ITUPublications

International Telecommunication Union
Standardization Sector

ITU-T Technical Specification

(10/2026)

FG-EAI D2.2

Physical-semantic and affordance-aware organization of 3D assets and scenes for embodied artificial intelligence

CAUTION!
PREPUBLISHED TECHNICAL SPECIFICATION

This prepublication is an unedited version of a recently approved Technical Specification. It will be replaced by the published version after editing. Therefore, there will be differences between this prepublication and the published version.

Technical Specification ITU-T FG-EAI D2.2

Physical-semantic and affordance-aware organization of 3D assets and scenes for embodied artificial intelligence

Summary

Technical Specification ITU-T FG-EAI D2.2 defines how 3D assets and scenes are organized, declared and checked so that they can be consumed by embodied artificial intelligence (EAI) systems without per-supplier reprocessing.

Existing 3D asset repositories are geometry-centric. When such assets are loaded into a physics simulator and acted upon by an embodied agent, four classes of failure recur: required information is absent without being declared absent; independently authored layers of the same asset disagree with one another; conventions such as unit, up-axis and reference frame are assumed rather than stated; and an asset that is adequate for rendering is inadequate for manipulation. Each failure is discovered only after download and manual inspection, and each consumer repairs the same defects independently.

This Technical Specification defines a declaration and checking layer that exposes these defects before payload retrieval. It specifies a layered organization model for assets, scenes, datasets and registries; a descriptor and header separated from payload; a three-state declaration discipline in which silent omission is non-conformant; machine-checkable consistency predicates between layers of an asset; affordance contracts that bind a declared interaction capability to a required field set; and conformance profiles with a machine-readable conformance report.

This Technical Specification does not repair assets, does not define trajectory or demonstration data formats, and does not define planning, control or learning methods.

Keywords

Affordance, conformance, consistency, embodied AI, physical grounding, simulation-ready, 3D asset.

Note

This is an informative ITU-T publication. Mandatory provisions, such as those found in ITU-T Recommendations, are outside the scope of this publication. This publication should only be referenced bibliographically in ITU-T Recommendations.

Change Log

This document contains Version 1.0 of the ITU-T FG-EAI Technical Specification on *Physical-semantic and affordance-aware organization of 3D assets and scenes for embodied AI* approved at the 3rd meeting of the ITU-T Focus Group of Embodied AI for Multimedia Technologies (FG-EAI) held on 13-16 October 2026.

Acknowledgements

This Technical Specification was developed in WG2 of FG-EAI, with inputs from Peng Cheng Laboratory and Harbin Institute of Technology as contribution to the ITU Focus Group on Embodied AI for Multimedia Technologies (FG-EAI). The development of this document was coordinated by Kei Kawamura (KDDI) as FG-EAI Working Group 2 (WG2) Chair, Qiongqian Yang (Hangzhou Embodied Intelligence Pilot Base Technology) as WG2 Vice-Chair and Editors. In addition, special thanks to Yuntao Wang (FG-EAI Chair), the FG-EAI management team, and the FG-EAI Secretariat.

Editor:	Ruotong Li Peng Cheng Laboratory P.R. China	Tel.: +86 13691691716 E-mail: lirt@pcl.ac.cn
Editor:	Xiang Tang Harbin Institute of Technology P.R. China	Tel.: +86 18655530970 E-mail: 24B951063@stu.hit.edu.cn
Editor:	Fan Xu Peng Cheng Laboratory P.R. China	Tel.: +86 15207146562 E-mail: xuf05@pcl.ac.cn
WG2 Chair:	Kei Kawamura KDDI Corporation Japan	Tel.: +81-80-5066-9075 E-mail: ki-kawamura@kddi.com
WG2 Vice-chair:	Qiongqian Yang Hangzhou Embodied Intelligence Pilot Base Technology Co., Ltd. China	Tel.: +86 15116306264 E-mail: yangqiongqian@jszn.cn

© ITU 2026

Some rights reserved. This publication is available under the Creative Commons Attribution-Non Commercial-Share Alike 3.0 IGO licence (CC BY-NC-SA 3.0 IGO; https://creativecommons.org/licenses/by-nc-sa/3.0/igo).

