

Individual Submission
Internet-Draft
Intended status: Informational
Expires: 17 October 2026

W. Ackerman
Vertiv Holdings Co.
15 April 2026

Temporal Integrity Metadata (TIM) for Infrastructure Telemetry
draft-ackerman-temporal-integrity-metadata-00

Abstract

Distributed computing systems generate timestamped events from components whose clocks operate under fundamentally different synchronization conditions. Existing logging and observability standards -- including RFC 3164, RFC 5424, SNMP, NETCONF, and OpenTelemetry -- define message formats and telemetry schemas but provide no standard mechanism for an event source to declare the provenance, confidence, or synchronization state of its timestamps. Every platform that must correlate events across components independently invents a proprietary temporal reconciliation layer. These systems fail silently, cannot be validated against a published standard, and are not interoperable.

This document defines Temporal Integrity Metadata (TIM): a transport-agnostic structure that any event-emitting system may attach to its telemetry to declare how its timestamp was generated, the synchronization state of its clock, a bounded uncertainty interval, the temporal reference domain, and a monotonic sequence token for ordering events when wall-clock time is unavailable. TIM is backward-compatible with existing protocols, implementable on constrained embedded hardware, and applicable from internet-scale distributed services to air-gapped and orbital deployments.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 17 October 2026.

Copyright Notice

Copyright (c) 2026 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Table of Contents

1. Introduction	3
1.1. Background and Motivation	4
1.2. Scope	4
1.3. Positioning vs. Adjacent Standards	5
1.4. Why This Standard, Why Now	7
1.5. Requirements Language	7
2. Problem Statement	8
2.1. Internet-Scale Scope of the Problem	8
2.2. The Temporal Fragmentation Problem	8
2.3. The Missing Declaration Standard	9
2.4. Impact Across Application Domains	9
2.4.1. Security Incident Timeline Reconstruction	9
2.4.2. Distributed Transaction Ordering	9
2.4.3. Telecom Call Trace Correlation	10
2.4.4. Representative Emerging Efficiency Metric (e.g., AI Inference tok/W)	10
2.5. The Precision Discard Problem	10
2.6. The Temporal Reference Domain Problem	10
3. Terminology	11
4. Temporal Integrity Metadata (TIM) Specification	11
4.1. Overview	12
4.1.1. Semantics of event_time and emission_time	12
4.1.2. Causality and TIM	13
4.1.3. Computing uncertainty_ns in Practice	13
4.2. Required Fields	15
4.3. Conditionally Required Fields	15
4.4. Optional Fields	16
4.5. Collector-Populated Fields	18
5. Temporal Confidence Classes	19
6. Sync State Definitions and Transitions	20
7. Temporal Reference Domains	21
7.1. Motivation	21
7.2. Defined Domains	21

7.3.	Cross-Domain Correlation Requirements	22
8.	TIM Schema and Examples	22
8.1.	Class A Example -- PTP-Synchronized Device	22
8.2.	Class D Example -- Internet NTP Device	23
8.3.	Class F Example -- Minimal TIM (No Clock, Sequence Only)	23
8.4.	Class E Example -- Holdover in Progress	24
8.5.	Class F Extended -- Unsynchronized RTC	24
8.6.	Mobile / IoT Example -- 5G-Connected Device (OS NTP vs. Network Precision)	25
8.7.	Wi-Fi FTM Example -- Relative Time Domain	25
8.8.	Orbital Example -- LEO with Relativistic Correction	26
9.	Implementation Guidance -- Event Sources	26
9.1.	Minimum Viable Implementation	27
9.2.	Sync Source Hierarchy for New Designs	27
9.3.	Sequence Token Requirements	28
9.4.	Holdover Uncertainty Reporting	29
10.	Implementation Guidance -- Consuming Platforms	29
10.1.	Temporal Reference Manager (TRM)	29
10.2.	Backward Compatibility -- Devices Without TIM	29
10.3.	Cross-Class Efficiency Metric Computation	30
11.	Relationship to Existing Standards	30
11.1.	RFC 3164 / RFC 5424 Syslog	30
11.2.	SNMP	30
11.3.	NETCONF/YANG	30
11.4.	IEEE 1588 (PTP)	30
11.5.	SMPTE ST 12-1	30
11.6.	Google TrueTime	31
11.7.	OpenTelemetry	31
11.8.	W3C Trace Context	31
11.9.	IEEE 802.11 -- Wi-Fi Timing Mechanisms	31
11.10.	3GPP -- LTE/5G Cellular Timing	32
12.	Security Considerations	32
13.	IANA Considerations	33
14.	References	33
14.1.	Normative References	33
14.2.	Informative References	33
Appendix A.	Deployment Scenarios	34
A.1.	Commercial AI Data Center	34
A.2.	Air-Gapped DOD / DOE Facility	34
A.3.	Aeronautical Platform	35
A.4.	Orbital Installation (LEO)	35
A.5.	Lunar Installation	35
A.6.	5G Edge Computing Site	35
Author's Address	35

1. Introduction

1.1. Background and Motivation

Distributed computing systems generate timestamped events from components that do not share a common time reference. This is not an edge case -- it is the default condition of every networked system at scale. A web request traverses a CDN edge node, a load balancer, an application cluster, a distributed cache, and a database, each with independently synchronized clocks, generating log entries that are later correlated to diagnose performance or reconstruct an incident. The assumption embedded in that correlation -- that the timestamps are comparable -- is almost never verified and almost never declared.

The same assumption fails in security operations (SIEM platforms correlating events across firewall, endpoint, identity, and network logs), in financial systems (distributed transaction processors ordering trades across geographically separated nodes), in telecommunications (call trace platforms reconciling CDRs across SIP proxies, media servers, and billing systems), and in industrial and infrastructure environments. In every case, the receiving system has no standard basis for knowing how trustworthy the timestamps in its event streams actually are.

This document observes that the problem has been solved repeatedly, in isolation, for specific domains. The broadcast and motion picture industry solved an analogous problem in 1969 with SMPTE 12M, establishing the principle that a timecode label is not the same as actual time, and that synchronization state must be explicitly declared. Google's TrueTime [SPANNER] solved it for globally distributed databases in 2012 by representing timestamps as bounded uncertainty intervals rather than point values. What has never existed is a single, transport-agnostic standard that any event-emitting system can use to declare the quality of its timestamps. This document defines that standard.

