Open Geospatial Consortium

Submission Date: <2021-mm-dd>

Approval Date: <yyyy-mm-dd>

Publication Date: <yyyy-mm-dd>

External identifier of this OGC® document: http://www.opengis.net/doc/{doc-type}/{standard}/{m.n}

Internal reference number of this OGC® document: 21-056r6

Version: 1.0

Category: OGC® Implementation

Editor: Carl Stephen Smyth

Draft OGC GeoPose 1.0 Data Exchange Standard

Warning

This document is not an OGC Standard. This document is distributed for review and comment. This document is subject to change without notice and may not be referred to as an OGC Standard.

Recipients of this document are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

Document type:	OGC® Standard
Document subtype:	Implementation Specification
Document stage:	Draft
Document language:	English

License Agreement

Table of Contents

i. Abstract

GeoPose 1.0 is an OGC Implementation Standard for exchanging the location and orientation of real or virtual geometric objects (Poses) within reference frames anchored to the earth's surface (Geo) or within other astronomical coordinate systems. The standard specifies two Basic forms with no configuration options for common use cases, an Advanced form with more flexibility for more complex applications, and five composite GeoPose structures that support time series plus chain and graph structures.

These eight Standardization Targets are independent. There are no dependencies between Targets and each may be implemented as needed to support a specific use case.

The Standardization Targets share an implementation-neutral Logical Model which establishes the structure and relationships between GeoPose components and also between GeoPose data objects themselves in composite structures. Not all of the classes and properties of the Logical Model are expressed in individual Standardization Targets nor in the specific concrete data objects defined by this standard. Those elements that are expressed are denoted as implementation-neutral Structural Data Units (SDUs). SDUs are aliases for elements of the Logical Model, isolated to facilitate specification of their use in encoded GeoPose data objects for a specific Standardization Target.

For each Standardization Target, each implementation technology and corresponding encoding format defines the encoding or serialization specified in a manner appropriate to that technology.

GeoPose Version 1.0 specifies a single encoding: IETF RFC 8259 / ECMA JavaScript Object Notation (JSON). Each Standardization Target has a JSON-Schema:2019-9 encoding specification. The key standardization requirements specify that concrete JSON-encoded GeoPose data objects must conform to the corresponding JSON-Schema definition. The individual elements identified in the encoding specification are composed of SDUs, tying the specifications back to the Logical Model.

The GeoPose 1.0 Standard makes no assumptions about the interpretation of external specifications, for example, of reference frames. Nor does it assume or constrain services or interfaces providing conversion between GeoPoses of difference types or relying on different external reference frame definitions.

ii. Keywords

The following are keywords to be used by search engines and document catalogues.

ogc, ogcdoc, OGC document, pose, geopose, ar, reference frame

iii. Preface

NOTE	This is a draft implementation specification for GeoPose 1.0. Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights. Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

iv. Submitting organizations

The following organizations submitted this Document to the Open Geospatial Consortium (OGC):

NOTE	The organizations, submitters, participants, and acknowledgements are open to changes and additions - just email a chair or editor.

Organization name(s)

This Document was submitted to the Open Geospatial Consortium (OGC) by the members of the GeoPose Standards Working Group of the OGC. Amongst others, this comprises the following organizations:

- Norkart AS
- Open Site Plan
- Open AR Cloud Association
- Ordnance Survey
- School of Management and Engineering Vaud, HES-SO University of Applied Sciences and Arts Western Switzerland

v. Submitters

All questions regarding this submission should be directed to the editors or the submitters:

Table 1. Submission Contact Points

Name	Institution
Nicolas Blanc	School of Management and Engineering Vaud, HES-SO University of Applied Sciences and Arts Western Switzerland
Kyoung-Sook Kim	National Institute of Advanced Industrial Science and Technology
Jeremy Morley	Ordnance Survey
Christine Perey	Open AR Cloud Association
Mahmoud SAKR	Université Libre de BRUXELLES
Scott Simmons	Open Geospatial Consortium
Carl Stephen Smyth	OpenSitePlan, USA

Name	Institution
Jan-Erik Vinje	Open AR Cloud Association

vi. Participants in development

The following individuals contributed to the GeoPose 1.0 development:

Table 2. Participants in Development

Name	Institution
Nazih Fino	Global Nomad
Josh Lieberman	Open Geospatial Consortium
Mikel Salazar	Augmented Interaction
Maxime Schoemans	Université Libre de Bruxelles
Marco Tillmann	HERE Technologies

vii. Acknowledgements

The GeoPose SWG wishes to thank the 3D Information Management (3DIM) Working Group of the OGC as well as Augmented City, Immersal, Trevor F. Smith, Transmutable, Khronos Group, SEDRIS, Roger Lott and Chris Little for their support and insight.

Chapter 1. Scope

The OGC GeoPose 1.0 Standard defines requirements (rules) for the interoperable exchange of the location and orientation of real or virtual geometric objects (poses) within reference frames anchored to the earth's surface (Geo) or within other astronomical coordinate systems. The Standard specifies

- A basic form with no configuration options for common use cases,
- An advanced form with more flexibility for more complex applications, and
- Composite GeoPose structures to support time series chain, and graph structures.

The GeoPose Standard is based on an implementation-neutral Logical Model(LM). This LM is a formalization of a Conceptual Model(CM). The CM consists of a linked set of terms and definitons, defining a domain odf discourse for the various geometrric, geographic, and physical concepts related to GeoPoses. The LM formalizes the relationships between the implementable parts and aspects of the CM. The LM further establishes the structure and relationships between GeoPose components and also between GeoPoses data objects themselves in composite structures.

Note that the concrete GeoPose data objects defined by this standard correspond to only certain classes and properties of the LM. These classes and properties are identified as implementation-neutral Structural Data Units (SDUs). SDUs are aliases for the implementable elements of the LM. SDUs are grouped to define the implementation-neutral form of the GeoPose Standardization Targets: the specific implementation that the Standard addresses. For each Standardization Target, each implementation technology will have the definition of the encoding or serialization specified in a manner appropriate to that technology.

GeoPose Version 1.0 defines only one of many possible encoding methods for implementation of any or all of the eight STandardization Targets: JavaScript Object Notation (JSON). Each Standardization Target has a JSON-Schema:2019-9 definition. Most of the GeoPose standardization requirements are that concrete JSON GeoPose data objects shall conform to the corresponding JSON-Schema definition. The individual elements identified in the encoding specifications are SDUs that refer to one or more classes or attributes of the LM.

The GeoPose 1.0 Standard excludes assumptions about the interpretation of external specifications such as reference frames. Further, the Standard does not assume or constrain services or interfaces providing conversion between GeoPoses of difference types or relying on different external reference frame definitions.

Chapter 2. Conformance

Conformance with this standard shall be checked using all the relevant tests specified in Annex A (Normative) of this document. The framework, concepts, and methodology for testing, and the criteria to be achieved to claim conformance are specified in the OGC Compliance Testing Policies (https://portal.ogc.org/files/?artifact_id=55234) and Procedures and the OGC Compliance Testing web site (https://www.ogc.org/compliance). GeoPose 1.0 JSON encodings are specified via JSON-Schema:2019-9 and most of the requirements are that conforming encoded data objects shall validate against the corresponding schema.

In order to conform to this OGC® Standard, a software implementation shall choose to implement any one or more of the eight Standardization Targets specified in Annex A (normative).

All requirements-classes and conformance-classes described in this document are owned by the standard(s) identified.

2.1. Modularity

This standard describes eight Standardization Targets. These targets are independent and a conforming implementation may implement one or more of the targets.

2.2. Conformance Classes

This standard identifies eight conformance classes. One conformance class is defined for each corresponding set of Structural Data Units (SDUs) where each SDU is linked to the Logical Model as an alias for a class or attribute. Additionally, each of the eight standardization targets is represented by a conformance class as defined by a corresponding requirements class. The tests in Annex A are organized by requirements class. An implementation of a conformance class must pass all tests specified in Annex A for the corresponding requirements class.

No conformance class has a dependency on another conformance class.

The Logical Model is the root normative part of this standard.

2.3. Standardization Targets

There are eight independent standardization targets. Each addresses the specific requirements of one or more individual use cases. The Basic and Advanced Targets share in the use of an EPSG 4979/3D WGS-84 **Outer Frame** but differ in the level of options and flexibility in specication of the **Inner Frame**. The Composite Targets offer approaches to packaging sequenced or linked Frame Transforms. The eight targets are denoted by bold terms in the following categories:

1. Basic - Satisfy most use cases - EPSG 4979 Outer Frame
   a. Local Tangent Plane East, North, Up(LTP-ENU) **Inner Frame** oriented by Yaw, Pitch, and Roll (YPR) rotations about z, y, x axes: **Basic-YPR** Target
   b. LTP-ENU **Inner Frame** oriented by unit quaternion: **Basic-Quaternion** Target

2. Configurable **Inner Frame** oriented by unit quaternion - Flexible enough for complex use cases: **Advanced** Target
3. Composite - Efficient structures for linked and sequential GeoPoses
   a. Linked linear sequence of poses: **Chain** Target
   b. General linked poses: **Graph** Target
   c. Sequence
      i. Series
         A. Timeseries with constant time spacing: **Regular** Timeseries Target
         B. Timeseries with per-GeoPose time: **Irregular** Timeseries Target
      ii. Open-ended sequence of time-stamped GeoPoses: **Stream** Target

Chapter 3. References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this document. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. For undated references, the latest edition of the normative document referred to applies.

1. Bray, T., Ed., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, December 2017, https://www.rfc-editor.org/info/std90
2. H. Andrews, Ed. 2019. "JSON-Schema Specification", Draft 2019-09" https://json-schema.org/specification-links.html#draft-2019-09-formerly-known-as-draft-8
3. A. Wright, Ed. 2019. "JSON-Schema Validation Recommendation", Draft 2019-09. https://tools.ietf.org/html/draft-handrews-json-schema-validation-02
4. ISO 19101 Conceptual Modelling
5. ISO / TC 211: ISO 19115-1:2014 Geographic information — Metadata — Part 1: Fundamentals (2014)
6. ISO / TC 211: ISO 19157:2013 Geographic information — Data quality (2013)
7. ISO / TC 211: ISO 19139:2007 Geographic information — Metadata — XML schema implementation (2007)
8. ISO / TC 154: ISO 8601-1:2019 Date and time — Representations for information interchange — Part 1: Basic rules (2019)
9. ISO / TC 154: ISO 8601-2:2019 Date and time — Representations for information interchange — Part 1: Extensions (2019)
10. OGC: OGC 15-097 OGC Geospatial User Feedback Standard. Conceptual Model (2016)
11. OGC: OGC 12-019, OGC City Geography Markup Language (CityGML) Encoding Standard (2012)
12. OGC: OGC 14-005r3, OGC IndoorGML (2014)
13. OWL Time: https://www.w3.org/TR/2020/CR-owl-time-20200326/
14. JSON Schema 2019-09 http://json-schema.org/specification.html
15. ISO 19162:2019 Geographic information — Well-known text representation of coordinate reference systems https://www.iso.org/standard/76496.html
16. ISO 19111:2019 Geographic information — Referencing by coordinates https://www.iso.org/standard/74039.html
17. EPSG https://epsg.org
18. NASA SPICE https://naif.jpl.nasa.gov/naif/ Navigation and ancillary information facility
19. YPR https://www.gwg.nga.mil/misb//docs/standards/ST0601.8.pdf
20. Ref frame https://en.wikipedia.org/wiki/Frame_of_reference
21. Pose https://en.wikipedia.org/wiki/Pose_(computer_vision)

Chapter 4. Terms and Definitions

This document uses the terms defined in Sub-clause 5.3 of [OGC 06-121r8], which is based on the ISO/IEC Directives, Part 2, Rules for the structure and drafting of International Standards. In particular, the word “shall” (not “must”) is the verb form used to indicate a requirement to be strictly followed to conform to this standard.

For the purposes of this document, the following additional terms and definitions apply.

The GeoPose Conceptual Model(CM) consists of linked definitions of terms denoting concepts expressed in the GeoPose LM and structural data unit(SDU) specifications for the standardization targets. The CM describes a (non-normative) domain of discourse for terms used in defining a precise and normative Logical Model(LM) expressed as a Unified Modelling Language (UML) class diagram.

The scope of the standardization targets is a subset of the scope of the LM. The scope of the LM is a subset of the scope of the Conceptual Model. The standardization targets are mutually independent implementations of subsets of the LM. The standardization targets are expressed in Extended Backus-Naur Form where all terminal symbols reference attributes of classes in the LM.

“Conceptual model: a description of common concepts and their relationships, particularly in order to facilitate exchange of information between parties within a specific domain [CEN ENV 1613:1994]. A conceptual model is explicitly chosen to be may be informed by, but independent of design or implementation concerns.”

Conventions

Defined terms are in bold caps. Underlined and bolded terms are linked to the defined term.

4.1. Spatial Concepts

A **position** (OGC **direct position**) is a set of coordinates of a point in a 3D Euclidean space and associated **reference frame**.

Orientation is the rotational relationship between two **reference frames**.

A **pose** is a representation of a **frame transform** mapping the space of an **outer (reference) frame** to the space of an **inner (reference) frame**. A **pose** may be associated with additional non-geometrical properties such as time of observation or validity. **Poses** in computer graphics often have an **Outer Frame** defined by a parent node in a scenegraph and an **Inner Frame** define by a **position** and an orientation.

A **GeoPose** is a **pose** whose associated **outer frame** or a **pose chain** whose associated **outermost** frame is a topocentric reference frame defined by an extrinsic specification related to the ephemeris object planet Earth.

A **(reference) frame** is a system of location and measurement often defined by a frame specification usually including a coordinate system to be used within a corresponding space.

A **frame transform** consists of a pair of reference frames and a bi-continuous coordinate transformation relating points in the corresponding spaces. The two **frames** are called outer frame (domain) and inner frame (range). Only an outer frame may have an extrinsic specification. [A **frame transform** functions as a directed edge in a frame graph representation of the transformational relationship between frames.]

An **outer frame** is the first of two reference frames associated with a frame transform.

NOTE	In the NASA SPICE system, the outer frame is referred to as the **from** frame. In the ROS SDF documentation, the outer frame is referred to as the **Parent** frame. In ISO 19162, the outer frame is referred to as the **base** frame.

An **inner frame** is the second of two reference frames associated with a frame transform. An **inner frame** may not be a topocentric frame.

NOTE	In the NASA SPICE system, the inner frame is referred to as the **to** frame. In the ROS SDF documentation, the inner frame is referred to as the **child** frame. In ISO 19162, the inner frame is referred to as the **derived** frame.

An **outermost frame** is the outer frame of the first frame transform in a pose chain.

An **Innermost frame** is the inner frame of the last frame transform in a pose chain.

An **ephemeris object** is a physical object or manifestation of a physical object that can be characterized by an externally-defined (possibly time-dependent) location and orientation in a 3-dimensional space.

A **topocentric (reference) frame** is a frame that has an extrinsic specification associated with a location on or near the surface of a natural body, such as planet Earth. [This is the definition used in the NASA SPICE system.] In connection with a GeoPose, one way that a **topocentric frame** may be realized is by a local tangent plane east-north-up frame (LTP-ENU) attached to the surface of a body, to a gravitational equipotential surface (geoid in the case of planet Earth), or to a mathematical surface such as an ellipsoid approximating a geoid.

A **frame specification** is data that completely and uniquely defines a reference frame. In the context of Poses, there are extrinsic specifications defined by an external data source, and derived specifications defined by a transformation from another reference frame.

An **extrinsic frame specification** relates a reference frame to an ephemeris object or other external reference, which may be based on joint properties of a group of objects, such as the center of mass of the Earth-Moon system.

A **derived frame specification** relates a reference frame to another frame by a frame transform or its inverse.

