

Technical White Paper on Network Video Interface (NVI)

(2024 Edition)

China Media Group

2024.9

Preface

Traditional broadcasting and television systems use coaxial cables and Serial Digital Interface (SDI) for video transmission. However, this approach has several drawbacks, including complex cabling, high maintenance difficulty, and limited scalability. With the continuous expansion of Ultra High Definition (UHD) technology in the industry, the deployment based on the SDI architecture has become more complex and costly. To meet the development trend of converged media, the broadcasting and television industry is transitioning towards IP-based systems to achieve a more flexible and integrated platform. IP transmission technology is highly favored for its high reliability, high quality, and low complexity.

At present, media convergence is accelerating and deepening. The increasing richness of scenarios such as collaborative production between large screens and new media platforms, remote production, and cloud-based collaboration has led to the emergence of various IP-based transmission technologies tailored to different industry needs, such as ST 2110, SRT (Secure Reliable Transport), and NDI (Network Device Interface). ST 2110 is mainly aimed at localized professional broadcasting and television production, with relatively high costs for construction, maintenance, and upgrades. SRT supports high-quality point-to-point transmission over the Internet but lacks the professional production functions required by broadcasting and television, such as multicast transmission. NDI performs well in both local area network (LAN) and Internet transmission, but its closed-source nature means it does not support customized service development.

To promote the lightweight production of UHD signals and meet the needs of converged media and cloud-based deployment, China Media Group has jointly developed the Network Video Interface (NVI) technology with related enterprises. NVI integrates the two core functions of transmission and video encoding/decoding, providing a simple and efficient solution for the transmission of audio and video data over LAN and Internet. NVI technology offers two transmission modes, NVI-HF (light compression) and NVI-HC (high compression), to ensure high-quality and low-latency video transmission under different network conditions. In addition, NVI

technology supports remote control of PTZ (Pan-Tilt-Zoom) cameras and can carry important additional information such as Tally and Alpha during video transmission, offering more possibilities for different application scenarios such as professional media broadcasting and Internet live streaming. NVI technology uses standard Ethernet connections and supports automatic registration and plug-and-play of connected devices, effectively reducing the costs of system construction, deployment, and operation and maintenance.

By integrating open-source protocols, low-latency transmission, support for multicast functionality, flexible data encapsulation, and multiple encoding options, NVI provides an efficient and reliable professional video transmission solution. It not only meets the needs of modern media production and real-time broadcasting but also has the potential to be extended and promoted to other industries such as education, government and enterprise, and security.

Main drafting organizations of this white paper:

China Media Group, Chengdu Sobey Digital Technology Co., Ltd.

Main contributors to this white paper:

Jiang Wenbo, Xu Jin, Wang Wei, Liu Di, Wang Zhenzhong, Luo Tian, Wang Juan, Qi Yi, Zhang Yu, Xu Xiang, Lin Fei, Guo Chenghu, Li Jiawei, Zhang Jinsha, Duan Xin, Gao Jiangjie

Contents

1 Scope.....	1
2 Normative References	1
3 Abbreviations	2
4 Technical Features and Application Scenarios of NVI	3
4.1 Technical Features of NVI	3
4.2 Application Scenarios	5
4.2.1 Lightweight Converged Media Broadcast Studio.....	5
4.2.2 Cloud-based Broadcasting and Production	6
4.2.3 Collaboration Production for Horizontal and Vertical Screens	7
4.2.4 Cross-Industry Expansion and Application	8
5 NVI Technical Architecture	9
5.1 Overall Technical Architecture	9
5.2 Transport Plane	10
5.2.1 LAN Multicast Design	10
5.2.2 LAN Low-Latency Retransmission Mechanism	11
5.2.3 LAN High Bitrate Transmission Performance Optimization	13
5.3 Data Plane	13
5.3.1 Encapsulation Layer	13
5.3.2 Encoding Layer.....	17
5.3.3 Management Layer.....	21
5.4 Control Plane	22
5.4.1 mDNS	22
5.4.2 NVI DNS	24
6 Summary and Outlook	25
1 Scope.....	27
2 Normative References	27
3 Conventions.....	28
4 Format Definition of NVI Audio-Video Stream in TS PMT	28
5 Encapsulation Position and Constraints of NVI Metadata in TS Packet Header	29
6 Syntax Description of NVI Meta.....	31

6.1	Syntax Definition of TS_private_data	31
6.2	Syntax Definition of NVI_metadata	31
6.3	Syntax Definition of NVI_video_metadata.....	32
6.4	Syntax Definition of NVI_audio_metadata	33
6.5	Syntax Definition of NVI_roi_metadata	33
6.6	Syntax Definition of NVI_hdr_static_metadata.....	33
6.7	Syntax Definition of NVI_hdr_dynamic_metadata.....	34
6.8	Syntax Definition of NVI_private_metadata	34
7	Semantic Description of NVI Metadata.....	35
7.1	Semantic Description of TS_private_data	35
7.2	Semantic Description of NVI_metadata	35
7.3	Semantic Description of NVI_video_metadata.....	36
7.4	Semantic Description of NVI_audio_metadata.....	40
7.5	Semantic Description of NVI_roi_metadata	42
7.6	Semantic Description of NVI_hdr_static_metadata.....	42
7.7	Semantic Description of NVI_hdr_dynamic_metadata.....	43
7.8	Semantic Description of NVI_private_metadata.....	43
8	Definition of NVI Metadata Not Encapsulated in TS.....	43
8.1	Definition of NVI_Tally JSON	43
8.2	Definition of PTZ JSON	44
9	NVI TS Audio-Video Data and Metadata Storage Diagram	45

1 Scope

This White Paper explains the concept and technical principles of Network Video Interface (NVI) technology, defines the basic resources, core functions, and performance requirements for constructing system solutions using NVI technology, and describes the application scenarios of NVI technology.

This White Paper is applicable for guiding the construction of systems based on NVI technology, as well as the application and promotion of related products.

2 Normative References

The following documents are essential for the application of this White Paper. For dated references, only the version specified is applicable to this standard. For undated references, the latest version (including all amendments) is applicable to this White Paper.

- GY/T 398.1-2024, "Light Compression Video Encoding Part 1: Ultra High Definition Video Hierarchical Encoding"
- ISO/IEC 13818-1:2023, "Information technology — Generic coding of moving pictures and associated audio information Part 1: Systems"
- GY/T 358—2022, "Technical Requirements for Display Adaptation Metadata of High Dynamic Range Television Systems"
- GB/T 41808-2022, "Image Parameter Values for High Dynamic Range Television Program Production and Exchange"
- GB/T 41809-2022, "Program Production and Exchange Parameter Values for Ultra High Definition Television Systems"
- GY/T 155-2000, "Video Parameter Values for High Definition Television Program Production and Exchange"
- GY/T 363-2023, "3D Audio Encoding, Decoding, and Rendering"
- T/UWA 012.2—2023, "UHD Audio-Video Broadcasting System: Audio-Video Encoding - System"
- "The SRT Protocol draft-sharabayko-srt-01"
- ITU-T H.264-2021, "Advanced video coding for generic audiovisual services"

- ITU-T H.265-2024, "High efficiency video coding"
- EBU DVB ETSI TS 101 154 V2.8.1 (2023-07), "Specification for the use of Video and Audio Coding in Broadcast and Broadband Applications"
- ISO/IEC 14496-3:2019, "Information technology — Coding of audio-visual objects — Part 3: Audio"
- ETSI TS 102 366 v.1.4.1, "Digital Audio Compression (AC-3, Enhanced AC-3) Standard"
- GY/T 257.1-2012, "Advanced Audio-Video Encoding for Broadcasting and Television Part 1: Video"
- GB/T 33475.2-2016, "Information Technology — High Efficiency Multimedia Coding Part 2: Video"
- GY/T 368-2023, "Advanced High Efficiency Video Coding"
- GB/T 20090.1-2012, "Information Technology — Advanced Audio-Video Coding Part 1: Systems"

3 Abbreviations

The following abbreviations apply to this White Paper.

- **NVI** Network Video Interface Protocol
- **ACK** Acknowledgment
- **AVS** Audio Video coding Standard
- **AVS-HFM** Audio Video coding Standard - High Fidelity Model)
- **DNS** Domain Name System
- **FPGA** Field-Programmable Gate Array
- **H.264** Advanced Video Coding
- **H.265** High Efficiency Video Coding (HEVC)
- **HDR** High Dynamic Range
- **mDNS** Multicast Domain Name System
- **MPEG-TS** Moving Picture Experts Group - Transport Stream
- **NAK** Negative Acknowledgment
- **NTP** Network Time Protocol
- **NVI** Network Video Interface
- **NVI RDS** NVI Remote Discovery Service

- **NVI-HC** NVI-High Compression
- **NVI-HF** NVI-High Fidelity
- **PTP** Precision Time Protocol
- **ROI** Region of Interest
- **RTMP** Real-Time Messaging Protocol
- **RTT** Round-Trip Time
- **SRT** Secure Reliable Transport
- **SRT-NVI** Secure Reliable Transport - NVI
- **SUVC** Scalable Ultra-HD Video Coding
- **PTZ** Pan-Tilt-Zoom

4 Technical Features and Application Scenarios of NVI

NVI is a network transmission protocol designed for ultra-high-definition/high-definition audio-video streams. It closely integrates the two core functions of transmission and video encoding/decoding, providing a simple and efficient solution for the transmission of audio-video data over local area networks (LANs) and the Internet. By offering two transmission modes—light compression (NVI-HB, NVI-High Bandwidth) and heavy compression (NVI-HC, NVI-High Compression)—NVI caters to different production scenarios and ensures high-quality and low-latency transmission of ultra-high-definition/high-definition video.

4.1 Technical Features of NVI

The application-oriented characteristics of NVI technology can be summarized in the following six aspects:

1) **Compatibility with Internet and LAN Transmission**

- Supports Internet cross-domain point-to-point transmission scenarios.
- Supports LAN multicast and point-to-point transmission scenarios.

2) **Low Latency**

- For UHD video transmission, the end-to-end transmission latency (including encoding and decoding) can be as low as 100ms in LANs and up to 200ms on

the Internet.

3) **Lightweight**

Compared to the ST2110 standard, NVI's defined transcoding, gateways, picture splitting, production switching, and packaging hardware and software have significantly lower system resource consumption. The network bandwidth usage is also correspondingly reduced. NVI can fully meet the functional requirements for UHD production and broadcasting in terms of image quality, supported encoding formats, and application functions. The system resource requirements for application products are inevitably lighter than those defined by other technical standards, offering better cost-performance ratios.

4) **High Quality**

- For Internet transmission scenarios, it supports deep compression formats such as H.264, H.265, AVS+, AVS2, AVS3, etc.
- For LAN transmission scenarios, it supports light compression formats, which can be selected based on business needs, such as JPEG XS, SUVC, NVI-HF, AVS-HFM, etc.
- Supports large-format transmission of 4K, 8K, and custom formats.
- Supports proxy bitrate transmission and HDR transmission.
- Supports transmission of stereo, surround sound, and 3D audio.

5) **Professional Media Attributes**

- Supports transmission of audio-video, pure video, pure audio, and Alpha information.
- Supports transmission of metadata such as timecode, PTZ (Pan-Tilt-Zoom), and camera tracking data.
- Supports transmission of custom metadata.
- Supports precise clock synchronization with NTP (Network Time Protocol) and PTP (Precision Time Protocol).
- Supports metadata transmission for collaborative production between horizontal and vertical screens.

