Open Geospatial Consortium

Submission Date: <yyyy-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: 21-005

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

Table of Contents

cd ..\.. **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, …

Chapter 1. Scope

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 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 and chain and graph structures.

The GeoPose standard consists of an implementation-neutral Logical Model which establishes the structure and relationships between GeoPose components and also between GeoPoses data objects themselves in composite structures. Classes and properties of the Logical Model that are expressed in concrete data objects defined by this standard are identified as implementation-neutral Structural Data Units (SDUs), aliases for elements of the Logical Model. SDUs are grouped to define the implementation-neutral form of the GeoPose Standardization Targets. 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 has a single encoding, JavaScript Object Notation (JSON). Each Standardization Target has a JSON-Schema:2019-9 specification. Most of the standardization requirements are that concrete JSON GeoPose data objects must conform to the corresponding JSON-Schema definition. The individual elements identified in the encoding specification are SDUs, tying the specifications back to the Logical Model.

The GeoPose 1.0 standard excludes 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.

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 and Procedures and the OGC Compliance Testing web site. 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® interface standard, a software implementation shall choose to implement any one 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 Standarization Targets. These targets are independent and a conforming implementation may implement one or more of them.

2.2. Conformance Classes

This standard identifies eight conformance classes. One conformance class is defined for each correspoding set of Structural Data Units, each linked to the Logical Model. Additionally, each of the eight standarization targets is represented by a conformance class, defined by a 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.

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.

ISO / TC 211: ISO 19115-1:2014 Geographic information — Metadata — Part 1: Fundamentals (2014)

ISO / TC 211: ISO 19157:2013 Geographic information — Data quality (2013)

ISO / TC 211: ISO 19139:2007 Geographic information — Metadata — XML schema implementation (2007)

ISO / TC 154: ISO 8601-1:2019 Date and time — Representations for information interchange — Part 1: Basic rules (2019)

ISO / TC 154: ISO 8601-2:2019 Date and time — Representations for information interchange — Part 1: Extensions (2019)

OGC: OGC 15-097 OGC Geospatial User Feedback Standard. Conceptual Model (2016)

OGC: OGC 12-019, OGC City Geography Markup Language (CityGML) Encoding Standard (2012)

OGC: OGC 14-005r3, OGC IndoorGML (2014)

OWL Time: https://www.w3.org/TR/2020/CR-owl-time-20200326/

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.

[NOTE] Terms moved to annex-glossary.

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

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

NOTE

Although the issue does not seem to be totally resolved in the OGC OAB, our usage follows the spirit of OAB issue 1310 where the OAB resolved to include the following definitions in the TC Policies and Procedures (meeting of July 9, 2019):

“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 **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.

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). [The corresponding names in the NASA SPICE system are **(from frame** (domain) and **(to 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**. In the NASA SPICE system, the **inner frame** is referred to as the **to frame**. 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 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 tranformations 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 location 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 Logical Model 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 OGC 16-071r3. 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 a **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**.

4.3.1. 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 sections 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. Design

5.2.1. Use Cases, Concepts, Logical Model, Implementation Targets, Encodings

5.3. UML Notation

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)..

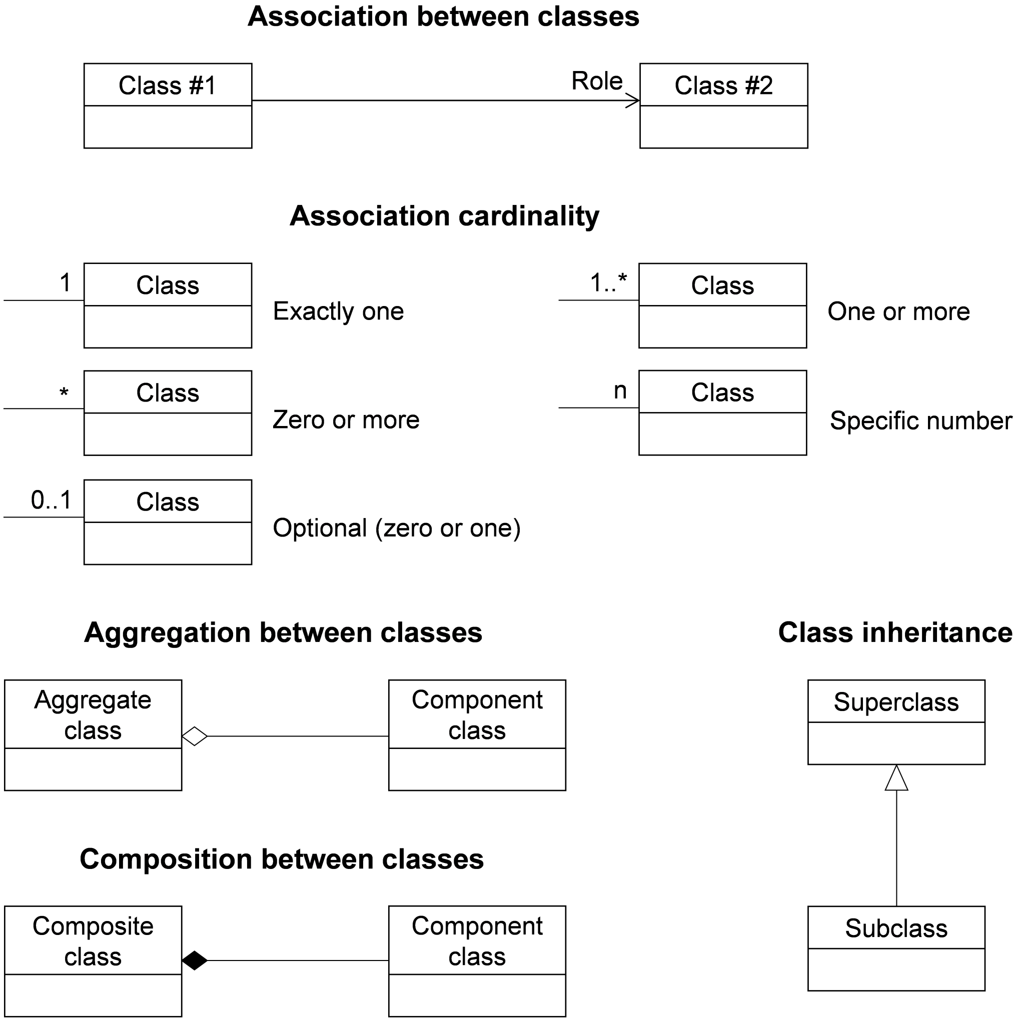
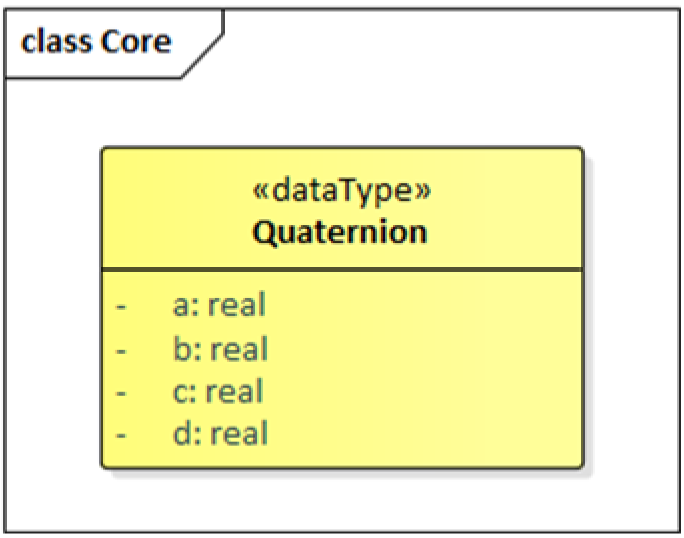


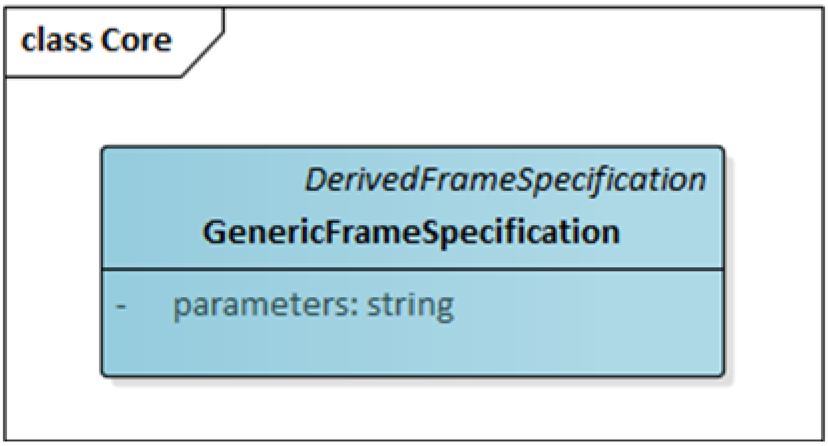
Figure 1. UML notation (see ISO TS 19103, Geographic information - Conceptual schema language).

