

Independent Submission
Internet-Draft
Intended status: Informational
Expires: 16 April 2026

J. Tansey
Independent
J. Tansey
Cisco
13 October 2025

Adaptive Layered Voice Container (ALVC) for Constrained Networks
draft-joetansey-alvc-codec-02

Abstract

This document specifies the Adaptive Layered Voice Container (ALVC), a codec-agnostic framing and metadata container that enables progressive voice delivery in constrained and lossy networks. ALVC defines a Base layer that is intelligible on its own at sub-kilobit rates, and optional Enhancement layers that improve quality when additional capacity is available. The container supports store-and-forward operation, progressive enhancement, unequal error protection signaling, and receiver behavior for seamless splice-and-improve playback. ALVC does not define a new speech coding algorithm; it multiplexes existing voice coders within a layered container.

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 16 April 2026.

Copyright Notice

Copyright (c) 2025 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. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Requirements Language	2
3. Container Overview	2
4. ALVC Frame Format	3
5. Receiver Behavior	3
6. Sender Behavior	4
7. Examples	4
8. Security Considerations	4
9. IANA Considerations	4
10. Changes since -01	4
11. References	4
11.1. Normative References	4
11.2. Informative References	5
Authors' Addresses	5

1. Introduction

Low-power and highly constrained links (for example LPWANs) cannot sustain traditional conversational streaming. ALVC provides a simple container for layered audio so that a small Base layer is delivered first for intelligibility, with optional Enhancements sent later. The container is transport-agnostic and can be mapped to different networks; a companion document describes a SCHC mapping for LPWAN [ALVC-SCHC].

2. 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.

3. Container Overview

ALVC multiplexes one Base stream and zero or more Enhancement streams across time-aligned windows (for example 20 ms or 40 ms). Each window is a self-contained set of frames that can be played as soon as the Base frame is available. Enhancements, when present, refine the audio for the same window.

ALVC is codec-agnostic: the Base frame is typically produced by a very low bitrate coder (for example Codec2), while Enhancements may be produced by a higher-fidelity coder (for example LPCNet or Opus) configured to refine the same speech segment. The precise codec choices are outside the scope of this document.

4. ALVC Frame Format

Each ALVC frame carries structured metadata followed by codec payload. Fields:

`ts` Unsigned timestamp of the start of the covered window, in milliseconds since an agreed origin.

`win_dur_ms` Window duration in milliseconds.

`layer_id` 0 for Base; positive integers for Enhancements, where higher numbers indicate higher quality.

`codec_id` Identifier indicating which codec produced the payload (for example a registry or vendor space).

`frag_seq`, `frag_total` Optional per-frame fragmentation indices when a frame is split over multiple transport units.

`flags` Bit flags including: `end_of_clip`, `parity_present`, `auth_present`.

`payload_len` Length of codec payload in bytes.

`payload` Opaque codec bytes for this layer and window.

The Base layer for a window MUST be decodable on its own. Enhancement layers for a window MUST refine, but MUST NOT be required for basic intelligibility. Receivers MUST render the best available layer for each window as data arrives.

5. Receiver Behavior

Receivers maintain per-window state. On arrival of a Base frame for window N, the window becomes immediately playable. If one or more Enhancements for window N later arrive, the receiver SHOULD splice-in the improved audio without glitch, using a crossfade or codec-specific switch. Missing or invalid Enhancements MUST NOT block Base playback. Implementations SHOULD expose progress to the application layer, such as "Base-only", "Enhanced L1", "Enhanced L2".

6. Sender Behavior

Senders SHOULD prioritize timely delivery of upcoming Base windows to sustain continuous intelligible playback, then transmit earliest-missing Enhancements for already-playable windows. Senders MAY include parity or forward error correction and indicate this with the `parity_present` flag. Transport-specific scheduling (for example SCHC fragmentation or channel hopping) is out of scope here but is discussed in the SCHC mapping document [ALVC-SCHC].

7. Examples

**Example timing:* A clip is encoded into 20 ms windows. The Base stream averages about 0.5 kb/s. Enhancement L1 averages 1.0 kb/s. During constrained periods, only Base is delivered; when capacity improves, the sender backfills L1 for the earliest windows missing enhancement.

8. Security Considerations

ALVC frames SHOULD be protected end-to-end using an authenticated encryption scheme. Integrity failures in Enhancement frames MUST NOT affect Base playback; such frames are discarded. Metadata should be minimized consistent with receiver needs.

9. IANA Considerations

This document does not create any new IANA registries. If a public registry of ALVC codec identifiers is later desired, it can be defined in a follow-up document.

10. Changes since -01

Added BCP 14 requirements-language, clarified container scope and receiver behavior, added explicit field list and examples, aligned text with SCHC mapping companion, ASCII cleanup.

11. References

11.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.

11.2. Informative References

[ALVC-SCHC]

Tansey, J., "Carrying ALVC over LPWAN using SCHC Fragmentation and Priorities", 2025, <<https://datatracker.ietf.org/doc/draft-joetansey-alvc-schc-lpwan/>>.

Authors' Addresses

Joe Tansey
Independent
Email: joe.tansey@gmail.com

Joe Tansey
Cisco
Email: joetanse@cisco.com