6) **Low Cost**

- Supports cloud-based deployment.
- Supports connection with general-purpose network devices.
- Automatic discovery and registration, plug-and-play.

7) **Security and Reliability**

- Supports encrypted transmission.
- Supports frame retransmission and reliable transmission.
- Provides device management and control solutions.

4.2 Application Scenarios

4.2.1 Lightweight Converged Media Broadcast Studio

The architecture of a lightweight converged media broadcast studio based on NVI is shown in Figure 1. The common signal sources are mainly divided into three categories. The first category is acquisition devices that already support NVI signal output, such as certain PTZ (Pan-Tilt-Zoom) robotic cameras. These types of devices can directly output NVI video streams to the studio production system. The second category consists of other commonly used video cameras that still employ traditional baseband signals. These devices can encode the baseband signals into NVI streams through an encoder, which can then be transmitted via wired or wireless networks to the production system. The third category includes live acquisition signals available on the network, such as those in SRT and RTMP formats. These signals can be converted into NVI signals by a stream conversion device and then fed into the production system.

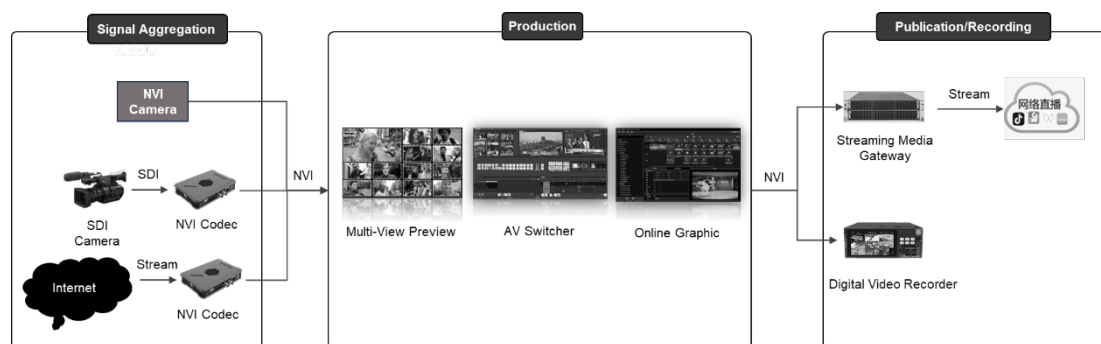


Figure 4-1 Schematic Diagram of a Lightweight Converged Media Studio System Based on NVI

The production system can schedule signals according to requirements: for example, using multi-view monitoring systems and video switchers for program production, as well as for graphics and packaging. After production is completed, the program can be pushed to a traditional SDI broadcasting system via an NVI decoder for broadcast, or it can be converted into other protocols through a video gateway for

multi-platform streaming live.

A converged media broadcast studio built on NVI technology can avoid the high costs, complex cabling, and special equipment configurations associated with traditional SDI transmission solutions, which often lead to long setup times and difficult maintenance.

Moreover, in scenarios with high demands for on-site coverage, such as breaking news or small-scale events, the lightweight nature of the system also supports deployment in mobile production vehicles and EFP (Electronic Field Production), enabling mobile and lightweight broadcast scenarios.

4.2.2 Cloud-based Broadcasting and Production

Based on NVI technology, services such as audio-video switching, production, monitoring, graphics packaging, and rendering can be deployed in the cloud. This allows the production end to flexibly connect and start production. This greatly reduces the deployment difficulty and complexity for scenarios such as on-site activities with remote sub-venues and external signal inputs. The architecture is shown in Figure 4-2. Remote directors and editors can directly use computers or laptops to access the cloud via the network to complete tasks such as cloud-based production, signal switching control, and picture monitoring.

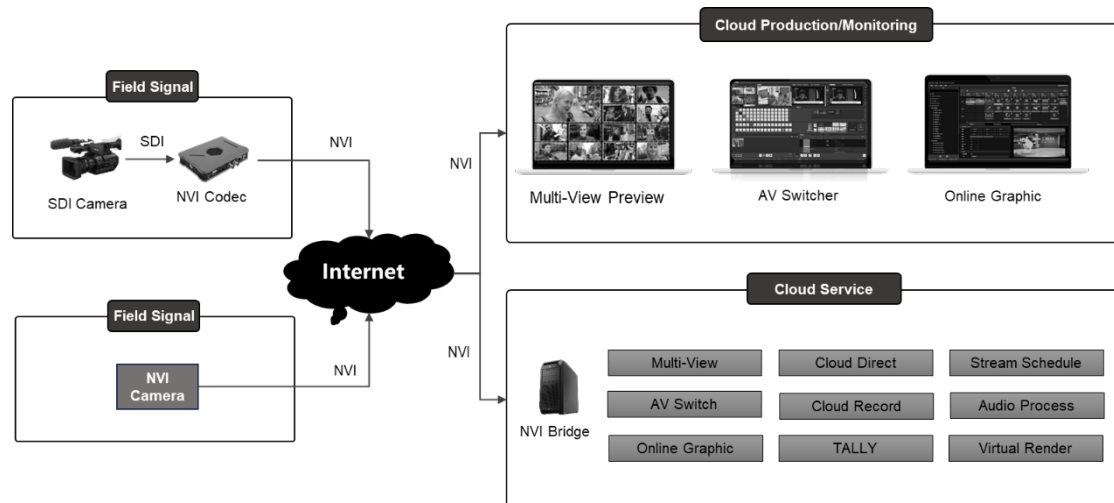


Figure 4-2 Schematic Diagram of Cloud-Based Production System Based on NVI

In this solution, remote and local signals are integrated through the cloud-based NVI Bridge via point-to-point or multicast methods. Meanwhile, the locally deployed audio-video switcher can instantly switch any signal to the display screen for SDI

monitoring and record the signals using a hard disk recorder. NVI offers various encoding formats for users to choose from. In the case of Internet transmission, considering the pressure on network conditions and bandwidth, the NVI-HC (NVI-High Compression) deep compression format can be used for production. For production within a local area network (LAN), given the stable and reliable network environment, the NVI-HB (NVI-High Bandwidth) light compression method can be adopted to ensure a higher quality of image experience.

Moreover, if the main production process is based within the studio, the production services can also be directly deployed offline. Remote signals can interact with the in-studio production system through the cloud-based NVI Bridge, thus realizing a production mode primarily based within the studio.

4.2.3 Collaboration Production for Horizontal and Vertical Screens

Two solutions are provided to support collaborative production for horizontal and vertical screens.

Solution One:

Multiple front-end cameras capture footage in landscape mode. The captured signals are converted into NVI IP signals and sent to the NVI encoder. Using the tool software provided by NVI, the NVI IP signals can be marked with a vertical frame to select the Region of Interest (ROI). The selected vertical screen information is recorded as extended metadata ROI information in the NVI IP signal and sent back to the NVI encoder for signal transmission. When the signal enters the NVI switcher, the switcher can directly read the ROI metadata information from the NVI IP signal and

crop the signal to output a vertical screen signal.

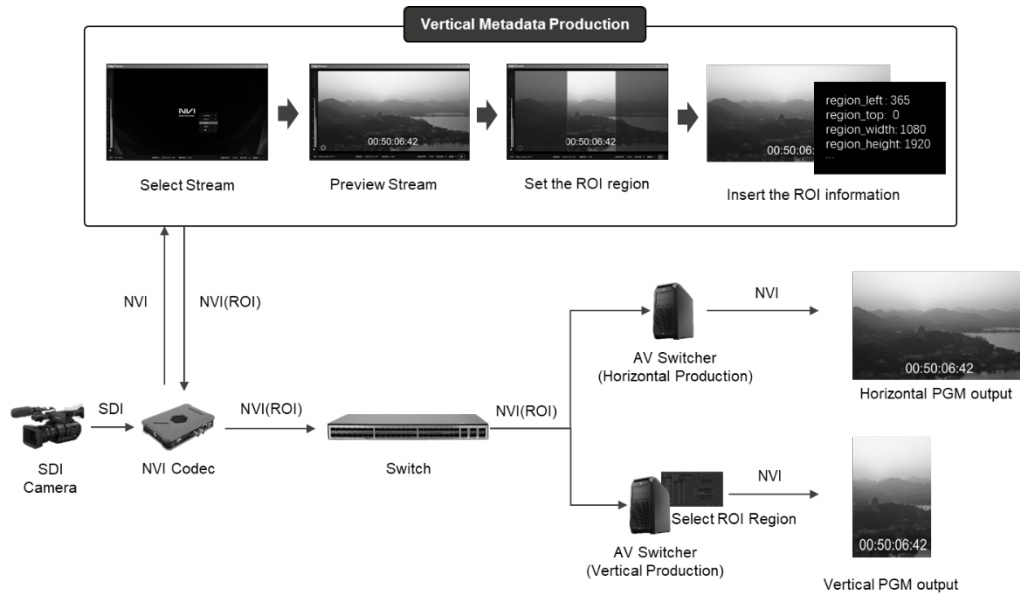


Figure 4-3 Vertical ROI Information Transmission

For the transmission of ROI information, the capability of marking ROI information can be directly embedded into the front-end signal acquisition devices. By utilizing AI technology to achieve automatic ROI information marking, the entire vertical screen production process can be made more efficient and convenient.

Solution Two:

Multiple front-end cameras capture footage in landscape mode. The captured signals are converted into NVI IP signals and sent to the back-end NVI switcher. The NVI switcher used for vertical screen production can crop the landscape signals to generate multiple vertical screen materials, which are then switched and packaged to output a vertical PGM (Program) signal. Meanwhile, the NVI switcher used for horizontal screen production can directly package the landscape signals to output a horizontal PGM signal.

4.2.4 Cross-Industry Expansion and Application

NVI leverages Ethernet for high-quality video stream transmission, supporting low latency and high resolution. This makes NVI suitable for video applications that require real-time monitoring and control, such as security surveillance and medical imaging.

By adopting standard IP network protocols, NVI eliminates the complex cabling

and hardware devices found in traditional video transmission systems. This not only reduces deployment costs and time but also enhances system scalability and flexibility. For industry applications that require extensive coverage, such as live education, government and enterprise broadcast studios, and new media live streaming, NVI provides flexible solutions.

Moreover, NVI is compatible with multiple operating systems and devices, including Windows, macOS, and Linux. This enables users across various industries to integrate NVI with their existing equipment and infrastructure, facilitating cross-platform video transmission and collaboration. Examples of such applications include corporate video conferencing and online training.

5 NVI Technical Architecture

5.1 Overall Technical Architecture

The NVI technical framework is shown in Figure 4. The transport plane is based on the open-source SRT protocol. On the basis of SRT's original transport capabilities, three modules have been additionally developed for multicast functionality, delay optimization, and high bitrate optimization to form the SRT-NVI version protocol. In the data plane, the encapsulation layer uses MPEG-TS to encapsulate audio-video streams and defines multiple types of metadata information to support professional media production needs. The encoding layer addresses the NVI-HC and NVI-HB modes and introduces a joint source and channel coding mechanism for slice-based transmission. The management plane provides support for clock synchronization and real-time stream transmission and reception services. The control plane addresses

networking and production metadata support capabilities.