1.2. Scope

This specification defines the Temporal Integrity Metadata (TIM) standard. The following are within scope:

- * The TIM structure: a transport-agnostic metadata block for declaring timestamp provenance, synchronization state, uncertainty bounds, temporal reference domain, and sequence ordering
- * Sync State definitions: six states covering all operational scenarios from GPS-locked to sequence-only
- * Temporal Confidence Classes A through F: derived from declared sync state and uncertainty bounds

- * Temporal Reference Domains: covering terrestrial UTC, orbital, cislunar, and facility-autonomous environments
- * Implementation guidance for event sources and consuming platforms
- *Fundamental design principle:* _This specification does not require the existence of a globally synchronized time source. It defines a framework for expressing temporal information under conditions where such a source may be unavailable, unreliable, or undesired._
- *Non-Goals -- this specification explicitly does NOT:*
- * Synchronize clocks, discipline oscillators, or improve the accuracy of any time source
- * Replace or supersede existing timestamp fields in any protocol, log format, or telemetry schema
- * Guarantee the accuracy of declared values -- TIM declares the emitting system's temporal state at the time of emission; it does not certify objective accuracy
- * Define transport mechanisms for telemetry or modify existing message formats
- * Provide causal ordering guarantees -- TIM enables systems to reason about the bounds within which causal ordering can and cannot be established; it does not guarantee causal correctness

1.3. Positioning vs. Adjacent Standards

TIM occupies a specific and previously unoccupied layer in the distributed systems standards stack:

Layer	Standard(s)	What It Addresses
Time synchronization	NTP ([RFC5905]), PTP (IEEE 1588)	Making clocks accurate
Telemetry schema	OpenTelemetry	Defining what telemetry data looks like
Distributed correlation	W3C Trace Context	Propagating trace/span IDs
Temporal trust	TIM (this document)	Declaring how trustworthy timestamps are

Table 1: TIM in the Distributed Systems Standards Stack

TIM does not replace any of them; it provides the missing fourth layer.

*NTP (RFC 5905) and PTP (IEEE 1588):*These protocols synchronize clocks. TIM declares the quality of whatever clock a system has, regardless of whether it has been synchronized. A system with perfect PTP synchronization should declare TIM Class A. A system with no time reference should declare TIM Class F. Both are valid and useful declarations.

*OpenTelemetry (OTel):*OTel defines schemas for traces, metrics, and logs with timestamp fields but provides no mechanism for declaring how trustworthy those timestamps are. TIM operates orthogonally to observability schemas: it annotates time, not telemetry semantics. TIM is the missing provenance layer for OTel timestamps. TIM fields SHOULD be expressed as OTel resource attributes, extending OTel rather than replacing it.

*W3C Trace Context:*Distributed tracing carries trace and span IDs but contains no timestamp information. Timestamps in distributed traces are assigned by each service independently. TIM provides what distributed tracing assumes but never defines: a standard for declaring the reliability of those timestamps.

*RFC 3164 / RFC 5424 Syslog:*RFC 3164 deferred timestamp quality to the application layer. RFC 5424 added structured formatting but left the same gap. TIM fills this gap without modifying either RFC.

1.4. Why This Standard, Why Now

This problem has existed since the first distributed computing system was assembled. Four developments make standardization critically important now:

***AI systems, automation, and decisions without human validation:** AI inference pipelines, autonomous vehicles, robotic control systems, and AI agents making real-time decisions operate on event streams without a human in the loop. When an autonomous system misorders two events, it may make an incorrect inference with no opportunity for correction before the next decision cycle. As the tempo of automated decision-making accelerates and human oversight decreases, the accuracy of event ordering becomes a safety and reliability property. The absence of a timestamp quality standard means every AI system must either assume all timestamps are reliable (they are not) or invent its own assessment (no two systems agree).

***Zero-trust security and audit requirements:** The zero-trust model requires forensically defensible audit trails. An audit trail constructed from timestamps of unknown quality is legally and forensically weak. Regulatory frameworks -- MiFID II, NERC CIP, HIPAA, the EU AI Act -- are converging on timestamp accuracy declaration requirements. TIM provides the standard vocabulary.

***Edge computing and IoT at scale:** Edge deployments and IoT systems involve billions of devices with highly variable clock quality. Aggregating their events for anomaly detection and operational intelligence requires honest temporal integrity declarations at scale.

***Regulatory convergence:** Multiple regulatory bodies are independently requiring timestamp accuracy declarations. The White House OSTP directive on cislunar time standardization [OSTP-LTC] is one example. A single open standard that serves all these contexts is preferable to parallel proprietary solutions.

1.5. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Problem Statement

2.1. Internet-Scale Scope of the Problem

The absence of a timestamp quality declaration standard is not a niche infrastructure problem. It is a structural deficiency present in every distributed computing system on the internet.

The global volume of timestamped events generated across internet-scale systems is measured in trillions per day. The fraction for which the receiving system can state, with any declared confidence, the accuracy of the timestamp, is effectively zero. This is not because the problem is unsolvable -- it is because no standard declaration vocabulary has existed. This document provides that standard.

2.2. The Temporal Fragmentation Problem

Event-emitting systems operate under fundamentally different clock synchronization conditions:

Class	Accuracy	Protocol	Representative Systems
1	+/-10 ns	PTP IEEE 1588	AI compute fabric, financial trading systems, 5G base stations
2	+/-1-250 ms	NTP [RFC5905]	Application servers, cloud VMs, PDU management cards, IoT gateways
3	Relative only	SNMP sysUpTime	Legacy network devices, embedded controllers
4	Unknown drift	Unsync'd RTC	Offline devices, isolated OT networks
5	None	None	Air-gapped systems, constrained embedded devices

Table 2: Clock Synchronization Classes in Distributed Systems

All five classes are present simultaneously in any large-scale deployment. No existing standard requires any device to declare which class applies to its timestamps.

2.3. The Missing Declaration Standard

RFC 3164 [RFC3164] acknowledges that not all devices can timestamp their messages but offers no mechanism for a device to declare this condition. RFC 5424 [RFC5424] adds structured timestamp formatting but provides no vocabulary for declaring uncertainty, sync state, or source. The consequence: a monitoring platform cannot distinguish, from message content alone, between a timestamp accurate to +/-10 nanoseconds and one accurate to +/-250 milliseconds.