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.

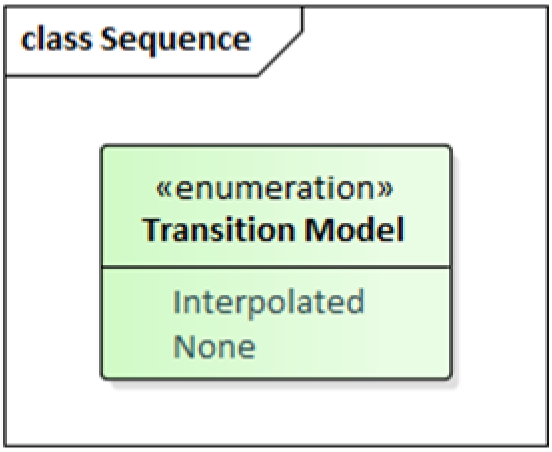
In order to enhance the readability of the GeoPose UML diagrams, classes are depicted in different colors. The following coloring scheme is applied:



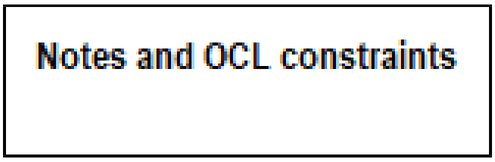
Classes painted in yellow represent data types.



Classes painted in blue are represented as Structural Data Units or have properties that are represented in Structural Data Units and in encodings of those SDUs.



Enumerations are painted in green.



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

The example UML diagram in [figure-2] demonstrates the UML notation and coloring scheme used throughout this specification.

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.

Chapter 6. Informative Material

6.1. Document Structure

The structure of this document flows from the GeoPose use cases to the JSON-Schema specification of eight GeoPose data object types. A conceptual domain of discourse based on section XX, 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 are independent of serialization or encoding format. Version 1.0 of this standard specifies one realization of the SDUs as JavaScript Object Notation (JSON), using the JSON-Schema 2019-9 specification.

A key idea is that specific use cases are tied to the standardization targets, i.e. the part of the document that prescribes the structure and content of GeoPose data objects. These use cases are linked to these data objects as well as implementation examples that appear in other documents.

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

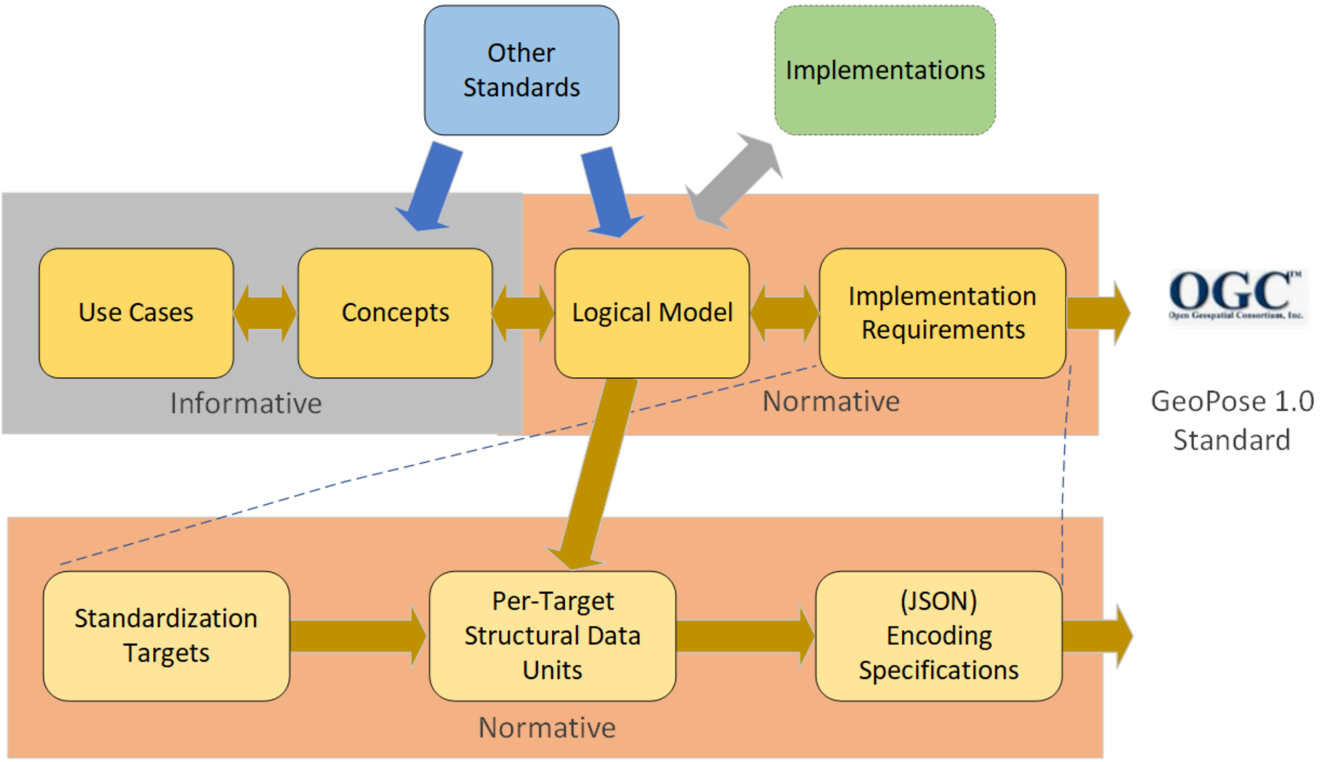


Figure 6.1: Document Structure Overview

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 GeoPoses is to convey location 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 all together in a geographic reference frame.

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-E, Basic-Q, 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-E, Basic-Q, 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-E, Basic-Q
/geopose/1.0/use_case/ar/05	Input to ray-casting and line-of-sight calculations	Basic-E, Basic-Q, Chain
/geopose/1.0/use_case/ar/06	Input to proximity calculations	Basic-E, Basic-Q,
/geopose/1.0/use_case/ar/07	In time series, input and output to and from trajectory projection calculations	Regular Timeseries, Irregular Timeseries, Stream

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 locating and orienting a moving object. Sensors attached to mobile elements have their own poses and a chain of reference frame transformtions 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 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-E, Basic-Q
/geopose/1.0/use_case/av/02	Calculate parameters such as distances and routes (reference to OGC Moving Features?)	Basic-E, Basic-Q, 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 objests 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 precise position and orientation of objects both those visible to and hidden from the user's view (e.g. underground, embedded in walls or ceilings)	Basic-E, Basic-Q
/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-E, Basic-Q

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 environement that contains a fixed background of air, land, water, vegetation, built objects, and other non-moving elements. The assembly is animated in time to display a visualization or analytical results of the evolving state of the modelled environment. Synthetic environments support training, rehearsal, and archive 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	Instantaeous and time series locations and orientations of mobile objects	Basic-E, Basic-Q, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/02	Instantaeous and time series location and orientation of an optical imaging device simultaneous location and mapping (SLAM)	Basic-E, Basic-Q, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/03	Instantaeous and time series location and orientation of an optical imaging device (Visual Odometry)	Basic-E, Basic-Q, Advanced, Regular Timeseries, Irregular Timeseries, Stream
/geopose/1.0/use_case/im/04	Instantaeous and time series location and orientation of an optical imaging device used for photogrammetry	Regular Timeseries, Irregular Timeseries, Stream

Chapter 7. Standardization Targets (Normative)

The core abstraction in the GeoPose Standard is the Frame Transform. This 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 location 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.