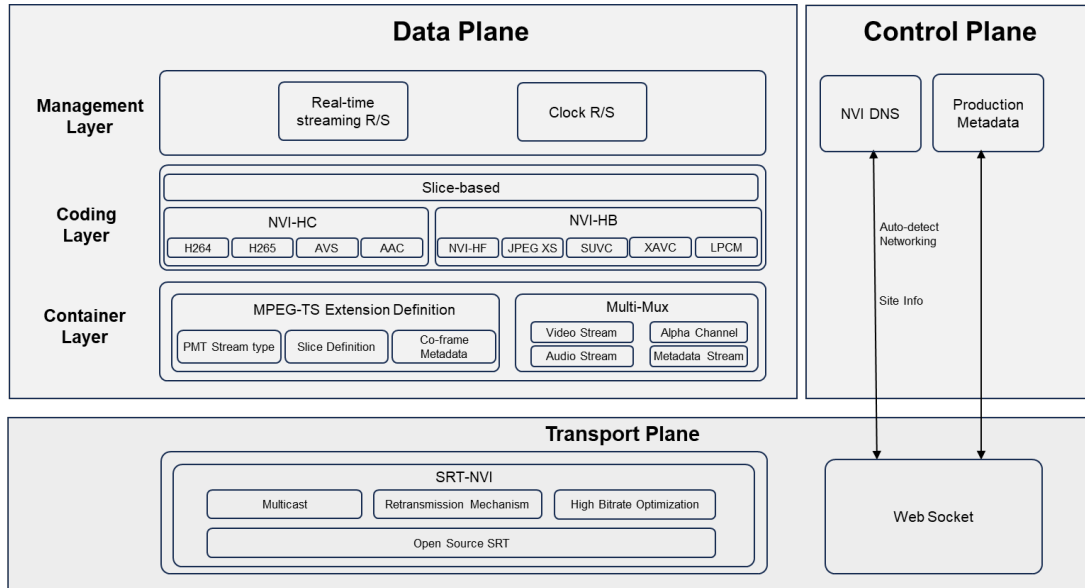


Figure 5-1 NVI Technical Architecture Diagram

5.2 Transport Plane

The NVI real-time media stream transmission system is built on a three-layer IP network, using SRT as the transport control protocol. However, since SRT is primarily designed for Internet transmission, its direct application in LAN transmission can encounter some issues. Therefore, NVI uses the modified and optimized SRT-NVI protocol in LANs to better meet the requirements of LAN environments.

5.2.1 LAN Multicast Design

Using multicast technology in LANs can effectively reduce network bandwidth usage and improve transmission efficiency. Multicast allows a single data stream to be delivered to multiple receivers simultaneously, without the need to send data separately to each receiver, thereby reducing network load and latency. NVI has improved the SRT protocol to efficiently support multicast transmission in LAN environments and has developed its own transport layer architecture, SRT-NVI.

When NVI's transmitting end performs multicast transmission, it continuously sends data to the default or agreed multicast address. Upon coming online, the receiving end discovers the existence of the multicast source through registration and can choose to join a specific multicast group to start receiving data from that group.

An important distinction between NVI multicast and point-to-point transmission is the connection establishment handshake mechanism. For point-to-point transmission, there is a clear handshake process before transmitting audio and video data to establish the connection and exchange transmission parameters. In contrast,

NVI multicast simplifies the handshake process. NVI multicast streams use periodic handshake messages to carry the transmitter's transmission parameters and current status. The receiver can join at any time, obtain these parameters and status information from the transmitter, complete the multicast handshake process, align the transmission state, and prepare to receive the real-time audio and video stream. NVI multicast still retains a retransmission mechanism. NVI multicast does not use SRT's ACK (Acknowledgment) confirmation mechanism but retains the NAK (Negative Acknowledgment) retransmission request mechanism. When an NVI receiver detects packet loss, it directly sends a NAK message to the multicast source requesting retransmission. Upon receiving the NAK message, the source retransmits the lost data packet in multicast mode. Each receiver that receives the retransmitted data packet will determine whether it is a packet lost by itself. If it is, the receiver will accept it; if not, the packet will be ignored.

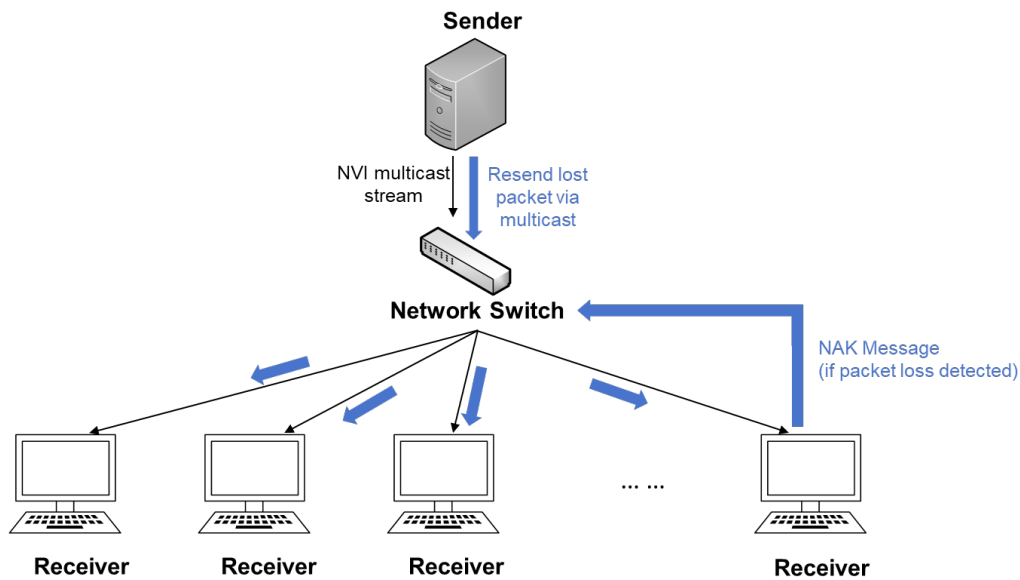


Figure 5-2 NVI Multicast Retransmission Mechanism

5.2.2 LAN Low-Latency Retransmission Mechanism

The retransmission mechanism of the original SRT protocol was primarily designed to address high latency and instability in Internet transmission. However, in

LAN environments, latency requirements are much stricter. Therefore, NVI has optimized the retransmission mechanism of the SRT protocol and introduced the SRT-NVI version suitable for LANs.

In real-time audio and video transmission protocols, implementing a packet loss retransmission mechanism through NAK (Negative Acknowledgment) messages is crucial. The ideal sending interval of NAK messages is strongly correlated with the network round-trip time (RTT). In the original SRT protocol, the minimum sending interval for NAK messages is 20ms, which is a reasonable setting for Internet transmission. However, for LANs with $RTT < 1\text{ms}$, such a minimum interval for NAK messages is excessively long. This results in lost packets not being retransmitted in a timely manner, affecting the timeliness and reliability of data transmission in LANs.

Packet loss detection in the original SRT protocol relies solely on packet sequence numbers. This method has shortcomings in detecting packet loss at the end of a frame, as it requires the arrival of the next frame to determine whether a packet has been lost. This misses the optimal retransmission opportunity.

The original SRT protocol also employs a fixed delay mechanism, where the delay value needs to be set externally. Given the extremely low packet loss rate in LANs, an excessively high delay setting negatively impacts overall transmission performance and prevents achieving the most ideal low-latency transmission effect.

To address the limitations of the original SRT protocol, SRT-NVI has made targeted optimizations:

- 1) **NAK Message Trigger Interval:** SRT-NVI reduces the minimum trigger interval for NAK messages to 2ms. This increases the frequency of multiple retransmissions and significantly reduces latency in LANs with higher packet loss scenarios.
- 2) **Packet Loss Detection Mechanism:** SRT-NVI improves the timeliness of packet loss detection by carrying the sequence number of the last packet sent in periodic ACKACK messages (sent every 10ms). This allows for more rapid identification and retransmission of lost packets, even those at the end of a frame, thereby enhancing retransmission efficiency.
- 3) **Flexible Delay:** SRT-NVI supports a maximum flexible delay of up to 60ms under ultra-low delay configurations. When network conditions are stable, packets stay in the receiver's buffer for a short time before being output,

minimizing latency. When network fluctuations occur, the receiver's buffer does not immediately output packets that are lost and timed out. Instead, it waits until the maximum delay deadline before outputting, ensuring reliable transmission.

5.2.3 LAN High Bitrate Transmission Performance Optimization

In LAN environments, professional video production requires visually lossless, high-quality images, often employing lightly compressed video formats with bitrates reaching several hundred Mbps. In such cases, the stable transmission of large amounts of data becomes a critical issue. When a large number of UDP packets are simultaneously sent into the kernel protocol stack, the kernel schedules them at the maximum rate of the network card. This scheduling method can lead to unstable transmission traffic, which in turn causes network congestion and traffic jitter, affecting the stable transmission of high-bitrate video streams.

To address the above issues, NVI has made the following optimizations:

1) **Removing Data Size Limitations in SRT Live Mode:**

NVI eliminates the data size restrictions on the sending interface in SRT Live mode. Instead, it uses a packet segmentation mechanism within the transport layer to reduce the scheduling overhead caused by multiple small data packet calls, thereby enhancing the stability of the transmission interface.

2) **Optimizing Bandwidth Control Algorithms for Transmission Queues:**

Within the transport plane, NVI optimizes the bandwidth control algorithm for the sending queue. This provides a more stable packet transmission interval cycle for high-bitrate streams, allowing the network to handle large volumes of data packets at a more uniform rate. This avoids the negative impacts of sudden traffic spikes.

5.3 Data Plane

5.3.1 Encapsulation Layer

MPEG-TS (Transport Stream) is commonly used as the data encapsulation format for low-latency real-time audio and video data transmission. It is one of the most widely used media content formats in the fields of streaming media and broadcasting. NVI also adopts it as the media content format for real-time media stream transmission. However, some NVI audio and video formats and the metadata required for NVI transmission are not defined in the TS standard. Therefore, NVI

extends it to address these issues.

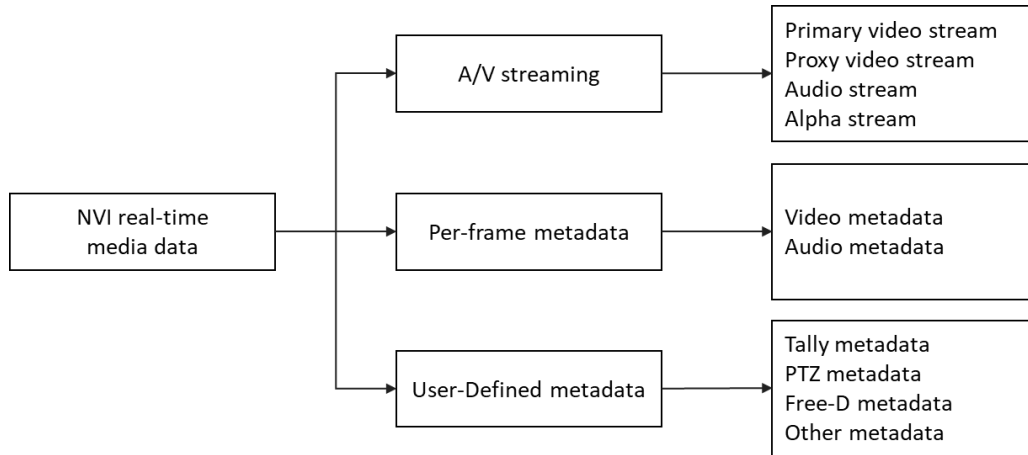


Figure 5-3 NVI Realtime Media Metadata Structure

The real-time media data content of NVI mainly consists of three categories:

- 1) **A/V Streaming:** This part contains a large volume of data and represents the core data of NVI.
- 2) **Per-Frame Metadata:** This is the key data that enables NVI to achieve low-latency transmission and rich professional features.
- 3) **User-Defined Metadata:** This refers to data customized by applications, such as specific Tally/PTZ/Free-D protocol data used by users. NVI supports transparent transmission without parsing these data.

