

Independent Submission
Internet-Draft
Intended status: Informational
Expires: 2 November 2026

G.A. Aldana Flores
Aldana Innovations
1 May 2026

Smart Traffic Synchronization Protocol (STSP) Version 1.0
draft-aldana-stsp-00

Abstract

This document defines the Smart Traffic Synchronization Protocol (STSP), version 1.0 -- an open, extensible protocol for real-time coordination of urban traffic signal infrastructure across distributed networks of intersections.

STSP defines a standard message format, node state machine, synchronization algorithm, and inter-node communication model that enables any compliant traffic controller to participate in a federated intelligent network -- regardless of manufacturer, city, or country of deployment.

The key contribution of STSP is the definition of the Infrastructure-to-Infrastructure (I2I) coordination layer -- a communication model between traffic signal nodes that does not exist as an open standard in any currently published specification. Existing standards such as SAE J2735 SPaT/MAP and ETSI ITS-G5 address Vehicle-to-Infrastructure (V2I) communication. STSP addresses the orthogonal problem: how nodes coordinate with each other.

This document is placed in the public domain under CC BY 4.0 with Open Implementation Clause. Any city, government, company, or individual may implement STSP freely, provided that original authorship is attributed in all implementations, documentation, and derivative works as follows: "Implements STSP, designed by Gustavo Angel Aldana Flores (draft-aldana-stsp)."

Author's Note on Priority

This Internet-Draft constitutes the first formal public disclosure of the STSP protocol specification. The conceptual foundation was developed independently by the author beginning approximately in 2015. The first working implementation was completed on 1 May 2026 and validated on ARM64 hardware (Raspberry Pi 4, 8GB RAM) running a full Docker stack including the STSP engine, TimescaleDB, Apache Kafka, Redis, and Grafana. The author asserts perpetual moral rights to this work under Mexican copyright law (Ley Federal del Derecho de Autor, as a Mexican citizen born 21 July 1992) and under the Berne Convention for the Protection of Literary and Artistic Works.

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 2 November 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	4
2. Problem Statement	4
2.1. Fragmentation of Existing Systems	4
2.2. The Technical Opportunity	5
3. Terminology	5
4. Protocol Overview	6
4.1. Node Layer	6

4.2. Corridor Layer	6
4.3. Federation Layer	6
5. Message Format (STSP/1.0)	6
5.1. Standard STSP Message	6
5.2. Compact Form	7
5.3. Phase Enumeration	7
5.4. Node ID Format	8
6. Node State Machine	8
6.1. Fixed Mode	8
6.2. Adaptive Mode	8
6.3. Emergency Override	8
7. Green Wave Synchronization Algorithm	9
7.1. Offset Calculation	9
7.2. Bidirectional Propagation	9
7.3. Priority Resolution	9
8. Adaptive Phase Control	9
8.1. Demand Ratio Policy (Normative Baseline)	9
8.2. Early Termination	9
8.3. Reinforcement Learning Extension (Optional)	10
9. Federated Network Architecture	10
9.1. Data Sovereignty	10
9.2. Federated Learning	10
9.3. Inter-Region Protocol	10
10. Degraded Mode and Failsafe	10
10.1. Degraded Mode Triggers	11
10.2. Degraded Mode Behavior	11
10.3. Recovery	11
11. Security Considerations	11
11.1. Message Authentication	11
11.2. Emergency Override Authorization	12
11.3. Denial of Service	12
11.4. Physical Security	12
12. IANA Considerations	12
13. Reference Implementation	12
14. References	13
14.1. Normative References	13
14.2. Informative References	13
Appendix A: RL State Space (Non-Normative)	14
Appendix B: Glossary	14
Author's Declarations	14
Declaration of Priority	15
Declaration of Open Intent	15
Declaration of Independence	15
Acknowledgements	15
Author's Address	15

1. Introduction

Urban traffic congestion represents one of the most pervasive and quantifiable quality-of-life challenges in modern cities. According to INRIX Global Traffic Scorecard data, the average driver in a major metropolitan area loses between 50 and 150 hours per year to traffic delays. The aggregate economic cost exceeds hundreds of billions of US dollars annually worldwide.