In the absence of declared temporal integrity, applications independently implement proprietary mechanisms for event ordering, correlation, and suppression. These mechanisms are not interoperable, cannot be validated against a published standard, and produce unreliable causal inference across system boundaries. Temporal integrity is a data-plane concern; causality is an application concern. SMPTE ST 12-1 established this boundary for broadcast infrastructure in 1969. TIM establishes it for networked computing.

2.4. Impact Across Application Domains

2.4.1. Security Incident Timeline Reconstruction

A SIEM investigating a data exfiltration incident correlates events from a firewall (NTP, +/-100ms), an authentication service (cloud-hosted, +/-50ms), an endpoint agent (Windows time service, +/-500ms), and a database audit log (GPS-disciplined NTP, +/-5ms). If the firewall alert and the authentication event are 200ms apart, the analyst cannot determine which came first -- the combined +/-600ms uncertainty makes the ordering ambiguous.

*With TIM:*Each event carries a declared Temporal Confidence Class. The analyst knows which events can be trusted for millisecond-level ordering and which cannot. The SIEM presents the incident timeline with per-event confidence indicators rather than presenting all events as equally reliable.

2.4.2. Distributed Transaction Ordering

A financial platform processes trades across distributed nodes. Node A (GPS-disciplined, +/-10ns) records a price update at T=14:23:45.100. Node B (NTP, +/-200ms) records a trade execution at T=14:23:45.050. The trade appears to precede the price update by 50ms -- possibly evidence of front-running. But the +/-200ms NTP uncertainty means the trade may have occurred up to 200ms later. With TIM, the compliance system automatically flags that Node B is Class D and the 50ms lead is within the uncertainty window --

inconclusive rather than suspicious.

2.4.3. Telecom Call Trace Correlation

A VoIP operator correlates events from a SIP proxy (PTP, +/-1us), a media transcoder (NTP, +/-50ms), a billing CDR system (NTP, +/-200ms), and a QoS monitor (SNMP sysUpTime, no wall-clock reference). Without declared uncertainty, the operator cannot determine whether a codec error caused packet loss or was caused by it. With TIM, the SIP proxy events anchor the timeline at Class A, the transcoder events declare Class C, and the QoS monitor declares Class F -- sequence ordering only.

2.4.4. Representative Emerging Efficiency Metric (e.g., AI Inference tok/W)

A representative example of an emerging efficiency metric requiring cross-domain temporal correlation is token throughput per watt (tok/W) for AI inference infrastructure. Computing this metric requires correlating primary compute throughput (PTP-synchronized, +/-10ns) with power consumption (NTP-synchronized, +/-250ms) and cooling load (Modbus, no time reference). TIM enables such metrics to be computed as bounded intervals derived from the declared uncertainty of constituent measurements, rather than as point values with false precision. This pattern generalizes to any multi-domain efficiency metric requiring temporal coherence across heterogeneous components.

2.5. The Precision Discard Problem

A pervasive pattern: each layer of the software stack systematically discards precision that the hardware layer below it provides. A 5G base station synchronizes to GNSS to +/-1.5 microseconds. The mobile device connected to that base station uses NTP, achieving +/-100 milliseconds. TIM makes this gap visible, quantifiable, and actionable.

2.6. The Temporal Reference Domain Problem

As computing extends to aeronautical, orbital, and cislunar environments, relativistic effects mean that clocks at different gravitational potentials do not tick at the same rate. For an observer on the lunar surface, an Earth-based clock loses approximately 56.7 microseconds per Earth day [OSTP-LTC]. No existing infrastructure logging RFC addresses this. TIM defines a Temporal Reference Domain field enabling any device to declare its relativistic and administrative time context.

3. Terminology

The following terms are used throughout this document:

Temporal Provenance The complete characterization of how a timestamp was generated, including its source, synchronization state, uncertainty bounds, and temporal reference domain. Temporal provenance is the concept; TIM is the standard for declaring it.

Event Time (T_{event}) The time at which the condition being reported actually occurred at the source. This is the time of interest for event correlation.

Emission Time (T_{emit}) The time at which the device generated and transmitted the message. MAY differ from Event Time if the device queues or batches events.

Ingestion Time (T_{ingest}) The time at which the collecting platform received the message. Populated by the collector, not the originating device.

Temporal Integrity Metadata (TIM) The structured metadata block defined in this document, attached to or associated with a telemetry emission to declare its temporal provenance.

Sync State A declared operational state indicating the relationship between a device's clock and an external time reference at the moment of emission. See Section 6.

Temporal Confidence Class A single-letter classification (A through F) summarizing overall timestamp quality implied by declared sync state and uncertainty bounds. See Section 5.

Uncertainty Interval A closed interval [earliest, latest] guaranteed to contain the absolute time of the event. Inspired by and modeled after Google TrueTime [SPANNER].

Temporal Reference Domain A declared context specifying the time scale and relativistic reference frame in which timestamps are expressed. See Section 7.

Causal Anchor A high-confidence timestamp from a correlated event in a different domain, used to bound the uncertainty interval of a lower-confidence timestamp. A platform-level concept; not carried in TIM itself.

4. Temporal Integrity Metadata (TIM) Specification

4.1. Overview

The TIM structure is a metadata block that SHOULD accompany every telemetry emission. It is transport-agnostic and may be carried as:

- * application/json -- in REST API responses and webhook payloads
- * Structured data in RFC 5424 syslog messages using a registered SD-ID
- * OID-value pairs in SNMP responses and notifications
- * A YANG data node in NETCONF notifications and RESTCONF responses
- * Any other structured representation appropriate to the transport

TIM does not replace existing message formats. It augments them with temporal integrity metadata that existing formats cannot express.

4.1.1. Semantics of event_time and emission_time

***event_time:** The time at which the reported condition occurred. This value MAY be directly observed or inferred. When inferred, the `uncertainty_ns` field MUST reflect the inference uncertainty in addition to clock quality uncertainty. Devices SHOULD declare `event_time` null rather than declare a value they cannot support with any uncertainty bound.

***emission_time:** The time at which the device generated and transmitted this message. This value MAY be later than `event_time` due to queuing, batching, or scheduling latency. The gap between `event_time` and `emission_time` is observable evidence of reporting latency.