These data types have high requirements for timeliness and reliability. Audio-video data should be transmitted quickly and in sequence, while auxiliary data must correspond perfectly with audio-video frames. NVI addresses these challenges by closely integrating the processing of the transport control protocol, data content format, and audio-video encoding/decoding, delivering a low-latency and reliable real-time media streaming experience.

5.3.1.1 Multiplexing

The audio-video streams encapsulated by NVI in MPEG-TS for real-time transmission typically include at least the primary video stream and the audio stream. In addition to these, NVI's audio-video streams may also contain proxy video streams and Alpha streams.

1) **Primary Video Stream**

The primary video stream is the main data stream in NVI, responsible for

carrying the core video content. This stream is encoded at a higher resolution and quality to ensure that the receiving end can perform final processing and high-quality display.

2) **Proxy Video Stream**

The proxy video stream serves as an auxiliary video stream with lower resolution and quality compared to the primary video stream. This design allows for smooth playback experiences in scenarios where network bandwidth is limited or the playback device's performance is insufficient.

3) **Audio Stream**

The audio stream is responsible for transmitting the synchronized sound information associated with the video content. It ensures the consistency between the audio and video, providing the audience with a complete audio-visual experience.

4) **Alpha Stream**

The Alpha stream is a special type of video stream that contains transparency information (Alpha Channel) related to the primary video stream. This stream is crucial for achieving advanced image compositing and visual effects.

NVI defines specific PES (Packetized Elementary Stream) **stream_id** and PMT (Program Map Table) **stream_type** for each type of audio-video stream encapsulated in MPEG-TS to ensure correct identification and processing of the streams. For details, refer to Appendix A, "Network Video Interface (NVI) Metadata Encapsulation Specification (Version 1.0)." As an efficient transport protocol, MPEG-TS ensures the real-time and reliable transmission of data by segmenting audio-video data into small packets. NVI embeds specific identifiers in the MPEG-TS stream to achieve precise management and synchronization of different types of audio-video streams.

5.3.1.2 Per-Frame Metadata

Per-frame metadata is information closely associated with each frame of audio and video data, used to enhance transmission quality, achieve audio-video synchronization, and perform specific processing and effects.

1) **Video Metadata**

The basic information in video metadata includes: video format, timestamp, frame size, frame rate, bit depth, chroma sampling, frame type, rotation information, color gamut, gamma information, etc. Video slice information includes: slice division mode, number of slices, slice number, slice start and end markers, etc. HDR metadata

includes static and dynamic metadata compliant with various HDR standards. ROI metadata includes the position, size, and labeling of multiple ROI (Region of Interest) areas.

2) Audio Metadata

Per-frame audio metadata includes information such as audio format, timestamp, sampling rate, bit depth, and channel configuration.

The storage location of NVI per-frame metadata is in the Transport private data area of the Adaptation field in the MPEG-TS packet header. The basic encapsulation method of metadata is consistent with Annex D of EBU DVB ETSI TS 101 154 V2.8.1.

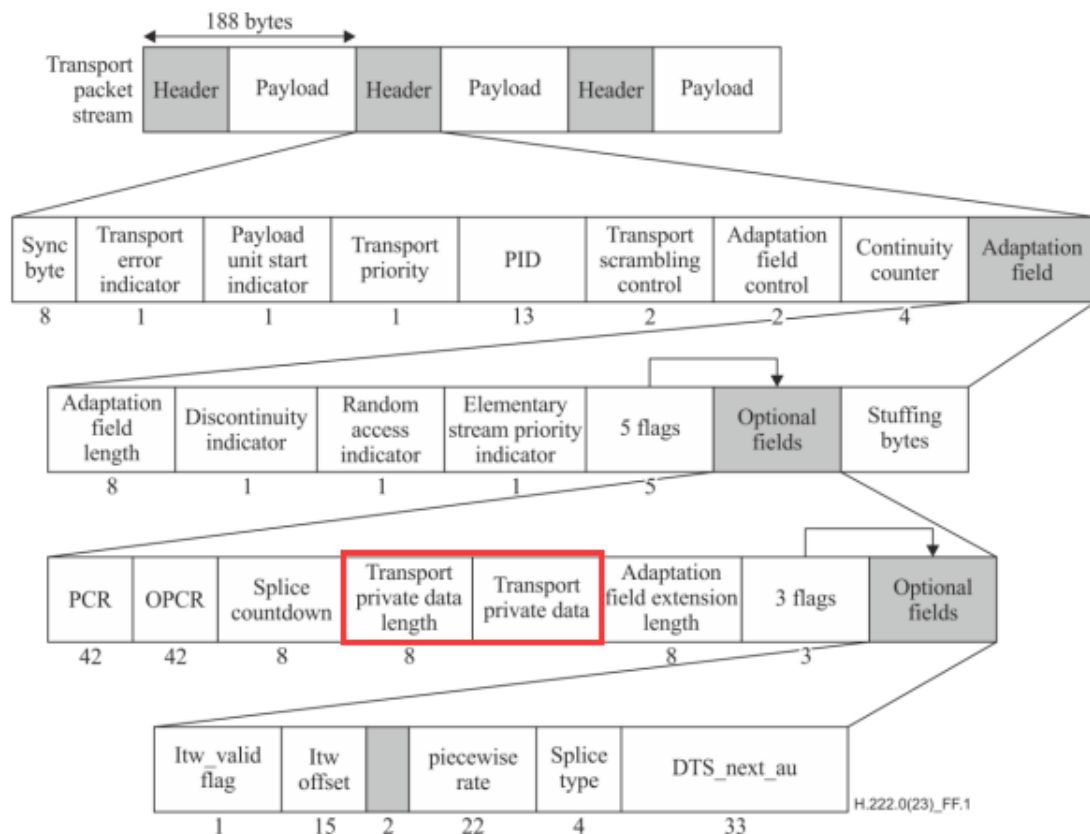


Figure 5-4 Storage Location of NVI Per-Frame Metadata

5.3.1.3 User-Defined Metadata

NVI supports user-defined metadata, which is additional information embedded in the MPEG-Transport Stream (MPEG-TS) audio-video stream. This metadata is defined by application developers or users according to the specific needs of their application scenarios. These metadata include Tally data, PTZ control information,

Free-D audio positioning data, etc., and are designed to meet specific functions in professional production, monitoring, or interactive applications.

During transmission, NVI performs transparent transmission of user-defined metadata without parsing or modifying it, ensuring the integrity and accuracy of the data from the sender to the receiver. The receiving end device or application can then extract this metadata and process it according to predefined logic, thereby achieving advanced control and analysis of the audio-video stream.

This mechanism significantly enhances the flexibility and scalability of NVI, allowing users to customize metadata according to their own needs and expand the application scope of the audio-video stream. At the same time, it also ensures system compatibility because NVI acts merely as a data transmission channel without interfering with the specific content of user data. This approach accommodates diverse audio-video transmission requirements while maintaining the system's openness and ease of use.

5.3.2 Encoding Layer

5.3.2.1 Slice Transmission Adaption

To reduce transmission latency in the encoding layer, a video frame can be divided into multiple slices for transmission.

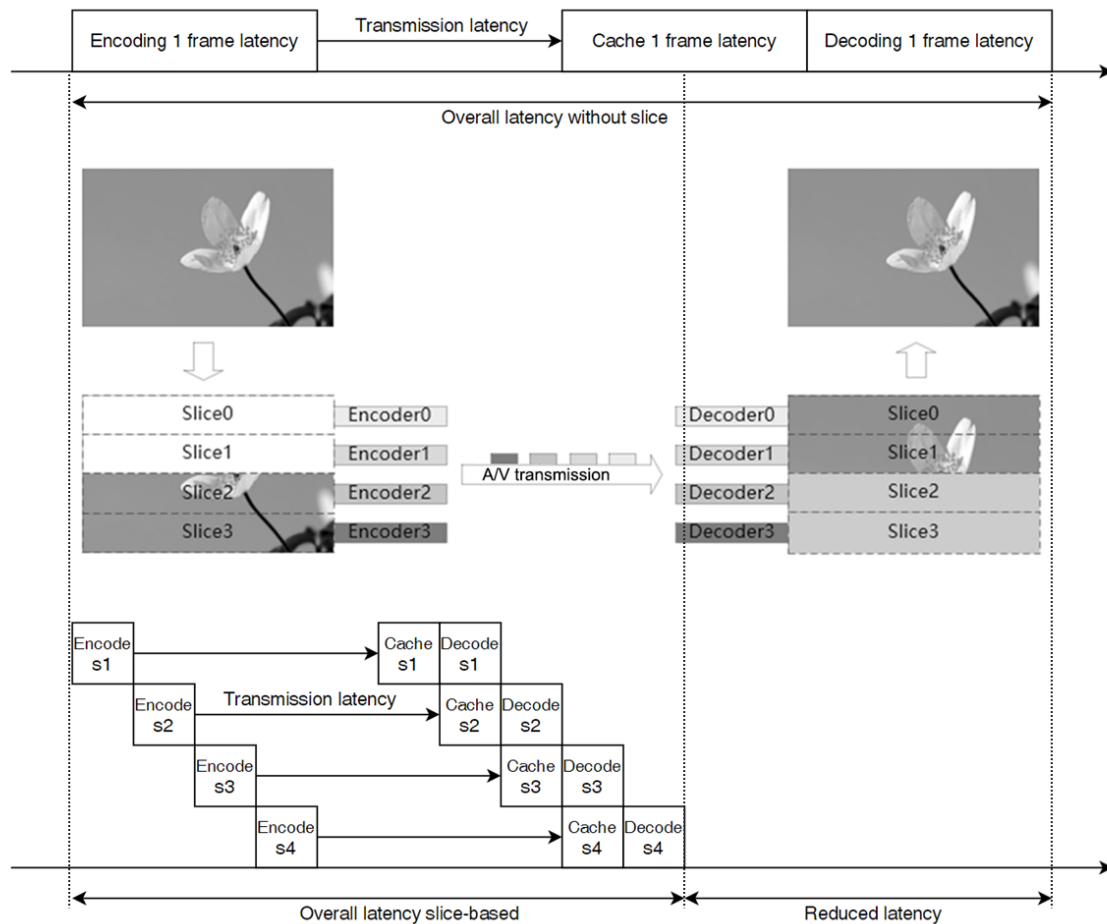


Figure 5-5 Transmission Mechanism Based on Slices

1. Video Transmission

Video data is encoded, transmitted, and decoded on a per-frame basis. The overall latency consists of encoding delay, transmission delay, buffering delay, and decoding delay, all of which are serial in nature.

2. Intra-Frame Multi-Slice Transmission

By dividing a frame into multiple slices, a pipeline processing mechanism can be formed. This allows encoding and transmission to be processed in parallel, thereby reducing overall latency.