A **frame graph** is a directed acyclic graph representation of the transformational relationships between **reference frames**. **frames** are the nodes or vertices of the graph. **frame transforms** are the edges of the graph, directed from the **outer frame** to the **inner frame**. [Note that there may be zero, one, or many paths between two distinct vertices, i.e. **frames**. This is by design, even though the corresponding linked **frame transforms**, when composed into single transformations between the same starting **outer frames** and the same **inner frame**. This corresponds to real-world situations with, for example, redundant line-of-sight links in point-to-point radio networks used in communication systems.]

A **pose chain** is a directed path in a **frame graph** connecting an **outermost frame** to an **innermost frame**. The sequence of **frame transforms** in a **pose chain** may be combined in a single composite transformation. [There may exist multiple **pose chains** linking the same **outermost frame** and **innermost frame** and the corresponding composite transformations may not agree. This is intentional, representing real-world configurations and capabilities of sensors and communication links.]

4.2. Sequence and Stream Concepts

A **(GeoPose) sequence** is a set of **(member) poses** ordered by **valid time** and pertaining to the same underlying physical object or construct. Each successive **(member) pose** must have a **valid time** after its predecessor.

Inter-pose duration is the time **duration** between consecutive **poses** in a **sequence**. The member poses in a sequence

A **closed (pose) sequence** is a **GeoPose sequence** of fixed length with specific meta-data that fully characterize the sequence and its **members**.

A **regular (GeoPose) sequence** is a **closed sequence** with a constant **inter-pose duration**.

An **irregular (GeoPose) sequence** is a **closed sequence** with a variable **inter-pose duration**. Each **pose** in an **irregular sequence** has an associated **valid time**. A **GeoPose stream** is an **irregular sequence** of unbounded length.

A **(sequence) header** is metadata essential for interpretation of the following **members** of a **sequence**.

A **transition model** is metadata that indicates whether or how it may be possible to estimate **poses** in the interval between consecutive **poses** in a **sequence**.

NOTE

Poses always represent the position and orientation of a real or virtual physical entity. There is temporal continuity of pose for any such entity. On the other hand, there is no condition on consecutive poses in a sequence. There are two causes. First, the poses themselves may be representative of a physical object only at the instant assigned to the pose. Consider a service that provides a sequence of predicted timed poses of a camera that would observe a satellite flare (specular reflection of sunlight) for a specific satellite at a specific earth location. Poses between the member poses of the sequence are meaningless. Second, the sampling of poses may not support computation of intermediate poses. Consider poses that are sampled at a rate much slower than the rate of change of the pose of an underlying externally controlled (such as an airplane controlled by a pilot) physical entity. The sampled poses do not constrain or otherwise provide computational control for estimating intermediate poses. Alternatively, the provider of the sequence may declare via metadata whether it is possible and/or reasonable to compute intermediate poses. The provider is in a position to know this information, which may be binary: "none" ⇒ the data do not support the computation of intermediate poses or "interpolate" ⇒ the data do support the computation of intermediate poses - though the method is not prescribed. These are the two values in the enumeration in the LM TransitionModel datatype. I know from my experience with the "fair fight" issue in distributed simulations that there are a lot of possibilities in defining how to interpolate and these are themselves as complex as GeoPose. That’s why I suggest postponing definition of more comprehensive metadata to a later version but leaving this as an enumeration that we can expand to include additional possibilities beyond the binary "none" and "interpolate".

A **(sequence) trailer** is metadata essential for validation of the preceding **members** of a **sequence**.

4.3. Temporal Concepts

These terms are intended to align with terms used in OWL-TIME (https://www.w3.org/TR/owl-time/) . The only temporal frame used in this GeoPose standard is "Unix Time": seconds since the Unix Epoch of 1 January 1970 measured by a virtual "Unix clock", ticking once per "Unix second", and omitting any corrections such as leap seconds. Times before 1 January 1972 are not precisely related to another temporal frame but the value at UTC 1 January 1972 was +63,072,000. This allows precise conversion to and from modern temporal frames. Note that the GeoPose standard does not reference a calendar and encoded values are representations of the count of seconds, rather than a calendar-relative date and time. These times may be converted to UTC and expressed as text (e.g. with ISO 8601-1:2019 and ISO 8601-2:2019) relative to a specific calendar but this is outside the GeoPose scope.

A **temporal frame** is a specification for the interpretation of points on a **Time Line** as **Instants** in relation to a specified **epoch**.

A **time line (time axis)** is a one-dimensional **euclidean space** whose points represent an ordered sequence of **instants** directed from the past to the future.

An **instant** is a specific point on a **time line**.

An **interval** is the timespan between two **Instants** on a **time line**, interpreted in context of the associated **temporal frame**. A **duration** is semi-open: It includes the earlier **instant** but not the later **instant**.

The **duration** of an **interval** is the one-dimensional signed distance between its bounding **instants**. The magnitude of a **length** value depends on the **temporal frame**.

An **epoch** is a specified **instant** that can be used as a reference point to calculate **temporal relationships** and **durations** between **instants**.

A **temporal relationship** between two **instants** is one of: **before**, **coincident**, or **after**. **temporal relationships** are only valid within the context of a specific **temporal frame**.

4.4. Temporal Database Concepts

---

valid time is a **time line** where the time of changes in the existence or validity of real-world objects or property values are located. **instants** in **valid time** mark the temporal location of real-world transitions in existence, property values, or their validity.

transaction time is a **time line** where the time of changes in the presence or validity of the representations of real-world objects or their properties in an information system are located. **instants** in **transaction time** mark the temporal location of actions that create, update, or delete representations of objects or properties.

NOTE	Both of the terms **valid time**> and **transaction time** are used in ways that can refer to **instants** or to **time lines**.

bitemporality is a property of a data representation that denotes that it carries both **valid** and **transaction times**.

Chapter 5. Conventions

This section provides details and examples for conventions used in this document.

5.1. Identifiers

The normative provisions in this document are denoted by the URI

http://www.opengis.net/spec/GeoPose/1.0

All requirements and conformance tests that appear in this document are denoted by partial URIs which are relative to this base.

5.2. Use Cases, Concepts, Logical Model, Standardization Targets, Encodings

5.3. UML Notation

The logical structure of the elements used in the GeoPose Standard is presented in this document in diagrams using the Unified Modeling Language (UML) static structure diagram (see Booch et al. 1997). The UML notations used in this standard are described in the diagram in UML notation (see ISO TS 19103, Geographic information - Conceptual schema language)..

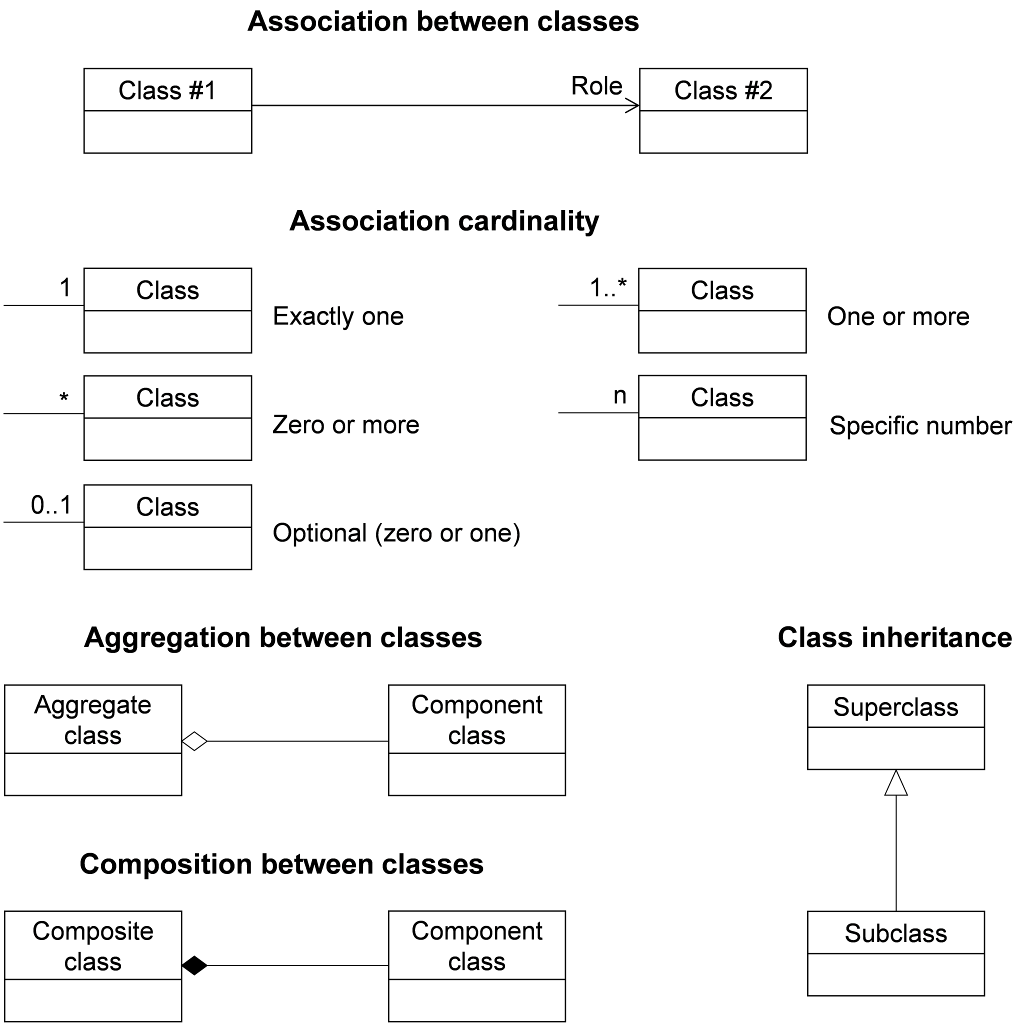

Figure 1. UML notation (see ISO TS 19103, Geographic information - Conceptual schema language).

All associations between model elements in GeoPose are uni-directional. Thus, associations in GeoPose are navigable in only one direction. The direction of navigation is depicted by an arrowhead. In general, the context an element takes within the association is indicated by its role. The role is displayed near the target of the association. If the graphical representation is ambiguous though, the position of the role has to be drawn to the element the association points to.

In order to enhance the readability of the GeoPose UML diagrams, classes are depicted in different colors. The following coloring scheme is applied:

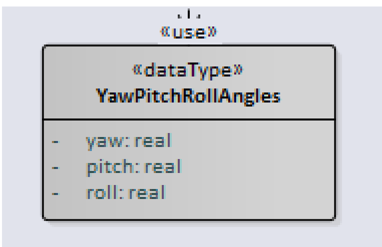

Figure 2. Data Types

Classes painted in grey represent data types.

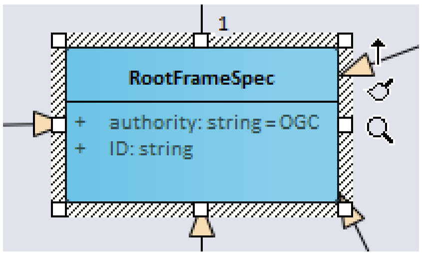

Figure 3. Blue color Denotes Abstract or More General Classes

Classes painted in blue are internal less-derived elements that are not themselves directly encoded as concrete data objects.

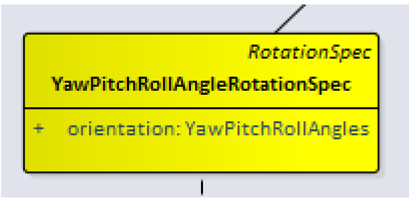

Figure 4. Yellow Color Denotes Structural Data Units

Classes painted in yellow correspond to Structural Data Units or have properties that are represented in Structural Data Units and in encodings of those SDUs. Only data types and classes painted yellow are encoded as concrete data objects.

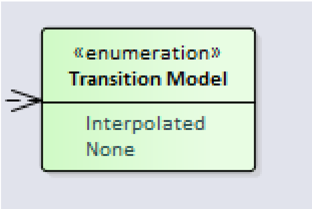

Figure 5. Green Color Denotes Enumeration

Enumerations are painted in green.

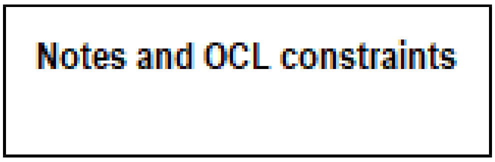

Figure 6. White Color Denotes a Note or Constraint

The color white is used for notes and OCL constraints that are provided in the UML diagrams.

5.4. Conceptual Modelling

ISO 19101 defines universe of discourse to be a view of the real or hypothetical world that includes everything of interest. That standard then defines conceptual model to be a model that defines concepts of a universe of discourse.

The goal of this GeoPose Standard is to establish and document a common set of concepts that spans the targeted use cases. This does not attempt to redefine application concepts, but merely present a common set of concepts from and to which their concepts can be understood and mapped.

The GeoPose conceptual model is a graph of related concepts. One technology-independent realization is given by the GeoPose Logical Model.

Chapter 6. Informative Material

6.1. Document Structure

The structure of the GeoPose Standard document flows from

- use cases to the definition of a conceptual domain of discourse comprehensive enough to support those use cases,
- a realization of a portion of that conceptual domain with an implementation-neutral but specific and normative logical data model expressed in UML, and
- the normative derivation of specific structural data units that represent abstract implementation and standardization targets.

These Structural Data Units(SDU)s are abstract: they are independent of implementation or delivery technology and serialization or encoding formats. GeoPose Version 1.0 specifies one of many possible realizations of the structural data units in JSON.

A key aspect of the GeoPose Standard is that specific use cases are tied to the standardization targets, which prescribe the structure and content of GeoPose data objects. Corresponding implementation examples appear in other documents.

Of course, GeoPose must incorporate or align with other relevant existing standards and common practices. The goal is to fill an interoperability gap in existing standards without reinventing technology in a way that encourages interoperability.

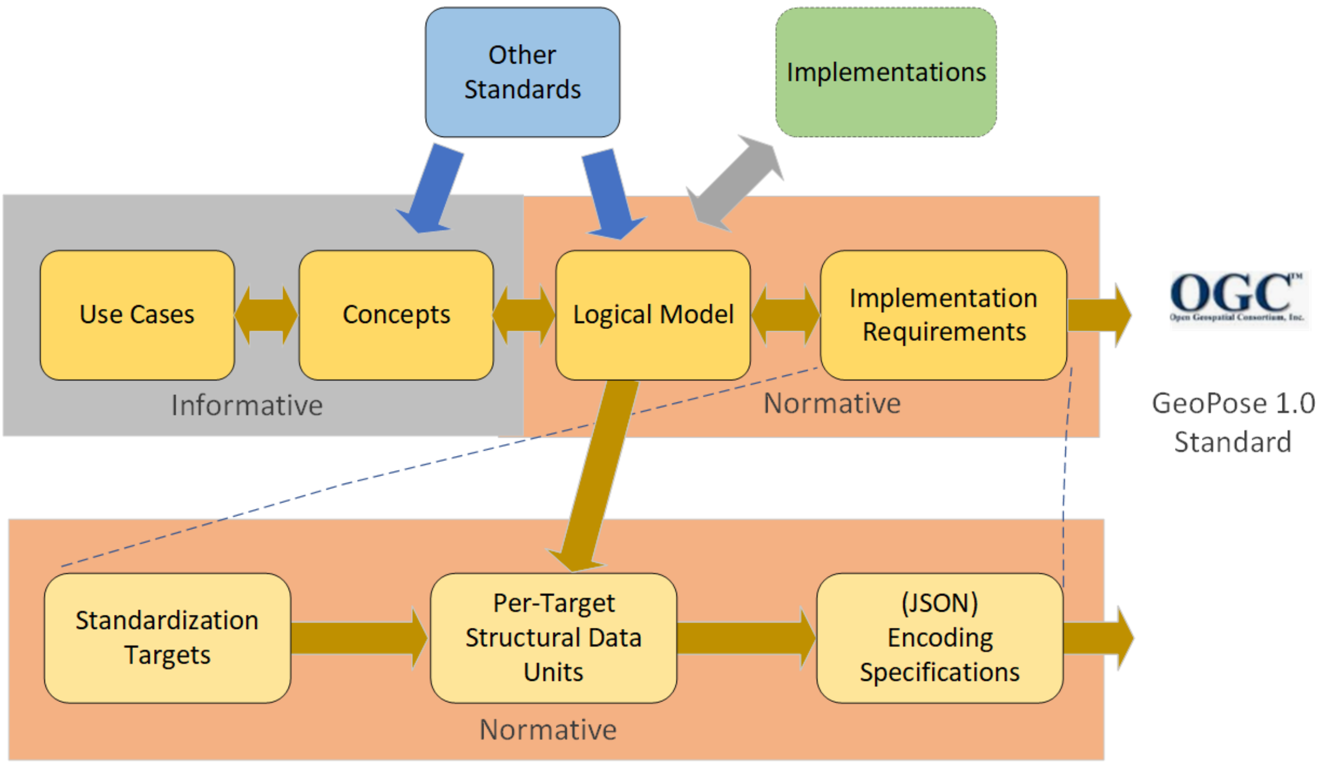

Figure 7. Document Structure Overview

6.2. Use Case Summary

The GeoPose use cases involve interactions between information systems or between an information system and a storage medium. The essential role of a GeoPoses is to convey the position and orientation of a real or virtual object. The possibility of chained transformational relationships and cross-linkages between chains affords representation of complex pose relationships and a way to bring a collection of related GeoPoses in a common geographic reference frame.