There are eight independent Standardization Targets to specifically address the requirements of individual use cases. The Basic and Advanced 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. The eight targets are denoted by bold terms in the following categories:

1. Basic - Satisfy most use cases
    a. Orientation by Euler angles: **Basic-Euler** Target
    b. Orientation by 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

7.1. Methodology

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 aliases for elements (classes or attributes) in the Logical Model. SDUs may have additional Requirements limiting the range, multiplicity, representation or other constraining and testable characteristics. SDUs are combined to express each of the Standarization Targets using Extended Backus-Naur Form (EBNF) notation as described in EBNF with the following minor differences. The eight independent SDU combinations that define each corresponding Target are expressed as encoding-neutral Requirements. Finally, for each encoding format, there are requirements for the expression of the grouped SDUs.

Poses are structurally two frames, each identified by a corresponding frame specification. The edge connecting the two frame specifications represents the pose’s frame transform and is marked with a thick blue vertical line.

EBNF: ISO/IEC 14977:1996, Information technology — Syntactic metalanguage — Extended BNF https://www.w3.org/TR/2010/REC-xquery-20101214/#XML

The EBNF notation is described in more detail in: https://www.w3.org/TR/2010/REC-xquery-20101214/#EBNFNotation

All named non-terminal symbols have a name that begins with an uppercase letter.

All terminal symbols have the name of the corresponding element of the UML GeoPose Logical Model.

Comments or extra-grammatical constraints on grammar productions are between '/' **and** '/' symbols.

NOTE Only EBNF-derived diagrams (and not even all of them yet) are in the SDU sections now. Adding the actual EBNF allows explicit and machine-readable specification. That’s why explicit EBNF is still needed.

7.2. Logical Model (Normative)

The Logical Model, expressed in UML static class diagrams is normative for defining the various data elements that are either GeoPose instances or groupings of GeoPose instances in composite structures. The detailed expression of elements from the UML model are specified in top level "structural" requirements as Structural Data Units (SDUs), which are terminal symbols in an Extended Backus-Naur Form production system. The SDUs are then expressed in one or more serialization or encoding formats, and there is one requirement per mapping from SDU to a specific encoding, such as JSON.

The normative expression of the UML model is a Sparx Systems Enterprise Explorer project ("eap") file, made available **link goes here** by OGC.

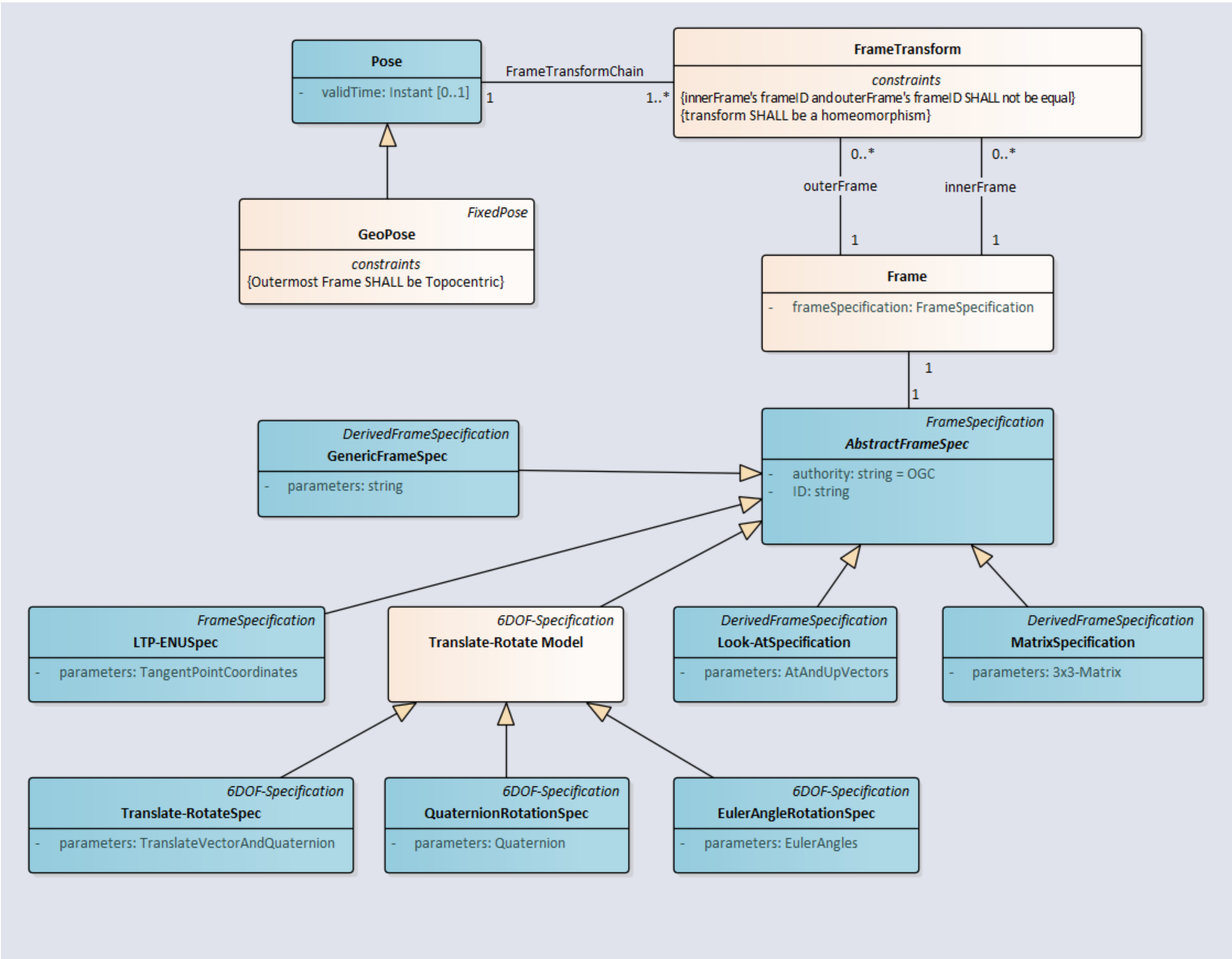
The Logical Model consists of three packages: Core, Time, and Sequence. The Basic GeoPose targets rely on only the Core package. The Advanced GeoPose also incorporates the Time Package. Composite GeoPoses incorporate all three Packages.

The coloring of the classes indicates whther they are normative in terms of concrete structural data units (SDUs) or the corresponding encodings (blue), provide context for interpreting the SDUs (light yellow), or represent data types (bright yellow).

NOTE	Implementers of software using encoded SDUs must agree with the logical description of the blue model elements. The other elements are not expressed in the concrete data objects in the GeoPose 1.0 standard. Therefore they are not normative in the sense of requiring implementation of a specific internal representation. Stated another way, an implementation is free to use any internal representation needed for efficient implementation but the interpretation of the data objects themselves must conform to the standard, including the Logical Model.

