# An Overview of ATSC 3.0 STL Standard

Daniel Pedroso Neri

Vinicius Yukio Shinzato Tricoli

George Henrique M. Garcia de Oliveira

Cristiano Akamine Digital

CITE THIS ARTICLE

Daniel Pedroso Neri, Vinicius Yukio Shinzato Tricoli, George Henrique M. Garcia de Oliveira, Cristiano Akamine Digital; 2025. An Overview of ATSC 3.0 STL Standard. SET INTERNATIONAL JOURNAL OF BROADCAST ENGINEERING. ISSN Print: 2446-9246 ISSN Online: 2446-9432, doi: 10.18580/setijbe.2025.8 web link: https://dx.doi.org/10.18580/setijbe.2025.8

COPYRIGHT This work is made available under the Creative Commos - 4.0 International Licence. Reproduction in whole or in part is permitted provided the source is acknowledged

© 2025 SET – Brazilian Society of Television Engineering | ISSN (Print): 2446-9246 | ISSN (Online): 2446-9432

# An Overview of ATSC 3.0 STL Standard

Daniel Pedroso Neri, Digital TV Research Laboratory Mackenzie Presbyterian University
Vinicius Yukio Shinzato Tricoli, Digital TV Research Laboratory Mackenzie Presbyterian University
George Henrique M. Garcia de Oliveira, Digital TV Research Laboratory Mackenzie Presbyterian University
Cristiano Akamine Digital, TV Research Laboratory Mackenzie Presbyterian University

**Abstract—The Studio-to-Transmitter Link (STL) is a critical subsystem within the Advanced Television Systems Committee (ATSC) 3.0 broadcasting architecture, serving as the bridge between studio originated content and the physical transmission infrastructure. It handles the encapsulation, timing, synchroniza- tion, and reliable delivery of essential data streams, including Baseband Packets (BBPs), Preamble information, and Timing and Management (T&M) data from the Broadcast Gateway to one or more transmitters. To support robust and synchronized emissions, the STL employs transport protocols derived from the Common Tunneling Protocol (CTP), like Data Source Transport Protocol (DSTP), ATSC Link-Layer Protocol Transport Protocol (ALPTP) and Studio-to-Transmitter Link Transport Protocol (STLTP), these protocols ensure that data is delivered in the correct order and at the correct time, even across complex networks.**

**Index Terms—ATSC 3.0, STL**

## I. INTRODUCTION

The history of Television (TV) broadcasting in Brazil has been marked by continuous technological evolution, driven by the pursuit of improved audiovisual quality, spectrum efficiency, and user experience. The first TV broadcasts in the country began in 1950, with black-and-white images and monophonic sound, characterizing what is conventionally known as TV 1.0. Over the following decades, improvements such as color transmission, stereo sound, and closed captions culminated in TV 1.5, maintaining backward compatibility while enhancing the viewing experience.

A major leap occurred in 2006, when the Brazilian govern- ment, supported by the Brazilian Digital Terrestrial Television System (SBTVD-T) Forum, selected the Integrated Services Digital Broadcasting Terrestrial - Version B (ISDB-TB) stan- dard for its first generation of Digital Terrestrial Television Broadcasting (DTTB) system. This transition to digital broad- casting, known as TV 2.0, introduced high-definition video, surround sound, mobile reception, and interactive services, the system began broadcasting in 2007. Since its implementation, the SBTVD-T system updated around 2020, incorporating optional features such as High Dynamic Range (HDR) video and immersive audio, referred to as TV 2.5.

Although internet and streaming platforms have become increasingly popular, television continues to play a central role in Brazilian households. Recognizing the rapid technological advancements in broadcasting and broadband, as well as the increasing demand for personalized, high-quality content, the SBTVD-T Forum, published a Call for Proposals (CfP) de- tailing the requirements for Brazil's next generation of DTTB system [1], initiating the TV 3.0 project, also called DTV+. This new generation of DTTB aims to deliver significant improvements in audiovisual quality, spectrum efficiency, and service integration, incorporating technologies such as Ultra High Definition (UHD), 4K and 8K, immersive audio, inter- active applications, personalized content, targeted advertising, geographic-based content segmentation, and a seamless inte- gration between broadcast and broadband networks through an IP-based infrastructure, all while working with an object oriented structure.