***ingestion_time:** The time at which the collecting platform received the message. Populated by the collector, never by the originating device.

***Critical:** None of these fields are guaranteed to represent objective ground truth. Each represents the best available estimate at the respective stage, qualified by `uncertainty_ns`. Consuming systems MUST NOT treat any field as exact without verifying the confidence class.

4.1.2. Causality and TIM

TIM does not guarantee causal correctness. It enables systems to reason about the bounds within which causal ordering can and cannot be established. Two events whose uncertainty intervals do not overlap can be confidently ordered. Two events whose uncertainty intervals overlap cannot be confidently ordered -- their causal relationship is ambiguous within the declared bounds.

This is the same principle established by Google's TrueTime [SPANNER]: when uncertainty intervals overlap, ordering ambiguity must be resolved explicitly. TIM makes this uncertainty visible for any event type.

4.1.3. Computing `uncertainty_ns` in Practice

The `uncertainty_ns` field MUST represent a conservative upper bound on the absolute difference between `event_time` and the true occurrence time. Implementations SHOULD use the following reference models:

Sync Source	Recommended Computation	Notes
PTP_IEEE1588	<code>grandmaster_clockAccuracy + path_delay_asymmetry</code>	Use IEEE 1588-2019 <code>clockAccuracy</code> field. Add 10% margin for uncompensated asymmetry.
GPS_GNSS	<code>gnss_receiver_accuracy_spec</code>	Use manufacturer-specified accuracy. Typical: +/-10-30 ns.
NTP_GPS_DISCIPLINED	<code>2 x ntp_root_delay + ntp_root_dispersion</code>	chrony and ntpd expose <code>root_delay</code> and <code>root_dispersion</code> . Conservative bound. Typical: +/-1-5 ms.
NTP_INTERNET	<code>2 x ntp_root_delay + ntp_root_dispersion</code>	Same formula. If statistics unavailable, RECOMMENDED default: 250,000,000 ns.

IRIG_B	manufacturer_spec + cable_delay_estimate	Typically +/-1 us. Add approx 5 ns/m cable delay for runs over 100 m.
Holdover (OCXO)	drift_rate_ppb x holdover_duration_s x 1e9	Example: 50 ppb x 3600 s = 180,000 ns. Update at 60-second intervals.
Holdover (Rubidium)	drift_rate_ppb x holdover_duration_s x 1e9	Example: 0.1 ppb x 86400 s = 8,640 ns per day.
FREEWHEEL / UNKNOWN	null -- MUST NOT be fabricated	When no defensible bound exists, uncertainty_ns MUST be null. Implementations MAY provide an operator-defined fallback bound but MUST set uncertainty_source: "operator_fallback" to distinguish it from measured uncertainty. Consuming platforms receiving this flag MUST NOT use it for Class promotion above F without explicit operator authorization or policy override.

Table 3: uncertainty_ns Computation Guide by Sync Source

*Inference uncertainty:*When event_time is inferred from a polling interval, the inference uncertainty MUST be added to clock uncertainty. Example: 30-second polling + NTP +/-250 ms = total uncertainty_ns of approximately 30,250,000,000 ns.

4.2. Required Fields

The following fields are REQUIRED in every TIM emission:

Field	Type	Description
tim_version	String	TIM version implemented. Current value: "1.0". MUST be present to identify the schema.
sync_state	Enum	Synchronization state at emission time. MUST be one of: LOCKED, HOLDOVER, RECOVERING, FREEWHEEL, AUTONOMOUS, UNKNOWN. See Section 6.
sequence_token	Uint64	Monotonically increasing counter scoped to this device. MUST increment for every emission. Provides event ordering independent of wall-clock time.

Table 4: TIM Required Fields

4.3. Conditionally Required Fields

The following fields are REQUIRED under the stated conditions:

Field	Type	Description
event_time	[RFC3339]	REQUIRED unless sync_state is UNKNOWN or FREEWHEEL with no wall-clock reference. MUST include timezone designator (Z for UTC).
uncertainty_earliest	[RFC3339]	OPTIONAL. Earliest bound of the uncertainty interval for event_time. Derived from event_time and uncertainty_ns: uncertainty_earliest = event_time − uncertainty_ns. If present and inconsistent with uncertainty_ns,

		uncertainty_ns governs.
uncertainty_latest	[RFC3339]	OPTIONAL. Latest bound of the uncertainty interval for event_time. Derived from event_time and uncertainty_ns: uncertainty_latest = event_time + uncertainty_ns. If present and inconsistent with uncertainty_ns, uncertainty_ns governs.
uncertainty_ns	Uint64	REQUIRED when event_time is present. The canonical normative uncertainty half-width in nanoseconds. uncertainty_earliest and uncertainty_latest are OPTIONAL derived representations; uncertainty_ns is normative. Implementations MAY omit interval fields entirely when consumers can derive them from uncertainty_ns. MUST be null when no defensible bound can be established.

Table 5: TIM Conditionally Required Fields

4.4. Optional Fields

The following fields are OPTIONAL but RECOMMENDED where available:

Field	Type	Description
emission_time	[RFC3339]	Time at which this message was generated and transmitted, if different from event_time.
sync_source	Enum	Type of synchronization

		source. Values: PTP_IEEE1588, CELLULAR_5G_PRTC, GPS_GNSS, NTP_GPS_DISCIPLINED, NTP_INTERNET, IRIG_B, HAVEQUICK, INS_HOLDOVER, WIFI_TSF, WIFI_FTM, MOBILE_5G, MOBILE_LTE, OSCILLATOR_OCXO, OSCILLATOR_RUBIDIUM, OSCILLATOR_CESIUM, LTC_LUNAR, SPACECRAFT_ATOMIC, NONE.
grandmaster_id	String	For PTP sources: IEEE EUI-64 identity of the PTP grandmaster clock.
holdover_duration_s	Uint32	For HOLDOVER state: seconds since external reference was lost.
time_domain	Enum	Temporal reference domain. Default if absent: UTC_TERRESTRIAL. See Section 7.
relativistic_correction_applied	Boolean	Whether relativistic correction has been applied to event_time. Relevant for orbital and cislunar deployments.
relativistic_correction_ns_per_day	Int64	Declared relativistic correction rate in nanoseconds per day.
domain_id	String	Operator-defined

		identifier for isolated time domains (e.g., classification domains in DOD/DOE environments).
--	--	--

Table 6: TIM Optional Fields

4.5. Collector-Populated Fields

The following fields MUST NOT be populated by the originating device. They MUST be populated by the collecting platform upon receipt:

Field	Type	Description
<code>_collector.ingestion_time</code>	[RFC3339]	Time the collecting platform received this message. Always from the collector's TRM.
<code>_collector.confidence_class</code>	String	Temporal Confidence Class (A-F) assigned by the collector based on <code>sync_state</code> and <code>uncertainty_ns</code> .
<code>_collector.device_tim_native</code>	Boolean	True if the device emitted native TIM; false if TIM was inferred by the collector.
<code>_collector.anomaly_flags</code>	Array	List of anomaly codes detected: SEQUENCE_VIOLATION, STALE_HOLDOVER, DOMAIN_MISMATCH, ANOMALOUS_LOCKED.