For any uses of this publication that are not included in this licence, please seek permission from ITU by contacting TSBmail@itu.int.

Table of Contents

		Page
1	Scope	1
	1.1 In scope	1
	1.2 Out of scope	1
	1.3 Applicability	2
2	References	2
3	Definitions	2
	3.1 Terms defined elsewhere	2
	3.2 Terms defined in this Technical Specification	2
4	Abbreviations and acronyms	2
5	Conventions	2
6	Overview and principles	3
	6.1 Classes of failure addressed	3
	6.2 Design principles	3
7	Layered organization model	4
8	Descriptor and header	5
9	Physical grounding	5
10	Affordance and interaction representation	6
11	Consistency constraints	7
12	Conformance profiles and reporting	7
13	Derived state	8
Appendix I – Illustrative consistency predicates		9
Appendix II – Illustrative interaction contracts		10

Technical Report ITU-T FG-EAI D2.2

Physical-semantic and affordance-aware organization of 3D assets and scenes for embodied artificial intelligence

1 Scope

This Technical Specification describes the organization, declaration and checking of 3D assets and scenes intended for use by embodied artificial intelligence (EAI) systems within multimedia environments for embodied AI.

1.1 In scope

This Technical Specification specifies:

– a layered organization model covering assets (L0), scenes (L1), datasets (L2) and registries (L3), and the referencing rules between layers;

– the separation of descriptor from payload, and the content of the self-describing header;

– a three-state declaration discipline distinguishing a declared value, a declared absence and an undeclared field;

– the physical-semantic metadata required to ensure spatio-temporal consistency, including absolute scale, mass properties, surface properties and collision representation, etc.;

– the explicit declaration of conventions, including unit of length, up-axis, forward-axis and reference frame;

– the representation of an interaction capability as a contract, associating a declared capability with the information a consumer needs in order to attempt the interaction;

– consistency conditions between independently authored layers of the same asset or scene;

– conformance profiles and the reporting of an evaluation.

This Technical Specification describes the organization model and the declaration discipline. The illustrative material in Appendices I and II is provided to show that the declarations described here can be made concrete and machine-checkable. It is not intended to be exhaustive, and the field-level detail of a future specification is not fixed by this document.

1.2 Out of scope

This Technical Specification does not specify:

– the internal structure of trajectory, demonstration or episode data;

– planning, control, policy or learning methods, or any method of training an embodied agent;

– the numerical methods by which a physics engine solves contact, articulation or dynamics;

– the repair, regeneration or re-authoring of non-conformant assets;

– any guarantee that a runtime implementation will keep visual and physical behaviour synchronized; this Technical Specification describes how the relationship is declared and checked, not how it is enforced at runtime;

– the semantic plausibility of a declared value; whether a stated friction coefficient is realistic for the material it claims to represent is outside the scope of the checks described here;

– a new geometry, material or scene interchange format; this Technical Specification is format-neutral and is intended to be carried by existing host formats;

– the declaration of levels of detail; this is for further study.

1.3 Applicability

This Technical Specification applies to assets and scenes that are intended to be consumed within a simulation environment for embodied artificial intelligence.

2 References

[ITU-T F.748.66] Recommendation ITU-T F.748.66 (2025), *Requirements and framework for embodied artificial intelligence systems*.

[ISO/IEC 12113] ISO/IEC 12113, *Information technology – Runtime 3D asset delivery format – Khronos glTF 2.0*.

[ISO/IEC 23090-14] ISO/IEC 23090-14, *Information technology – Coded representation of immersive media – Part 14: Scene description*.