In 2024, after two test phases, Advanced Television Systems Committee (ATSC) 3.0 was chosen as the technology for the physical layer of TV 3.0 alongside Real-time Object delivery over Unidirectional Transport (ROUTE)/Dynamic Adaptive Streaming over HTTP (DASH), that in 2021 was chosen as the technology for Transport Layer (TL).

This paper aims to provide an overview of ATSC Studio-to-Transmitter Link (STL) standard and its integration with the ATSC Link-Layer Protocol (ALP), Signaling, and Physical Layer (PL) standards, and it will be organized as the following, II describes the TL for the TV 3.0 project, Section III explains the workings of the Common Tunneling Protocol (CTP), it will also contain details of the other transport protocol that use CTP inside the ATSC standards, Section IV describes how STL works, Section V presents the conclusion of the paper.

## II. TRANSPORT LAYER IN TV 3.0 (DTV+)

The technology adopted for the TV 3.0 TL was the ROUTE/DASH transport method, with modifications and extensions from the ATSC 3.0 system [3]. For Brazil, the use of this technology marks the transition to an IP-based DTTB system.

Adopted by many major streaming services, like Netflix and YouTube [4], DASH became a popular solution for delivering audiovisual content over broadband networks. Designed by Moving Picture Experts Group (MPEG), DASH enables the transmission of multiple streams through a server-based ar- chitecture, providing flexibility and scalability for on-demand content delivery. To adapt this technology for broadcast ap- plications, DASH was integrated with ROUTE, a protocol developed by the Internet Engineering Task Force (IETF) [5], making it possible to direct a stream with a desired fixed bit rate for Over-The-Air (OTA) transmission.

ROUTE/DASH carries various types of services, such as Link Mapping Table (LMT), Service List Table (SLT), and Service Layer Signaling (SLS), as well as audio and video segments. To ensure efficient and organized delivery, ALP is used as an abstraction layer, encapsulating and standardizing this information so it can be transmitted and processed uni- formly by the receiver.

Each ALP packet is composed of a ALP header and payload, the function of the ALP header is to convey

This open access article is distributed under a Creative Commos Attribution (CC-BY) license.
https://set.org.br/ijbe/ doi:10.18580/setijbe.2025.8 Web Link: https://dx.doi.org/10.18580/setijbe.2025.8

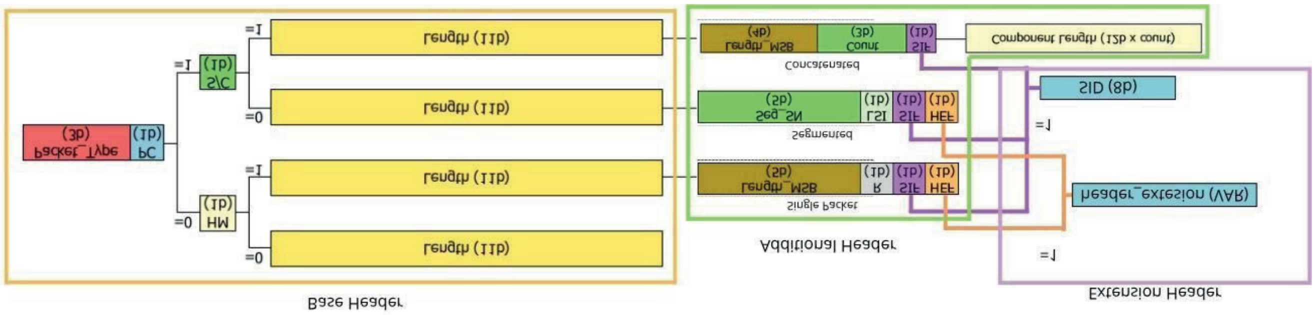



Fig. 1. ALP Header Structure [2]

