for all A/V experiential elements with the virtual environment, including audio and video.
3. Audio: M Audio channels (generated by or passed through the AIM), placement of audio channels, including attachments to objects or characters, Virtual Audio device commands, e.g., MIDI.
4. Video: N Video channels (generated by or passed through the AIM), Placement of Video channels, including mapping to objects and characters, Virtual VJ console commands/data, Virtual Video mixing console commands.
5. **Virtual Environment Venue Specification** An input to the Virtual Experience Generation AIM defining protocols, and data and command structures for the specific Virtual Environment Venue.

**Real Environment Experience Generation** processes Action Descriptors and produces commands that are actionable in the Real Environment:

1. **Real Environment Venue Specification** defines protocols, data, and command structures for the specific Real Environment Venue. This could include number, type, and placement of lighting fixtures, special effects, sound, and video display resources.
2. **FX (Effects)** Commands and data for all FX generators (e.g., fog, rain, pyro, mist, etc. machines, 4D seating, stage props, rigging etc.), typically using various standard protocols. FX systems may include video and audio provided by Real Experience Generation.
3. **Lighting** Commands and data for all lighting systems, devices, and elements, typically using the DMX protocol or similar. Lighting systems may include video provided by Real Experience Generation.
4. **Audio-Visual (A/V)** Data and commands for all A/V experiential elements, including audio, video, and capture cameras/microphones. They include:
5. Camera control: Camera #n Camera on/off, Keyframe based (Spatial Attitude, Optical parameters (aperture, focus, zoom), Frame rate, Camera)
6. Audio: M Audio channels (generated by or passed through the AIM), Audio source location designation (channel number or spatial orientation of STEM), MIDI device commands, Audio server commands, Mixing console commands.
7. Video: N Video channels (generated by or passed through the AIM), Video display location designation (display number or spatial orientation and mapping details), Video server commands, VJ console commands/data, Video mixing console commands.