The root cause of this problem is not vehicle density alone -- it is the absence of coordination between traffic signal controllers. The vast majority of traffic signals deployed worldwide operate on fixed-cycle timers programmed once and rarely updated. These controllers have no knowledge of real-time queue conditions, no awareness of neighboring intersections, and no mechanism for inter-city coordination.

Existing adaptive systems (SCATS, SCOOT, Siemens SiTraffic, Kapsch) address portions of this problem but share critical limitations: they are proprietary, city-siloed, non-interoperable, and incapable of federated coordination across administrative boundaries.

This document specifies STSP -- a protocol designed to be for urban mobility infrastructure what TCP/IP is for digital communication: an open, implementable, universally adoptable standard that enables any conforming node to participate in a global intelligent traffic network.

The key insight motivating STSP is the distinction between two communication layers that existing standards conflate or ignore:

- * *V2I (Vehicle-to-Infrastructure):* the layer that informs vehicles of the current signal phase. Standards such as SAE J2735 SPaT/MAP and ETSI ITS-G5 address this layer.
- * *I2I (Infrastructure-to-Infrastructure):* the layer that enables traffic signals to coordinate with each other in real time. No open standard exists for this layer. STSP fills this gap.

2. Problem Statement

2.1. Fragmentation of Existing Systems

Current traffic management systems worldwide share the following structural deficiencies, each of which STSP is designed to resolve:

Proprietary protocols prevent interoperability. A city deploying System A cannot coordinate with an adjacent city running System B, even when both cities share the same physical corridor.

Static timing plans do not respond to real demand. Fixed green cycles optimized for average conditions perform poorly during peak hours, incidents, special events, or adverse weather.

Absence of federation means no city benefits from another city's operational data. Cities independently develop solutions that could be shared. No equivalent of internet routing exists for urban traffic.

2.2. The Technical Opportunity

The convergence of edge computing, 5G connectivity, computer vision, and reinforcement learning creates the technical conditions for a fundamentally different approach. STSP specifies the coordination layer -- the protocol -- that allows these technologies to compose into a coherent, federated, open system.

3. Terminology

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.

Node A single traffic signal controller implementing STSP.

Region A set of nodes within a single administrative boundary (city, district).

Federation A collection of Regions participating in the STSP global network.

Phase The current signal state of a Node. One of: NS_GREEN, NS_YELLOW, EW_GREEN, EW_YELLOW.

Green Wave A coordinated sequence of phase transitions across adjacent Nodes timed to allow vehicles traveling at design speed to encounter consecutive green signals.

Degraded Mode The autonomous fixed-cycle operation a Node reverts to when network connectivity is lost.

I2I Infrastructure-to-Infrastructure. The communication layer

between STSP Nodes. Distinct from V2I (Vehicle-to-Infrastructure) as defined in SAE J2735 and ETSI ITS-G5.

STSP/1.0 The wire format and protocol version defined in this document.

4. Protocol Overview

STSP operates across three hierarchical layers. Each layer functions independently. Higher layers augment but do not replace lower layers.

4.1. Node Layer

Each STSP Node maintains an autonomous state machine governing its phase transitions. The Node broadcasts its current state via the STSP message format at a configurable interval (default: 200ms). A Node MUST be capable of operating indefinitely in Degraded Mode without network connectivity.

4.2. Corridor Layer

Nodes within communication range of one another form a corridor. The Green Wave algorithm (Section 7) coordinates phase transitions across corridors based on vehicle travel time between Nodes.

4.3. Federation Layer

Regions (cities) connect to the STSP federation via a standardized gRPC interface. Federation enables cross-city route optimization, shared model training via Federated Learning, and global traffic event dissemination. Participation in the federation is OPTIONAL for Node and Corridor layer functionality.

5. Message Format (STSP/1.0)

All STSP messages are encoded as JSON objects transmitted over WebSocket [RFC6455] or MQTT (ISO/IEC 20922). Future versions MAY specify binary encoding for bandwidth-constrained deployments.