TABLE I
Code Values for packet type [2]

| packet type Value | Meaning |
|---|---|
| 000 | IPv4 Packet |
| 001 | Reserved |
| 010 | Compressed IP Packet |
| 011 | Reserved |
| 100 | Link-layer Signaling Packet |
| 101 | Reserved |
| 110 | Packet Type Extension |
| 111 | MPEG-2 Transport Stream |

TABLE II
SYNTAX OF ADDITIONAL HEADER FOR CONCATENATION [2]

| Syntax | No. of Bits | Format |
|---|---|---|
| concatenation hdr(){ | | |
| length MSB | 4 | uimsbf |
| count | 3 | uimsbf |
| SIF | 1 | bslbf |
| for (i=0; i<count+1; i++){ component length | 12 | uimsbf |
| } | | |
| if ((count & 1) == 0){ | | |
| stuffing bits | 4 | ‘0000’ |
| } | | |
| } | | |

essential details about the structure of the payload, including the type of information it contains, its length, and whether the data is complete, concatenated or segmented. Figure 1 shows how the ALP header is constructed, the **Packet Type** field details which type of packet the payload is made of, the Table I describes what options are available for this field. The **payload configuration (PC)** field describes the payload configuration, where a value of ‘0’ indicates that the ALP carries a single packet and a value of ‘1’ indicates the ALP carries either a segment of a larger packet, or multiple packets inside its payload [2].

When the **PC** field is equal to ‘0’, the next field represents the **header_mode (HM)**, a value of ‘0’ in this field indicates that there is no Additional Header for single packet, and therefore indicates that this packet size is less than 2048 bytes, where a value of ‘1’ indicates that the Additional Header for single packet is present, and that the size of the length of this payload for this ALP is larger than 2048 bytes. The Additional Header for single packet has a length of 1 byte, and contains a length MSB 5-bit field, this field need to be concatenated with the **Length** field in order to obtain the total length of the ALP payload. **The Sub-stream Identifier Flag (SIF),** and **Header Extension Flag (HEF)**, fields are 1-bit length fields that signal the presence of optional data. The SIF, when set to 1, indicates that a sub stream identification field follows, allowing differentiation between sub-streams within the same ALP flow. The **HEF**, when set to 1, signals that a header extension is present for additional information or future enhancements. When either **SIF** or **HEF** flags are 0, the corresponding optional field is absent, keeping the header compact [2].

When the **PC** field is equal to ‘1’, the **Segmentation/ Concatenation (S/C)**, segmentation or concatenation, field appears, a value of ‘0’ indicates that the payload carries a segment of an input packet and a Additional Header for segmentation is present, while a value of ‘1’ indicates that the payload carries more than one complete input packet a

TABLE III
RTP Header Field Definitions for CTP [6]

| Syntax | No. of Bits | Format |
|---|---|---|
| RTP Fixed Header () { | | |
| version (V) | 2 | ‘10’ |
| padding (P) | 1 | bslbf |
| extension (X) | 1 | bslbf |
| CSRC count (CC) | 4 | ‘0000’ |
| marker (M) | 1 | bslbf |
| payload type (PT) | 7 | uimsbf |
| sequence number | 16 | uimsbf |
| if (PT == DSTP \|\| PT ==ALPTP) { | | |
| timestamp min () | 32 | Table 6.3 of [6] |
| } | | |
| else if (PT == STLTP) { | | |
| timestamp () | 32 | Table 9.2 of [6] |
| } | | |
| else { | | |
| reserved | 32 | for(i=0, i<32, i++) ‘0’ |
| } | | |
| protocol version | 2 | uimsbf |
| if (PT == STLTP) { | | |
| redundacy | 2 | uimsbf |
| number of channels | 2 | uimsbf |
| reserved | 10 | for(i=0, i<10, i++) ‘0’ |
| } | | |
| else { | | |
| reserved | 14 | for(i=0, i<14, i++) ‘0’ |
| } | | |
| packet offset | 16 | uimsbf |
| } | | |