Each use case is identified by a unique ID, has a brief description, and a list of the relevant Standardization Targets

6.2.1. Augmented and Mixed Reality [AR]

Description: Augmented reality (AR) integrates synthetic objects or synthetic representations of real objects with a physical environment. Geospatial AR experiences can use GeoPose to position synthetic objects or their representations in the physical environment. The geospatial connection provides a common reference frame to support integration in AR.

Use Cases:

ID	Description	Standardization Target
/geopose/1.0/use_case/ar/01	Stored representation of synthetic objects	Basic-YPR, Basic-Quaternion, Advanced
/geopose/1.0/use_case/ar/02	Positioning information to support integration of synthetic object data in a representation or visualization of the physical environment	Basic-YPR, Basic-Quaternion, Advanced
/geopose/1.0/use_case/ar/03	Report of position and orientation from a mobile device to an AR network service	Advanced (time)
/geopose/1.0/use_case/ar/04	Input to visual occlusion calculations	Basic-YPR, Basic-Quaternion
/geopose/1.0/use_case/ar/05	Input to ray-casting and line-of-sight calculations	Basic-YPR, Basic-Quaternion, Chain
/geopose/1.0/use_case/ar/06	Input to proximity calculations Basic-YPR, Basic-Quaternion,	/geopose/1.0/use_case/ar/07

6.2.2. Autonomous Vehicles [AV]

Description: Autonomous vehicles are mobile objects that move through water, across a water surface, in the air, through the solid earth (tunnel boring machine), on the land surface, or in outer space without real-time control by an independent onboard operator. A pose captures the essential information in positioning and orienting a moving object. Sensors attached to mobile elements have their own poses and a chain of reference frame transformations enables common reference frames to be used for data fusion. The possibility of relating the vehicle to other elements of the environment via a common reference frame is essential.

ID	Description	Standardization Target
/geopose/1.0/use_case/av/01	Provide accurate visual positioning and guidance based on one or more services based on a 3D representation of the real world combined with real time detection and location of real world objects	Basic-YPR, Basic-Quaternion
/geopose/1.0/use_case/av/02	Calculate parameters such as distances and routes (reference to OGC Moving Features?)	Basic-YPR, Basic-Quaternion, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/av/03	Record the trajectory of a moving vehicle.	Regular Timeseries, Irregular Timeseries, Stream

6.2.3. Built Environment: [BE]

Description: The built environment consists of objects constructed by humans and located in physical space. Buildings, roads, dams, railways, and underground utilities are all part of the built environment. The location and orientation of built objects, especially those whose view is occluded by other objects is essential information needed for human interaction with the built environment. A common reference frame tied to the earth's surface facilitates the integration of these objects when their representations are supplied by different sources.

ID	Description	Standardization Target
/geopose/1.0/use_case/be/01	Specify the position and orientation of visible objects and objects that are underground or hidden within a construction.	Basic-YPR, Basic-Quaternion
/geopose/1.0/use_case/be/02	Compactly and consistently specify or share the location and pose of objects in architecture, design and construction.	Basic-YPR, Basic-Quaternion

6.2.4. Synthetic Environments [SE]

Description: Synthetic environments contain collections of moving objects, which themselves may be composed of connected and articulated parts, in an animation or simulation environment that contains a fixed background of air, land, water, vegetation, built objects, and other non-moving elements. The assembly is animated over some time period to provide visualizations or analytical results of the evolving state of the modelled environment. Synthetic environments support training, rehearsal, and archival of activities and events. The location and orientation of the movable elements of a scene are the key data controlling animation of in a synthetic environment. Since there are may be multiple possible animations consistent with observations, storage of the sequences of poses of the actors, vehicles, and implements is a direct and compact way of representing the variable aspects of the event. Access to one or more common reference frames through a graph of frame transformations make a coherent assembly possible

ID	Description	Standardization Target
/geopose/1.0/use_case/se/01	Record pose relationships of all mobile elements in an environment	Graph
/geopose/1.0/use_case/se/02	Control animation of mobile elements in an environment using stored pose time sequences	Graph, Regular Timeseries, Irregular Timeseries, Stream

6.2.5. Image Understanding [IM]

Description: Image understanding is the segmentation of an image or sequence of images into inferred 3D objects in specific semantic categories, possibly determining or constraining their motion and/or geometry. One important application of image understanding is the recognition of moving elements in a time series of images. A pose is a compact representation of the key geometric characteristics of a moving element. In addition to moving elements sensed by an imaging device, it is often useful to know the pose of the sensor or imaging device itself. A common geographic reference frame integrates the objects into a single environment.

ID	Description	Implementation Target
/geopose/1.0/use_case/im/01	Instantaneous and time series locations and orientations of mobile objects	Basic-YPR, Basic-Quaternion, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/02	Instantaneous and time series location and orientation of an optical imaging device using Simultaneous Location And Mapping (SLAM)	Basic-YPR, Basic-Quaternion, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/03	Instantaneous and time series estimation of the changes in location and orientation of an object using an optical imaging device (Visual Odometry)	Basic-YPR, Basic-Quaternion, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/04	Instantaneous and time series location and orientation of an optical imaging device used for photogrammetry	Regular Timeseries, Irregular Timeseries, Stream

Chapter 7. Logical Model

7.1. Description (Informative)

The Frame Transform is the core abstraction in the GeoPose Standard. The Frame Transform is a representation of the transformation taking an Outer Frame coordinate system to an Inner Frame coordinate system. This abstraction is constrained in GeoPose v 1.0 to only allow transformations involving translation and rotation. The intention is to match the usual concept of a pose as a position and orientation. The formalism that expresses a GeoPose Frame Transform is a pair of Reference Frames, Outer and Inner, each defined by a Frame Specification. The Logical Model relates these elements to represent different types of GeoPose data objects and also defines structures built of time series and linked GeoPoses.

7.2. UML Logical Model (Normative)

The normative expression of the UML model is a Sparx Systems Enterprise Explorer project ("eapx")file.

The Logical Model consists of four top-level packages: Core, Time, Sequence, and Targets. The Targets package contains two detail packages: Basic and Composite. The Composite package is in turn subdivided into a Linked package and a Sequence package.The Basic GeoPose targets depend on only the Core package. The Advanced GeoPose target also depends on the Time Package. Composite GeoPoses depend on all four top-level packages.

The coloring of the classes indicates their role in the logical design. Note that the classes and data types defined in the Target packages are the source of structural data units (SDUs) that may be realized as concrete data objects.

7.2.1. Core

The Logical Model Core contains the essential elements specific to the GeoPose modelled as a transformation between an anchoring Outer Frame and one or more derived Inner Frames.

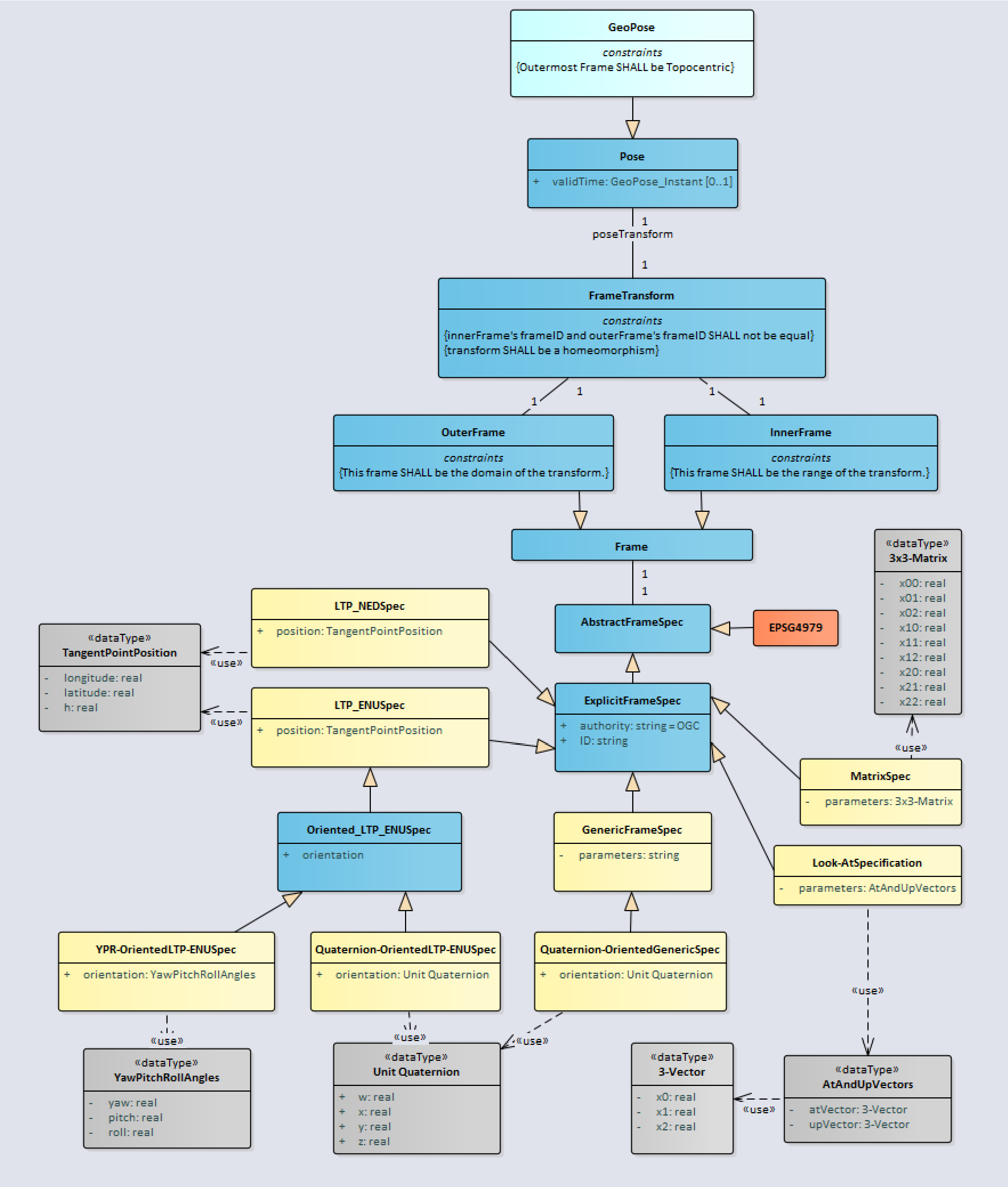

Figure 8. Core Logical Model

7.2.2. Time

The time logical model is based on the OWL Time document.

Only relevant classes, properties, and associations are included. GeoPose v1.0 has a very restricted idea of time position, limited to seconds of UNIX Time.

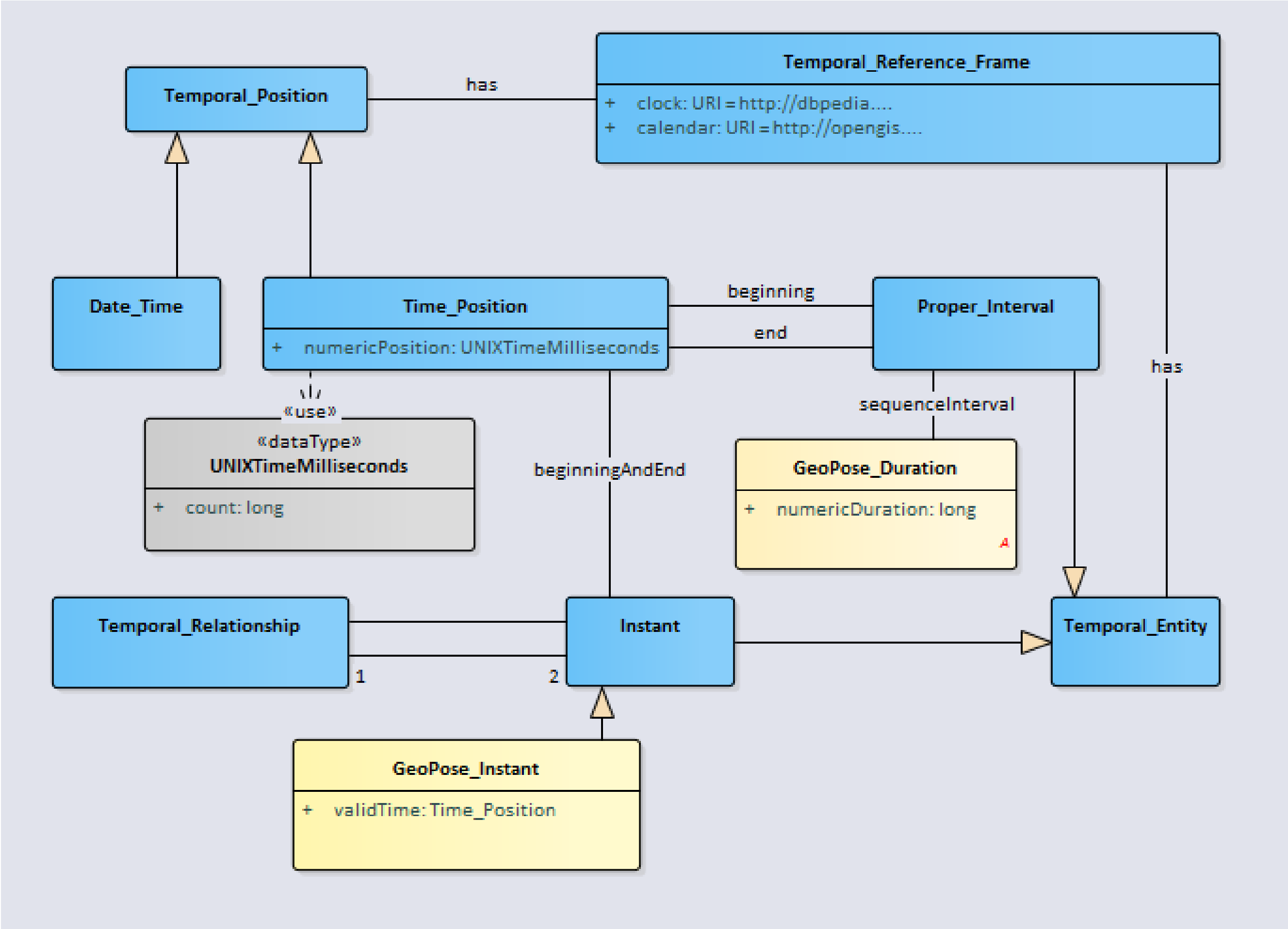

Figure 9. Time Logical model

7.2.3. Sequence

The sequence logical model defines a method for packaging of GeoPose data, where multiple GeoPoses in a sequence share the same **Outer Frame** and there is a time-dependent changing **Inner Frame**.