Table 7: TIM Collector-Populated Fields

5. Temporal Confidence Classes

Temporal Confidence Classes are a standardized interpretation layer enabling consistent cross-platform reasoning about timestamp quality -- comparable to DSCP QoS classes in IP networking or severity levels in syslog. The class boundaries are anchored to the accuracy regimes of common synchronization technologies rather than arbitrary numeric divisions: Classes A and B correspond to PTP/GNSS-quality synchronization; Class C to GPS-disciplined NTP; Class D to internet NTP; Class E to oscillator holdover with growing drift; and Class F to devices with no usable time reference.

Collecting platforms MUST assign a Temporal Confidence Class to each received telemetry event based on the declared `sync_state` and `uncertainty_ns` fields.

Class	Sync States	Uncertainty	Typical Source
A	LOCKED	< 1 us	PTP IEEE 1588 / GPS atomic
B	LOCKED / HOLDOVER	1 us - 100 us	GPS-disciplined NTP / IRIG-B / rubidium holdover < 24h
C	LOCKED	100 us - 10 ms	GPS-disciplined NTP server
D	LOCKED / FREEWHEEL	10 ms - 250 ms	Internet NTP
E	HOLDOVER / RECOVERING	> 250 ms or growing	Oscillator holdover, re- convergence
F	FREEWHEEL / UNKNOWN	Undefined	No reference / sequence only

Table 8: Temporal Confidence Classes

***Class F constraint:** Events with Confidence Class F MUST NOT be used for cross-domain temporal ordering without explicit operator authorization or policy override. Such authorization SHOULD be recorded as an audit event. Sequence tokens remain valid for Class F events and SHOULD be used for relative ordering within a single device stream.

***Holdover progression:** A device entering HOLDOVER begins at the class appropriate to its last locked uncertainty. The class degrades as accumulated drift increases `uncertainty_ns`. Devices SHOULD update `uncertainty_ns` at regular intervals (RECOMMENDED: every 60 seconds) during holdover.

6. Sync State Definitions and Transitions

Six sync states are defined. Devices MUST declare exactly one sync state in every TIM emission.

State	Description
LOCKED	Device clock is actively synchronized to an external reference within declared uncertainty bounds.
HOLDOVER	External reference was lost. Device is maintaining time using an internal oscillator. <code>holdover_duration_s</code> SHOULD be populated and <code>uncertainty_ns</code> MUST reflect accumulated drift.
RECOVERING	External reference has been restored after HOLDOVER or FREEWHEEL. Clock is converging. <code>uncertainty_ns</code> remains elevated until convergence completes.
FREEWHEEL	Device is operating on its internal clock with no recent synchronization. No external reference is available.
AUTONOMOUS	Device is operating on a local atomic reference with no external synchronization path. Applicable to orbital, cislunar, and deep space deployments. The <code>time_domain</code> field MUST be populated.
UNKNOWN	Device cannot determine its synchronization state. Implies sequence-only ordering.

Table 9: Sync State Definitions

Devices SHOULD emit a TIM with the updated `sync_state` whenever a state transition occurs. Devices that cannot emit unsolicited transition events MUST reflect the current `sync_state` in the next scheduled emission.

7. Temporal Reference Domains

7.1. Motivation

All existing infrastructure logging RFCs implicitly assume UTC on Earth's geoid. This assumption fails for infrastructure operating at significantly different gravitational potential or velocity. At ISS altitude (~400 km), the net relativistic correction is approximately +38 microseconds per Earth day. On the lunar surface, approximately +56 microseconds per Earth day.

Implementation note -- optionality: The `time_domain` field is OPTIONAL. Terrestrial deployments NEED NOT populate it; UTC_TERRESTRIAL applies implicitly. Implementations MAY ignore non-terrestrial domain values without loss of interoperability with other terrestrial TIM implementations. These domains are defined for future-proofing: (1) the field is optional and adds zero overhead to terrestrial deployments; (2) the White House OSTP issued a directive in April 2024 establishing Coordinated Lunar Time [OSTP-LTC], providing direct policy backing; (3) commercial orbital computing is actively being developed.

7.2. Defined Domains

Domain Value	Description
UTC_TERRESTRIAL	Standard UTC on Earth's surface. Default when <code>time_domain</code> is absent.
UTC_LEO_CORRECTED	UTC with applied relativistic correction for low Earth orbit (~+38 us/day at ISS altitude). <code>relativistic_correction_ns_per_day</code> MUST be populated.
LTC_LUNAR	Coordinated Lunar Time as defined by NASA/OSTP. Approx +56 us/day vs UTC_TERRESTRIAL.
LTC_CISLUNAR	Time standard for cislunar orbital operations. Correction varies with orbital parameters.
SPACECRAFT_TAI	International Atomic Time maintained by onboard atomic clock, without leap second corrections.

FACILITY_AUTONOMOUS	Facility-local time with no declared relationship to any international time scale. Events MUST NOT be cross-correlated with other domains without operator-defined conversion parameters.
WIFI_FTM_RELATIVE	Local relative time domain based on Wi-Fi FTM ranging. Timestamps represent elapsed nanoseconds from an anchor point and are NOT directly comparable to UTC-domain timestamps without calibration.