Additional Header for concatenation is present. The segmentation Additional Header is made of four fields, the **segment sequence number (Seg SN)**, or segment sequence number, a 5 bit field that indicates the order of the corresponding segment, the packet that carries the first segment will have this field as 0x0, and its value will increase by one on each additional segment. The **LSI** field is a bit that indicates the last segment of the packet when its value is ‘1’, otherwise the value is ‘0’. The final two fields

This open access article is distributed under a Creative Commos Attribution (CC-BY) license.

are **SIF** and **HEF** fields, with the same functionality as the ones on the Additional Header for single packet. The concatenation Additional Header is described by Table II, similar to the single packet Additional Header, the **length** MSB needs to be concatenated to the length field of the Base Header structure to obtain the total length of the payload, the count field indicates the number of packets included in the ALP packet, the **SIF** field is the same as the other Additional Packets, the **component length** field indicates the length of each individ- ual packet inside the payload, with the exception of the last component packet, if the the number of **component length** fields is odd, four stuffing bits will be added at the end of the header to pad the header [2].

In cases of a Link-layer Signaling Packet, an additional header for signaling information is added at the end of the ALP header.

## III. COMMON TUNNELING PROTOCOL

ATSC 3.0 utilizes many transport protocols, among then ATSC Link-Layer Protocol Transport Protocol (ALPTP), Data Source Transport Protocol (DSTP) and Studio-to-Transmitter Link Transport Protocol (STLTP), this transport protocols depend on a way to encapsulate its information in a manner that the transmitter and receiver understand. In order to solve this demand, a CTP is employed. The CTP encapsulates and abstracts the details of the many protocols inside a IP/UDP/RTP stream. This stream is divided into tunnel packets and tunneled packets, also called outer and inner packets, respectively [6]. Because UDP does not provide a way to ensure that the packets arrive in order, a modified RTP header, described in Table III, is used as one of the tunnel headers.

The **marker** field is what indicates if a new packet is starting within the tunnel payload, this case is indicated by a value of ‘1’ in the marker field, otherwise the value is ‘0’, indicating that the tunnel is, in its entirety, a continuation of a previous packet. Another key change in the RTP header, is the **packet offset** field, which, alongside other fields, are derived of a repurposed **SSRC ID**, the packet offset defines, when marker is equal to ‘1’, the remaining length of the previous packet, serving as a pointer to the beginning of the next packet. The **payload type** field is also useful, informing, in the case of an tunnel header, which transport protocol is being used and, in case of a tunneled header, which packet type is encapsulated.

The CTP tunnels are formed by encapsulated, or tunneled, packets and support both segmentation and concatenation mechanisms. Segmentation is applied when a tunneled packet is too large to fit entirely within the available space in a tunnel packet, while concatenation occurs when the tunneled packet length is smaller than the available length of the tunnel, allowing another packet to be partially or, in cases where the available length is greater than the packet size, entirely encapsulated in the tunnel. As a result, CTP tunnels can carry more than one tunneled packet or, in some cases, may be, in its entirety, a continuation of a previous tunneled packet [6]. Cases where a new tunneled packet starts within the tunnel are signaled inside the RTP header through the marker field and the number of tunneled packets inside a given tunnel can be determined by the lengths of each tunneled packet.

## IV. STUDIO-TO-TRANSMITTER LINK

Inside the STL there are four tunneled packet types, Pream- ble Packets, Timing and Management (T&M) Packets, Base band Packets (BBPs) and Security Packets. The Preamble and T&M packets carry synchronization and control information, essential for proper transmitter operation. To enhance the reliability of their delivery, the STL supports the use of Majority Logic.