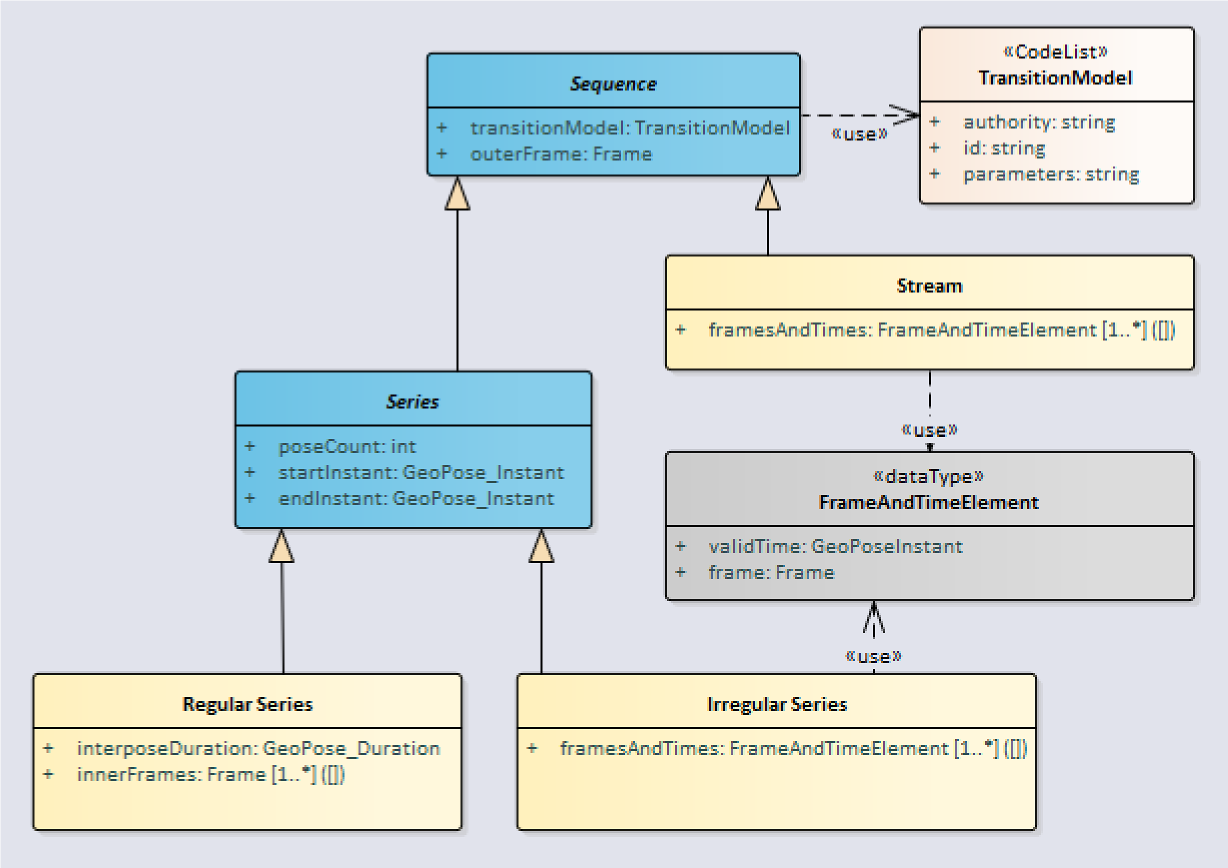

Figure 10. Sequence Logical Model

7.2.4. Targets

The Logical Model's Targets package specify the design of logical data objects and data types that are directly expressed in GeoPose data objects.

The Basic-YPR, Basic-Quaternion, and the Advanced GeoPose SDUs represent single GeoPose objects.

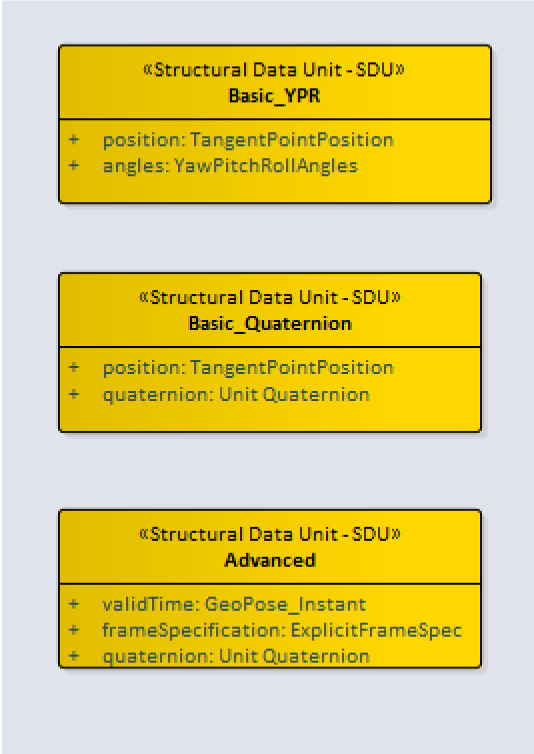

Figure 11. Basic and Advanced Structural Data Units

The Chain and the Graph GeoPose composite structures respectively represent linear or branching frame transformation relationships.

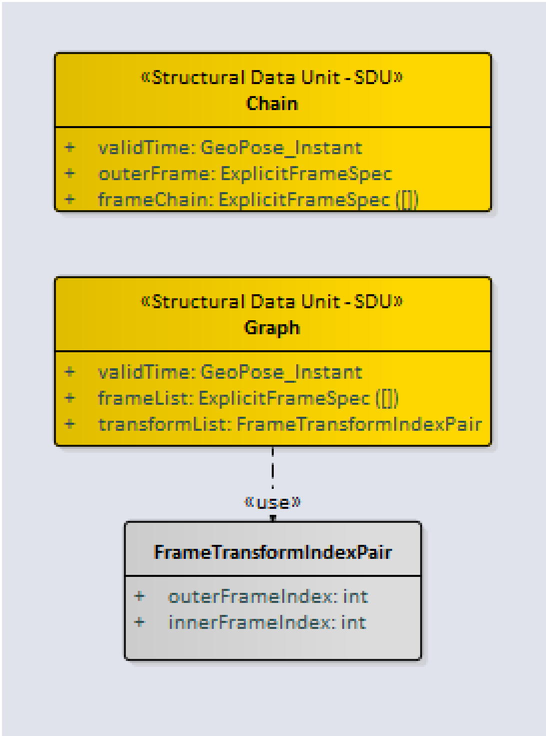

Figure 12. Chain and Graph Structural Data Units

The Stream and each of the two Series composite structures represent time series of a single evolving GeoPose.

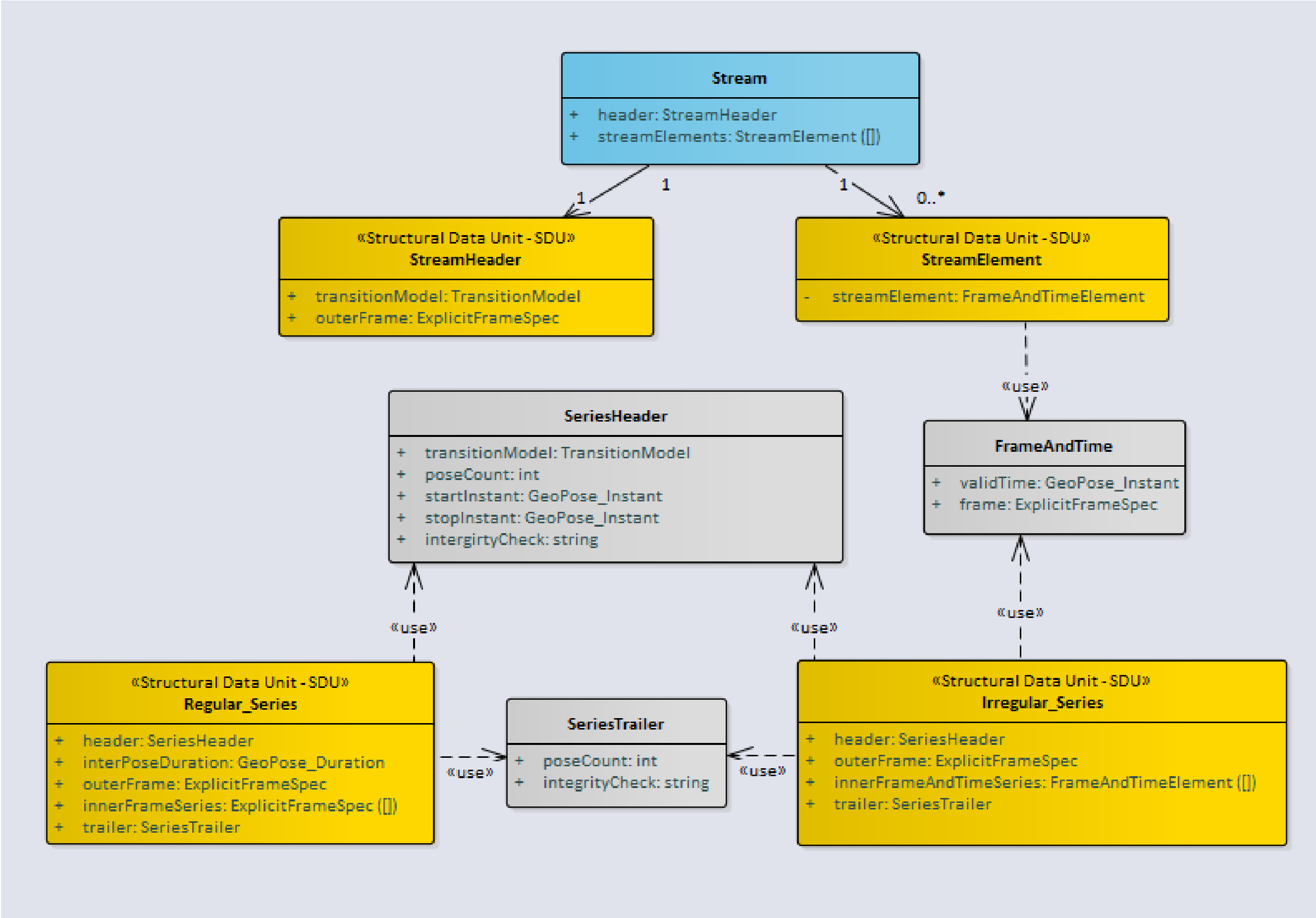

Figure 13. Series and Stream Structural Data Units

Chapter 8. Structural Data Units and Standardization Targets

8.1. Description (Informative)

Classes, attributes, and relationships of the GeoPose domain are specified in a (normative) GeoPose UML static class model - the GeoPose Logical Model. Standardization Targets are specified by encoding-neutral elements of the Logical Model. These Structural Data Units (SDUs) are elements (classes or attributes) in the Logical Model with the "Structural Data Unit - SDU" stereotype. SDUs may have additional Requirements limiting the range, multiplicity, representation or other constraining and testable characteristics. SDUs are used individually or in combination combined to express each of the Standardization Targets.

SDUs provide Standardization Targets that are independent of serialization/encoding format. This allows multiple equivalent serializations to be defined. Each SDU that may be expressed as a concrete data object is associated with a corresponding element (class or attribute) in the logical model.

The Basic and Advanced Standardization Targets differ in the level of options and flexibility in the Frame Specifications. The Composite Targets offer approaches to packaging Frame Transforms. The Targets are the data classes that are specified by the GeoPose Standard. There are eight Standardization Targets denoted by bold terms in the following categories:

1. Basic - Satisfy most use cases
    a. Orientation by Yaw, Pitch, and Roll (YPR) rotations about z, y, x axes: **Basic-YPR** Target
    b. Orientation by unit quaternion: **Basic-Quaternion** Target
2. Configurable - Flexible enough for complex use cases: **Advanced** Target
3. Composite - Efficient structures for linked and sequential GeoPoses
    a. Linked linear sequence of poses: **Chain** Target
    b. General linked poses: **Graph** Target
    c. Sequence
        i. Series
            A. Timeseries with constant time spacing: **Regular** Timeseries Target
            B. Timeseries with per-GeoPose time: **Irregular** Timeseries Target
        ii. Open-ended sequence of time-stamped GeoPoses: **Stream** Target

8.2. Global Requirements (Normative)

Global requirements apply to all SDUs and Standardization Targets.

Requirements Class:	Global
ID	/req/global
Target Type	SDU
Description	Global SDU Requirements
Dependency	None
Requirement G-Target	/req/target_dependency
Requirement G-SDU	/req/sdu_general
Requirement G-LM	/req/lm_general

The individual global requirements are defined as follows:

Requirement:	G-Target
ID	/req/target_dependency
There SHALL be no dependency between or among the individual Standardization Targets.	

Requirement	G-SDU
ID	/req/sdu_general
Implementations using encoded SDUs SHALL conform to the logical description of the Logical Model elements with the "Structural Data Unit - SDU" stereotype.	

Requirement	G-LM
ID	/req/lm_general
Implementations of concrete data conforming to this standard SHALL conform to all dependent or inherited classes, attributes, and associations, multiplicities, and data types in the Logical Model.	

8.2.1. Standardization Target 1: Basic-YPR

Summary (Informative)

The Basic-YPR Target has a simple structure with no options. Position is specified as a point in an LTP-ENU frame and rotation is specified by yaw, pitch, and roll angles specified in decimal degrees.

Structure (Normative)

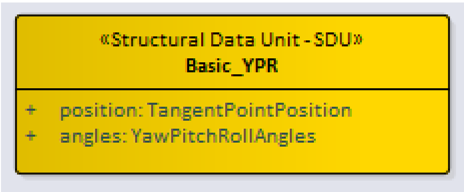

Figure 14. Basic YPR SDU

Requirements Class

Requirements Class:	Basic-YPR
ID	/req/basic/ypr/sdu
Target Type	Implemntation of Logical Model
Description	Basic-YPR Logical Model SDU
Dependency	/req/global
Requirement B-YPR-SDU	/req/basic/ypr/sdu
Requirement B-TP-Lon-SDU	/req/tangent_plane/longitude/sdu
Requirement B-TP-Lat-SDU	/req/tangent_plane/latitude/sdu
Requirement B-TP-h-SDU	req/tangent_plane/h/sdu
Requirement B-YPR-ANGLES-SDU	/req/orientation/ypr_angles/sdu

Details of the individual requirements:

Requirement	B-YPR-SDU
ID	/req/basic/ypr/sdu
An implementation of a Basic-YPR Target SHALL consist of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. The Inner Frame SHALL be a rotation-only transformation using Yaw, Pitch, and Roll angles.	

Requirement	B-TP-Lon-SDU
ID	/req/tangent_plane/longitude/sdu
An instance of a GeoPose tangentPoint.longitude attribute SHALL be expressed as decimal degrees.	

Requirement	B-TP-Lat-SDU
ID	/req/tangent_plane/latitude/sdu
An instance of GeoPose tangentPoint.latitude attribute SHALL be expressed as decimal degrees.	