7.2.1. Core

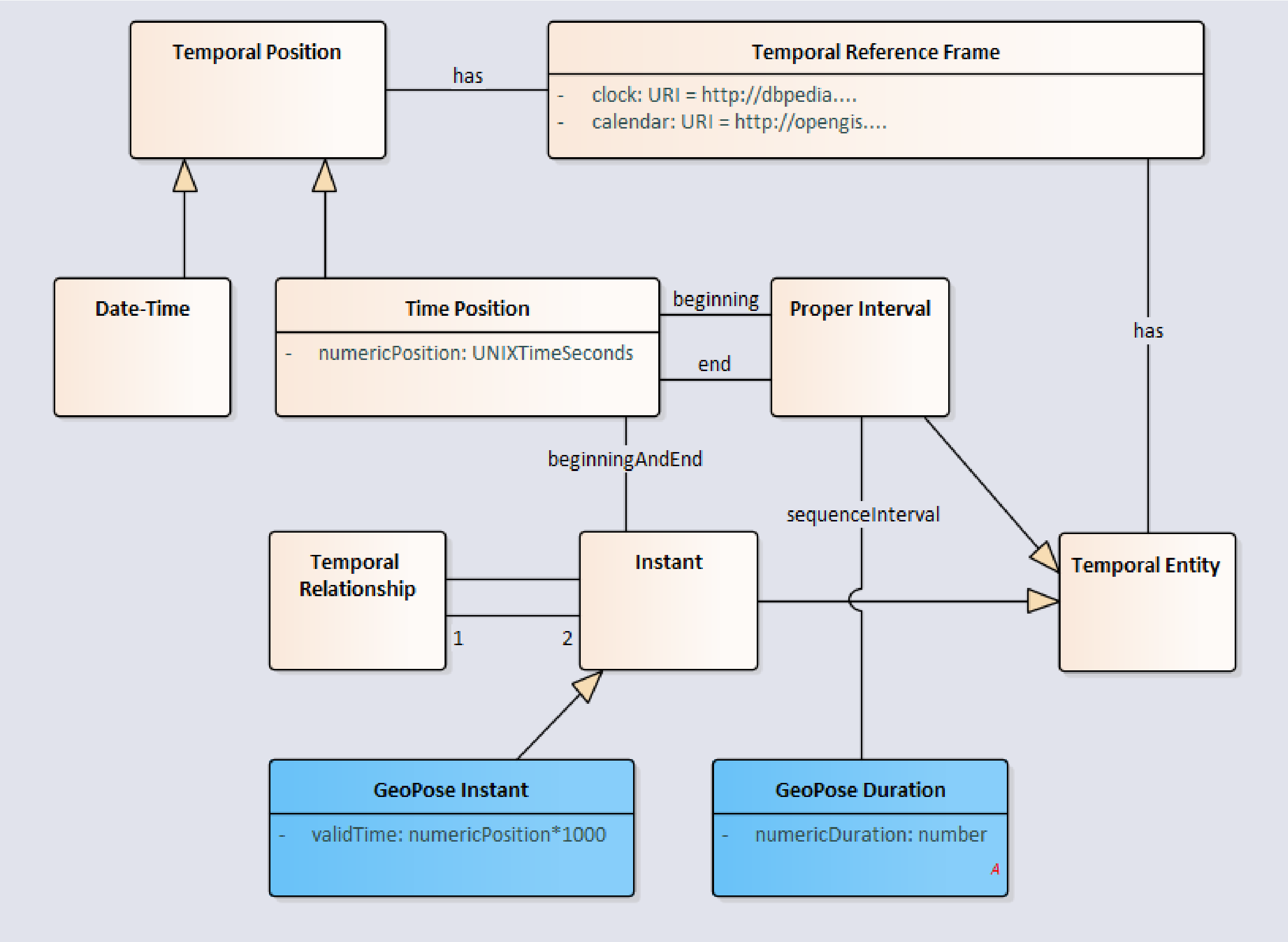
The Logical Model Core contains the essential elements specific to the GeoPose conceived as a transformation between an anchoring Outer Frame and one or more derived Inner Frames.



7.2.2. Time

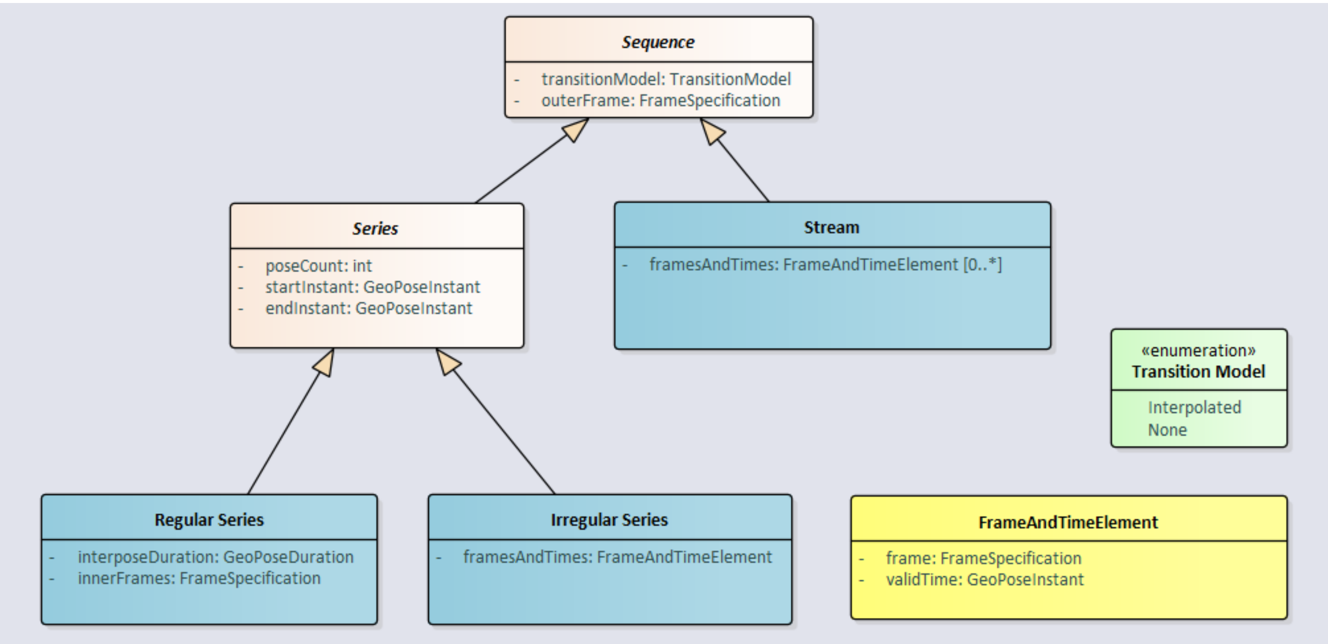
The time logical model is based on the OWL Time document.