3 Definitions

3.1 Terms defined elsewhere

This Technical Specification uses the following term defined elsewhere:

3.1.1 embodied artificial intelligence (EAI) [ITU-T F.748.66]: Artificial intelligence integrated into physical systems that interact autonomously with and adapt to the physical world.

3.2 Terms defined in this Technical Specification

None.

4 Abbreviations and acronyms

This Technical Specification uses the following abbreviations and acronyms:

AI Artificial Intelligence

EAI Embodied Artificial Intelligence

5 Conventions

The following conventions are used in this Technical Specification:

The keywords "it is critical" indicate a possible requirement which would be necessary to be fulfilled (e.g., by an implementation) and enabled to provide the benefits of the use case.

The keywords "it is expected" indicate a possible requirement which would be important but not absolutely necessary to be fulfilled (e.g., by an implementation). Thus, this possible requirement would not need to be enabled to provide complete benefits of the use case.

The keywords "it is of added value" indicate a possible requirement which would be optional to be fulfilled (e.g., by an implementation), without implying any sense of importance regarding its fulfilment. Thus, this possible requirement would not need to be enabled to provide complete benefits of the use case.

A declaration governed by this Technical Specification is in exactly one of the three states of Table 5-1. The distinction between a declared absence and an undeclared field is fundamental to this Technical Specification and is described in clause 8.

Table 5-1 – Declaration states

State	Meaning	Conformance
Declared value	A value is stated	Conformant
Declared absence	Absence is explicitly stated, by a reserved value	Conformant
Undeclared	Neither of the above	Non-conformant

Unless otherwise stated, all quantities are expressed in SI units. Length is expressed in metres.

6 Overview and principles

6.1 Classes of failure addressed

An asset that has been published, retrieved and opened may still be unusable. This Technical Specification addresses the four classes of failure listed in Table 6-1. Each class corresponds to a mechanism described in a later clause.

Table 6-1 – Classes of failure and the corresponding mechanisms

Class	What is not declared	Mechanism
1 Undeclared incompleteness	Information required to place an asset in a simulation environment is absent, and its absence is not stated, so that a consumer cannot distinguish "not provided" from "not applicable" without retrieving the payload.	Three-state declaration (clause 8)
2 Inconsistency between layers	Two or more independently authored layers of one asset disagree, although each layer is well formed on its own. The disagreement is invisible to any check that examines one layer at a time.	Consistency conditions (clause 11)
3 Undeclared conventions	Unit of length, up-axis and the frame in which a pose is expressed are assumed rather than declared, so that data assembled from several sources is misaligned in a way that no single artifact reveals.	Declaration of conventions (clause 9)
4 Undeclared unfitness	An asset adequate for rendering is not adequate for interaction, and this unfitness is neither declared nor available as a filter.	Conformance profiles (clause 12)

6.2 Design principles

6.2.1 Expose, do not repair

This Technical Specification makes defects visible in the declaration. It does not repair assets, and does not require an implementation to modify payload content in order to bring an artifact into conformance.

6.2.2 State, do not assume

Where a convention, a capability or an absence can be assumed, it is instead stated.

R-6-1: It is critical that, where this Technical Specification governs a field, the artifact declares either a value or an explicit absence for that field. An undeclared field is not conformant.

6.2.3 Declare before retrieval

Filtering and metadata-profile evaluation described here are performed on the descriptor alone; payload-dependent predicates (clause 11) may require payload. A consumer selecting assets for an embodied task cannot retrieve candidate payloads at the scale at which it must filter.

R-6-2: It is critical that a descriptor references payload by identifier and checksum, and does not embed payload content. It is critical that filtering, conformance evaluation and licence inspection are possible from the descriptor alone.

6.2.4 Fitness is relative to the consumer

This Technical Specification does not classify an artifact as good or bad. It exposes what an artifact declares and what it has been checked for, so that each consumer may determine fitness for its own use. A declaration is not a claim that a declared interaction is exercisable by any particular embodiment. Whether a given embodiment can exercise it is for the consumer to determine.