Table 10: Temporal Reference Domain Values

7.3. Cross-Domain Correlation Requirements

When correlating events from different Temporal Reference Domains, a consuming platform MUST:

1. Identify the declared domain of each event stream
2. Apply the appropriate conversion function and propagation delay correction
3. Compute resulting uncertainty as the sum of individual bounds plus conversion uncertainty
4. NEVER silently treat timestamps from different domains as directly comparable without conversion

8. TIM Schema and Examples

8.1. Class A Example -- PTP-Synchronized Device

```
{
  "tim_version":    "1.0",
  "event_time":     "2025-09-18T14:23:45.123456789Z",
  "uncertainty_ns": 10,
  "sync_state":     "LOCKED",
  "sync_source":    "PTP_IEEE1588",
  "grandmaster_id": "00:1a:2b:3c:4d:5e:6f:70",
  "sequence_token": 8472910,
  "time_domain":    "UTC_TERRESTRIAL",
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.610Z",
    "confidence_class": "A",
    "device_tim_native": true
  }
}
```

8.2. Class D Example -- Internet NTP Device

```
{
  "tim_version":    "1.0",
  "event_time":     "2025-09-18T14:23:45.123Z",
  "uncertainty_ns": 250000000,
  "sync_state":     "LOCKED",
  "sync_source":    "NTP_INTERNET",
  "sequence_token": 33201,
  "time_domain":    "UTC_TERRESTRIAL",
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.890Z",
    "confidence_class": "D"
  }
}
```

8.3. Class F Example -- Minimal TIM (No Clock, Sequence Only)

The absolute minimum valid TIM. No time reference. Only the `sequence_token` is populated, enabling event ordering within this device stream. Conformant baseline for the most constrained embedded devices.

```
{
  "tim_version":    "1.0",
  "event_time":     null,
  "uncertainty_ns": null,
  "sync_state":     "FREEWHEEL",
  "sync_source":    "NONE",
  "sequence_token": 12847,
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.610Z",
    "confidence_class": "F"
  }
}
```

8.4. Class E Example -- Holdover in Progress

```
{
  "tim_version":    "1.0",
  "event_time":     "2025-09-18T14:23:45.123456Z",
  "uncertainty_earliest": "2025-09-18T14:23:44.723Z",
  "uncertainty_latest":  "2025-09-18T14:23:45.524Z",
  "uncertainty_ns":   400200000,
  "sync_state":      "HOLDOVER",
  "sync_source":     "OSCILLATOR_OCXO",
  "holdover_duration_s": 3612,
  "sequence_token":  98341,
  "time_domain":     "UTC_TERRESTRIAL",
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.890Z",
    "confidence_class": "E"
  }
}
```

8.5. Class F Extended -- Unsynchronized RTC

Device has a battery-backed RTC but it has never been synchronized.
event_time is present and useful for human inspection but MUST NOT be
used for cross-domain ordering.


```
{
  "tim_version":    "1.0",
  "event_time":     "2025-09-18T14:23:45.000Z",
  "uncertainty_ns": null,
  "sync_state":     "FREEWHEEL",
  "sync_source":    "NONE",
  "sequence_token": 5501,
  "_advisory": {
    "note": "RTC present; last sync unknown. No bound established."
  },
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:46.001Z",
    "confidence_class": "F"
  }
}
```

8.6. Mobile / IoT Example -- 5G-Connected Device (OS NTP vs. Network Precision)

Illustrates the precision discard problem. Device is connected to a 5G network (GNSS-disciplined to +/-1.5 us) but the OS uses NTP. The `_advisory` block flags that better precision is available but not exposed.

```
{
  "tim_version":    "1.0",
  "event_time":     "2025-09-18T14:23:45.123Z",
  "uncertainty_ns": 250000000,
  "sync_state":     "LOCKED",
  "sync_source":    "NTP_INTERNET",
  "sequence_token": 884721,
  "_advisory": {
    "cellular_network_timing_available": true,
    "cellular_achievable_uncertainty_ns": 1500,
    "note": "OS does not expose cellular timing; NTP used instead"
  },
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.890Z",
    "confidence_class": "D"
  }
}
```

8.7. Wi-Fi FTM Example -- Relative Time Domain

Device using Wi-Fi FTM ranging. `event_time` is null because this is a relative domain. UTC correlation requires a separate calibration record.

```
{
  "tim_version": "1.0",
  "event_time": null,
  "uncertainty_ns": 2,
  "sync_state": "LOCKED",
  "sync_source": "WIFI_FTM",
  "sequence_token": 44102,
  "time_domain": "WIFI_FTM_RELATIVE",
  "_ftm": {
    "anchor_bssid": "aa:bb:cc:dd:ee:ff",
    "anchor_calibration_id": "cal-2025-09-18T00:00:00Z",
    "rtts_ns": 142847193
  },
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.890Z",
    "confidence_class": "A"
  }
}
```

8.8. Orbital Example -- LEO with Relativistic Correction

Compute node in low Earth orbit. GPS receiver provides Class A timestamps. The relativistic correction (+38 us/day at ISS altitude) has been applied to event_time before emission.

```
{
  "tim_version": "1.0",
  "event_time": "2025-09-18T14:23:45.000010Z",
  "uncertainty_ns": 10,
  "sync_state": "LOCKED",
  "sync_source": "GPS_GNSS",
  "sequence_token": 220194,
  "time_domain": "UTC_LEO_CORRECTED",
  "relativistic_correction_applied": true,
  "relativistic_correction_ns_per_day": 38000,
  "_collector": {
    "ingestion_time": "2025-09-18T14:23:45.900Z",
    "confidence_class": "A"
  }
}
```

9. Implementation Guidance -- Event Sources

9.1. Minimum Viable Implementation

A device with no time reference can still implement a conformant minimal TIM with three fields: `tim_version`, `sync_state`: "FREEWHEEL", and `sequence_token`. This provides sequence ordering and ensures consuming platforms do not silently treat absent timestamps as reliable.