3. Compression Formats

- **Deep Compression Techniques (e.g., H.264/H.265):** These techniques can divide a frame into 2 to 8 slices. Under the premise of minimal impact on encoding quality, latency is minimized as much as possible.
- **Light Compression Techniques:** Since there are no complex predictions and

dependencies within these techniques, more slices can be divided, thus achieving lower encoding and decoding latency. In practical applications, dividing a frame into 8 slices using light compression formats can generally meet the low-latency encoding and transmission requirements for most application scenarios.

Light Compression Formats	Minimum Independent Unit	Maximum Number of Slices (for 4K Frame Size)
XAVC-Intra	1/8 frame	8
JPEG XS	16 line	135
NVI-HF	16 line	135
AVS-HFM	8 line	270
SUVC	16 line	135

Tabel 5-1 Slice Division Structures for Different Light Compression Formats

Therefore, by dividing a video frame into multiple slices for parallel processing, the overall latency of video transmission can be significantly reduced. This approach is suitable for video transmission scenarios that require low latency and high efficiency, especially in real-time video communication and streaming media services.

5.3.2.2 Encoding and Decoding Formats

NVI-HC (High Compression) Encoding Format

NVI-HC is a highly compressed video format designed to significantly reduce the size of video files through a high compression ratio. This format is particularly suitable for bandwidth-constrained scenarios, such as video transmission over public networks. The actual encoding standards used by NVI-HC are the widely adopted international standards H.264 (AVC) and H.265 (HEVC), with support for the AVS standard as well.

For audio, formats such as AAC, OPUS, and Audio Vivid are used.

When encoding video using NVI-HC, users can choose between H.264 and H.265 encoding formats. Generally, H.264 offers better device compatibility and lower resource consumption, while H.265 provides a higher compression ratio, making it more suitable for ultra-high-definition HDR 10-bit encoding.

To achieve low latency, NVI-HC employs low-latency encoding modes of H.264/H.265 without using B-frames. Additionally, NVI-HC supports slice division within frames to further reduce latency.

Light Compression Encoding Formats

NVI-HB (High Bandwidth) is a low-complexity intra-frame compression technology that supports 4:2:2 8-bit/10-bit encoding required for professional production, providing more stable and higher image quality compared to deep compression. NVI currently supports multiple light compression formats, such as JPEG XS, SUVC, AVS-HFM, and NVI-HF, with audio encoded in formats like LPCM.

Among these, AVS-HFM is a light compression format developed by the AVS organization. Compared to JPEG XS, AVS-HFM achieves over 20% higher compression ratio without a significant increase in complexity.

NVI-HB is based on the MPEG2 I-frame encoding technology, with extensions and modifications focusing on ease of use and low resource consumption. Its compression ratio and speed metrics are comparable to NDI-FULL (SpeedHQ).

The NVI-HB encoding format has the following characteristics:

1) Extremely Low Complexity

The NVI-HF codec has an extremely low CPU usage rate. Real-time encoding of an HD video stream on a regular PC consumes less than 5% of the CPU. On a high-performance workstation, it can support real-time decoding of more than 20 4K 50p streams simultaneously. It is also easy to implement on an FPGA with minimal resource usage.

2) Relatively High Quality

In HD conditions, the quality of NVI-HF is comparable to that of formats like ProRes and DNxHD. In 4K conditions, the quality of NVI-HF is close to that of JPEG XS.

3) Low Latency

Each frame of NVI-HF is divided into multiple slices, with each slice having a height of 16 lines. Each slice can be independently encoded, decoded, and

transmitted, resulting in extremely low encoding and decoding latency.

Therefore, the NVI-HF light compression encoding technology offers significant advantages in terms of complexity, quality, and latency, making it suitable for efficient video processing needs in various professional production environments.

5.3.3 Management Layer

1) **System Clock Synchronization**

The NVI transmitter and receiver rely on the system clock as the core of their synchronization mechanism during operation. The accuracy of the system clock is crucial for ensuring the synchronization of the streams. To this end, the system clock can be aligned with high-precision time synchronization services such as Network Time Protocol (NTP) or Precision Time Protocol (PTP) to ensure the precision and consistency of time. For example, PTP is used for timing in LAN environments, while NTP is used for timing in Internet scenarios.

2) **Timestamp Embedding**

In the NVI stream, each video and audio frame is embedded with a timestamp. These timestamps reflect the exact time of frame generation. The precision of timestamps for audio frames is typically higher than that for video frames to meet the higher synchronization requirements of audio. The embedding of timestamps is a key technology for achieving stream synchronization, allowing the receiver to accurately identify the arrival time of each frame. For details on timestamp format and insertion location, refer to Appendix A, "Network Video Interface (NVI) Metadata Encapsulation Specification (Version 1.0)."

3) **Synchronization Processing at the Receiver**

The receiver utilizes the embedded timestamp information to realign the received audio and video streams. This process involves parsing the timestamps and scheduling the frames based on these timestamps to ensure synchronization across multiple streams.

4) **Advantages of Synchronization**

The use of system clocks and timestamp embedding provides an efficient synchronization solution for remote production and cloud-based video production. This method not only reduces synchronization errors but also enhances the flexibility and reliability of the production process. It allows production teams to collaborate across different geographical locations or via cloud services while maintaining a high

standard of audio-visual experience.

NVI technology achieves efficient synchronization of audio and video streams through precise system clock synchronization and timestamp embedding. This method is particularly important for remote and cloud production scenarios that require high-precision time synchronization, providing robust technical support for modern video production.

5.4 Control Plane

5.4.1 mDNS

Networking is a key step in the initialization of the NVI system. By establishing a communication network, it enables interconnectivity between devices and real-time invocation of audio-video media resources.

Multicast discovery utilizes multicast addresses and the DNS protocol to detect NVI site information. It automatically resolves hostnames to IP addresses, allowing devices capable of receiving NVI video streams to immediately detect the presence of a video source when an application for sending NVI video streams is launched.

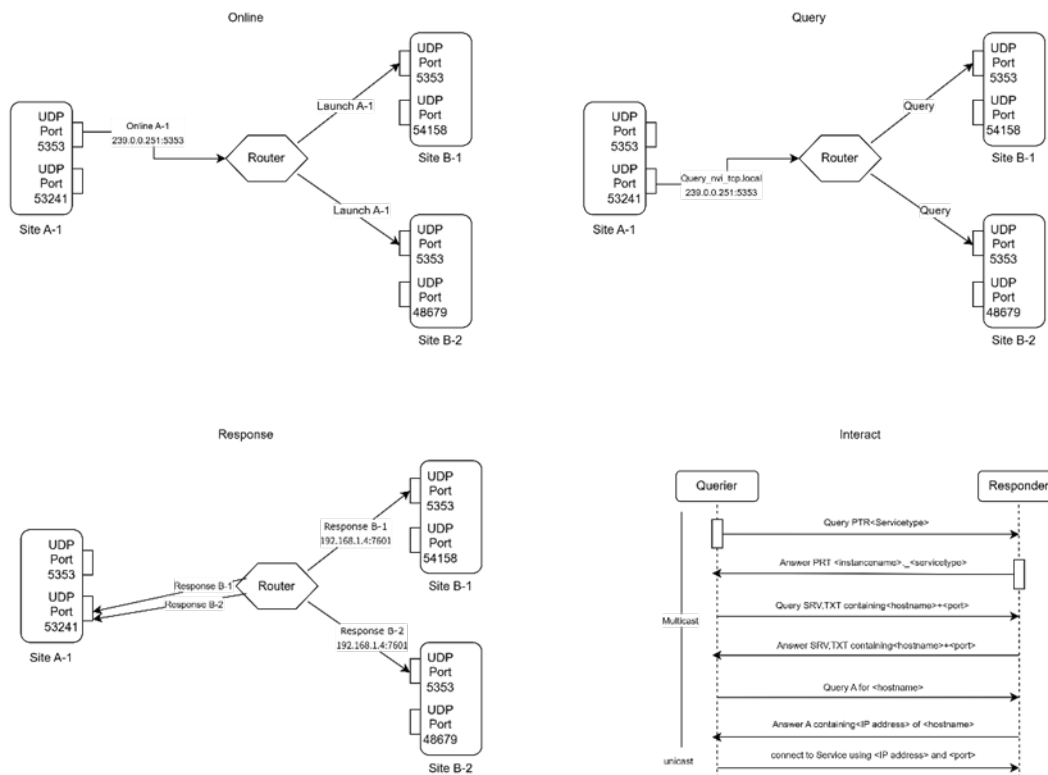


Figure 5-6 NVI mDNS Networking Process Design

NVI utilizes the Multicast DNS (mDNS) service to create a network environment that requires no configuration or minimal configuration. mDNS sends IP multicast messages to request hosts to identify themselves, and the target machine responds with a multicast message containing its IP address. This multicast message is received by all NVI-receiving devices on the LAN, which use this information to update their video source lists.

These multicast queries are sent to a specific multicast address, meaning that devices on the network do not need to have global knowledge but can discover services through local multicast messages. When a device recognizes a query for any service it cares about, it provides a response containing its service record information.

The primary advantage of using mDNS is that its setup requires almost no administration. Unless the network is specifically configured to block mDNS, NVI video sources will be automatically discovered. This mechanism is particularly effective in the absence of complex infrastructure or when infrastructure failures need to be bypassed.

In short, NVI combined with mDNS provides an efficient and automatic mechanism for video stream discovery and connection. It simplifies the configuration

process for video streams, making video communication within a LAN more convenient and flexible.

5.4.2 NVI DNS

NVI Remote Discovery Service (NVI RDS) is an advanced network protocol that provides an efficient, flexible, and easily manageable solution for constructing NVI network applications across subnets and for cloud-based production scenarios over the Internet. By configuring a single discovery service address and integrating DNS services, sites can easily access the network via domain names, thereby simplifying the access process. This service is not only responsible for registering and maintaining the host addresses of NVI sites but also enables dynamic networking between sites through peer-to-peer (P2P) connections.

Utilizing the WebSocket protocol, the discovery service can maintain persistent long connections with each site, ensuring that any dynamic changes at a site are pushed in real-time to other members of the network. This real-time communication mechanism greatly enhances the responsiveness and flexibility of the network. Moreover, since the discovery service can be deployed on any subnet that supports unicast packet routing, and it defaults to using port 7060 and the WebSocket protocol, the deployment of the service becomes extremely convenient.

Sites access the discovery service through local DNS resolution of domain names, which not only increases the flexibility of configuration but also reduces the reliance on static IP addresses. More importantly, the discovery service also has the capability of network group management. It can control the scope of networking and verify access permissions by defining service scopes and implementing authentication mechanisms. This enhances the flexibility of network management while also improving the overall security of the network.