The Majority Logic dictates that Preamble and T&M be sent before the BBP to which they refer. This logic also allows for repetition of these structures an N amount of times before the BBP, this approach ensures that even if part of the STL supply is interrupted, redundancy increases the chances that essential synchronization information will still reach the transmitters [6].

TABLE IV
Preamble Structure [6]

| Syntax | No. of Bits | Format |
|---|---|---|
| Preamble Payload() { | | |
| length | 16 | uimsbf |
| L1 Basic signaling () | 200 | Table 9.2 of [7] |
| L1 Detail signaling () | var | Table 9.8 of [7] |
| crc16 | 16 | uimsbf |
| } | | |

### A. Preamble Data

The Preamble stream contains the complete description of the transmitter configuration, including the processing param- eters. In order for the transmitters to be able to correctly apply these configurations, the corresponding Preamble must be sent by the Scheduler at least one physical layer frame before the frame it describes. The Preamble Structure is identified by the value 0x4D in the **payload type** of the tunneled packet RTP header, and its composition is as described by Table IV.

The **length** field of the preamble structure is a field containing the combined length of the **L1 Basic signaling** and **L1 Detail signaling**. The L1 Basic signaling is a fixed size structure, described by the Table 9.2 of [7], and contains the system’s most fundamental sig- nals, together with the parameters required to decode **L1 Detail signaling**. While **L1 Basic signaling** is a fixed size structure, **L1 Detail signaling** size varies according to the number of Physical Layer Pipes (PLPs), subframes and other configured parameters. It contains, in detail, signalization required to decode the transmitted data, and it is described in Table 9.8 of [7].

### B. Timing and Management

T&M information is needed to control the synchronization of the transmitter, bootstrap emission and similar transmission tasks. This information is fed into a T&M generator, following the scheduler’s instructions. The T&M structure is identified by the value 0x4C in the **payload type** of the tunneled packet RTP header, and its composition is as described by Table 9.3 of [6].

The T&M stream is distinct from content data streams, such as BBP Packets, it is encapsulated into its own packet and transmitted along with the Preamble and BBP packets

This open access article is distributed under a Creative Commos Attribution (CC-BY) license.
https://set.org.br/ijbe/ doi:10.18580/setijbe.2025.8 Web Link: https://dx.doi.org/10.18580/setijbe.2025.8

through the STLTP and is delivered to the transmitters before the data to which it refers. This early delivery ensures that the transmitter can be configured correctly in time for emission [6].

### *C. Baseband Packets*

BBPs are fundamental data units in the ATSC 3.0 transmission system that carry data of the TL to be transmitted by the physical layer. Its packets are identifiable by the value 0x4E in the **payload type** of the tunneled packet RTP header. When a new BBP is starting, after the IP/UDP/RTP headers, there is a BBP header, which is described in details at the PL standard [7] and depicted by Figure 2, inside this header there are information regarding the full structure of the BBP, the three main fields of the header are pointer, extension type and extension length.

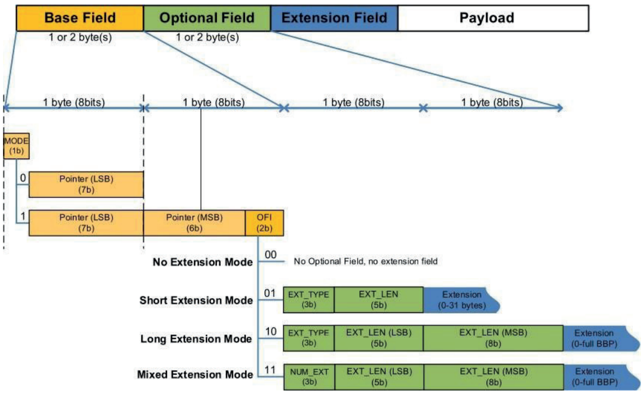


Fig. 2. BBP Header Structure [7]

The **pointer** field tells where does the next ALP header starts, in the case where the **MODE** bit field is equal to '1' the field is separated between Least Significant Bit (LSB) and Most Significant Bit (MSB). In this case the complete pointer field is a concatenation of the Pointer (MSB) field and the Pointer (LSB) field, the same is true in cases where the optional field is present and the **OFI** field is either '10' or '11'. In these cases the **extension length** field is divided between EXT LEN (LSB) and EXT LEN (MSB) [7].