9.2. Sync Source Hierarchy for New Designs

Device designers implementing TIM SHOULD implement sync source support in the following priority order, selecting the highest available tier:

Tier	Source	Accuracy	Context
1A	PTP IEEE 1588	+/-10 ns	High-performance data centers and compute environments with PTP-capable fabric
1B	IRIG-B	+/-1 us	Air-gapped military, DOE, industrial environments
1C	GPS/GNSS direct	+/-10 ns	Installations with GPS antenna access
1D	CELLULAR_5G_PRTC	+/-100 ns - 1.5 us	5G edge sites using base station T-GM
2	GPS-disciplined NTP	+/-1-5 ms	Facilities with OCP-TAP or equivalent
3	Internet NTP	+/-100-250 ms	Commercial deployments without local time infrastructure
4A	MOBILE_5G / MOBILE_LTE	+/-1.5-10 us	When OS or app layer explicitly exposes cellular timing
4B	WIFI_FTM	+/-1 ns relative	Wi-Fi FTM ranging; relative domain
5	Sequence only	N/A	Constrained devices with no time reference

Table 11: Sync Source Priority Tiers

9.3. Sequence Token Requirements

The `sequence_token` is the most important field for devices without reliable wall-clock time:

* ***MUST:** Increment the token for every emission without exception.

- * ***SHOULD:**Store the sequence token in non-volatile memory so it survives power cycles. Constrained devices [RFC7228] that cannot provide non-volatile storage MAY initialize to a random 64-bit value on each boot as a degraded-mode alternative, and **MUST** document this limitation.
- * ***MUST:**Initialize to a random 64-bit value after factory reset to reduce collision probability with prior device history.

9.4. Holdover Uncertainty Reporting

Devices entering HOLDOVER state SHOULD update uncertainty_ns at regular intervals (RECOMMENDED: every 60 seconds) to reflect accumulated drift. Constrained devices that cannot update on schedule MUST update uncertainty_ns on every emission during holdover.

10. Implementation Guidance -- Consuming Platforms

10.1. Temporal Reference Manager (TRM)

Consuming platforms that ingest TIM-annotated telemetry SHOULD implement a Temporal Reference Manager -- a component responsible for maintaining the best available time reference for ingestion_time stamping, managing failover between sources, and assigning confidence classes to incoming events.

10.2. Backward Compatibility -- Devices Without TIM

Consuming platforms MUST operate correctly when receiving telemetry from devices that do not implement TIM. For such events, the platform MUST:

- * Assign Confidence Class F
- * Use collector ingestion_time as the best available timestamp
- * Declare the event as having no declared temporal integrity metadata
- * NEVER treat the absence of TIM as equivalent to a Class A timestamp

10.3. Cross-Class Efficiency Metric Computation

When computing efficiency metrics from constituent measurements with different Temporal Confidence Classes, consuming platforms SHOULD compute the metric as a bounded interval rather than a point value. The interval width is determined by the least-confident constituent measurement, and the overall confidence class equals the minimum class across all inputs.

11. Relationship to Existing Standards

11.1. RFC 3164 / RFC 5424 Syslog

TIM is additive to RFC 5424 [RFC5424]. It MAY be carried as structured data using a registered SD-ID. The `TIMESTAMP` field carries `event_time`; the TIM structured data block carries provenance metadata. This document does not modify RFC 3164 or RFC 5424.

11.2. SNMP

SNMP notifications carry `sysUpTime` as the event time reference. TIM MAY be carried as additional variable bindings in SNMP traps and informs, using an OID allocated under a registered enterprise MIB. The `sysUpTime` value remains present for backward compatibility.

11.3. NETCONF/YANG

TIM MAY be represented as a YANG data node and included in NETCONF event notifications alongside the `eventTime` element. A YANG module definition for TIM is left for a companion document.

11.4. IEEE 1588 (PTP)

When a device's `sync_source` is `PTP_IEEE1588`, the `TIM_grandmaster_id` field SHOULD contain the IEEE EUI-64 grandmaster identity from the PTP domain. TIM does not replace or modify PTP; it declares the provenance of timestamps derived from PTP synchronization.

11.5. SMPTE ST 12-1

SMPTE ST 12-1 [SMPTE-ST12] established two principles that directly inform TIM: (1) a timecode label is distinct from actual time; (2) synchronization state should be explicitly declared. TIM formalizes both principles for infrastructure telemetry.

11.6. Google TrueTime

The uncertainty interval model in TIM is directly inspired by Google's TrueTime API [SPANNER], which represents timestamps as `TTinterval{earliest, latest}` guaranteed to contain the absolute event time. TrueTime's central insight -- that acknowledging and bounding uncertainty is superior to asserting false precision -- is the conceptual foundation of the TIM confidence class system.

11.7. OpenTelemetry

Anticipated reviewer question: "Why isn't this just part of OpenTelemetry?"

OpenTelemetry defines what telemetry data looks like. TIM defines how trustworthy the time dimension of that data is. These are distinct concerns. TIM operates orthogonally to observability schemas: it annotates time, not telemetry semantics. A TIM implementation requires no knowledge of whether the event is a log record, a span, a metric, or a physical sensor reading.

OTel defines timestamp fields but provides no mechanism for declaring their quality. TIM is the missing provenance layer for OTel timestamps. TIM fields **SHOULD** be expressed as OTel resource attributes and log record attributes, extending OTel rather than replacing it. OTel resource attribute names for TIM fields are left to a companion specification.

11.8. W3C Trace Context

W3C Trace Context propagates trace and span identifiers but carries no timestamp information. Timestamps in distributed traces are assigned by each service independently. TIM provides what distributed tracing assumes but never defines: a standard for declaring the reliability of those timestamps. TIM **SHOULD** be propagated alongside Trace Context headers.

11.9. IEEE 802.11 -- Wi-Fi Timing Mechanisms

***802.11 TSF:** The 64-bit microsecond counter synchronized across a BSS via beacon frames. TIM implementations using TSF **SHOULD** declare `sync_source: "WIFI_TSF"`. TSF resets **MUST** cause a transition to **FREEWHEEL** until a new anchor is established.

*802.11 FTM:*Enables nanosecond round-trip time measurement without clock synchronization. Devices declaring `sync_source: "WIFI_FTM"` MUST populate `time_domain: "WIFI_FTM_RELATIVE"`. FTM is subject to spoofing attacks [IEEE80211AZ] and TIM security considerations apply with heightened weight.

11.10. 3GPP -- LTE/5G Cellular Timing

*CELLULAR_5G_PRTC:*When a consuming platform directly subscribes to the timing output of a 5G base station's Telecom Grandmaster (T-GM) as a PTP source, `sync_source` SHOULD be declared as `CELLULAR_5G_PRTC`. The T-GM typically achieves +/-100 ns to +/-1.5 us accuracy per [ITU-G8275].