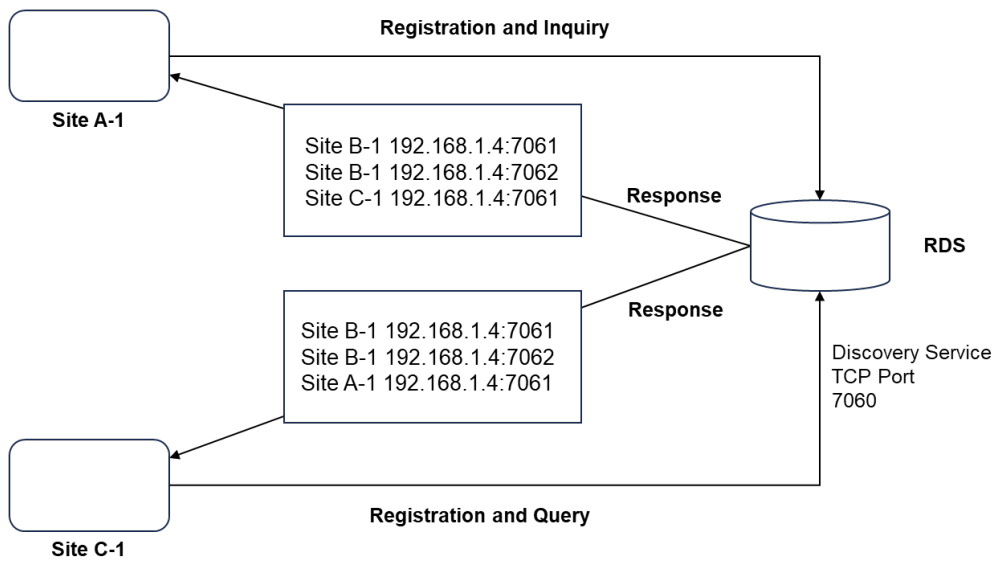


Figure 5-7 NVI RDS Networking Process Design

6 Summary and Outlook

NVI technology, with its efficient transmission performance, versatile application scenarios, and low cost, is gradually becoming one of the important technical solutions for video transmission in the professional media and Internet industries. It will play a key role in the transformation of the broadcasting and television industry and the convergence of media.

NVI technology has successfully integrated the two core functions of transmission and video encoding/decoding, providing a simple and efficient solution for the transmission of audio-video data over local area networks (LANs) and the Internet. Through two types of encoding and decoding modes—light compression and deep compression—combined with optimizations in the transport layer, NVI technology ensures high-quality and low-latency video transmission under different network conditions. In addition, NVI technology provides extended metadata information, such as Tally and Alpha, greatly enriching the expressiveness of audio-video content and offering more possibilities for different application scenarios, including professional media broadcasting and Internet live streaming.

Moving forward, the development of NVI technology will continue to focus on the following areas:

- 1) Standardization and Promotion: Further develop and perfect the industry

standards for NVI technology to promote its widespread application in the broadcasting and Internet industries. Standardization will enhance the compatibility and interoperability of NVI technology, fostering the formation of a robust industrial ecosystem.

- 2) **Product Ecosystem Development:** Accelerate the research and development of NVI-related products and build a comprehensive product ecosystem. This includes the development of NVI encoders, decoders, transmission equipment, and related software tools to meet video transmission needs in various scenarios.
- 3) **Technical Improvement and Innovation:** Continue to optimize the transmission performance and encoding/decoding efficiency of NVI on the existing basis. Investigate more efficient compression algorithms and lower-latency transport protocols to enhance the application effects of NVI in ultra-high-definition and low-latency video transmission. At the same time, explore the integration of NVI with other emerging technologies, such as AI-based intelligent encoding/decoding and edge computing, to expand the application scenarios of NVI.
- 4) **Enhanced Security:** Strengthen data security protection during the transmission process of NVI. Research and implement more sophisticated encryption and authentication mechanisms to ensure the security and reliability of video data during transmission, preventing data leakage and tampering.

Through continuous technological innovation and ecosystem development, NVI technology will bring more possibilities and higher efficiency to the field of video transmission, driving the development and transformation of the industry.

Appendix A: Specification for Metadata Encapsulation of Network Video Interface (NVI) (Version 1.0)

1 Scope

This document specifies the encapsulation method and detailed definitions of NVI metadata in MPEG-TS.

2 Normative References

- GY/T 398.1-2024, "Video Shallow Compression Coding Part 1: Ultra High Definition Video Hierarchical Coding"
- ISO/IEC 13818-1:2023, "Information technology — Generic coding of moving pictures and associated audio information — Part 1: Systems"
- GY/T 358—2022, "Technical Requirements for Display Adaptation Metadata of High Dynamic Range Television Systems"
- GB/T 41808-2022, "Image Parameter Values for High Dynamic Range Television Program Production and Exchange"
- GB/T 41809-2022, "Program Production and Exchange Parameter Values for Ultra High Definition Television Systems"
- GY/T 155-2000, "Video Parameter Values for High Definition Television Program Production and Exchange"
- GY/T 363-2023, "3D Audio Coding and Rendering"
- T/UWA 012.2—2023, "Ultra High Definition Audio-Visual Coding for the 'Hundred Cities, Thousand Screens' Ultra High Definition Audio-Visual Dissemination System: System"
- "The SRT Protocol draft-sharabayko-srt-01"
- ITU-T H.264-2021, "Advanced video coding for generic audiovisual services"
- ITU-T H.265-2024, "High efficiency video coding"
- EBU DVB ETSI TS 101 154 V2.8.1 (2023-07), "Specification for the use of Video and Audio Coding in Broadcast and Broadband Applications"
- ISO/IEC 14496-3:2019, "Information technology — Coding of audio-visual objects — Part 3: Audio"

- ETSI TS 102 366 v.1.4.1, "Digital Audio Compression (AC-3, Enhanced AC-3) Standard"

3 Conventions

The bitstream description method used in this document is similar to the C language. The syntax elements of the bitstream are indicated in bold type, and each syntax element is described by its name, syntax, and semantics. The numbering and counting in this document start from 0.

The descriptors used in this document are shown in Table 3-1.

Table 3-1 Descriptor

Descriptor	Description
u(n)	An n-bit unsigned integer. Uses big-endian byte order.
i(n)	An n-bit signed integer.

The following functions are used in this document for syntax description:

Table 3-2 Function

Function	Description
more_data()	If the length of the unparsed data contained in the current syntax layer is greater than 0, the function returns TRUE. If the length of the unparsed data contained in the current syntax layer is equal to 0, the function returns FALSE.

4 Format Definition of NVI Audio-Video Stream in TS PMT

The NVI audio-video stream is encapsulated using the Transport Stream (TS) specification defined in ISO/IEC 13818-1. The ISO/IEC 13818-1 standard specifies only a few stream_type formats. The ETSI TS standard and the T/UWA standard have supplemented the definitions for some other formats. For formats not specified in these standards, NVI has adopted the standard extension method for extended definitions, uniformly using the stream_type value of 0xFE, and specifying the exact format using a four-character code in the es_info_descriptor. The definition of NVI audio-video formats in the TS Program Map Table (PMT) is shown in Table 4-1.

Further definitions may be extended subsequently.

Table-4 Definition of NVI Audio-Video Formats in TS PMT

Format	stream_type	es_info_descriptor	Notes
H.264 (High Compression Video)	0x1B		Defined in ISO/IEC 13818-1
H.265 (High Compression Video)	0x24		Defined in ISO/IEC 13818-1
JPEG XS (Light Compression Video)	0x32		Defined in ISO/IEC 13818-1
AAC (Audio)	0x0F		Defined in ISO/IEC 13818-1
AC-3, E-AC-3 (Audio)	0x81	AC-3	Defined in ETSI TS 102 366
AVS+	0x42		Defined in GB/T 20090.2
AVS2	0xD2		Defined in T/UWA 012.2
AVS3 (High Compression Video)	0xD4	avs3	Defined in ETSI TS 101 154
Audio Vivid (Audio)	0xD5	av3a	Defined in T/UWA 012.2
NVI-HF (Light Compression Video)	0xFE	hfvc	NVI Extended Definition
AVS-HFM (Light Compression Video)	0xFE	avsf	NVI Extended Definition
SUVC (Light Compression Video)	0xFE	suvc	NVI Extended Definition
JPEG LS (Alpha)	0xFE	jpls	NVI Extended Definition
LPCM (Audio)	0xFE	lpcm	NVI Extended Definition
OPUS (Audio)	0xFE	Opus	NVI Extended Definition

5 Encapsulation Position and Constraints of NVI Metadata in TS Packet Header

NVI metadata is encapsulated in the `transport_private_data` field within the `adaptation_field` of the TS packet header. As shown in Figure 5-1.

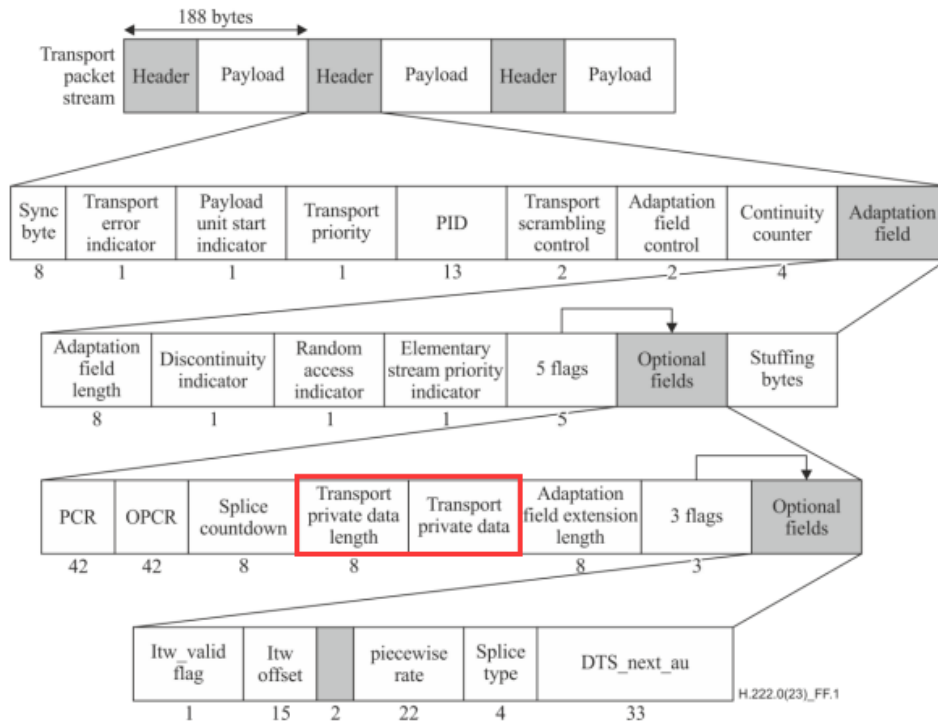


Figure 5-1 Schematic Diagram of TS Packet Header

When NVI metadata is encapsulated in the TS packet header, the relevant fields in the TS packet header shall comply with the limitations specified in Table 5-1.

Table 5-1 Constraints on TS Packet Header Fields for NVI Metadata Encapsulation

Field	Value	Meaning
adaptation_field_control	'10' or '11'	Indicates the presence of an adaptation field
adaptation_field_length	[9, 183]	Actual number of bytes in the adaptation field
PCR_flag	'0'	No PCR (Program Clock Reference) field present
OPCR_flag	'0'	No OPCR (Optional Program Clock Reference) field present
splicing_point_flag	'0'	No splicing field present
transport_private_data_flag	'1'	Indicates the presence of private data
adaptation_field_extension_flag	'0'	No adaptation field extension present
transport_private_data_length	[7, 181]	Actual number of bytes in the private data

6 Syntax Description of NVI Meta

6.1 Syntax Definition of TS_private_data

The definition of NVI metadata in TS_private_data is shown in Table 6-1.