6.2.5 Format neutrality

This Technical Specification describes what is declared and what relationships are to hold. It does not define a new interchange syntax, and is intended to be carried by existing host formats, such as the runtime 3D asset delivery format specified in [ISO/IEC 12113] and the scene description for immersive media specified in [ISO/IEC 23090-14].

6.2.6 Proportionate obligation on the producer

An obligation is placed on the producer only where the producer is the party that holds the information. The provisions concern declaration and not data holdings: a producer that does not hold a quantity conforms by declaring its absence, and is not required to measure, estimate or fabricate it.

7 Layered organization model

An artifact governed by this Technical Specification is organized as a graph of nodes and edges, carried together with a header. Four layers are defined. Each layer uses the same metamodel. A node at one layer can reference an artifact at the layer below: an entity in a scene references an asset, and a member of a dataset references an asset or a scene.

Table 7-1 – Layers of the model

Layer	Unit	What the layer carries
L0 asset	A single object, rigid or articulated	A tree of parts connected by joints; the physical-semantic metadata of clause 9 and the interaction capabilities of clause 10 are declared here
L1 scene	An arrangement of asset instances and regions in a shared coordinate system	The shared coordinate semantics, and whether a declared interaction is obstructed by the arrangement rather than by the object
L2 dataset	A reproducible collection of assets and scenes	Membership, splits, and index keys derived from what the layers below declare
L3 registry	A set of references to datasets held by one or more providers	Discovery, versioning and supersession across providers

The metamodel is a property graph. Attributes are carried on the edges as well as on the nodes. This matters because at these two layers much of the information sits on the relations. A joint carries an axis and its limits. A spatial relation carries a relative pose and a contact region. An interaction relation carries a reference to a capability and to the region at which it is exercised. Edges are classified as spatial, kinematic, semantic or interaction. The interaction family is what distinguishes

this metamodel from a scene graph used for rendering. It states not only what an agent can do, but what that action requires of the environment.

R-7-1: It is critical that a reference from one layer to the layer below is by identifier and checksum, and that the referenced content is not copied into the referencing artifact. It is critical that a referenced artifact is independently resolvable and independently evaluable.

8 Descriptor and header

A descriptor is the declarative content of an artifact. It carries no payload content, and has the three parts of Table 8-1.

Table 8-1 – Parts of a descriptor

Part	What it carries
Header	The identity of the artifact, its version, the conventions of clause 9, its licence, its provenance, and the conformance profile it is stated against. The header can be evaluated without resolving any other artifact.
Graph	The nodes and edges of clause 7.
Manifest	One entry for each payload item: its identity, the location at which it can be retrieved, and a checksum over its content. This is the only place from which payload is referenced.

This clause describes the mechanism addressing Class 1 of clause 6.1. One asset declares no mass because none was measured. Another declares no mass because the quantity does not apply to it. These are different artifacts, and in current practice they cannot be told apart without retrieving the payload.

R-8-1: It is critical that a declared absence is expressed by a reserved value. It is critical that an undeclared field causes evaluation against the declared conformance profile to fail, and that the failure is recorded, identifying the field.

R-8-2: It is critical that a consumer treats a declared absence as authoritative, and does not infer that a value exists in the payload. It is critical that a value taken from a tool default is not recorded as a declared value, unless the method by which it was obtained is also declared.

9 Physical grounding

This clause describes explicit declarations of physical-semantic metadata and foundational conventions, required to ensure spatio-temporal consistency and multi-sensory interactions within multimedia environments for embodied AI.

In such environments, EAI systems process multimodal data, including visual, spatial and haptic streams, in a continuous perception-action loop. The metadata specified in this clause registers these independently authored streams into a single spatial reference, so that the geometry an agent acts upon corresponds to the geometry it perceives. This addresses virtual-world datasets and native multimodal perception.

It describes what is declared and in what units. It does not specify dynamic mechanical equations, the algorithms by which any implementation solves contact or articulation, or the actuation and control parameters of a robot. How a consumer uses a declared value is outside the scope of this Technical Specification.