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



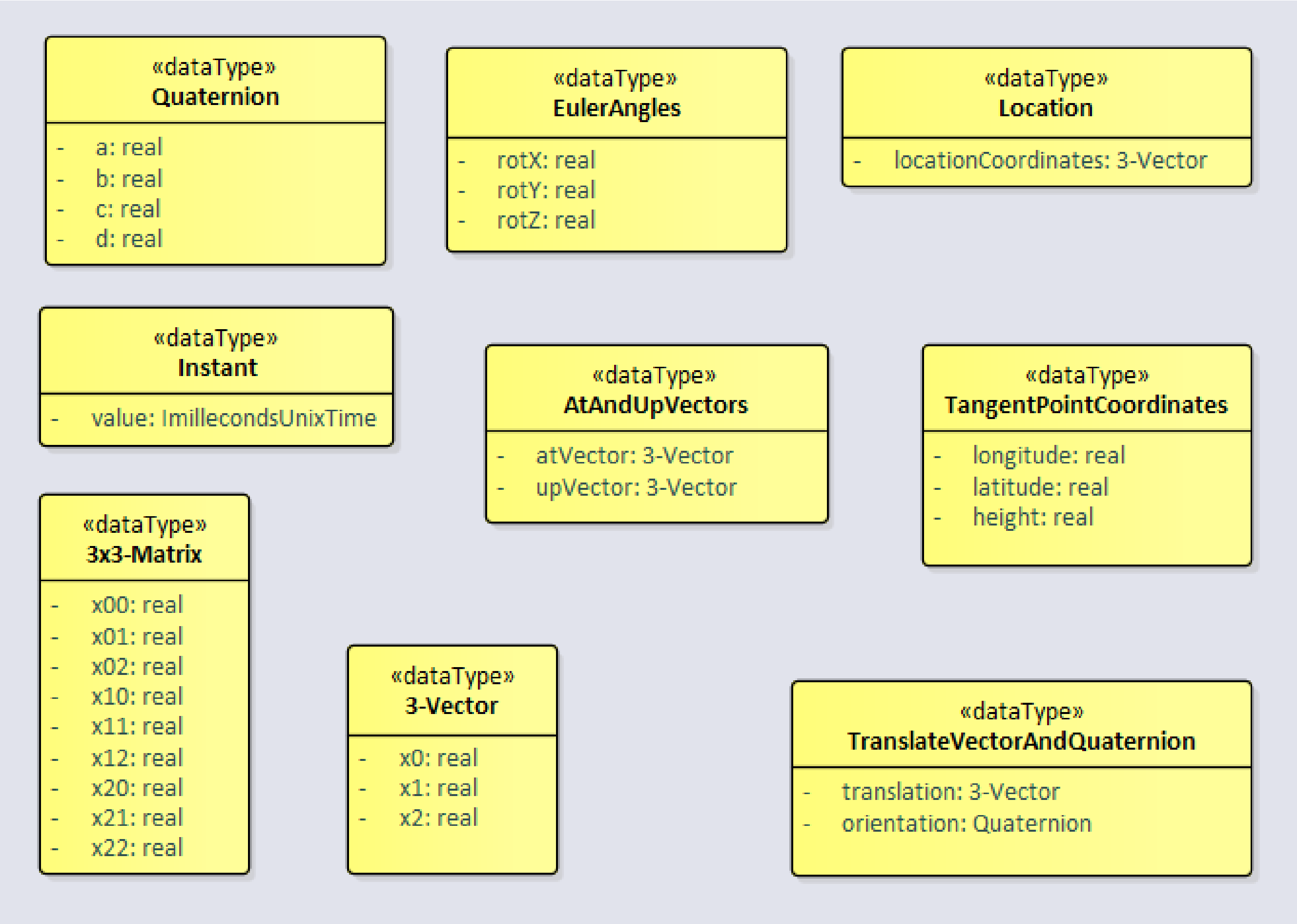
7.2.3. Sequence

The sequence logical model is almost entirely packaging of GeoPose data, where multiple GeoPoses in a sequence share the same Outer frame and there is a time-dependent changing inner frame.



7.2.4. Data Types

These data types appear in classes that have attributes expressed in standardization targets.



7.3. Structural Data Units (Normative)

NOTE	The purpose of SDUs is to provide a standardization target that is 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 exactly one element (class or attribute) in the logical model.
NOTE	EBNF notation from https://www.w3.org/TR/REC-xml/#sec-notation
NOTE	This is where the mapping of elements of the Logical Data Model to SDUs is specified. There is one requirement per assemblage of SDUs into a Standardization Target: "the implementation independent Standardization Target shall be defined by its EBNF specification".

NOTE	Requirements are modular by Standardization Target. This results in some SDU requirements being repeated between Targets. One alternative could be to make all of the requirements on SDUs separately, and then have some language that requires conformance and implementation only for the Targets implemented. Within the document, the SDU requirements are in separate files, included when appropriate in the sections for each of the Targets. This insures that the texts will be identical, even if it is not transparently obvious in the standard. For each Target, there is a requirement for how each data element that is mapped from the logical model to SDUs and then another for each group of SDUs are encoded in JSON. 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 v 1.0, with two different levels of JSON encoding strictness individually specified.

7.3.1. Standardization Target 1: Basic-Quaternion

The Basic-Quaternion GeoPose is a rotation-only transformation from a WGS-84-referenced local tangent plane east-north-up coordinate system. The rotation is specified with four components of a normalized quaternion.

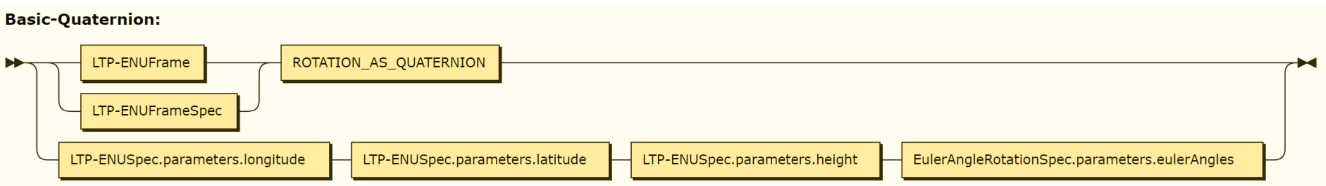
Requirements

Summary (Informative)

The Basic-Quaternion Target 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. Note that this is equivalent to a Generic Frame Specification, with the authority and ID as described above. The Inner Frame is a rotation-only transformation using a quaternion specification.

Structure (Normative)

Requirement BQ-S-1	**/req/basic/quaternion-permissive/structure**
Requirement	An implementation of a Basic-Quaternion 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.
ID	/req/basic/quaternion/structure




```
Basic-Quaternion
         ::= ( LTP-ENUFrame | LTP-ENUFrameSpec ) ROTATION_AS_QUATERNION
           | LTP-ENUSpec.parameters.longitude LTP-ENUSpec.parameters.latitude LTP-ENUSpec.parameters.height EulerAngleRotationSpec.parameters.eulerAngles
```

Structural Dependencies (Normative)

Requirement SDU-1	/req/sdu/tangent_plane/longitude
Requirement	An instance of a GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/longitude

Requirement SDU-2	/req/sdu/tangent_plane/latiitude
Requirement	An instance of GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/latitude

Requirement SDU-3	/req/sdu/tangent_plane/height
Requirement	An instance of a GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/height

Requirement SDU-4	/req/sdu/orientation/quaternion
Requirement	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.
ID	/req/sdu/orientation/quaternon

7.3.2. Standardization Target 2: Basic-Euler

The Basic-Euler GeoPose uses a rotation-only transformation from a WGS-84-referenced local tangent plane east-north-up coordinate system. The rotation is specified with three Euler angles denoting rotation angles about the three LTP-ENU axes. The local (rotated) coordinate system axis order is z, y, x. This is commonly referred to as the yaw, pitch, roll.

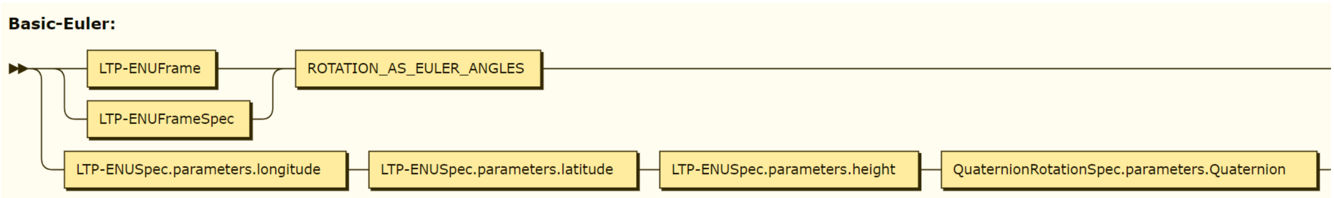
NOTE	Fix and document requirement on axes. Euler angles seem simple but there is wide variation on the axis order and whether the axes rotate in sequence or are fixed for the three rotations.

Summary (Informative)

The Basic-Euler 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. Note that this is equivalent to a Generic Frame Specification, with the authority and ID as described above. The Inner Frame shall be a rotation-only transformation using an Euler angles specification.

Structure (Normative)

Requirement BE-S-1	/req/basic/euler/structure
Requirement	An implementation of a Basic-Euler 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 an Euler angles specification.
ID	/req/basic/euler/structure




```
Basic-Euler
         ::= ( LTP-ENUFrame | LTP-ENUFrameSpec ) ROTATION_AS_EULER_ANGLES
           | LTP-ENUSpec.parameters.longitude LTP-ENUSpec.parameters.latitude LTP-ENUSpec.parameters.height QuaternionRotationSpec.parameters.Quaternion
```

Structural Dependencies (Normative)

Requirement SDU-5	/req/sdu/tangent_plane/longitude
Requirement	An instance of a GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/longitude

Requirement SDU-6	/req/sdu/tangent_plane/latiitude
Requirement	An instance of GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/latitude

Requirement SDU-7	/req/sdu/tangent_plane/height
Requirement	An instance of a GeoPose tangentPoint.longitude attribute shall be expressed as a real number.
ID	/req/sdu/tangent_plane/height

Requirement SDU-8	/req/sdu/orientation/euler_angles
Requirement	Euler angles shall be 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. The unit of measure shall be the degree.
ID	/req/sdu/orientation/euler_angles

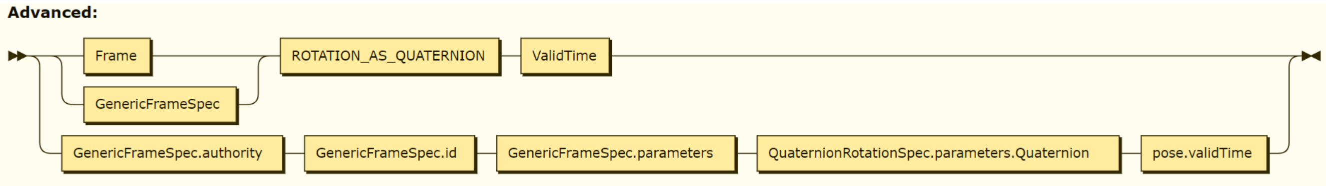
7.3.3. Standardization Target 3: Advanced