BBPs are created in the Broadcast Gateway by concatenating multiple ALP packets, which can vary in size, in its payload as efficiently as possible. These BBP packets are then assigned to specific PLPs, logical channels within the physical layer that can have different robustness and capacity configurations. Each PLP receives its own stream of BBPs, at the transmitter, BBPs are buffered until the correct emission time and then modulated into the waveform according to the configuration defined by the Preamble and T&M information.

### *D. Security*

Security packets within the STL are carried through a dedicated Security Data Stream encapsulated by the CTP. These packets contain cryptographic metadata, authentication tags, and key references that enable transmitters to verify the authenticity and integrity of the control and content data they receive. Each packet is signed by a Signing Entity using private keys, and verified by an Authenticating Entity at the transmitter side using the corresponding public key. The cryptographic system relies on standards such as Advanced Encryption Standard (AES), Galois/Counter Mode-based MAC (GMAC), and Elliptic Curve Digital Signature Algorithm (ECDSA), and follows a public/private key infrastructure with secure key exchange mechanisms. Security packets must be delivered in advance of the data they authenticate, requiring proper scheduling and buffering

## V. CONCLUSION

The paper provided an overview of the ATSC 3.0 STL standard and its integration with the ALP, Signaling, and PL standards of the ATSC 3.0 system.

The DTV+ TL adopted the ATSC 3.0 system with adapta- tions and extensions, described in the Brazilian PL [8] and TL [9] standards, in order to support the needs of the Brazilian system.

## References

[1] Brazilian Digital Terrestrial TV System Forum, "Call for Proposals: TV 3.0 Project," SBTVD-T Forum, Tech. Rep., July 2020. [Online]. Available: https://forumsbtvd.org.br/wp-content/uploads/2020/ 07/SBTVDTV-3-0-CfP.pdf

[2] A/330:2024-04, "ATSC standard: Link-layer protocol," Advanced Television Sistem Committee, Tech. Rep., 2024.

[3] A/331:2024-04, "ATSC standard: Signaling, delivery, synchronization, and error protection," Advanced Television Sistem Committee, Tech. Rep., 2024.

[4] Bitmovin, "Why YouTube & Netflix use MPEG-DASH in HTML5," February 2015. [Online]. Available: https://bitmovin.com/ blog/mpeg-dash-youtube-netflix-html5/

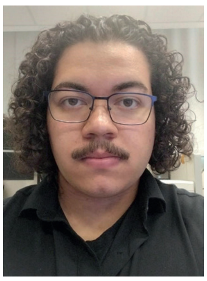
[5] W. Zia, T. Stockhammer, L. Chaponniere, G. Mandyam, and M. Luby, "Real-Time Transport Object Delivery over Unidirectional Transport (ROUTE)," RFC 9223, Apr. 2022. [Online]. Available: https://www.rfc-editor.org/info/rfc9223

[6] A/324:2025-04, "ATSC standard: Scheduler / studio to transmitter link," Advanced Television Sistem Committee, Tech. Rep., 2025.

[7] A/322:2024-09, "ATSC standard: Physical layer protocol," Advanced Tele- vision Sistem Committee, Tech. Rep., 2024.

[8] BRAZILIAN STANDARD: TV 3.0 – Over-the-air physical layer (NBR 25601), ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, Brasília, D.F, Dezembro 2024.

[9] BRAZILIAN STANDARD: TV 3.0 – Transport layer (NBR 25602), ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS, Brasília, D.F, Dezembro 2024.