Requirement	B-TP-h-SDU
ID	/req/tangent_plane/h/sdu
An instance of a GeoPose tangentPoint.h attribute SHALL be expressed as a height in meters above the WGS-84 ellipsoid.	

Requirement	B-YPR-ANGLES-SDU
ID	/req/orientation/ypr_angles/sdu

Requirement	B-YPR-ANGLES-SDU
Yaw, Pitch, and Roll (YPR) angles SHALL be expressed as three consecutive rotations of a reference frame oriented East-North-Up (ENU) coordinate system (where the coordinate axes East, North, and Up correspond to the axes X, Y, Z) about the local (rotated) axes z, y, and x, applied in that order, corresponding to the conventional Yaw, Pitch, and Roll angles. The unit of measure SHALL be the degree.	

8.2.2. Standardization Target 2: Basic-Quaternion

Summary (Informative)

The Basic-Quaternion Target has a simple structure with no options. Position is specified as a point in an LTP-ENU frame and rotation is specified as a unit quaternion.

Structure (Normative)

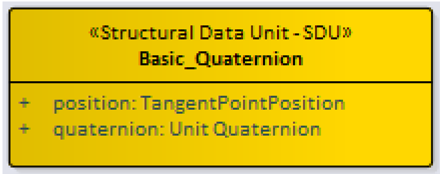

Figure 15. Basic Quaternion SDU

Requirements Class

Requirements Class:	**Basic-Quaternion**
ID	/req/basic/quaternion/sdu
Target Type	Implemntation of Logical Model
Description	Basic-Quaternion Logical Model SDU
Dependency	/req/global
Requirement B-TP-Lon-SDU	/req/tangent_plane/longitude/sdu
Requirement B-TP-Lat-SDU	/req/tangent_plane/latitude/sdu
Requirement B-TP-h-SDU	req/tangent_plane/h/sdu
Requirement B-Quaternion-SDU	/req/orientation/quaternion/sdu

Requirement	**B-TP-Lon-SDU**
ID	/req/tangent_plane/longitude/sdu
An instance of a GeoPose tangentPoint.longitude attribute SHALL be expressed as decimal degrees.	

Requirement	**B-TP-Lat-SDU**
ID	/req/tangent_plane/latitude/sdu
An instance of GeoPose tangentPoint.latitude attribute SHALL be expressed as decimal degrees.	

Requirement	**B-TP-h-SDU**
ID	/req/tangent_plane/h/sdu
An instance of a GeoPose tangentPoint.h attribute SHALL be expressed as a height in meters above the WGS-84 ellipsoid.	

Requirement	**Quaternion-SDU**
ID	/req/orientation/quaternion/sdu

An instance of a GeoPose Logical Model quaternion datatype value SHALL be expressed as four real numbers, representing four quaternion components w, x, y, z, in that sequential order. The sum of the squares of the individual components SHALL be as close to 1.0 as the real number representation allows. The quaternion SHALL be applied to an initial reference frame oriented East-North-Up (ENU) coordinate system where the coordinate axes East, North, and Up correspond to the axes X, Y, Z.

8.2.3. Requirements for Standardization Target 3: Advanced

Summary (Informative)

The Advanced Target has a more general structure, allowing flexible specification of Outer Frame and a Valid Time.

Structure (Normative)

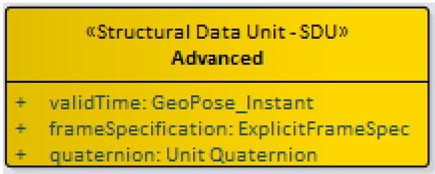

Figure 16. Basic Advanced SDU

Requirements Class

Requirements Class:	**Advanced**
ID	/req/advanced/sdu
Target Type	Implemntation of Logical Model
Description	Advanced Logical Model SDU
Dependency	/req/global
Requirement B-YPR-SDU	/req/advanced/sdu
Requirement B-TP-Lon-SDU	/req/tangent_plane/longitude/sdu
Requirement B-TP-Lat-SDU	/req/tangent_plane/latitude/sdu
Requirement B-TP-h-SDU	req/tangent_plane/h/sdu
Requirement B-Quaternion-SDU	/req/orientation/quaternion/sdu

Requirement	**GP-Instant-SDU**
ID	/req/time/geopose_instant/sdu
The Logical Model attribute GeoPoseInstant SHALL be expressed in Unix Time in seconds multiplied by 1,000. The unit of measure SHALL be milliseconds.	

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**
ID	/req/frame_specification/parameters/sdu
The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.	

Requirement	**Quaternion-SDU**
ID	/req/orientation/quaternion/sdu
An instance of a GeoPose Logical Model quaternion datatype value SHALL be expressed as four real numbers, representing four quaternion components w, x, y, z, in that sequential order. The sum of the squares of the individual components SHALL be as close to 1.0 as the real number representation allows. The quaternion SHALL be applied to an initial reference frame oriented East-North-Up (ENU) coordinate system where the coordinate axes East, North, and Up correspond to the axes X, Y, Z.	

8.2.4. Requirements for Standardization Target 4: Graph

Summary (Informative)

The Graph Target supports a network of object relative poses. The graph is a directed acyclic graph, each node must either be an Extrinsic Frame or reachable from an Extrinsic Frame.

Structure (Normative)

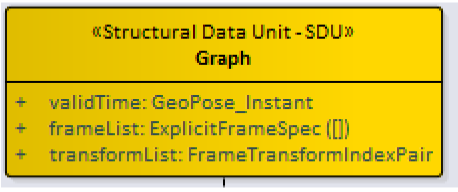

Figure 17. Graph SDU

Requirements Class	**Graph**
ID	/req/graph/sdu
Target Type	Implemntation of Logical Model
Description	Graph Logical Model SDU
Dependency	/req/global
Requirement Graph-SDU	/req/graph/sdu
Requirement B-TP-Lon-SDU	/req/tangent_plane/longitude/sdu
Requirement B-TP-Lat-SDU	/req/tangent_plane/latitude/sdu
Requirement B-TP-h-SDU	req/tangent_plane/h/sdu
Requirement GP-Instant-SDU	/req/time/geopose_instant/sdu

Requirement	**GP-Instant-SDU**
ID	/req/time/geopose_instant/sdu
The Logical Model attribute GeoPoseInstant SHALL be expressed in Unix Time in seconds multiplied by 1,000. The unit of measure SHALL be milliseconds.	

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**
ID	/req/frame_specification/parameters/sdu
The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.	

Requirement	**GP-Graph-Transform_Index-SDU**
ID	/req/chain/transform_index/sdu
Each index vaue in a FrameListTransformPair SHALL be a distinct integer value between 0 and one less than the number of elements in the frameList property.	

8.2.5. Requirements for Standardization Target 5: Chain

Summary (Informative)

The Chain Target supports relationships between a linear sequence of pose relationships. The first frame in the sequence must be an Outer Frame.

Structure

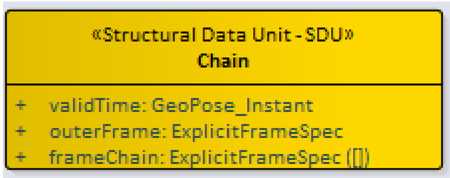

Figure 18. Chain SDU

Requirements Class	**Chain**
ID	/req/chain/sdu
Target Type	Implemntation of Logical Model
Description	Chain Logical Model SDU
Dependency	/req/global
Requirement Chain-SDU	/req/chain/sdu
Requirement B-TP-Lon-SDU	/req/tangent_plane/longitude/sdu
Requirement B-TP-Lat-SDU	/req/tangent_plane/latitude/sdu
Requirement B-TP-h-SDU	req/tangent_plane/h/sdu
Requirement GP-Instant-SDU	/req/time/geopose_instant/sdu

Requirement	**GP-Instant-SDU**
ID	/req/time/geopose_instant/sdu
The Logical Model attribute GeoPoseInstant SHALL be expressed in Unix Time in seconds multiplied by 1,000. The unit of measure SHALL be milliseconds.	

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**

ID	/req/frame_specification/parameters/sdu
The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.	

8.2.6. Requirements for Standardization Target 6: Regular Series

Summary (Informative)

The Regular (Time) Series Target represents the time evolution of a single GeoPose, with a constant time duration between successive inner frames.

Structure (Normative)

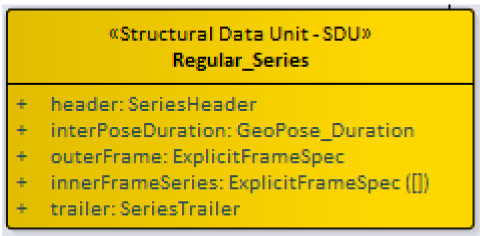

Figure 19. Regular Series SDU

Requirement	**S-Header-SDU**
ID	/req/regular_series/header/sdu
A header property SHALL be implemented as a SeriesHeader structure.	

Requirement	**GP-Duration-SDU**
ID	/req/time/geopose_duration/SDU
Requirement	An interposeDuration property SHALL be implemented as a GeoPoseDuration.

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**
ID	/req/frame_specification/parameters/sdu

The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.

Requirement	**S-Trailer-SDU**
ID	/req/series/regular/trailer/sdu
A trailer property SHALL be implemented as a SeriesTrailer structure.	

8.2.7. Requirements for Standardization Target 7: Irregular Series

Summary (Informative)

The Irregular (Time) Series Target represents the time evolution of a single GeoPose, with a variable time duration between successive inner frames.

Structure (Normative)

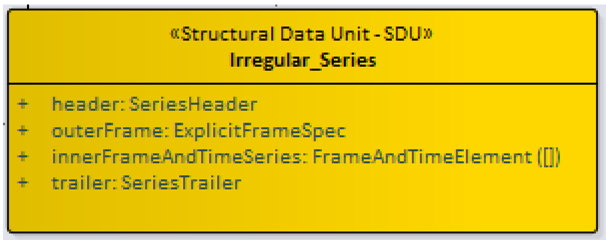

Figure 20. Irregular Series SDU

Requirement	**S-Header-SDU**
ID	/req/series/irregular/header/sdu
Requirement	A header property SHALL be implemented as a SeriesHeader.

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**
ID	/req/frame_specification/parameters/sdu
The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.	

Requirement S-FT-Element-SDU	**/req/IrregularSeries/frameandtime/SDU**
Requirement	An innerFrameAndTime property SHALL be implemented as an ExplicitFrameSpec and GeoPoseInstant pair.
ID	/req/IrregularSeries/frameandtime/SDU

Requirement S-Trailer-SDU	**/req/RegularSeries/trailer/SDU**

Requirement	A trailer property SHALL be implemented as a SeriesTrailer.
ID	/req/RegularSeries/trailer/SDU

8.2.8. Requirements for Standardization Target 8: Stream

Summary (Informative)

The Stream target consists of two parts: a single initial specification of a transition model and an outer frame (the Stream Header) and zero or more time-stamped frame specifications (the Stream Elements). In the delivery of a stream the Header and Elements are not part of a single data structure that exists at a single instant. Nevertheless, recording the Header and all of the Elements received up to some point in time in a single structure is possible. The result is that there are two kinds of data objects that may be involved in transmission of a stream: Headers and Elements and a third kind of object that represents a Recorded Stream.

Structure (Normative)

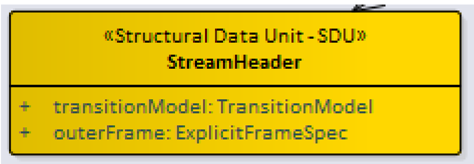

Figure 21. Stream Header SDU

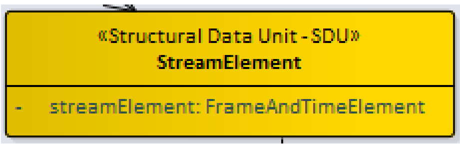

Figure 22. Stream Element SDU

Requirement	**S-TM-SDU**
ID	/req/transition_model/sdu
A TransitionModel property SHALL be implemented as one of the values in the TransitionModel enumeration.	

Requirement	**FS-Authority-SDU**
ID	/req/frame_specification/authority/sdu
The frame_specification.authority attribute SHALL contain a string uniquely specifying a source of reference frame specifications.	

***Requirement**	**FS-ID-SDU**
ID	/req/frame_specification/id/sdu
The frame_specification.ID attribute SHALL be a string uniquely defining a frame within the authority.	

Requirement	**FS-Parameters-SDU**
ID	/req/frame_specification/parameters/sdu
The frame_specification.parameter attribute SHALL contain all parameters needed for the corresponding authority and ID.	

Requirement	**S-FST-SDU**
ID	/req/stream/frame_time_element/sdu
Requirement	A FrameAndTimeElement property SHALL be implemented as as an ExplicitFrameSpec and a GeoPoseInstant.

Chapter 9. Requirements for Encodings

9.1. Description (Informative)

Requirements Classes are modularized based on the corresponding Standardization Target. This results in some SDU requirements being repeated between Targets. SDU requirements are abstract in the sense that SDUs are implemented as concrete data objects via serialization formats or encodings. Therefore, there are additional requirements that specify how each Target's group of SDUs are encoded. If there are multiple encodings of a Target, then there is a corresponding additional set of encoding requirements in the Target's section. This occurs only once in GeoPose 1.0, with two different levels of JSON encoding strictness individually specified for the Basic-Q Target.

9.2. JSON Encoding

9.2.1. Standardization Target 1: Basic-YPR (Normative)

NOTE	This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

Requirement	**B-YPR-Encoding-JSON**
ID	/req/basic/ypr/encoding/json
A JSON-encoded Basic-YPR GeoPose SHALL conform to the following Basic-YPR JSON-Schema 2019-9 definition.	

JSON-Schema:

```
{
  "description": "Basic-YPR: Basic GeoPose using yaw, pitch, and roll to specify
orientation",
  "definitions": {
    "angles": {
      "type": "object",
      "properties": {
        "yaw": {
          "type": "number"
        },
        "pitch": {
          "type": "number"
        },
        "roll": {
          "type": "number"
        }
      },
      "required": [
        "yaw",
        "pitch",
        "roll"
      ]
    },
    "Position": {
      "type": "object",
      "properties": {
        "lat": {
          "type": "number"
        },
        "lon": {
          "type": "number"
        },
```

```
        "h": {
          "type": "number"
        }
      },
      "required": [
        "lat",
        "lon",
        "h"
      ]
    }
  },
  "type": "object",
  "properties": {
    "position": {
      "$ref": "#/definitions/Position"
    },
    "angles": {
      "$ref": "#/definitions/angles"
    }
  },
  "required": [
    "position",
    "angles"
  ]
}
```

Instance (Informative)

```
{
  "position": {
    "lat": 47.7,
    "lon": -122.3,
    "h": 11.5
  },
  "angles": {
    "yaw": 5.514456741060452,
    "pitch": -0.43610515937237904,
    "roll": 0.0
  }
}
```

9.2.2. Standardization Target 2: Basic-Quaternion (Normative)

> **NOTE**
>
> Two JSON encodings are defined for the Basic-Quaternion Target: **Strict**, disallowing additional JSON properties not defined in the schema and **Extensible**, allowing additional JSON properties in addition to those required by the schema. All other targets follow the default and permit additional JSON properties.

Strict JSON Encoding (Normative)

Requirement	**B-Quaternion-Encoding-JSON-Strict**
ID	/req/basic/quaternion/encoding/json_strict
A JSON-encoded Basic-Quaternion (Strict) GeoPose SHALL conform to the following Basic-Quaternion (Strict) JSON-Schema 2019-9 definition. There SHALL be no encoded properties not explicitly defined in the JSON-Schema definition.	