Summary (Informative)

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

Structure (Normative)

Requirement AD-S-1	/req/advanced/structure
Requirement	An implementation of an Advanced Target shall consist of a
ID	/req/advanced/structure



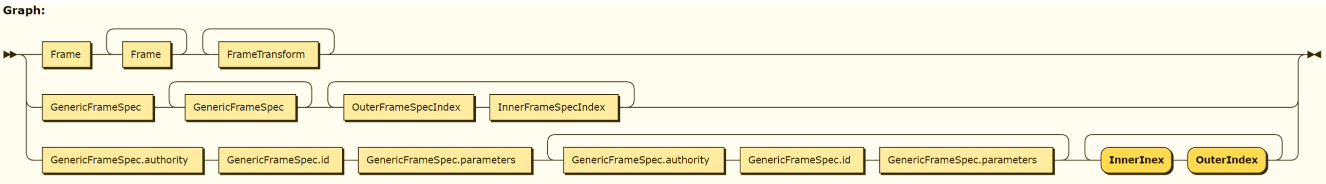

```
Advanced ::= ( Frame | GenericFrameSpec ) ROTATION_AS_QUATERNION ValidTime
           | GenericFrameSpec.authority GenericFrameSpec.id GenericFrameSpec.parameters QuaternionRotationSpec.parameters.Quaternion pose.validTime
```

7.3.4. 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.

Requirement Graph-S-1	**/req/Graph/structure**
Requirement	An implementation of a Graph GeoPose shall consist of a
ID	/req/Graph/structure




```
Graph    ::= Frame Frame+ FrameTransform+
           | GenericFrameSpec GenericFrameSpec+ ( OuterFrameSpecIndex InnerFrameSpecIndex )+
           | GenericFrameSpec.authority GenericFrameSpec.id GenericFrameSpec.parameters ( GenericFrameSpec.authority GenericFrameSpec.id GenericFrameSpec.parameters )+ ( 'InnerInex' 'OuterIndex' )+
```

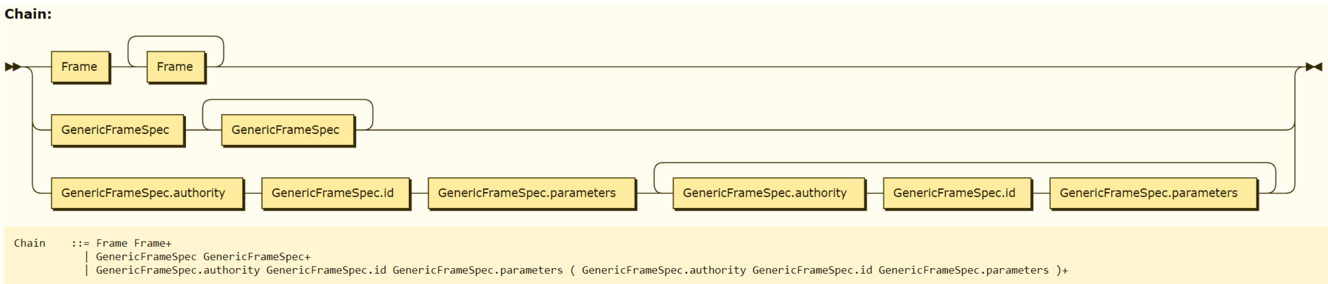
7.3.5. Standardization Target 5: Chain

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 (Normative)

Requirement Chain-S-2	**/req/Chain/structure**
Requirement	An implementation of a Chain GeoPose shall consist of a
ID	/req/Chain/structure



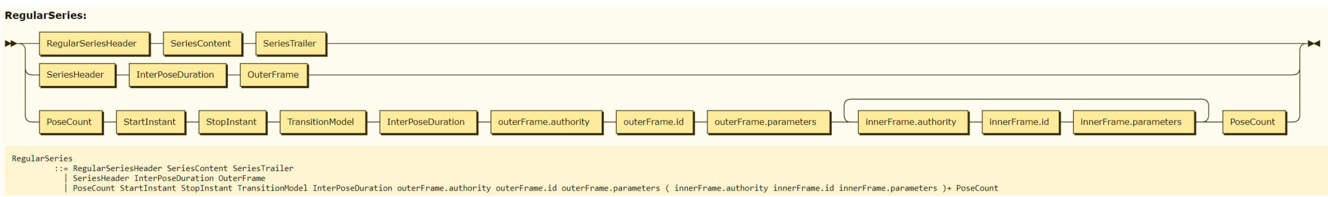
7.3.6. Standardization Target 6: Regular Series

Summary (Informative)

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

Structure (Normative)

Requirement Regular_Series-S-1	**/req/Regular_Series/structure**
Requirement	An implementation of a Regular Series GeoPose shall consist of a
ID	/req/Regular_Series/structure



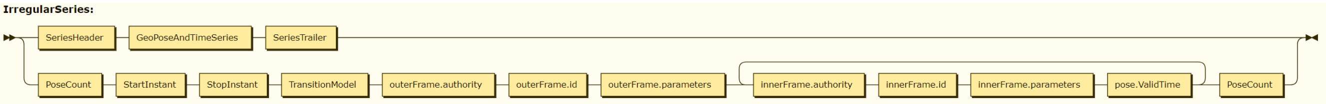
7.3.7. Standardization Target 7: Irregular Series

Summary (Informative)

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

Structure (Normative)

Requirement Irregular_Series-S-1	**/req/Irregular_Series/structure**
Requirement	An implementation of an Irregular Series GeoPose shall consist of a
ID	/req/Irregular_Series/structure




```
IrregularSeries
         ::= SeriesHeader GeoPoseAndTimeSeries SeriesTrailer
           | PoseCount StartInstant StopInstant TransitionModel outerFrame.authority outerFrame.id outerFrame.parameters ( innerFrame.authority innerFrame.id innerFrame.parameters pose.ValidTime )+ PoseCount
```

7.3.8. Standardization Target 8: Stream

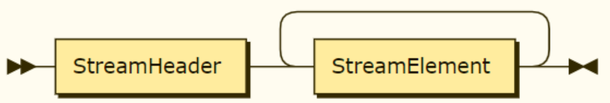
Summary (Informative)

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, it is possible to record the Header and all of the Elements received up to some point in time in a single structure. 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)

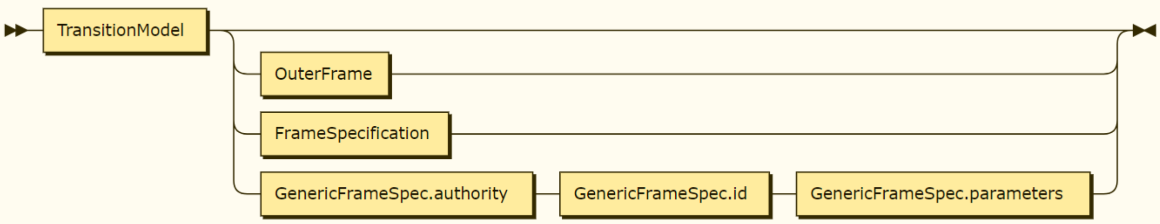
Requirement Stream-S-1	**/req/Stream/structure**
Requirement	An implementation of a Stream GeoPose shall consist of a
ID	/req/Stream/structure

Stream:



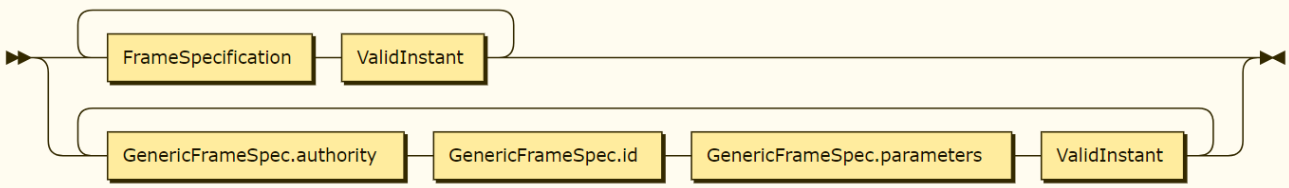

```
Stream   ::= StreamHeader StreamElement+
```

StreamHeader:




```
StreamHeader
         ::= TransitionModel ( OuterFrame | FrameSpecification | GenericFrameSpec.authority GenericFrameSpec.id GenericFrameSpec.parameters )?
```

StreamElement:




```
StreamElement
         ::= ( FrameSpecification ValidInstant )+
           | ( GenericFrameSpec.authority GenericFrameSpec.id GenericFrameSpec.parameters ValidInstant )+
```

7.4. JSON Encodings (Normative)

NOTE	Requirements are modular by Standardization Target. This results in some SDU requirements being repeated between Targets. One alternative could be to make all of the requirements on SDUs separately, and then have some language that requires conformance and implementation only for the Targets implemented. Within the document, the SDU requirements are in separate files, included when appropriate in the sections for each of the Targets. This insures that the texts will be identical, even if it is not transparently obvious in the standard. For each Target, there is a requirement for how each data element that is mapped from the logical model to SDUs and then another for each group of SDUs are encoded in JSON. 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 v 1.0, with two different levels of JSON encoding strictness individually specified.