5.1. Standard STSP Message

The following is the normative STSP/1.0 message schema:

```
{
  "stsp_version":      "1.0",
  "node_id":           "MX-QRO-CONST-042",
  "region_id":         "MX-QRO",
  "grid_row":          2,
  "grid_col":          4,
  "latitude":          20.5928,
  "longitude":         -100.4048,
  "phase":             "NS_GREEN",
  "phase_remaining_ms": 18400,
  "timestamp_utc":     1746123456.789,
  "queue_ns":          3,
  "queue_ew":          0,
  "density_ns":        0.30,
  "density_ew":        0.00,
  "neighbor_ids":      [ "MX-QRO-CONST-041",
                        "MX-QRO-CONST-043" ],
  "green_wave_offset_ms": 0,
  "emergency_override": false,
  "uptime_s":          86432.1,
  "firmware_ver":      "0.1.0",
  "degraded_mode":     false
}
```

5.2. Compact Form

For high-frequency broadcasts (greater than 10Hz), Nodes MAY transmit the compact form, which omits static identity fields:

```
{
  "id":  "MX-QRO-CONST-042",
  "ph":  "NS_GREEN",
  "rm":  18400,
  "qns": 3,
  "qew": 0,
  "ts":  1746123456.789
}
```

5.3. Phase Enumeration

STSP defines four canonical phase values. Implementations MUST NOT use any other values in the "phase" or "ph" fields:

```
NS_GREEN    -- North-South has right of way.
NS_YELLOW   -- Transition from NS_GREEN. Minimum: 3 seconds.
EW_GREEN    -- East-West has right of way.
EW_YELLOW   -- Transition from EW_GREEN. Minimum: 3 seconds.
```

Nodes with non-orthogonal street geometry SHOULD map their local phase names to the nearest STSP canonical phase and document the mapping in their configuration.

5.4. Node ID Format

Node IDs MUST follow the hierarchical format:

{COUNTRY}-{REGION}-{CORRIDOR}-{SEQUENCE}

Example: MX-QRO-CONST-042

MX -- ISO 3166-1 alpha-2 country code
QRO -- Region/city identifier (max 8 chars, uppercase)
CONST -- Corridor identifier (max 8 chars, uppercase)
042 -- Zero-padded sequence number within corridor

6. Node State Machine

Each STSP Node implements a deterministic four-state cyclic machine. The transition function $f(\text{phase})$ is defined as:

NS_GREEN --[timer_expired]--> NS_YELLOW
NS_YELLOW --[timer_expired]--> EW_GREEN
EW_GREEN --[timer_expired]--> EW_YELLOW
EW_YELLOW --[timer_expired]--> NS_GREEN

6.1. Fixed Mode

Phase durations are statically configured. Minimum yellow duration MUST be 3 seconds. Maximum green duration SHOULD NOT exceed 90 seconds. This mode is the mandatory Degraded Mode fallback. Default durations: NS_GREEN 26s, NS_YELLOW 4s, EW_GREEN 26s, EW_YELLOW 4s.

6.2. Adaptive Mode

Phase durations are computed dynamically by the Adaptive Controller (Section 8). MIN_GREEN default: 10 seconds. MAX_GREEN default: 45 seconds.

6.3. Emergency Override

When `emergency_override` is set to TRUE by an authenticated and authorized source, the Node MUST immediately transition to a predefined safe state and hold that state for the override duration. The override MUST expire automatically. Manual reset MUST be available at all times.

7. Green Wave Synchronization Algorithm

7.1. Offset Calculation

When Node N transitions to a green phase, it SHOULD notify its downstream neighbor N+1 to schedule its own green phase transition at time $T + \text{delta}$, where:

$$\text{delta} = D / V_{\text{design}}$$

D -- physical distance between N and N+1, in meters

V_design -- corridor design speed, in meters per second

The notified Node schedules the transition only if it has already served its minimum green duration (MIN_GREEN).

7.2. Bidirectional Propagation

Wave propagation is bidirectional. Nodes SHOULD compute offsets for both upstream and downstream neighbors to accommodate traffic in both directions of a corridor simultaneously.

7.3. Priority Resolution

When a Node receives conflicting wave triggers from two different corridors simultaneously, it MUST prioritize the trigger with the higher queue density on the requesting direction.