*MOBILE_5G / MOBILE_LTE:*These values apply when a device or OS layer explicitly synchronizes to cellular network timing. Devices that do not expose cellular timing SHOULD declare `NTP_INTERNET` and SHOULD populate the `_advisory` block to indicate that better timing is available but not exposed.

12. Security Considerations

TIM carries metadata about a device's clock synchronization state. Adversaries with write access to a device or its telemetry stream could manipulate TIM fields to misrepresent timestamp confidence, causing monitoring platforms to incorrectly weight events in correlation algorithms. Incorrectly elevated `confidence_class` values represent a high-impact integrity violation and SHOULD be treated as a security event by collecting platforms.

Implementations SHOULD sign TIM fields using a device-resident cryptographic key where hardware security capabilities are available. Consuming platforms SHOULD treat TIM metadata as untrusted input from unverified devices and apply independent validation where possible (e.g., comparing declared `grandmaster_id` against the known PTP domain topology).

The `sequence_token` provides tamper-evidence for event ordering: a gap in the sequence indicates either dropped events or tampering. Collecting platforms SHOULD alert on unexpected sequence gaps from devices operating in LOCKED state.

In multi-classification environments (DOD/DOE), the `domain_id` field identifies isolated time domains. Consuming platforms MUST enforce that events from different `domain_id` values are not cross-correlated without explicit operator authorization, as unauthorized cross-domain correlation may constitute an information security violation.

13. IANA Considerations

This document requests the following IANA actions:

- * Registration of a Syslog Structured Data ID for TIM fields under the IANA Syslog Parameters registry
- * Registration of a YANG module name in the IANA YANG Module Names registry
- * Allocation of an OID arc under the IANA-managed SMI Enterprise Numbers for TIM SNMP MIB objects

For the Informational submission, IANA coordination is not required. This section will be completed upon progression toward Standards Track consideration.

14. References

14.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC3339] Klyne, G. and C. Newman, "Date and Time on the Internet: Timestamps", RFC 3339, July 2002, <<https://www.rfc-editor.org/info/rfc3339>>.
- [RFC5424] Gerhards, R., "The Syslog Protocol", RFC 5424, March 2009, <<https://www.rfc-editor.org/info/rfc5424>>.
- [RFC5905] Mills, D., Martin, J., Burbank, J., and W. Kasch, "Network Time Protocol Version 4", RFC 5905, June 2010, <<https://www.rfc-editor.org/info/rfc5905>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

14.2. Informative References

- [IEEE80211AZ] IEEE, "IEEE Standard for Information Technology -- Amendment 4: Next Generation Positioning", IEEE Std 802.11az-2022, 2022.

[ITU-G8275]

ITU-T, "G.8275.1: Precision time protocol telecom profile for phase/time synchronization", ITU-T Recommendation G.8275.1, 2020.

[OSTP-LTC] White House Office of Science and Technology Policy, "Policy on Celestial Time Standardization in Support of the National Cislunar Science and Technology Strategy", April 2024.

[RFC3164] Lonvick, C., "The BSD Syslog Protocol", RFC 3164, August 2001, <<https://www.rfc-editor.org/info/rfc3164>>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, May 2014, <<https://www.rfc-editor.org/info/rfc7228>>.

[SMPTE-ST12]

SMPTE, "Time and Control Code", SMPTE ST 12-1:2014, 2014.

[SPANNER] Corbett, J., "Spanner: Google's Globally-Distributed Database", OSDI 2012, 2012.

Appendix A. Deployment Scenarios

This appendix provides non-normative guidance for deploying TIM-capable infrastructure in environments with specialized timing constraints.

A.1. Commercial AI Data Center

In a commercial data center with a PTP-capable compute fabric: the PTP grandmaster is provided by compute fabric switches; new management cards implementing TIM declare `sync_state` LOCKED; legacy devices without TIM are classified Class F by the collecting platform; an OCP-TAP Time Server elevates infrastructure card confidence from Class D to Class B; the collecting platform computes efficiency metrics as intervals, with width determined by the least-confident constituent measurement.

A.2. Air-Gapped DOD / DOE Facility

In an air-gapped facility with IRIG-B time distribution: devices with IRIG-B inputs declare `sync_source` IRIG_B, `sync_state` LOCKED, `uncertainty_ns` ~1000 (1 us), `confidence_class` B; the `domain_id` field is populated per classification domain; cross-domain correlation is blocked at the collector. No external connectivity is required.

A.3. Aeronautical Platform

In an airborne computing platform: GPS is the primary source at altitude (Class A); on GPS signal loss in contested airspace, INS provides holdover; uncertainty_ns grows at the INS drift rate during holdover; confidence degrades from A toward E. On GPS recovery, sync_state transitions to RECOVERING, then LOCKED.

A.4. Orbital Installation (LEO)

In a computing installation in low Earth orbit: the time_domain field MUST be set to UTC_LEO_CORRECTED; relativistic_correction_ns_per_day is populated with the correction for the current orbital altitude (approximately +38,000 for ISS orbit); during communication blackout, AUTONOMOUS state with an onboard atomic clock provides continued operation.

A.5. Lunar Installation

In a computing installation on the lunar surface: until LTC is formally defined, the time_domain field is set to FACILITY_AUTONOMOUS; onboard atomic clock ensemble provides the primary time reference; events MUST NOT be directly correlated with UTC_TERRESTRIAL timestamps without applying the ~56 us/day relativistic correction and propagation delay uncertainty; when LTC is defined, time_domain transitions to LTC_LUNAR.

A.6. 5G Edge Computing Site

In a computing deployment co-located with a 5G base station: the T-GM is GNSS-disciplined to +/-100 ns to 1.5 us and serves as Tier 1D TRM source; primary compute node telemetry achieves Class A; power distribution measurements achieve Class B; facility monitoring achieves Class C; the combined metric interval reflects the weakest link at the facility monitoring temporal coherence; on GNSS outage, the T-GM enters holdover and TIM-capable collectors transition to Class E/B while continuing to declare honest uncertainty throughout.

Author's Address

William "Trey" Ackerman
Vertiv Holdings Co.
3414 Governors Dr SW
Huntsville, AL 35805
United States of America
Email: william.ackerman@vertiv.com, treyackerman@hotmail.com