JSON-Schema:

```
{
  "description": "Basic-Quaternion-Strict: Basic GeoPose using quaternion to specify
orientation - no additional properties",
  "definitions": {
    "Position": {
      "type": "object",
      "properties": {
        "lat": {
          "type": "number"
        },
        "lon": {
          "type": "number"
        },
        "h": {
          "type": "number"
        }
      },
      "required": [
        "lat",
        "lon",
        "h"
      ]
    },
    "Quaternion": {
      "type": "object",
      "properties": {
        "x": {
          "type": "number"
        },
        "y": {
          "type": "number"
```

```
        },
        "z": {
          "type": "number"
        },
        "w": {
          "type": "number"
        }
      },
      "required": [
        "x",
        "y",
        "z",
        "w"
      ]
    }
  },
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "position": {
      "$ref": "#/definitions/Position"
    },
    "quaternion": {
      "$ref": "#/definitions/Quaternion"
    }
  },
  "required": [
    "position",
    "quaternion"
  ]
}
```

Instance-Strict (Informative)

```
{
  "position": {
    "lat": 47.7,
    "lon": -122.3,
    "h": 11.5
  },
  "quaternion": {
    "x": 0.20054473382601948,
    "y": -0.08111675703887213,
    "z": 0.3660908114262869,
    "w": -0.9050852994339209
  }
}
```

Permissive JSON Encoding (Normative)

NOTE	This JSON encoding of the Basic-Quaternion GeoPose is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**. This encoding is intended to be the default GeoPose.

Requirement	**B-Quaternion-Encoding-JSON**
ID	/req/basic/quaternion/encoding/json
A JSON-encoded Basic-Quaternion GeoPose SHALL conform to the following Basic-Quaternion JSON-Schema 2019-9 definition.	

JSON-Schema:

```
{
  "description": "Basic-Quaternion-Strict: Basic GeoPose using quaternion to specify
orientation - no additional properties",
  "definitions": {
    "Position": {
      "type": "object",
      "properties": {
        "lat": {
          "type": "number"
        },
        "lon": {
          "type": "number"
        },
        "h": {
          "type": "number"
        }
      },
      "required": [
        "lat",
        "lon",
        "h"
      ]
    },
    "Quaternion": {
      "type": "object",
      "properties": {
        "x": {
          "type": "number"
        },
        "y": {
          "type": "number"
        },
        "z": {
          "type": "number"
        },
        "w": {
```

```
            "type": "number"
          }
        },
        "required": [
          "x",
          "y",
          "z",
          "w"
        ]
      }
    },
    "type": "object",
    "additionalProperties": false,
    "properties": {
      "position": {
        "$ref": "#/definitions/Position"
      },
      "quaternion": {
        "$ref": "#/definitions/Quaternion"
      }
    },
    "required": [
      "position",
      "quaternion"
    ]
  }
```

Instance (Informative)

```
{
  "position": {
    "lat": 47.7,
    "lon": -122.3,
    "h": 11.5
  },
  "quaternion": {
    "x": 0.20054473382601948,
    "y": -0.08111675703887213,
    "z": 0.3660908114262869,
    "w": -0.9050852994339209
  }
}
```

9.2.3. Standardization Target 3: Advanced GeoPose (Normative)

NOTE This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

Requirement	Advanced-Encoding-JSON
ID	/req/advanced/encoding/json
A JSON-encoded Advanced GeoPose SHALL conform to the following Advanced JSON-Schema 2019-9 definition.	

JSON-Schema:

```
{
  "description": "Advanced: Advanced GeoPose allowing flexible outer frame
specification, quaternion orientation, and valid time.",
  "definitions": {
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
```

```
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "Quaternion": {
      "type": "object",
      "properties": {
        "x": {
          "type": "number"
        },
        "y": {
          "type": "number"
        },
        "z": {
          "type": "number"
        },
        "w": {
          "type": "number"
        }
      },
      "required": [
        "x",
        "y",
        "z",
        "w"
      ]
    }
  },
  "type": "object",
  "properties": {
    "frameSpecification": {
      "$ref": "#/definitions/FrameSpecification"
    },
    "quaternion": {
      "$ref": "#/definitions/Quaternion"
    },
    "validTime": {
      "type": "integer"
    }
  },
  "required": [
    "frameSpecification",
    "quaternion"
  ]
}
```

Instance (Informative)


```
{
  "frameSpecification": {
    "authority": "/geopose/1.0",
    "id": "LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  },
  "quaternion": {
    "x": 0.20056154657066608,
    "y": -0.08111602541464237,
    "z": 0.36606032744426537,
    "w": -0.9050939692261301
  },
  "validTime": 1630560671227
}
```

9.2.4. Standardization Target 4: Graph (Normative)

NOTE	This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

Requirement	**Graph-Encoding-JSON**
ID	/req/graph/encoding/json
A JSON-encoded GeoPose Graph SHALL conform to the following GeoPose Graph JSON-Schema 2019-9 definition.	

JSON-Schema:

```
{
  "description": "Graph: An general structure modelling the pose relationship between
frames (nodes) and transforms (edges) in a graph structure.",
  "definitions": {
    "FrameSpecification": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "FrameTransformPair": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "link": {
          "type": [
            "array",
```

```
          "null"
        ],
        "items": {
          "type": "integer"
        }
      }
    },
    "required": [
      "link"
    ]
  }
},
"type": "object",
"properties": {
  "validTime": {
    "type": "integer"
  },
  "frameList": {
    "type": "array",
    "items": {
      "$ref": "#/definitions/FrameSpecification"
    },
    "minItems": 2
  },
  "transformList": {
    "type": "array",
    "items": {
      "$ref": "#/definitions/FrameTransformPair"
    },
    "minItems": 1
  }
},
"required": [
  "validTime",
  "frameList",
  "transformList"
]
}
```

Instance (Informative)


```
{
  "validTime": 1630560671313,
  "frameList": [
    {
      "authority": "/geopose/1.0",
      "id": "/Extrinsic/LTP-ENU",
      "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    }
  ],
  "transformList": [
    {
      "link": [
```

```
        0,
        1
      ]
    },
    {
      "link": [
        1,
        2
      ]
    },
    {
      "link": [
        2,
        3
      ]
    },
    {
      "link": [
        0,
        4
      ]
    },
    {
      "link": [
        4,
        5
      ]
    },
    {
      "link": [
        5,
        6
      ]
    }
  ]
}
```

9.2.5. Standardization Target 5: Chain (Normative)

NOTE	This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

JSON-Schema:

Requirement	**Chain-Encoding-JSON**
ID	/req/chain/encoding/json
A JSON-encoded GeoPose Chain SHALL conform to the following GeoPose Chain JSON-Schema 2019-9 definition.	

```
{
  "description": "Chain: An outer frame and a sequence of transformations to a final
innermost frame.",
  "definitions": {
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "FrameSpecification-1": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
```

```
          "parameters": {
            "type": "string"
          }
        },
        "required": [
          "authority",
          "id",
          "parameters"
        ]
      }
    },
    "type": "object",
    "properties": {
      "validTime": {
        "type": "integer"
      },
      "outerFrame": {
        "$ref": "#/definitions/FrameSpecification"
      },
      "frameChain": {
        "type": "array",
        "items": {
          "$ref": "#/definitions/FrameSpecification-1"
        },
        "minItems": 2
      }
    },
    "required": [
      "validTime",
      "outerFrame",
      "frameChain"
    ]
  }
```

Instance (Informative)

```
{
  "validTime": 1630560671263,
  "outerFrame": {
    "authority": "/geopose/1.0",
    "id": "/Extrinsic/LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  },
  "frameChain": [
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "/Intrinsic/Translate-Rotate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[0.69291, 0.69291, 0.14097,
0.14097]"
    }
  ]
}
```

9.2.6. Standardization Target 6: Regular Series (Normative)

NOTE This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

JSON-Schema:

Requirement	**Regular-Series-Encoding-JSON**
ID	/req/series/regular/encoding/json
A JSON-encoded GeoPose Regular Series SHALL conform to the following GeoPose Regular JSON-Schema 2019-9 definition.	

```
{
  "description": "Regular Series: Regular GeoPose time series with constant inter-pose
```

```
duration.",
  "definitions": {
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "FrameSpecification-1": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "SeriesHeader": {
      "type": "object",
      "properties": {
        "poseCount": {
          "type": "integer"
        },
        "integrityCheck": {
          "type": [
```

```
          "string",
          "null"
        ]
      },
      "startInstant": {
        "type": "integer"
      },
      "stopInstant": {
        "type": "integer"
      },
      "transitionModel": {
        "$ref": "#/definitions/TransitionModel"
      }
    },
    "required": [
      "poseCount",
      "startInstant",
      "stopInstant",
      "transitionModel"
    ]
  },
  "SeriesTrailer": {
    "type": "object",
    "properties": {
      "poseCount": {
        "type": "integer"
      },
      "integrityCheck": {
        "type": [
          "string",
          "null"
        ]
      }
    },
    "required": [
      "poseCount"
    ]
  },
  "TransitionModel": {
    "type": "object",
    "properties": {
      "authority": {
        "type": "string"
      },
      "id": {
        "type": "string"
      },
      "parameters": {
        "type": "string"
      }
    },
```

```
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    }
  },
  "type": "object",
  "properties": {
    "header": {
      "$ref": "#/definitions/SeriesHeader"
    },
    "interPoseDuration": {
      "type": "integer"
    },
    "outerFrame": {
      "$ref": "#/definitions/FrameSpecification"
    },
    "innerFrameSeries": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/FrameSpecification-1"
      },
      "minItems": 1
    },
    "trailer": {
      "$ref": "#/definitions/SeriesTrailer"
    }
  },
  "required": [
    "header",
    "interPoseDuration",
    "outerFrame",
    "innerFrameSeries",
    "trailer"
  ]
}
```

Instance (Informative)


```
{
  "header": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}",
    "startInstant": 1630560671367,
    "stopInstant": 1630560716367,
    "transitionModel": {
      "authority": "/geopose/1.0",
      "id": "none",
      "parameters": ""
    }
  },
  "interPoseDuration": 1000,
  "outerFrame": {
    "authority": "/geopose/1.0",
    "id": "LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  },
  "innerFrameSeries": [
    {
      "authority": "/geopose/1.0",
      "id": "RotateTranslate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "RotateTranslate",
      "parameters": "translation=[0.5, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
    },
    {
      "authority": "/geopose/1.0",
      "id": "RotateTranslate",
      "parameters": "translation=[1.0, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
    }
  ],
  "trailer": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}"
  }
}
```

9.2.7. Standardization Target 7: Irregular Series (Normative)

NOTE	This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

JSON-Schema:

Requirement	**Irregular-Series-Encoding-JSON**
ID	/req/series/irregular/encoding/json
A JSON-encoded GeoPose Irregular Series SHALL conform to the following GeoPose Irregular JSON-Schema 2019-9 definition.	

```
{
  "description": "Irregular Series: Irregular GeoPose time series with variable inter-
pose duration.",
  "definitions": {
    "FrameAndTime": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "frame": {
          "$ref": "#/definitions/FrameSpecification"
        },
        "validTime": {
          "type": "integer"
        }
      },
      "required": [
        "frame"
      ]
    },
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
```

```
        "authority",
        "id",
        "parameters"
      ]
    },
    "SeriesHeader": {
      "type": "object",
      "properties": {
        "poseCount": {
          "type": "integer"
        },
        "integrityCheck": {
          "type": [
            "string",
            "null"
          ]
        },
        "startInstant": {
          "type": "integer"
        },
        "stopInstant": {
          "type": "integer"
        },
        "transitionModel": {
          "$ref": "#/definitions/TransitionModel"
        }
      },
      "required": [
        "poseCount",
        "startInstant",
        "stopInstant",
        "transitionModel"
      ]
    },
    "SeriesTrailer": {
      "type": "object",
      "properties": {
        "poseCount": {
          "type": "integer"
        },
        "integrityCheck": {
          "type": [
            "string",
            "null"
          ]
        }
      },
      "required": [
        "poseCount"
      ]
    },
```

```json
    "TransitionModel": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    }
  },
  "type": "object",
  "properties": {
    "header": {
      "$ref": "#/definitions/SeriesHeader"
    },
    "outerFrame": {
      "$ref": "#/definitions/FrameSpecification"
    },
    "innerFrameAndTimeSeries": {
      "type": "array",
      "items": {
        "$ref": "#/definitions/FrameAndTime"
      },
      "minItems": 1
    },
    "trailer": {
      "$ref": "#/definitions/SeriesTrailer"
    }
  },
  "required": [
    "header",
    "outerFrame",
    "innerFrameAndTimeSeries",
    "trailer"
  ]
}
```

Instance (Informative)

```
{
  "header": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}",
    "startInstant": 1630560671429,
    "stopInstant": 1630560716429,
    "transitionModel": {
      "authority": "/geopose/1.0",
      "id": "none",
      "parameters": ""
    }
  },
  "outerFrame": {
    "authority": "/geopose/1.0",
    "id": "LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  },
  "innerFrameAndTimeSeries": [
    {
      "frame": {
        "authority": "/geopose/1.0",
        "id": "RotateTranslate",
        "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
      },
      "validTime": 1630560671429
    },
    {
      "frame": {
        "authority": "/geopose/1.0",
        "id": "RotateTranslate",
        "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
      },
      "validTime": 1630560671429
    },
    {
      "frame": {
        "authority": "/geopose/1.0",
        "id": "RotateTranslate",
        "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[1.0, 0.0, 0.0, 0.5]"
      },
      "validTime": 1630560671429
    }
  ],
  "trailer": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}"
  }
}
```

9.2.8. Standardization Target 8: Stream (Normative)

NOTE	This JSON encoding is extensible because the JSON-Schema "addtionalProperties" property is set to the default value of **true**.