8. Adaptive Phase Control

8.1. Demand Ratio Policy (Normative Baseline)

For each green phase, the controller MUST compute:

$$\text{ratio} = \text{queue_active} / (\text{queue_active} + \text{queue_waiting})$$
$$\text{duration} = \text{MIN_GREEN} + \text{ratio} * (\text{MAX_GREEN} - \text{MIN_GREEN})$$

queue_active -- vehicles queued in the currently green direction

queue_waiting -- vehicles queued in the opposing direction

8.2. Early Termination

A green phase MAY be terminated early (but not before MIN_GREEN) if the ratio of queue_waiting to queue_active exceeds SWITCH_IMBALANCE (default: 1.8) AND queue_active is zero or near-zero (less than 1 vehicle) AND MIN_GREEN has been served.

8.3. Reinforcement Learning Extension (Optional)

Implementations MAY replace the demand ratio policy with a full reinforcement learning controller, subject to the following normative constraints:

State: $S = (q_{ns_bucket}, q_{ew_bucket}, current_phase)$
Bucket size: 3 vehicles. Buckets: $\{0,1,2,3,4\}$
Actions: $A = \{EXTEND_GREEN, SWITCH_NOW\}$
Reward: $R = \alpha * throughput - \beta * weighted_wait$
Default: $\alpha=0.8, \beta=1.0$

The RL controller MUST respect all hard bounds defined in Sections 6.1 and 6.2. Minimum yellow duration is non-negotiable.

9. Federated Network Architecture

9.1. Data Sovereignty

Individual vehicle trajectories and personally identifiable information MUST NOT be transmitted in any STSP message or shared across regional boundaries. Only aggregate queue metrics and phase states are federated. This requirement is normative and reflects a core design principle of STSP.

9.2. Federated Learning

Regions participating in the federation MAY share model gradients (not training data) for improving adaptive controllers globally. This approach follows the Federated Learning paradigm and ensures that no city's raw traffic data leaves its administrative boundary.

9.3. Inter-Region Protocol

Regions communicate via gRPC over TLS 1.3. The inter-region message format extends the compact STSP message with additional fields for regional routing and model versioning. Full specification is deferred to STSP/2.0.

10. Degraded Mode and Failsafe

This section specifies MANDATORY failsafe behavior. Implementations that do not conform to this section MUST NOT be deployed in production traffic infrastructure.

10.1. Degraded Mode Triggers

A Node MUST enter Degraded Mode automatically upon any of the following conditions:

- * Network connectivity loss exceeding 30 seconds
- * Regional engine heartbeat timeout
- * Hardware sensor failure detected by watchdog
- * Watchdog timer expiry
- * Explicit authenticated operator command

10.2. Degraded Mode Behavior

In Degraded Mode, the Node MUST:

- * Revert to fixed-cycle operation with pre-configured durations
- * Continue operating indefinitely without network connectivity
- * NOT require operator intervention to enter Degraded Mode
- * Log the trigger event with UTC timestamp and cause code

10.3. Recovery

When connectivity is restored, the Node MAY re-synchronize after a mandatory stabilization period of not less than one complete phase cycle. Re-synchronization MUST be gradual and MUST NOT cause abrupt phase changes visible to road users.

11. Security Considerations

Traffic signal infrastructure is critical public safety infrastructure. Implementations MUST address all threat vectors described in this section.

11.1. Message Authentication

All STSP messages MUST be authenticated using HMAC-SHA256 or equivalent. Unauthenticated messages MUST be silently discarded. Message replay attacks MUST be mitigated via timestamp validation with a maximum tolerance window of 5 seconds.

11.2. Emergency Override Authorization

The `emergency_override` field MUST only be accepted from authenticated sources with explicit override authorization. Unauthorized override attempts MUST be logged and reported to the regional engine immediately.

11.3. Denial of Service

Nodes MUST implement rate limiting on incoming STSP messages. A Node receiving more than 100 messages per second from a single source SHOULD enter protective throttle mode and alert the regional engine.

11.4. Physical Security

Node hardware MUST be deployed in tamper-evident enclosures rated minimum IP67. Physical access MUST require multi-factor authentication. All tamper events MUST be logged with timestamp and reported to the regional engine.

12. IANA Considerations

This document requests the following IANA registrations:

- * WebSocket Subprotocol "stsp": for STSP/1.0 over WebSocket connections per the WebSocket subprotocol registry ([RFC6455]).
- * MQTT Topic Namespace "stsp/": reserved for STSP message routing per MQTT specification (ISO/IEC 20922).
- * A dedicated TCP port number for STSP engine-to-engine communication is requested (specific number pending IANA assignment).