Table 6-1 Definition of NVI Metadata in TS_private_data

TS_private_data() {	Descriptor
while (more_data()) {	
DVB_private_data_field_tag	u(8)
DVB_private_data_field_length	u(8)
NVI_metadata()	
}	
}	

6.2 Syntax Definition of NVI_metadata

The definition of NVI_metadata is shown in Table 6-2.

Table 6-2 The Definition of NVI_metadata

NVI_metadata() {	Descriptor
version	u(8)
metadata_type	u(8)
cur_packet_idx	u(8)
last_packet_flag	u(1)
reserved_1	u(7)
if (type==1) {	
NVI_video_metadata()	
}	
else if (type==2) {	
NVI_audio_metadata()	
}	
else if (type==16) {	
NVI_roi_metadata()	
}	
else if (type == 17) {	
NVI_hdr_static_metadata()	
}	
else if (type == 18) {	

NVI_hdr_dynamic_metadata()	
}	
else if (type==255) {	
NVI_private_metadata()	
}	
}	

6.3 Syntax Definition of NVI_video_metadata

The definition of NVI_video_metadata is shown in Table 6-3.

Table 6-3 Definition of NVI_video_metadata

NVI_video_metadata() {	Descriptor
video_fourcc	u(32)
video_frame_number	u(32)
video_frame_length	u(32)
video_tick	u(64)
video_time	i(64)
frame_rate_num	u(32)
frame_rate_den	u(32)
width	u(32)
height	u(32)
sar_width	u(16)
sar_height	u(16)
chroma_format	u(8)
bit_depth	u(8)
frame_type	u(8)
frame_field_flag	u(8)
rotation	u(8)
reserved_2	u(24)
colour primaries	u(8)
transfer_characteristics	u(8)
matrix_coefficients	u(8)
video_full_range_flag	u(8)
slice_mode	u(16)
slice_total_count	u(16)
slice_count_in_cur_pes	u(16)
cur_slice_idx	u(16)
nominal_bitrate	u(32)
}	

6.4 Syntax Definition of NVI_audio_metadata

The definition of NVI_audio_metadata is shown in Table 6-4.

Table 6-4 Definition of NVI_audio_metadata

NVI_audio_metadata() {	Descriptor
audio_fourcc	u(32)
audio_frame_number	u(32)
audio_frame_length	u(32)
audio_tick	u(64)
audio_time	i(64)
samples_rate	u(32)
bit_depth_format	u(16)
channels_configuration	u(16)
audio_bitrate	u(16)
}	

6.5 Syntax Definition of NVI_roi_metadata

The definition of NVI_roi_metadata is shown in Table 6-5.

Table 6-5 Definition of NVI_roi_metadata

NVI_roi_metadata() {	Descriptor
region_count	u(8)
for (i=0; i<region_count; i++) {	
region_label_length	u(8)
for (j=0; j<region_label_length; j++)	
region_label[j]	u(8)
region_left	u(16)
region_top	u(16)
region_width	u(16)
region_height	u(16)
}	
}	

6.6 Syntax Definition of NVI_hdr_static_metadata

The definition of NVI_hdr_static_metadata is shown in Table 6-6.

Table 6-6 Definition of NVI_hdr_static_metadata

NVI_hdr_static_metadata() {	Descriptor
hdr_static_type	u(8)
if (hdr_static_type==1)	
SMPTE_ST2086_static_metadata()	
else if (hdr_static_type==2)	
hdr_vivid_static_metadata()	
}	

Among them, SMPTE_ST2086_static_metadata() refers to the PQ static metadata, which is consistent with the mastering_display_colour_volume() defined in Annex D.2.28 of the ITU-T H.265 (V9) (09/2023) standard.

6.7 Syntax Definition of NVI_hdr_dynamic_metadata

The definition of NVI_hdr_dynamic_metadata is shown in Table 6-7.

Table 6-7 Definition of NVI_hdr_dynamic_metadata

NVI_hdr_dynamic_metadata() {	Descriptor
hdr_dynamic_type	u(8)
if (hdr_dynamic_type==2)	
hdr_vivid_dynamic_metadata()	
}	

The syntax definition of hdr_vivid_dynamic_metadata() can be found in Table 11 of the standard GY/T 358-2022

6.8 Syntax Definition of NVI_private_metadata

The definition of NVI_private_metadata is shown in Table 6-8.

Table 6-8 Definition of NVI_private_metadata

NVI_private_metadata() {	Descriptor
for (i=0; i< N; i++) {	
private_metadata_payload[i]	u(8)
}	

7 Semantic Description of NVI Metadata

7.1 Semantic Description of TS_private_data

- **DVB_private_data_field_tag**

An 8-bit unsigned integer. This field specifies the type of private data as defined in Annex D of DVB ETSI TS 101 154. For NVI metadata, the value shall be 0xC0.

- **DVB_private_data_field_length**

An 8-bit unsigned integer. This field indicates the length of the private data as defined in Annex D of DVB ETSI TS 101 154. For NVI metadata, the value shall be the actual length of the NVI metadata contained in the TS packet.

7.2 Semantic Description of NVI_metadata

- **Version**

An 8-bit unsigned integer. Indicates the version number of the NVI metadata. The current value should be 1.

- **metadata_type**

An 8-bit unsigned integer. Indicates the type of the NVI metadata.

Table 7-1 Metadata Types

Value	Meaning
0	Metadata type is empty. No data follows the reserved_1 field.
1	Metadata type is video basic metadata. NVI_video_metadata data follows immediately after the reserved_1 field.
2	Metadata type is audio basic metadata. NVI_audio_metadata data follows immediately after the reserved_1 field.
16	Metadata type is region of interest (ROI) metadata. NVI_roi_metadata data follows immediately after the reserved_1 field.
17	Metadata type is HDR static metadata. NVI_hdr_static_metadata data follows immediately after the reserved_1 field.
18	Metadata type is HDR dynamic metadata. NVI_hdr_dynamic_metadata data follows immediately after the reserved_1 field.
255	Metadata type is private metadata. NVI_private_metadata data follows immediately after the reserved_1 field.
Other values	Reserved

- **cur_packet_idx**

An 8-bit unsigned integer. Indicates the index number of the current metadata

packet. If the current metadata is split into multiple TS packets (n packets), the value of `cur_packet_idx` in each TS packet will be 0, 1, ..., n-1, respectively.

- **last_packet_flag**

A 1-bit unsigned integer. Indicates the end-of-PES flag or the end-of-metadata-packet flag.

If `metadata_type` is 0, `last_packet_flag` indicates whether this TS packet is the last packet of a PES (Packetized Elementary Stream). If the value is 0, it indicates that this TS packet is not the last packet of the PES. If the value is 1, it indicates that this TS packet is the last packet of the PES.

If `metadata_type` is not 0, `last_packet_flag` indicates whether this TS packet is the last packet of the metadata split. If the value is 0, it indicates that this TS packet is not the last packet of the metadata split. If the value is 1, it indicates that this TS packet is the last packet of the metadata split.

- **reserved_1**

A 7-bit unsigned integer. Currently reserved.

7.3 Semantic Description of NVI_video_metadata

NVI_video_metadata is used to describe the basic properties of the video stream.

- **video_fourcc**

A 32-bit unsigned integer. Indicates the video encoding format.

Table 7-2 Video Encoding Formats

c		Meaning
0x68323634 ('h', '2', '6', '4')		H.264
0x68323635 ('h', '2', '6', '5')		H.265
0x61767333 ('a', 'v', 's', '3')		AVS3
0x73757663 ('s', 'u', 'v', 'c')		SUVC
0x68667663 ('h', 'f', 'v', 'c')		NVI-HF
0x61767366 ('a', 'v', 's', 'f')		AVS HFM
0x6A787376 ('j', 'x', 's', 'v')		JPEG XS
Other values		Reserved

- **video_frame_number**

A 32-bit unsigned integer. Indicates the video frame number.

- **video_frame_length**

A 32-bit unsigned integer. Indicates the number of bytes in the current video

frame (or the current slice of the video).

- **video_ticz**

A 64-bit unsigned integer. Indicates the number of ticks elapsed, with each tick representing 1/90000 of a second.

- **video_time**

A 64-bit signed integer. Indicates the absolute time, measured in microseconds, starting from 00:00:00 UTC on January 1, 1970.

- **frame_rate_num**

A 32-bit unsigned integer. Indicates the numerator of the video frame rate.

- **frame_rate_den**

A 32-bit unsigned integer. Indicates the denominator of the video frame rate.

- **width**

A 32-bit unsigned integer. Indicates the actual width of the video.

- **height**

A 32-bit unsigned integer. Indicates the actual height of the video.

- **sar_width**

A 16-bit unsigned integer. Indicates the width component of the display aspect ratio.

- **sar_height**

A 16-bit unsigned integer. Indicates the height component of the display aspect ratio.

- **chroma_format**

An 8-bit unsigned integer. Indicates the chroma sampling attributes of the video.

Table 7-3 Chroma Sampling

Values	Meaning
0	Grayscale
1	4: 2: 0
2	4: 2: 2
3	4: 4: 4
Other values	Reserved

- **bit_depths**

An 8-bit unsigned integer. Indicates the bit depth of the video.

Table 7-4 Video Bit Depth

Values	Meaning
8	8bit depth
10	10bit depth

12	12bit depth
14	14bit depth
16	16bit depth
Other values	Reserved

- **frame_type**

An 8-bit unsigned integer. Indicates the type of the current video frame. NVI video types should only use I-frames and P-frames, and B-frames should not be used. If the value is 1, it indicates that the current video frame is an I-frame. If the value is 2, it indicates that the current video frame is a P-frame.

- **frame_field_flag**

An 8-bit unsigned integer. Indicates the progressive or interlaced attribute of the video. If the value is 0, it indicates that the video is progressive. If the value is 1, it indicates that the video is interlaced.

- **Rotation**

An 8-bit unsigned integer. Indicates the rotation attribute of the video.

Table 7-5 Rotation Attributes

Values	Meaning
0	No rotation
1	Rotate 90 degrees clockwise
2	Rotate 180 degrees clockwise
3	Rotate 270 degrees clockwise
Other values	Reserved

- **reserved_2**

A 24-bit unsigned integer. Currently reserved.

- **colour primaries**

An 8-bit unsigned integer. Indicates the colour primaries of the video. Values are defined in Table E.3 of ITU-T H.265. Common values are shown in the following table.

Table 7-6 Common Values for colour primaries

Values	Meaning
1	Rec. ITU-R BT.709-6
5	Rec. ITU-R BT.601-7 625
9	Rec. ITU-R BT.2020-2
Other values	Refer to Rec. ITU-T H.265 Table E.3

- **transfer_characteristics**

An 8-bit unsigned integer. Indicates the transfer characteristics (gamma function) of the video. Values are defined in Table E.4 of ITU-T H.265. Common values are

shown in the following table.

Table 7-7 Common Values for transfer_characteristics

Values	Meaning
1	Rec. ITU-R BT.709-6
6	Rec. ITU-R BT.601-7
14、15	Rec. ITU-R BT.2020-2
16	Rec. ITU-R BT.2100-2 perceptual quantization (PQ) system
18	Rec. ITU-R BT.2100-2 hybrid loggamma (HLG) system
Other vlues	Refer to Rec. ITU-T H.265 Table E.4

- **matrix_coefficients**

An 8-bit unsigned integer. Indicates the RGB/YUV conversion matrix used in the video. Values are defined in Table E.5 of ITU-T H.265. Common values are shown in the following table.