JSON Encoding (Normative)

JSON-Schemata:

Requirement	**Stream-Header-Encoding-JSON**
ID	/req/stream/header/encoding/json
A JSON-encoded GeoPose Stream Header SHALL conform to the following GeoPose Stream Header JSON-Schema 2019-9 definition.	

```
{
  "description": "Composite: StreamHeader - appears once at the beginning of a
stream.",
  "definitions": {
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "TransitionModel": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
```

```
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    }
  },
  "type": "object",
  "properties": {
    "transitionModel": {
      "$ref": "#/definitions/TransitionModel"
    },
    "outerFrame": {
      "$ref": "#/definitions/FrameSpecification"
    }
  },
  "required": [
    "transitionModel",
    "outerFrame"
  ]
}
```

Requirement	**Stream-Element-Encoding-JSON**
ID	/req/stream/element/encoding/json
A JSON-encoded GeoPose Stream Element SHALL conform to the following GeoPose Stream Element JSON-Schema 2019-9 definition.	

```json
{
  "description": "Stream Element: The repeated information streamed at irregular
times.",
  "definitions": {
    "FrameAndTime": {
      "type": "object",
      "properties": {
        "frame": {
          "$ref": "#/definitions/FrameSpecification"
        },
        "validTime": {
          "type": "integer"
        }
      },
      "required": [
        "frame"
      ]
    },
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    }
  },
  "type": "object",
  "properties": {
    "streamElement": {
      "$ref": "#/definitions/FrameAndTime"
    }
  },
  "required": [
    "streamElement"
  ]
}
```

Requirement	Stream-Record-Encoding-JSON
ID	/req/stream/record/encoding/json
A JSON-encoded GeoPose Stream Record (a recorded Stream) SHALL conform to the following GeoPose Stream Record JSON-Schema 2019-9 definition.	

```
{
  "description": "Stream: GeoPose stream is an open-ended irregular timeseries
suitable for streaming from a sensor or information service.",
  "definitions": {
    "FrameAndTime": {
      "type": "object",
      "properties": {
        "frame": {
          "$ref": "#/definitions/FrameSpecification"
        },
        "validTime": {
          "type": "integer"
        }
      },
      "required": [
        "frame"
      ]
    },
    "FrameSpecification": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    },
    "StreamElement": {
      "type": [
        "object",
        "null"
      ],
      "properties": {
        "streamElement": {
```

```
          "$ref": "#/definitions/FrameAndTime"
        }
      },
      "required": [
        "streamElement"
      ]
    },
    "StreamHeader": {
      "type": "object",
      "properties": {
        "transitionModel": {
          "$ref": "#/definitions/TransitionModel"
        },
        "outerFrame": {
          "$ref": "#/definitions/FrameSpecification"
        }
      },
      "required": [
        "transitionModel",
        "outerFrame"
      ]
    },
    "TransitionModel": {
      "type": "object",
      "properties": {
        "authority": {
          "type": "string"
        },
        "id": {
          "type": "string"
        },
        "parameters": {
          "type": "string"
        }
      },
      "required": [
        "authority",
        "id",
        "parameters"
      ]
    }
  },
  "type": "object",
  "properties": {
    "header": {
      "$ref": "#/definitions/StreamHeader"
    },
    "streamElements": {
      "type": [
        "array",
        "null"
```

```
      ],
      "items": {
        "$ref": "#/definitions/StreamElement"
      },
      "minItems": 1
    }
  },
  "required": [
    "header",
    "streamElements"
  ]
}
```

Instances (Informative)

Valid JSON encoding of a Stream Header instance:

```
{
  "transitionModel": {
    "authority": "/geopose/1.0",
    "id": "interpolate",
    "parameters": ""
  },
  "outerFrame": {
    "authority": "/geopose/1.0",
    "id": "LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  }
}
```

Valid JSON encoding of a Stream Element instance:

```
{
  "streamElement": {
    "frame": {
      "authority": "/geopose/1.0",
      "id": "RotateTranslate",
      "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.90510, 0.20057,
-0.08112, 0.36605]"
    },
    "validTime": 1630560671474
  }
}
```

Valid JSON encoding of a Recorded Stream:

```
{
  "header": {
    "transitionModel": {
      "authority": "/geopose/1.0",
      "id": "interpolate",
      "parameters": ""
    },
    "outerFrame": {
      "authority": "/geopose/1.0",
      "id": "LTP-ENU",
      "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
    }
  },
  "streamElements": [
    {
      "streamElement": {
        "frame": {
          "authority": "/geopose/1.0",
          "id": "RotateTranslate",
          "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.90510, 0.20057,
-0.08112, 0.36605]"
        },
        "validTime": 1630560671474
      }
    },
    {
      "streamElement": {
        "frame": {
          "authority": "/geopose/1.0",
          "id": "RotateTranslate",
          "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.90566, 0.20166,
-0.08106, 0.36406]"
        },
        "validTime": 1630560683820
      }
    },
    {
      "streamElement": {
        "frame": {
          "authority": "/geopose/1.0",
          "id": "RotateTranslate",
          "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.90622, 0.20275,
-0.08101, 0.36207]"
        },
        "validTime": 1630560696165
      }
    }
  ]
}
```

Chapter 10. Media Types for JSON Encoding

application/json: http://www.iana.org/assignments/media-types/application/json

Annex A: Abstract Test Suite (Normative)

A.1. Introduction

GeoPose 1.0 specifies a eight Standardization Targets using elements of the Logical Model. These elements are Structural Data Units (SDUs) and they have the stereotype "Structural Data Unit - SDU". Each SDU is an element of the Logical Model that will be expressed in concrete data objects encoded using specific encoding or serialization technologies.

Although implementation of the Standardization Targets, expressed as SDUs, is independent of the Logical Model, GeoPose 1.0 also defines one of many possible implementations, single encoding in JavaScript Object Notation (JSON). The encodings of the eight targets are specified using JSON-Schema-2019-09. To keep the individual Standardization targets independent, there are some SDU requirements and corresponding conformance tests that appear in more than one requirement or conformance class. This structure is based on the judgement that it is easier to understand the independence of targets with complete definitions than would be the case if the definitional requirements of the SDUs were factored out and referenced indirectly by individual encodings.

A.2. Global Conformance Class

Conformance with the Global Requirements is required for all implementations.

Conformance Class:	**Global**
ID	/conf/global
Target Type	SDU
Description	Conformance with Global SDU requirements
Dependency	None
Test: G-Target	/conf/target_dependency
Test: G-SDU	/conf/sdu_general
Test: G-LM	/conf/lm_general

The individual global tests are:

Test:	**G-Target**
ID	/conf/target_dependency
There SHALL be no dependency between or among the individual Standardization Targets.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: G-Target	/req/target_dependency
Test Type	Conformance

Test:	G-SDU
ID	/conf/sdu_general
Implementations using encoded SDUs SHALL conform to the logical description of the Logical Model elements with the "Structural Data Unit - SDU" stereotype.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: G-SDU	/req/sdu_general
Test Type	Conformance

Test:	G-LM
ID	/conf/lm_general
Implementations of concrete data conforming to this standard SHALL conform to all dependent or inherited classes, attributes, and associations, multiplicities, and data types in the Logical Model.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: G-LM	/req/lm_general
Test Type	Conformance

A.3. Structural Data Unit (SDU) Conformance

There are some universal requirements on values that appear in a concrete implementation using a specific encoding technology. For example, angles may be constrainted to fall within a range of values correponding to a circle. Because these are independent of encoding technology, they are specified here at a logical level. Tests of an implementation at the SDU level generally only be done by inspection.

A.4. Basic-YPR SDU Conformance Class

Conformance Class:	Basic-YPR SDU
ID	/conf/basic_ypr_sdu
Target Type	SDU
Description	To confirm that a Basic-YPR GeoPose consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/global
Test: B-YPR-SDU	/conf/basic_ypr/sdu
Test: B-TP-Lon-SDU	/conf/tangent_plane/longitude/sdu

Conformance Class:	Basic-YPR SDU
Test: B-TP-Lat-SDU	/conf/tangent_plane/latitude/sdu
Test: B-TP-h-SDU	/conf/tangent_plane/h/sdu
Test: YPR-Angles	/conf/ypr_angles/sdu

The Basic-YPR SDU member tests are the following:

Test:	B-YPR-SDU
ID	/conf/basic_ypr/sdu
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

Test:	B-TP-Lon-SDU
ID	/conf_tangent_plane_longitude_sdu
To confirm that a GeoPose tangentPoint.longitude attribute is expressed as an angle in decimal degrees.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: Tangent-Plane-Longitude	/req/tangent_plane_longitude_sdu
Test Type	Conformance

Test:	B-TP-Lat-SDU
ID	/conf_tangent_plane_latitude_sdu
To confirm that a GeoPose tangentPoint.latitude attribute is expressed as an angle in decimal degrees.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: Tangent-Plane-Latitude	/req/tangent_plane_latitude_sdu
Test Type	Conformance

Test:	B-TP-h-SDU
ID	/conf_tangent_plane_h_sdu
To confirm that a GeoPose tangentPoint.h attribute is expressed as a height in meters above the WGS-84 ellipsoid.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: Tangent-Plane-h	/req/tangent_plane_h_sdu
Test Type	Conformance

Test:	YPR-Angles
ID	/conf_ypr_angles_sdu
To confirm that GeoPose YPR angles are expressed as three consecutive rotations about the local axes Z, Y, and X, in that order, corresponding to the conventional Yaw, Pitch, and Roll angles and that the unit of measure is the degree.	
Purpose	Verify that this requirement is satisfied
Method	Inspection
Reference: YPR-Angles	/req/ypr_angles_sdu
Test Type	Conformance

A.4.1. Basic-Q SDU Conformance Class

Abstract Test 1	/conf/basic/quaternion/sdu
Test Purpose	To confirm that a Basic-Q GeoPose consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using a unit quaternion.
Requirement	/req/basic/quaternion/sdu
Test Method	Inspection

Abstract Test 2	/conf/tangent_plane/longitude/sdu
Test Purpose	To confirm that a GeoPose tangentPoint.longitude attribute is expressed as a real number.
Requirement	/req/tangent_plane/longitude/sdu
Test Method	Inspection

Abstract Test 3	/conf/tangent_plane/latitude/sdu
Test Purpose	To confirm that a GeoPose tangentPoint.latitude attribute is expressed as a real number.

Requirement	/req/tangent_plane/latitude/sdu
Test Method	Inspection

Abstract Test 4	**/conf/tangent_plane/h/sdu**
Test Purpose	To confirm that a GeoPose tangentPoint.h attribute is expressed as a real number.
Requirement	/req/tangent_plane/h/sdu
Test Method	Inspection

Abstract Test 5	**/conf/orientation/quaternion/sdu**
Test Purpose	To confirm that GeoPose YPR angles are expressed as three consecutive rotations about the local axes Z, Y, and X, in that order, corresponding to the conventional Yaw, Pitch, and Roll angles and that the unit of measure is the degree..
Requirement	/req/orientation/quaternion/sdu
Test Method	Inspection

A.4.2. Advanced SDU Conformance Class

Abstract Test 6	**/conf/geoposeinstant/sdu**
Test Purpose	To confirm that a Logical Model attribute GeoPoseInstant is Unix Time in seconds multiplied by 1,000 and that the unit of measure is milliseconds.
Requirement	/req/geoposeinstant/sdu
Test Method	Inspection

Abstract Test 7	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 8	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 9	**/conf/frame_specification_parameters/sdu**

Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.
Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

Abstract Test 10	**/conf/orientation/quaternion/sdu**
Test Purpose	To confirm that GeoPose YPR angles are expressed as three consecutive rotations about the local axes Z, Y, and X, in that order, corresponding to the conventional Yaw, Pitch, and Roll angles and that the unit of measure is the degree..
Requirement	/req/orientation/quaternion/sdu
Test Method	Inspection

A.4.3. Graph SDU Conformance Class

Abstract Test 11	**/conf/geoposeinstant/sdu**
Test Purpose	To confirm that a Logical Model attribute GeoPoseInstant is Unix Time in seconds multiplied by 1,000 and that the unit of measure is milliseconds.
Requirement	/req/geoposeinstant/sdu
Test Method	Inspection

Abstract Test 12	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 13	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 14	**/conf/frame_specification_parameters/sdu**
Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.

Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

Abstract Test 15	**/conf/chain/index/sdu**
Test Purpose	To confirm that each index vaue in a FrameListTransformPair is a distinct integer value between 0 and one less than the number of elements in the frameList property.
Requirement	/req/Chain/index/sdu
Test Method	Inspection

A.4.4. Chain SDU Conformance Class

Abstract Test 16	**/conf/geoposeinstant/sdu**
Test Purpose	To confirm that a Logical Model attribute GeoPoseInstant is Unix Time in seconds multiplied by 1,000 and that the unit of measure is milliseconds.
Requirement	/req/geoposeinstant/sdu
Test Method	Inspection

Abstract Test 17	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 18	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 19	**/conf/frame_specification_parameters/sdu**
Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.
Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

A.4.5. Regular Series SDU Conformance Class

Abstract Test 20	**/conf/Series/header/sdu**
Test Purpose	To confirm that a Series Header is implemented in accordance with the Logical Model.
Requirement	/req/Series/header/sdu
Test Method	Inspection

Abstract Test 21	**/conf/geoposeduration/sdu**
Test Purpose	To confirm that a Logical Model attribute GeoPoseduration is expressed in seconds multiplied by 1,000 and that the unit of measure is milliseconds.
Requirement	/req/RegularSeries/duration/sdu
Test Method	Inspection

Abstract Test 22	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 23	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 24	**/conf/frame_specification_parameters/sdu**
Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.
Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

Abstract Test 25	**/conf/Series/trailer/sdu**
Test Purpose	To confirm that a Series Trailer is implemented in accordance with the Logical Model.
Requirement	/req/Series/trailer/sdu
Test Method	Inspection

A.4.6. Irregular Series SDU Conformance Class

Abstract Test 26	**/conf/Series/header/sdu**
Test Purpose	To confirm that a Series Header is implemented in accordance with the Logical Model.
Requirement	/req/Series/header/sdu
Test Method	Inspection

Abstract Test 27	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 28	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 29	**/conf/frame_specification_parameters/sdu**
Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.
Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

Abstract Test 30	**/conf/Series/frame_and_time/sdu**
Test Purpose	To confirm that a Series frame_and_time is implemented as an innerFrameAndTime property with an ExplicitFrameSpec and GeoPoseInstant pair.
Requirement	/req/Series/frame_and_time/sdu
Test Method	Inspection

Abstract Test 31	**/conf/Series/trailer/sdu**
Test Purpose	To confirm that a Series Trailer is implemented in accordance with the Logical Model.
Requirement	/req/Series/trailer/sdu
Test Method	Inspection

A.4.7. Stream SDU Conformance Class

Abstract Test 32	**/conf/transition_model/sdu**
Test Purpose	To confirm that a transition_model attribute is one of the values in the TransitionModel enumeration.
Requirement	/req/transition_model/sdu
Test Method	Inspection

Abstract Test 33	**/conf/frame_specification_authority/sdu**
Test Purpose	To confirm that a FrameSpecification.authority attribute contains a string uniquely specifying a source of reference frame specifications.
Requirement	/req/frame_specification_authority/sdu
Test Method	Inspection

Abstract Test 34	**/conf/frame_specification_id/sdu**
Test Purpose	To confirm that a FrameSpecification.id attribute is a string uniquely defining a frame within the authority.
Requirement	/req/frame_specification_id/sdu
Test Method	Inspection

Abstract Test 35	**/conf/frame_specification_parameters/sdu**
Test Purpose	To confirm that a FrameSpecification.parameters attribute contains contain all parameters needed for the corresponding authority and ID.
Requirement	/req/frame_specification_parameters/sdu
Test Method	Inspection

Abstract Test 36	**/conf/stream/frame_and_time/sdu**
Test Purpose	To confirm that a Stream frame_and_time is implemented as an innerFrameAndTime property with an ExplicitFrameSpec and GeoPoseInstant pair.
Requirement	/req/Stream/fst/sdu
Test Method	Inspection

A.5. Encodings Conformance

Each encoding technology has its own independent test suite. There is one cornformance class per Standardization target per encoding technology. The GeoPose Standard 1.0 has one encoding technology: JSON.

A.5.1. JSON Conformance

The **Basic-YPR GeoPose** is the JSON encoding intended for widest use.

Conformance Class:	**Basic-YPR Encoding-JSON**
ID	/conf/basic_ypr_encoding_json
Target Type	JSON object
Description	To confirm that a Basic-YPR GeoPose consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: B-YPR-Encoding-JSON	/conf/basic_ypr/encoding/json

The Basic-YPR JSON Encoding member test is the following:

Test:	**B-YPR-Encoding-JSON**
ID	/conf/basic_ypr/encoding/json/test
To confirm that a JSON encoding of a Basic-YPR GeoPose conforms with the corresponding JSON-Schema definition.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **Basic-Quaternion GeoPose** JSON encoding is intended for applications using quaternions. It comes in two sub-versions: normal and strict. The only difference is that a strict sub-version does not allow additional JSON members.

Conformance Class:	**Basic-Quaternion Encoding-JSON**
ID	/conf/basic_ypr_encoding_json
Target Type	JSON object
Description	To confirm that a Basic-YPR GeoPose consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: B-YPR-Encoding-JSON	/conf/basic_ypr/encoding/jsontest

The **Basic-Quaternion** JSON Encoding member test is the following:

Test:	**B-YPR-Encoding-JSON**
ID	/conf/basic_ypr/encoding/json/test
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **Basic-Quaternion (Strict) GeoPose** JSON encoding does not allow additional JSON members.

Conformance Class:	**Basic-Quaternion Encoding-JSON (Strict)**
ID	/quaternion_encoding_json_strict
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: B-Quaternion-Encoding-JSON-Strict	/conf/basic_ypr/encoding/json

The Basic-Quaternion JSON (Strict) Encoding member test is the following:

Test:	**B-YPR-Encoding-JSON (Strict)**
ID	/conf/basic_ypr/encoding/json_strict
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-Quaternion (Strict) GeoPose data objects conform to the Basic-Quaternion (Strict) JSON-Schema definition.
Method	JSON-Schema validation.