The author requests that IANA reserve the "stsp" identifier across relevant registries to prevent namespace collision with future incompatible uses.

13. Reference Implementation

A complete reference implementation of STSP/1.0 exists and is operational as of the date of this document. The implementation includes:

- * Python 3.12 asyncio engine with WebSocket server (RFC 6455)
- * Intersection state machine (models/intersection.py)

- * Traffic network model -- 5x4 node grid (models/network.py)
- * Adaptive controller with demand ratio and Q-Learning derived policy (algorithms/adaptive.py)
- * Green Wave coordinator with offset calculation (algorithms/green_wave.py)
- * STSP protocol encoder/decoder (protocol/stsp.py)
- * TimescaleDB schema for time-series metrics persistence
- * Docker Compose stack: engine + Kafka KRaft + Redis + Grafana
- * REST API: /health, /state, /stats, /mode
- * Validated on ARM64 (Raspberry Pi 4, 8GB RAM) and Apple M3

Repository: <https://github.com/smartflow-stsp/reference> [Publication pending]

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>>.
- [RFC6455] Fette, I. and A. Melnikov, "The WebSocket Protocol", RFC 6455, December 2011, <<https://www.rfc-editor.org/info/rfc6455>>.
- [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

- [ETSI-ITS] ETSI, "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications", ETSI EN 302 665, 2019.
- [McMahan2017] McMahan, H.B., "Communication-Efficient Learning of Deep Networks from Decentralized Data", AISTATS 2017, 2017.

[SAE-J2735]

SAE International, "Dedicated Short Range Communications (DSRC) Message Set Dictionary", SAE Standard J2735, 2020.

Appendix A: RL State Space (Non-Normative)

State: $S = (q_{ns_bucket}, q_{ew_bucket}, current_phase)$

Actions: $A = \{EXTEND_GREEN, SWITCH_NOW\}$

Reward: $R = 0.8 * throughput - 1.0 * weighted_wait$

Bucket mapping:

0 -- 0 vehicles	(empty)
1 -- 1-3 vehicles	(light)
2 -- 4-6 vehicles	(moderate)
3 -- 7-9 vehicles	(heavy)
4 -- 10+ vehicles	(saturated)

Appendix B: Glossary

IETF Internet Engineering Task Force

I2I Infrastructure-to-Infrastructure communication

V2I Vehicle-to-Infrastructure communication

STSP Smart Traffic Synchronization Protocol

SPaT Signal Phase and Timing (SAE J2735)

SCATS Sydney Coordinated Adaptive Traffic System

SCOOT Split Cycle Offset Optimization Technique

RL Reinforcement Learning

HMAC Hash-based Message Authentication Code

gRPC Google Remote Procedure Call (open protocol)

OT Operational Technology (industrial control systems)

HSM Hardware Security Module

Author's Declarations

Declaration of Priority

I, Gustavo Angel Aldana Flores, hereby declare that I am the sole inventor and author of the Smart Traffic Synchronization Protocol (STSP) as specified in this document. The conceptual foundation of this protocol was developed independently beginning approximately in 2015. The first working implementation was completed on 1 May 2026 and validated on physical ARM64 hardware (Raspberry Pi 4, 8GB RAM) running a full production stack. This Internet-Draft constitutes the first formal public disclosure of the STSP protocol specification.

Declaration of Open Intent

I hereby irrevocably dedicate the STSP protocol specification to the global commons under CC BY 4.0 with Open Implementation Clause. My intent is that this protocol serve as open infrastructure for the benefit of cities, governments, and citizens worldwide -- without commercial enclosure -- while preserving permanent attribution to its author.

Declaration of Independence

This protocol was developed independently, without funding from any government, corporation, or institution. No entity holds rights over this specification other than those expressly granted in the copyright notice. The author is not affiliated with any traffic signal manufacturer, smart city platform provider, or standards body at the time of this writing.

Acknowledgements

The author developed this protocol independently. This document was first drafted on 1 May 2026 in Peachtree City, Georgia, USA.

Author's Address

Gustavo Angel Aldana Flores
Aldana Innovations
Peachtree City, Georgia
United States of America
Email: g.aldana@aldanainnovations.com
URI: <https://github.com/smartflow-stsp>