Table 7-8 Common Values for matrix_coefficients

Values	Meaning
1	Rec. ITU-R BT.709-6
5	Rec. ITU-R BT.601-7 625
9	Rec. ITU-R BT.2020-2 non-constant luminance system
Other vlues	Refer to Rec. ITU-T H.265 Table E.5

- **video_full_range_flag**

An 8-bit unsigned integer. Indicates the video signal range. If the value is 0, it indicates that the video signal range is limited. If the value is 1, it indicates that the video signal range is full.

- **slice_mode**

An 8-bit unsigned integer. Indicates the mode used for video slice partitioning.

Table 7-9 Slice Modes

Values	Meaning
0	No multiple slices within a frame.
1	Multiple slices within a frame. Encoded by a single encoder.
2	Multiple slices within a frame. Encoded by multiple encoders.
Other Values	Reserved

- **slice_total_count**

A 16-bit unsigned integer. Indicates the total number of slices contained in the

video frame.

- **slice_count_in_cur_pes**

A 16-bit unsigned integer. Indicates the number of slices contained in the current PES (Packetized Elementary Stream).

- **cur_slice_idx**

A 16-bit unsigned integer. Indicates the index number of the current slice.

- **video_nominal_bitrate**

A 32-bit unsigned integer. Indicates the nominal bitrate of the video. The unit is kbps (kilobits per second).

7.4 Semantic Description of NVI_audio_metadata

NVI_audio_metadata is used to describe the basic properties of the audio stream.

- **audio_fourcc**

A 32-bit unsigned integer. Indicates the audio encoding format. Additional audio formats can be supported through extension.

Table 7-10 Audio Encoding Formats

Values	Meaning
0x6C70636D ('l', 'p', 'c', 'm')	LPCM
0x6D703461 ('m', 'p', '4', 'a')	AAC
0x47707573 ('O', 'p', 'u', 's')	OPUS
0x61763361 ('a', 'v', '3', 'a')	Audio Vivid
0x41432D33 ('A', 'C', '-', '3')	Dolby AC-3 E-AC-3
Other vlues	Reserverd

- **audio_frame_number**

A 32-bit unsigned integer. Indicates the audio frame number.

- **audio_frame_length**

A 32-bit unsigned integer. Indicates the number of bytes in the current audio frame.

- **audio_tick**

A 64-bit unsigned integer. Indicates the number of ticks elapsed, with each tick representing 1/(audio sampling rate) of a second.

- **audio_time**

A 64-bit signed integer. Indicates the absolute time, measured in microseconds, starting from 00:00:00 UTC on January 1, 1970.

- **samples_rate**

A 32-bit unsigned integer. Indicates the audio sampling rate.

- **bit_depth_format**

A 16-bit unsigned integer. Indicates the bit depth of each audio sample and the representation method.

Table 7-11 Audio Sample Bit Depth and Representation

Value	Meaning
0x0008	8bit integer
0x0010	16bit integer,host byte order
0x0018	24bit integer,host byte order
0x0020	32bit floating-point,host byte order
0x0110	16bit integer,little-endian byte order
0x0118	24bit integer,little-endian byte order
0x0120	32bit floating-point,little-endian byte order
0x0210	16bit integer,big-endian byte order
0x0218	24bit integer,big-endian byte order
0x0220	32bit floating-point,big-endian byte order
Other values	Reserved

- **channels_configuration**

A 16-bit unsigned integer. Indicates the number of audio channels and the channel configuration.

Table 7-12 Number of Audio Channels and Channel Configurations

Value	Meaning
0x0001	Mono (single channel)
0x0002	Stereo (two channels)
0x0006	5.1 channels
0x0007	6.1 channels
0x0008	7.1 channels
0x0106	FOA (Audio Vivid,4 channels)
0x0107	5.1.2 channels
0x0108	5.1.4 channels
0x0109	7.1.2 channels
0x010a	7.1.4 channels
0x010b	Third-order HOA (Audio Vivid,16 channels)
0x010c	Third-order HOA (Audio Vivid,9 channels)
Other values	Reserved

- **audio_bitrate**

A 16-bit unsigned integer. Indicates the bitrate of the audio. The unit is kbps (kilobits per second).

7.5 Semantic Description of NVI_roi_metadata

The NVI_roi_metadata structure is used to describe regions of interest (ROI) in the video frame, which can be applied for applications such as landscape/portrait area cropping and AI object labeling.

- **region_count**
An 8-bit unsigned integer. Indicates the number of ROI regions.
- **region_label_length**
An 8-bit unsigned integer. Indicates the length of the label for this ROI region.
- **region_label**
The label for this ROI region. Its length is specified by region_label_length.
- **region_left**
A 16-bit unsigned integer. Indicates the x-coordinate of the top-left corner of this ROI region.
- **region_top**
A 16-bit unsigned integer. Indicates the y-coordinate of the top-left corner of this ROI region.
- **region_width**
A 16-bit unsigned integer. Indicates the width of this ROI region.
- **region_height**
A 16-bit unsigned integer. Indicates the height of this ROI region.

7.6 Semantic Description of NVI_hdr_static_metadata

The NVI_hdr_static_metadata structure is used to store HDR static metadata.

- **hdr_static_type**
An 8-bit unsigned integer. Indicates the type of HDR static metadata. If hdr_static_type is 1, it indicates that the HDR static metadata type is SMPTE ST2086 PQ static metadata. The detailed semantic description is specified in Annex D.3.28 of ITU-T H.265 (V9) (09/2023). If hdr_static_type is 2, it indicates that the HDR static metadata type is HDR Vivid static metadata. The detailed semantic description is specified in Table 10 of GY/T 358-2022.

7.7 Semantic Description of NVI_hdr_dynamic_metadata

The NVI_hdr_dynamic_metadata structure is used to store HDR Vivid dynamic metadata.

- **hdr_dynamic_type**

An 8-bit unsigned integer. Indicates the type of HDR dynamic metadata. If hdr_dynamic_type is 2, it indicates that the dynamic metadata type is HDR Vivid dynamic metadata. The structure of hdr_vivid_dynamic_metadata is consistent with the HDR Vivid dynamic metadata definition specified in Table 11 of GY/T 358-2022. The length of the dynamic metadata should be DVB_private_data_field_length - 5.

7.8 Semantic Description of NVI_private_metadata

NVI_private_metadata is used to store user-defined metadata. NVI does not parse this part of the data and only performs transparent transmission.

- **private_metadata_payload**

The payload of the private metadata. The length should be DVB_private_data_field_length - 4. The specific content of the data is defined by the user.

8 Definition of NVI Metadata Not Encapsulated in TS

In addition to metadata encapsulated and transmitted using TS (Transport Stream), NVI also handles some metadata that is not closely related to real-time audio and video streams. For this type of metadata, NVI does not use TS encapsulation. Instead, it uses JSON format for encapsulation and transmits it through NVI's WebSocket transport channel.

8.1 Definition of NVI_Tally JSON

NVI_Tally is the root node of the JSON structure, and the definitions of its child nodes are shown in Table 8-1.

Table 8-1 Definition of NVI_Tally Child Nodes

Child Node	Property	Meaning	Value
------------	----------	---------	-------

Technical White Paper on NVI

device	index	Tally signal number	[0, 65535]
	text	Display text	String,length <=255
	color	Tally light display	0: OFF; 1: RED; 2: GREEN; 3: AMBER

8.2 Definition of PTZ JSON

NVI_PTZ is the root node of the JSON structure, and the definitions of its child nodes are shown in Table 8-2.

Table 8-2 Definition of NVI_PTZ Child Nodes

Child Node	Property	Meaning	Value
ptz_zoom	zoom	Zoom factor of the frame	[0.0, 1.0]
ptz_zoom_speed	zoom_speed	Speed of zooming	[-1.0, 1.0]
ptz_pan_tilt	pan	Horizontal control of framing direction	[-1.0, 1.0]
	tilt	Tilt control of framing direction	[-1.0, 1.0]
ptz_pan_tilt_speed	pan_speed	Horizontal movement speed	[-1.0, 1.0]
	tilt_speed	Tilt movement speed	[-1.0, 1.0]
ptz_focus	mode	ocus control mode	auto: Automatic manual: Manual
	distance	Focus control distance	[-1.0, 1.0]
ptz_white_balance	mode	White balance mode	auto: Automatic indoor: indoor outdoor: outdoor one_push: 18% gray card manual: Manual
	red	Red coefficient	[0.0, 1.0]
	blue	Blue coefficient	[0.0, 1.0]
ptz_exposure	mode	Exposure mode	auto: Automatic manual: Manual
	iris	Aperture	[0.0, 1.0]
	gain	Gain	[0.0, 1.0]
	shutter	Shutter	[0.0, 1.0]
ptz_recall_preset	index	Preset number to	[0, 255]

		recall	
	speed	Movement speed	[0.0, 1.0]

9 NVI TS Audio-Video Data and Metadata Storage Diagram

NVI real-time audio and video stream data is encapsulated in PES (Packetized Elementary Stream) according to the MPEG-TS standard for transmission. For real-time video streams, if a video frame is divided into multiple slices, each PES contains the data of one slice. If the video frame is not divided into multiple slices, each PES contains the data of one frame. For real-time audio streams, each PES contains the data of one audio frame.

NVI metadata is stored in the TS packet header and corresponds to the PES. Before each TS packet containing a PES header, the corresponding NVI video metadata or NVI audio metadata will appear. In the TS packet containing the end of a PES packet, NVI metadata indicating the end of a frame or slice (`last_packet_flag=1`) will appear.

The following diagram illustrates a scenario with video (each frame divided into 4 slices), audio, and metadata. In the diagram, "afc" is the abbreviation for the `adaptation_field_control` field in the TS packet header, and "tpdl" is the abbreviation for the `transport_private_data_flag` field.

Technical White Paper on NVI

TS packet 188 bytes

TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=1 last_flag=0	NVI_video_metadata cur_slice_idx=0	Padding Data	
TS header afc=0	Video PES header	Video slice 0 data (part 1 / n)			
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=2 last_flag=0	NVI_audio_metadata	Padding Data	
TS header afc=0	Audio PES header	Audio Frame Data (part 1 / m)			
TS header afc=0	Video slice 0 Data (part 2 / n)				
TS header afc=0	Audio Frame Data (part 2 / m)				
.....					
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=0 last_flag=1	Video slice 0 Data (part n / n)		Padding Data
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=1 last_flag=0	NVI_video_metadata cur_slice_idx=1	Padding Data	
TS header afc=0	Video PES header	Video slice 1 Data(part 1 / n)			
.....					
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=0 last_flag=1	Video slice 1 Data(part 1 / n)		Padding Data
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=1 last_flag=0	NVI_video_metadata cur_slice_idx=2	Padding Data	
TS header afc=0	Video PES header	Video slice 2 Data(part 1 / n)			
.....					
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=0 last_flag=1	Video slice 2 Data(part n / n)		Padding Data
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=1 last_flag=0	NVI_video_metadata cur_slice_idx=3	Padding Data	
TS header afc=0	Video PES header	Video slice 3 Data (part 1 / n)			
.....					
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=0 last_flag=1	Video slice 3 Data (part n / n)		Padding Data
TS header afc=10 tpdl=1	DVB tag & len	NVI_metadata meta_type=0 last_flag=1	Audio Frame Data (part m / m)		Padding Data

Figure 9-1 Diagram of NVI TS Audio-Video Data and Metadata Storage