Test:	B-YPR-Encoding-JSON (Strict)
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **Advanced GeoPose** JSON encoding has an optional time stamp and a flexible Outer Frame specification.

Conformance Class:	Basic-Quaternion Encoding-JSON (Strict)
ID	/quaternion_encoding_json_strict
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: B-Quaternion-Encoding-JSON-Strict	/conf/basic_ypr/encoding/json

The **Advanced GeoPose** JSON (Strict) Encoding member test is the following:

Test:	Advanced-Encoding-JSON
ID	/conf/advanced/encoding/json
To confirm that an implementation of an Advanced GeoPose consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **GeoPose Chain** JSON encoding has an optional time stamp and a flexible Outer Frame specification.

Conformance Class:	Basic-Quaternion Encoding-JSON (Strict)
ID	/quaternion_encoding_json_strict

Conformance Class:	**Basic-Quaternion Encoding-JSON (Strict)**
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: B-Quaternion-Encoding-JSON-Strict	/conf/basic_ypr/encoding/json

The Basic-Quaternion JSON (Strict) Encoding member test is the following:

Test:	**B-YPR-Encoding-JSON**
ID	/conf/basic_ypr/encoding/json
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **GeoPose Regular Series** JSON encoding ....

The **GeoPose Regular Series** JSON encoding ....

Conformance Class:	**GeoPose Regular Series Encoding-JSON (Strict)***
ID	/quaternion_encoding_json_strict
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.

Conformance Class:	GeoPose Regular Series Encoding-JSON (Strict)*
Dependency	/conf/basic_ypr_sdu
Test: B-Quaternion-Encoding-JSON-Strict	/conf/basic_ypr/encoding/json

The **GeoPose Regular Series** JSON Encoding member test is the following:

Test:	GeoPose Regular Series-Encoding-JSON
ID	/conf/basic_ypr/encoding/json
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Basic-YPR-SDU	/req/basic/ypr/sdu
Test Type	Conformance

The **GeoPose Irregular Series** JSON encoding has an optional time stamp and a flexible Outer Frame specification.

Conformance Class:	Irregular GeoPose Series Encoding-JSON*
ID	/quaternion_encoding_json_strict
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: Chain	/conf/basic_ypr/encoding/json

The **GeoPose Irregular Series** JSON Encoding member test is the following:

Test:	Series-Irregular-Encoding-JSON
ID	/conf/basic_ypr/encoding/json

Test:	**Series-Irregular-Encoding-JSON**
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Series-Irregular	/req/basic/ypr/sdu
Test Type	Conformance

The **GeoPose Stream** JSON encoding has an optional time stamp and a flexible Outer Frame specification.

Conformance Class:	**Stream Encoding-JSON (Strict)**
ID	/quaternion_encoding_json_strict
Target Type	JSON object
Description	To confirm that a Basic-Quaternion GeoPose (Strict) consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters defining the tangent point and that the Inner Frame is a rotation-only transformation using Yaw, Pitch, and Roll angles.
Dependency	/conf/basic_ypr_sdu
Test: Stream-Encoding-JSON-Strict	/conf/basic_ypr/encoding/json

The **GeoPose Stream** JSON Encoding member test is the following:

Test:	**Stream-Encoding-JSON**
ID	/conf/basic_ypr/encoding/json
To confirm that an implementation of a Basic-YPR consists of an Outer Frame specified by an implicit WGS-84 CRS and an implicit EPSG 4461-CS (LTP-ENU) coordinate system and explicit parameters to define the tangent point. To confirm that the Inner Frame is expressed as a rotation-only transformation using Yaw, Pitch, and Roll angles.	
Purpose	To confirm that Basic-YPR GeoPose data objects conform to the Basic-YPR JSON-Schema definition.
Method	JSON-Schema validation.
Reference: Stream JSON	/req/basic/ypr/sdu
Test Type	Conformance

Annex B: GeoPose Local Frame of Reference Specifications (Informative)

This annex has example Frame Specifications for the LTP-ENU and LTP-NED Reference Frames.

B.1. Local Tangent Plane - East North Up (LTP-ENU)

LTP-ENU ISO 19162 WKT

```
BASEGEOGCRS["WGS 84",
    DATUM["World Geodetic System 1984",
        ELLIPSOID["WGS 84",6378137,298.257223563,
    PRIMEM["Greenwich",0,
        ANGLEUNIT["degree",0.0174532925199433]],
    ID["EPSG",4979]],
CONVERSION["To LTP-ENU",
    METHOD["Geographic/topocentric conversions",
        ID["EPSG",9837]],
    PARAMETER["Latitude of topocentric origin",<latitude>,
        ANGLEUNIT["degree",0.0174532925199433],
        ID["EPSG",8834]],
    PARAMETER["Longitude of topocentric origin",<longitude>,
        ANGLEUNIT["degree",0.0174532925199433],
        ID["EPSG",8835]],
    PARAMETER["Ellipsoidal height of topocentric origin",<height>,
        LENGTHUNIT["metre",1],
        ID["EPSG",8836]]],
CS[Cartesian,3],
    AXIS["topocentric East (U)",east,
        ORDER[1],
        LENGTHUNIT["metre",1]],
    AXIS["topocentric North (V)",north,
        ORDER[2],
        LENGTHUNIT["metre",1]],
    AXIS["topocentric height (W)",up,
        ORDER[3],
        LENGTHUNIT["metre",1]],
USAGE[
    SCOPE["unknown"],
    AREA["To be specified"],
    BBOX[-90,-180,90,180]],
]
```

B.2. Local Tangent Plane - North East Down (LTP-NED)

LTP-NED ISO 19162 WKT

```
BASEGEOGCRS["WGS 84",
    DATUM["World Geodetic System 1984",
        ELLIPSOID["WGS 84",6378137,298.257223563,
    PRIMEM["Greenwich",0,
        ANGLEUNIT["degree",0.0174532925199433]],
    ID["EPSG",4979]],
CONVERSION["To LTP-NED",
    METHOD["Geographic/topocentric conversions",
        ID["EPSG",9837]],
    PARAMETER["Latitude of topocentric origin",<latitude>,
        ANGLEUNIT["degree",0.0174532925199433],
        ID["EPSG",8834]],
    PARAMETER["Longitude of topocentric origin",<longitude>,
        ANGLEUNIT["degree",0.0174532925199433],
        ID["EPSG",8835]],
    PARAMETER["Ellipsoidal height of topocentric origin",<height>,
        LENGTHUNIT["metre",1],
        ID["EPSG",8836]]],
CS[Cartesian,3],
    AXIS["topocentric North (U)",north,
        ORDER[1],
        LENGTHUNIT["metre",1]],
    AXIS["topocentric East (V)",east,
        ORDER[2],
        LENGTHUNIT["metre",1]],
    AXIS["topocentric depth (W)",down,
        ORDER[3],
        LENGTHUNIT["metre",1]],
USAGE[
    SCOPE["unknown"],
    AREA["To be specified"],
    BBOX[-90,-180,90,180]],
]
```

Annex C: GeoPose Use and Interpretation of UNIX Time

The GeoPose Standard has adopted a variation UNIX time as the method for denoting the location of Instants on a timeline. The reasons for this specific choice include the widespread availability of UNIX time in computer operating systems, the straightforward conversion to UTC at the level of precision required by the use cases considered in GeoPose 1.0: 1 millisecond.

Clearly, applications requiring higher precision and the recognition of non-Newtonian physical processes would require a more complex treatment of time. This has been left to possible future versions of the GeoPose Standard.

C.1. Intended Precision

The intended precision of UNIX time in GeoPose 1.0 is 1 millisecond. Representations and encodings are based on the use of integer numbers of milliseconds.

C.2. Scaling

Time vales are represented and encoded as integer values in GeoPose 1.0.

C.3. Non-negative Time Positions

Times at or after the UNIX epoch of 1 January 1970 are represented as though clocks ticked forward with the same duration of a second as at the epoch. Conversion to time reference systems and calendars requires the consideration of the generally decreasing rate of rotation of the earth with time increasing into the future. UTC, for example, makes use of leap seconds applied as needed either at 31 December or 30 June.

C.4. Negative Time Positions

Times before the UNIX epoch of 1 January 1970 are represented as though clocks ticked backward with the same duration of a second as at the epoch. Conversion to time reference systems and calendars requires the consideration of the generally increasing rate of rotation of the earth with time decreasing into the past. The rate is about 0.015 millisecond/year. The accumulated time error is about 0.6 second/year in the recent past.

C.5. Positive Time Positions before 1 January 1972 UTC

International timekeeping switched from an astronomical basis to a reference based on atomic processes in 1967. The details were in flux at the UNIX time epoch of 1 January 1970 until 1972, when the current system relating atomic time and UTC were adopted.

Annex D: Glossary

The following terms and concepts have appeared in the discussion of the use cases and design of the GeoPose standard but not part of the terms defined in Terms and Definitions. These

- **Acceleration:** The time rate of change of velocity.
- **Accelerometer:** A sensor that can measure Acceleration. Low cost, accurate sensors for measuring 3 mutually perpendicular components of acceleration are widely deployed in vehicles, communications devices, and other connected devices.
- Angular Acceleration:* The time rate of change of rotational velocity.
- **Application Domain:** A context within which some technology or device is usefully applied.
- **Associated Reference Frame (Pose Frame):** A Euclidean reference frame that is defined by the location and orientation of a Pose. A Pose defines the origin of its Associated Reference Frame, and its Orientation defines the orientation of its Associated Reference Frame. Associated Reference Frames are useful in many simulation and graphics applications where Poses are most naturally defined in terms of another (parent) object's pose.
- **Attribute:** A property associated with an object. In object modelling, it is the same as a property or data member.
- **Barometric Pressure:** The ambient pressure of the atmosphere at a location. Low cost, accurate sensors for barometric pressure are widely deployed in connected devices. Sensing of changes in Barometric Pressure over time periods of minutes or less is enables estimation of vertical relative position.
- **Bluetooth Indoor Positioning Services:** Indoor Positioning Services based on Bluetooth signal strength and/or triangulation allow precise determination of location and orientation inside smaller spaces. The location of a Bluetooth transceiver may be specified with respect to a Geographic Coordinate System and it may be possible to compute a GeoPose from interactions with multiple BT transceivers or other sensors.
- **(3D) Cartesian Coordinate System:** A system of geometrical reference using three mutually perpendicular axes where a point location is described by three numbers giving the perpendicular distance to each of the axes, all in the same numerical scale.
- **Class:** A template for the data structure and methods for operating on those data structures for objects belonging to the Class.
- **Compass:** A sensor for measuring the relative orientation of a device to an ambient magnetic field. Accurate and low-cost Compasses are widely deployed in connected devices. Coordinate Reference System A coordinate reference system is a coordinate system referenced to a Datum.
- **Data Type:** A representational form for a concrete data element such as a number, character, or colour.
- **Datum:** A reference point, line or surface used to establish measurements of position. A geodetic datum defines the measurement of horizontal position (latitude and longitude) and/or vertical position (height). datum is a set of parameters that define the position of the origin, the scale, and the orientation of a coordinate system.
- **Ellipsoid:** A mathematical surface that may be used as a datum in defining a Geographic

Coordinate System. An ellipsoid is usually established by fitting the parameters of the ellipsoid to measurements of a gravitational equipotential surface (Geoid) that approximates mean sea level.

- **East-North-Up Local Tangent Plane Coordinate System:** A Euclidean 3-dimensional coordinate system aligned with the Z axis increasing upward, the X axis aligned toward the direction east, and the Y axis aligned toward north. Not defined at the poles because there is no inherent orientation.
- **Euler Angles:** A simple way to describe the orientation of one Euclidean Reference Frame to another by specifying the rotations about each of the three axes respectively to bring one in alignment with the other.
- **Geographic Coordinates:** A 3-dimensional reference system based on a reference ellipsoid. Two of the coordinates are angles with respect to the axis of the ellipsoid and to a plane containing the axis of the ellipsoid and a specified point (principle point) on the ellipsoid surface. The third coordinate is a linear measure of height above the ellipsoidal surface.
- **Geographic Position:** A point defined in Geographic Coordinates.
- **Geoid:** An approximation of surface of equal gravitational force, usually attempting to match average sea-level. A Geoid is defined by measurements and is always inexact. The Ellipsoid used in Geographic Coordinate Systems is usually a mathematical approximation to a specific Geoid.
- **Gyro:** A sensor that measures the rate of rotation. Low-cost, accurate Gyros are widely deployed in connected devices.
- **Kinematics:** The properties of location, velocity, and acceleration of a body without regard to any forces acting on the body.
- **Local Tangent Plane (LTP) Coordinate System:** A right-hand Euclidean Coordinate System with a vertical (Z) axis extending from an origin at a point defined by Geographic Coordinates with respect to an Ellipsoid. Often specialized to an east-north-up (ENU) system, where the X axis is aligned toward east and the Y axis toward north. While a LTP Coordinate System can be established at any location, an ENU cannot be defined at the poles because it cannot be oriented.
- **Position:** The location of a point with respect to the Origin of a specific Reference Frame.
- **Property:** An attribute associated with an object. In object modelling, it is the same as an Attribute or data member.
- **Quaternion:** Quaternions are an extension of complex numbers that have (among many other things) some convenient properties for computing with rotations, in particular smooth interpolation and avoidance of “gimbal lock” possible with Euler Angles.
- **Rotation:** The angular relationship between a reference frame’s axes and a direction in that reference frame. Euler Angles, Rotation Matrices, and Quaternions are three ways to specify a rotation.
- **(Digital) Sensor:** A device that converts environmental properties into data suitable for computation.
- **Topographic Surface:** The interface between the liquid or solid surface of a planet and its atmosphere or surrounding empty space. This surface is always approximate. It may be measure with reference to a gravitational equipotential surface (such as a Geoid) or a mathematical reference surface (such as an Ellipsoid).

- **Velocity:** The time rate of change of Position.
- **Vertical datum:** A reference level from which elevation or altitude can be measured. The Topographic Surface, a Geoid, a level of constant Barometric Pressure, or an Ellipsoid are examples.

Annex E: Revision History

Date	Release	Editor	Primary clauses modified	Description
2020-11-21	0.1.0	Steve Smyth	all	initial integrated version
2021-08-25	0.5.0	Steve Smyth	all	assigned doc number 21-056
2021-08-26	0.5.0	Steve Smyth	all	doc number 21-056r1
2021-09-08	0.6.0	Steve Smyth	all	doc number 21-056r2
2021-09-26	0.6.1	Steve Smyth	fix links and minor edits to all - does not include Unix Time in seconds	doc number 21-056r3
2021-09-28	0.6.2	Steve Smyth	fix section heading 7.3 (YPR)	2022-01-07
0.6.2	Steve Smyth	public review comment resolution	2022-01-27	0.7.0

Annex F: Bibliography

[1] OGC: OGC Testbed 12 Annex B: Architecture. (2015).

[2] PHILIP J. SCHNEIDER, DAVID H. EBERLY, Geometric Tools for Computer Graphics, Morgan Kaufmann, 2003.

[3] OGC Testbed-14: CityGML and AR Engineering Report (https://docs.ogc.org/per/18-025.html)

[4] Interoperable Simulation and Gaming Sprint Engineering Report (http://docs.ogc.org/per/20-087.html)

[5] ISG Year 2 ER (https://github.com/opengeospatial/OGC-ISG-Sprint-Sep-2020/tree/master/Sprint%20Report)

[6] GeoVolumes charter (https://portal.ogc.org/files/97171)