7.4.1. JSON Encodings for Standardization Target 1: Basic-Quaternion

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.

Strict JSON Encoding (Normative)

Requirement BQ-S-2	**/req/basic/quaternion-strict/JSON_Encoding**
Requirement	JSON encoded Basic-quaternion-strict GeoPose data elements as a whole shall conform to the GeoPose Basic-Quaternion-Strict JSON-Schema 2019-9 definition. There shall be no encoded properties not explicitly defined in the JSON-Schema definition.
ID	/req/basic/quaternion-strict/JSON_Encoding

```
{
  "description": "Basic-Quaternion-Strict: Basic GeoPose using quaternion to specify
orientation - no additional properties",
  "type": "object",
  "additionalProperties": false,
  "properties": {
    "longitude": {
      "type": "number"
    },
    "latitude": {
      "type": "number"
    },
    "height": {
      "type": "number"
    },
    "quaternion": {
      "type": "array",
      "items": {
        "type": "number"
      },
      "minItems": 4,
      "maxItems": 4
    }
  },
  "required": [
    "longitude",
    "latitude",
    "height",
    "quaternion"
  ]
}
```

Instance-Strict (Informative)

```
{
  "longitude": -122.3,
  "latitude": 47.7,
  "height": 11.5,
  "quaternion": {
    "x": 0.24935508863486583,
    "y": -0.13561967913187356,
    "z": 0.4581349646468323,
    "w": -0.8423429803661402
  }
}
```

Permissive JSON Encoding (Normative)

NOTE	This JSON encoding 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 BQ-3	**/req/basic/quaternion/JSON_Encoding**
Requirement	Basic-quaternion GeoPose data elements as a whole shall conform to the GeoPose Basic-Quaternion JSON-Schema 2019-9 definition.
ID	/req/basic/quaternion/JSON_Encoding

```
{
  "description": "Basic-Quaternion: Basic GeoPose using quaternion to specify
orientation",
  "type": "object",
  "properties": {
    "longitude": {
      "type": "number"
    },
    "latitude": {
      "type": "number"
    },
    "height": {
      "type": "number"
    },
    "quaternion": {
      "type": "array",
      "items": {
        "type": "number"
      },
      "minItems": 4,
      "maxItems": 4
    }
  },
  "required": [
    "longitude",
    "latitude",
    "height",
    "quaternion"
  ]
}
```

Instance (Informative)

```json
{
  "longitude": -122.3,
  "latitude": 47.7,
  "height": 11.5,
  "quaternion": {
    "x": 0.24935508863486583,
    "y": -0.13561967913187356,
    "z": 0.4581349646468323,
    "w": -0.8423429803661402
  }
}
```

7.4.2. JSON Encoding for Standardization Target 2: Basic-Euler

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

JSON Encoding (Normative)

Requirement BE-2	**/req/basic/EulerAngle/JSON_Encoding**
Requirement	A JSON-encoded Basic-Euler GeoPose shall conform to the Basic-Euler JSON-Schema 2019-9 definition.
ID	/req/basic/EulerAngle/JSON_Encoding

```json
{
  "description": "Basic-Euler: Basic GeoPose using Euler angle rotations to specify orientation",
  "type": "object",
  "properties": {
    "longitude": {
      "type": "number",
      "minimum": -180.0,
      "maximum": 180.0
    },
    "latitude": {
      "type": "number",
      "minimum": -90.0,
      "maximum": 90.0
    },
    "height": {
      "type": "number"
    },
    "rotations": {
      "type": "array",
      "items": {
        "type": "number"
      },
      "minItems": 3,
      "maxItems": 3
    }
  },
  "required": [
    "longitude",
    "latitude",
    "height",
    "rotations"
  ]
}
```

Instance (Informative)

```json
{
  "longitude": -122.3,
  "latitude": 47.7,
  "height": 11.5,
  "eulerAngles": {
    "rotX": -0.5756131639306046,
    "rotY": 0.0,
    "rotZ": 5.2869177931177145
  }
}
```

7.4.3. JSON Encoding for Standardization Target 3: Advanced GeoPose

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:

```
{
  "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"
      ]
    }
  },
  "type": "object",
  "properties": {
    "frameSpecification": {
      "$ref": "#/definitions/FrameSpecification"
    },
    "quaternion": {
      "type": "array",
      "items": {
        "type": "number"
      },
      "minItems": 4,
      "maxItems": 4
    },
    "validTime": {
      "type": "integer"
    }
  },
  "required": [
    "frameSpecification",
    "quaternion"
  ]
}
```

Instance

```
{
  "frameSpecification": {
    "authority": "/geopose/1.0",
    "id": "LTP-ENU",
    "parameters": "longitude=-122.3000000&latitude=47.7000000&height=11.000"
  },
  "quaternion": {
    "x": 0.24936888089896878,
    "y": -0.13562001157583606,
    "z": 0.4581145403168927,
    "w": -0.842349952009026
  },
  "validTime": 1614921636567
}
```

7.4.4. JSON Encoding for Standardization Target 4: Graph

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:

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

Instance

```
{
  "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
      ]
    }
  ]
}
```

7.4.5. JSON Encoding for Standardization Target 5: Chain

> **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:

```
{
  "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

```
{
  "validTime": 1614921636596,
  "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]"
    }
  ]
}
```

7.4.6. JSON Encoding for Standardization Target 6: Regular Series

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:

```
{
  "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": {
          "type": "string"
        }
      },
      "required": [
        "poseCount",
        "startInstant",
        "stopInstant",
        "transitionModel"
      ]
    },
    "SeriesTrailer": {
      "type": "object",
      "properties": {
        "poseCount": {
          "type": "integer"
        },
        "integrityCheck": {
          "type": [
            "string",
            "null"
          ]
        }
      },
      "required": [
        "poseCount"
      ]
    }
  },
  "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

```
{
  "header": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}",
    "startInstant": 1614921636680,
    "stopInstant": 1614921681680,
    "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\"}"
  }
}
```

7.4.7. JSON Encoding for Standardization Target 7: Irregular Series

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:

```
{
  "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": {
          "type": "string"
        }
      },
      "required": [
        "poseCount",
        "startInstant",
        "stopInstant",
        "transitionModel"
      ]
    },
    "SeriesTrailer": {
      "type": "object",
      "properties": {
        "poseCount": {
          "type": "integer"
        },
        "integrityCheck": {
          "type": [
            "string",
            "null"
          ]
        }
      },
      "required": [
        "poseCount"
      ]
    }
  },
  "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

```
{
  "header": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}",
    "startInstant": 1614921636723,
    "stopInstant": 1614921681723,
    "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": 1614921636724
    },
    {
      "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": 1614921636724
    },
    {
      "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": 1614921636724
    }
  ],
  "trailer": {
    "poseCount": 2,
    "integrityCheck": "{\"SHA256\":
\"5556fb65f8bf9eddb3ace1329c9a6aeedd4833409965aeee3e6b61ed21f47858\"}"
  }
}
```

7.4.8. JSON Encoding for Standardization Target 8: Stream

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:

The Stream Header JSON encoding:

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

The Stream Element JSON ecndoing:

```
{
  "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"
  ]
}
```

JSON encoding of a recorded Stream:

```
{
  "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": {
          "type": "string"
        },
        "outerFrame": {
          "$ref": "#/definitions/FrameSpecification"
        }
      },
      "required": [
        "transitionModel",
        "outerFrame"
      ]
    }
  },
  "type": "object",
  "properties": {
    "header": {
      "$ref": "#/definitions/StreamHeader"
    },
    "streamElements": {
      "type": [
        "array",
        "null"
      ],
      "items": {
        "$ref": "#/definitions/StreamElement"
      },
      "minItems": 1
    }
  },
  "required": [
    "header",
    "streamElements"
  ]
}
```

Instance

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=[-0.84235, 0.24937,
-0.13562, 0.45811]"
    },
    "validTime": 1614921636772
  }
}
```

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=[-0.84235, 0.24937,
-0.13562, 0.45811]"
        },
        "validTime": 1614921636772
      }
    },
    {
      "streamElement": {
        "frame": {
          "authority": "/geopose/1.0",
          "id": "RotateTranslate",
          "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.84302, 0.25070,
-0.13565, 0.45614]"
        },
        "validTime": 1614921649117
      }
    },
    {
      "streamElement": {
        "frame": {
          "authority": "/geopose/1.0",
          "id": "RotateTranslate",
          "parameters": "translation=[0.0, 0.0, 0.0]&rotation=[-0.84369, 0.25202,
-0.13567, 0.45417]"
        },
        "validTime": 1614921661462
      }
    }
  ]
}
```

7.5. Required Implementations (Normative)

Table 3. Table Required Implementations (Normative)

Frame Specification Type	Authority	Implementation Requirements
LTP-ENU (WGS84 - EPSG 4979-CRS/EPSG 4461-CS)	https://epsg.org	Shall be accepted
LTP-NED (WGS84 - EPSG 4979-CRS/EPSG 4461-CS with first two coordinates interchanged and the third coordinate multiplied by -1.0: Procedurally EPSG Dataset coordinate operation method code 9837 followed by change of coordinates U' = V; V' = U; W' = -W for the direct transformation and preceded by the same change of coordinates U' = V; V' = U; W' = -W before the inverse transformation)	https://epsg.org	Shall be accepted
NASA SPICE	https://naif.jpl.nasa.gov/naif	Optional
Proj	https://proj.org	Optional
ISO 19162	https://www.iso.org/obp/ui/#iso:std:iso:19162:ed-2:v1:en	Optional

Chapter 8. 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). Each SDU is an alias for an element of the Logical Model that will be expressed in concrete data objects encoded using specific serialization technologies. GeoPose 1.0 has a single encoding technology: JavaScript Object Notation (JSON) and the encodings are specified using JSON-Schema. To keep the individual Standardization targets independent, there are some SDU requirements and corresponding conformance tests that appear in more than one conformance class. This structure is based on the judgement that it is easier to understand the independence of the Targets than would be the case if the definitional requirements of the SDUs were factored out and referenced indirectly by individual encodings.

A.2. + Basic Conformance Classes:

A.2.1. .Basic-Euler Conformance Class

Abstract Test 1	**/ats/basic/Euler**
Test Purpose	To validate that Basic-Euler GeoPose data objects conform to the JSON-Schema definition.
Requirement	/req/basic/Euler
Test Method	JSON-Schema validation

A.2.2. .Basic-Quaternion (Strict) Conformance Class

Abstract Test 2	**/ats/basic/quaternion**
Test Purpose	To validate that Basic-Quaternion GeoPose data objects conform to the JSON-Schema definition.
Requirement	/req/basic/quaternion
Test Method	JSON-Schema validation

A.2.3. .Basic-Quaternion (Permissive) Conformance Class

Abstract Test 3	**/ats/basic/quaternion/permissive**
Test Purpose	To validate that Basic-Quaternion GeoPose data objects conform to the permissive JSON-Schema definition.
Requirement	/req/basic/quaternion/permissive
Test Method	JSON-Schema validation

A.3. Conformance Class Advanced

Abstract Test 4	**/ats/advanced**
Test Purpose	To validate that Advanced GeoPose data objects conform to the JSON-Schema definition.
Requirement	/req/advanced
Test Method	JSON-Schema validation

A.4. Composite Conformance Classes:

A.4.1. . Graph Conformance Class

Abstract Test 5	**/ats/composite/graph**
Test Purpose	To validate that JSON-encoded GeoPose Graphs conform to the Graph JSON-Schema definition.
Requirement	/req/composite/graph
Test Method	JSON-Schema validation

A.4.2. . Chain Conformance Class

Abstract Test 6	**/ats/composite/chain**
Test Purpose	To validate that GeoPose data Chains conform to their JSON-Schema definition.
Requirement	/req/composite/chain
Test Method	JSON-Schema validation

A.4.3. + Sequence Conformance Classes:

A.4.4. ++ Series Conformance Classes:

A.4.5. .. Regular Series Conformance Class

Abstract Test 7	**/ats/composite/sequence/series/regular**
Test Purpose	To validate that JSON-encoded GeoPose series conform to the Regular Series JSON-Schema definition.
Requirement	/req/composite/sequence/series/regular
Test Method	JSON-Schema validation

A.4.6. .. Irregular Series Conformance Class

Abstract Test 8	**/ats/composite/sequence/series/irregular**

Test Purpose	To validate that JSON-encoded GeoPose series conform to the Irregular Series JSON-Schema definition.
Requirement	/req/composite/sequence/series/irregular
Test Method	JSON-Schema validation

A.4.7. .. Stream Conformance Class

Abstract Test 9	**/ats/composite/sequence/stream**
Test Purpose	To validate that GeoPose data stream elements conform to the JSON-Schema definition.
Requirement	/req/composite/sequence/stream
Test Method	JSON-Schema validation

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

XXX Frame Specification types are defined by the built in to GeoPose v1. The authority string for these is "geopose://v1/<frame specification name>".

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.1.1. Requirement 2

Requirement:	/req/framespec/LTP-ENU

Test purpose:	Verify that...
Test method:	Inspect...

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]],
]
```

B.2.1. Requirement 1

Test id:	/conf/conf-class-a/req-name-1
Requirement:	/req/req-class-a/req-name-1
Test purpose:	Verify that...

Test method:	Inspect…

Annex C: GeoPose Use and Interpretation of Euler Angles

Euler angles come in two forms: with reference to an initial set of unchanging world coordinates or with reference to local coordinates that undergo each of the rotations. In addition, the order of the axes about which the rotations are performed must be specified. Finally, the unit of measure must be specified for the amount of each of the rotations.

GeoPose shall use rotations about local coordinates, in the axis order z, y, x, and with the degree as the unit of measure.

Annex D: GeoPose Use and Interpretation of Quaternions

NOTE This section defines the interpretation of the four numbers representing a quaternion as w, x, y, z and ties back to the Foley and van Dam section on rotations.

Annex E: GeoPose Use and Interpretation of UNIX Time

GeoPose has adopted a variation UNIX time as the method for denoting the location of Instants on a time line. The reasons for this specific choice include the widespread availability of UNIX time in computer operating systems, the straighforward 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.

E.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 inegral numbers of milleseconds.

E.2. Scaling

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

E.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 seonds applied as needed either at 31 December or 30 June.

E.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.

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

International timekeeping switched from an astronomical basis to a reference based on atomic prosesses 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 F: 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 G: Revision History

Date	Release	Editor	Primary clauses modified	Description
2020-11-21	0.1	Steve Smyth	all	initial integrated version

Annex H: Bibliography

NOTE

Example Bibliography (Delete this note).

The TC has approved Springer LNCS as the official document citation type.

Springer LNCS is widely used in technical and computer science journals and other publications

- For citations in the text please use square brackets and consecutive numbers: [1], [2], [3]

– Actual References:

[n] Journal: Author Surname, A.: Title. Publication Title. Volume number, Issue number, Pages Used (Year Published)

[n] Web: Author Surname, A.: Title, http://Website-Url

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

```
HAMILTON, SIR WILLIAM ROWAN, "On quaternions; or on a new system of imaginaries in
algebra," Philosophical Magazine xxv pp. 10-13 (July 1844).
```