Table 9-1 – What is declared, and why

Declaration	Why it is needed
Unit of length, and geometry at absolute scale	An asset not expressed at absolute scale in a declared unit cannot be placed alongside assets from other sources.
Mass	Mass is the case in which information is most often absent without its absence being stated. It is declared as metadata, or declared absent.
Surface properties	Declared separately from any visual material, because a visual material describes appearance only. These are the properties on which a haptic representation of the same surface depends.
Collision representation	The geometry used for spatial reasoning, which may differ from the geometry authored for appearance.
Up-axis, forward-axis, handedness, angle unit, pose composition order, and the frame of a declared pose	A convention customary in a host format is not thereby declared. The customary value differs between formats and between tools, and a consumer assembling a scene from several sources cannot recover it from any single artifact. This addresses Class 3 of clause 6.1.

R-9-1: It is critical that an asset or scene declares each item of Table 9-1, or declares its absence in accordance with Table 5-1.

R-9-2: It is critical that the geometry of an asset is expressed at absolute scale in the declared unit. It is critical that a convention is declared in the artifact, and not established by reference to the customary value of a host format.

10 Affordance and interaction representation

An interaction capability declared by an asset is not a label. A declaration that an asset affords an interaction says that the asset carries the information a consumer needs in order to attempt it. That information is the contract of the capability, and is what makes the declaration usable rather than merely descriptive. A declaration that a surface affords support is of no use on its own; the consumer also needs the extent of that surface, its outward normal and its load limit. This Technical Specification specifies only the data structures of these affordances, and does not define the physical capability or execution constraints of any particular embodied agent.

Table 10-1 – Elements of a contract

Element	What it states
Capability identifier	The interaction the asset affords, such as support or the opening of a hinged part
Required field set	The fields a declaration of this capability is required to carry
Interaction region	The geometric region at which the capability is exercised, such as a support surface, a containment cavity or a grasp region
Applicable conditions	The consistency conditions of clause 11 that apply to a declaration of this capability
Target embodiment	The agent class to which the declaration applies, enabling parameterized interaction regions for different embodiments

A contract that is satisfied means that the declaration is complete. It does not mean that the interaction will succeed. Whether it succeeds depends on the acting system, on the scene in which the asset is placed, and on the implementation that executes the interaction. None of these is within the scope of this Technical Specification. Appendix II illustrates two contracts.

R-10-1: It is critical that a declaration of an interaction capability satisfies the contract of that capability, and that every field the contract requires is in one of the three states of Table 5-1.

11 Consistency constraints

This clause describes the mechanism addressing Class 2 of clause 6.1. Two or more independently authored parts of the same artifact are each well formed, and they disagree with one another. No check that examines one part at a time detects the disagreement.

Such a disagreement matters only where a system both perceives an artifact and acts upon it. A renderer is unaffected by a collision representation that does not match the visual geometry. A system that reaches for the surface it can see is not. The conditions are expressed as consistency predicates. Appendix I gives illustrative predicates.

R-11-1: It is critical that a consistency predicate is evaluable by machine over the declared content of an artifact and, where identified as payload-dependent, over its payload.

R-11-2: It is critical that the evaluation of a predicate yields one of the results pass, fail, not evaluated or not applicable. It is critical that a predicate that has not been evaluated is recorded as not evaluated, and not as pass.

12 Conformance profiles and reporting

This clause describes the mechanism addressing Class 4 of clause 6.1. An artifact declares one profile, is evaluated against that profile and no other, and satisfies every requirement of the profiles below it. The four profiles separate three things that are often treated as one: an artifact that can be displayed cannot always be simulated, and an artifact that can be simulated cannot always be interacted with.