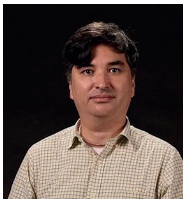
**Daniel Pedroso Neri** received his undergraduate degree in Electrical Engineering with an emphasis on Electronics and Telecommunications from Mackenzie Presbyterian University (2023) and his M.Sc. in Electrical and Computer Engineering from Mackenzie Presbyterian University (2025), where he is currently a Ph.D. student. He is currently an Researcher and participates in projects conducted by the Digital TV Laboratory of the School of Engineering at Mackenzie Presbyterian University.

He has participated in the national and international publication of articles relating to telecommunications and broadcasting. He has also participated as a researcher in the Physical and Transport Layers of the TV 3.0 project. https://orcid.org/0009-0007-6998-0077

This open access article is distributed under a Creative Commos Attribution (CC-BY) license.

**Vinicius Yukio Shinzato Tricoli** has a degree in Electrical Engineering, with an emphasis on Electronics, Telecommunications and Automation from Mackenzie Presbyterian University (2025) and is currently pursuing his M.Sc. in Electrical and Com- puter Engineering at Mackenzie Presbyterian Uni- versity. He is currently a Researcher and participates in projects conducted by the Digital TV Laboratory of the School of Engineering at Mackenzie Presbyterian University https://orcid.org/0009-0002-3200-5490

**George Henrique Maranhão Garcia de Oliveira** received his Ph.D. and M.Sc. in Electrical and Computer Engineering from Mackenzie Presbyterian University in 2025 and 2017, respectively. Also received bachelor's degree in Electrical Engineering with an emphasis on Electronics and Telecommunications from Mackenzie Presbyterian University (2015) and technical degree in telecommunications from Liceu de Artes e Of´ıcios de Sa˜o Paulo (2008). Since 2016, works as an engineer and researcher at the Digital TV Laboratory of the School of Engineering at Mackenzie Presbyterian University. Has experience and pub- lished articles in the field of Electrical Engineering, with an emphasis on the physical and transport layers of digital terrestrial broadcasting systems, digital communication, channel coding, and software-defined radio. Reviewer for IEEE Transactions on Broadcasting and SET International Journal of Broadcast Engineering. His line of research focuses on the TV 3.0 transport layer, involving DASH encoding, the ROUTE/DASH transport method, the ALP link layer, the STLTP interface, and the LCI/GSL technique. https://orcid.org/0000-0001-8211-9045



**Cristiano Akamine** received the Ph.D. degree in electrical engineering from the State University of Campinas, Brazil, in 2011. He is a professor and researcher at the Digital TV Research Laboratory of Mackenzie Presbyterian University and at Mack Graphe – the Mackenzie Institute for Research in Graphene and Nanotechnologies. He completed a research internship at the NHK Science and Technology Research Laboratories (STRL) and served as a Visiting Specialist Professor at the Faculty of Technology of the University of Campinas (Unicamp).

He currently works in the Department of Electrical Engineering and in the Postgraduate Program in Electrical and Computer Engineering (PPGEEC) at Mackenzie Presbyterian University, where he also heads the Digital TV Research Laboratory. He serves as a board member and coordinator of the Physical and Transport Layer Working Group of the Brazilian Digital Terrestrial Television System (SBTVD) Forum, specifically contributing to the development of TV 3.0 (DTV+), is a member of the Brazilian Society of Television Engineering (SET), an AdCom member of the IEEE Broadcast Technology Society (BTS), and an Associate Editor of the IEEE Transactions on Broadcasting. He holds several patents and has published extensively in his field. His research spans multiple areas within electrical engineering, with emphasis on digital television, digital communications, channel coding, embedded systems, reconfigurable logic, and software-defined radio.

https://orcid.org/0000-0002-3161-4668

Received in 2025-07-14 | Approved in 2025-08-04

This open access article is distributed under a Creative Commos Attribution (CC-BY) license.