Table 12-1 – Conformance profiles

Profile	Identifier	What the profile evidences
Metadata	PSA3D-M	The artifact is identified, licensed and attributed, its conventions are declared, and its references resolve
Renderable	PSA3D-R	The artifact additionally carries visual geometry at absolute scale
Simulatable	PSA3D-S	The artifact additionally carries collision, mass and surface declarations, and its layers have been checked against one another
Interactable	PSA3D-I	The artifact additionally declares interaction capabilities whose contracts are complete

The result of an evaluation is recorded in a conformance report, whose content is given in Table 12-2. The report is how work already performed on an artifact is reused rather than repeated, and it is readable without retrieving payload.

Table 12-2 – Content of a conformance report

Element	Description
Subject	The artifact evaluated, identified by identifier and checksum
Profiles	The profile declared by the artifact, and the profile found to be satisfied
Evaluator	The party that performed the evaluation, and the tool used
Undeclared fields	The fields found to be undeclared
Predicate results	The result of each predicate evaluated
Outcome	Conformant, not conformant, or partially evaluated

R-12-1: It is critical that, where a required predicate fails or a required field is undeclared, the outcome is recorded as not conformant. It is critical that, where nothing fails but one or more required predicates have not been evaluated, the outcome is recorded as partially evaluated.

R-12-2: It is critical that an evaluation result is bound to the checksum of the artifact evaluated, and that it does not apply to any other version of that artifact. This Technical Specification does not define an accreditation scheme and does not confer authority on any evaluating party.

13 Derived state

An EAI system changes the environment it acts in. Derived state records a changed environment without duplicating it, representing a static snapshot of changed configurations. It is an ordered set of deltas anchored to an immutable base artifact, and each delta declares what it targets and what value it sets. A delta may record a joint value, a pose, a change to a relation, or a replacement value for a declared field.

R-13-1: It is critical that a derived state references its base artifact by identifier and checksum, and does not modify that artifact. It is critical that, where the two checksums do not match, the derived state is treated as not applicable to that artifact.

External trajectory, demonstration or episode data may reference a base artifact and its derived states through these identifiers. This Technical Specification does not define the internal structure of such data.

Appendix I

Illustrative consistency predicates

This appendix shows that the conditions described in clause 11 can be stated in a form that a validator can evaluate. It is not a complete register, and the identifiers, tolerances and profile assignments given here are illustrative.

Identifier	Condition asserted	Payload required
CP-REF-01	Every reference resolves and the recorded checksum matches.	No
CP-SCALE-01	The declared bounding box corresponds, within a declared tolerance, to the extent of the geometry under the declared unit of length.	Yes
CP-VISCOL-01	The extents of the visual geometry and of the collision representation correspond within a declared tolerance.	Yes
CP-ARTCOL-01	The asset does not self-intersect at any configuration within its declared joint limits.	Yes
CP-VISPHY-01	Where a non-zero surface velocity is declared for a region, the visual representation of that motion is declared or declared absent. This applies wherever declared physical motion is independent of the pose of the part carrying it, including conveying surfaces, rotating surfaces, treadmills and escalators.	No

CP-VISPHY-01 covers the conveyor case. The belt surface is declared to transport objects, but the texture applied to it is static. An object is therefore carried across a belt whose pattern does not move. Each layer of such an asset is well formed on its own, and no check that examines one layer at a time detects the disagreement.

Appendix II

Illustrative interaction contracts

This appendix illustrates the form of a contract as described in clause 10. Two capabilities are shown. The field sets given here are illustrative and are not intended to be exhaustive.

support

Interaction region	one region of kind surface
Required fields	outward normal; planar extent; load limit
Applicable conditions	reference integrity

open_revolute

Applies to	a part
Interaction region	one actuation region
Required fields	joint reference; joint limits; opening direction
Applicable conditions	reference integrity; articulation

The difference between the two shows why a contract is bound to a capability rather than to an asset. Support is a property of a surface. Opening a hinged part refers to an articulation, and the volume the part sweeps is determined by the joint limits already declared. The same asset may therefore be usable in one scene and obstructed in another.