

FBMF Standard

(기술보고서는 FBMF Technical Report)

미래방송미디어포럼표준(국문표준)

FBMF-STD-033

제정일: 2026. 04. 10.

방송 제작을 위한 원격기기 제어 인터페이스 표준

Remote Device Control Interface in
Broadcasting Production



표준초안 검토 위원회 UHD융합기술 분과위원회

표준안 심의 위원회 운영위원회

	성명	소속	직위	위원회 및 직위
표준(과제) 제안	김순철	ETRI	책임	UHD융합기술분과위원회/위원
표준 초안 에디터	오혜주	ETRI	책임	UHD융합기술분과위원회/위원
	이재호	KBS	수석	UHD융합기술분과위원회/위원

사무국 담당

-

본 문서에 대한 저작권은 미래방송미디어표준포럼에 있으며, 미래방송미디어표준포럼과 사전 협의 없이 이 문서의 전체 또는 일부를 상업적 목적으로 복제 또는 배포해서는 안 됩니다.

본 표준 발간 이전에 접수된 지식재산권 약삭서 정보는 본 표준의 '부록(지식재산권 약삭서 정보)'에 명시하고 있으며, 이후 접수된 지식재산권 약삭서는 미래방송미디어표준포럼 웹사이트에서 확인할 수 있습니다.

본 표준과 관련하여 접수된 약삭서 외의 지식재산권이 존재할 수 있습니다.

발행인 : 미래방송미디어표준포럼 의장

발행처 : 미래방송미디어표준포럼

06130, 서울특별시 강남구 테헤란로 7길 22 신관 1108호

Tel : 02-568-3556, Fax : 02-568-3557

발행일 : 2026.04

서 문

1 표준의 목적

본 표준의 목적은 로컬의 방송 제작 기기들(예: 카메라, 마이크, 조명 등)과 원격의 비디오 스위칭 서버(혹은 클라우드 비디오 스위처)간 장치제어 및 관리를 위한 시스템 구성요소들 간 인터페이스를 정의하는 데 있다. 이를 통해, 원격으로 라이브 방송 제작과 송출을 수행하는 참여자들 간의 제작 장비 사용 접근과 제어를 용이하도록 지원한다. 본 문서에서 원격이라 함은, 물리적 거리뿐만 아니라 제작자들 간에 비대면 제작 협업 환경을 포함한다. 예를 들어, 제작자는 촬영장에 직접 가지 않고서도 원격의 촬영장에 있는 카메라의 팬/틸트/줌을 수행하거나 PRV/PGM 스위칭 제어에 따른 탈리 신호를 전송할 수 있으며, 촬영장의 조명기기의 파라미터를 변경하는 등의 장치 제어를 할 수 있으며, 촬영된 영상을 실시간으로 모니터링할 수 있다.

2 주요 내용 요약

본 표준은 방송프로그램 제작자(프로듀서, 1인창작자 등)가 비디오 스위칭 서버에서 원격에서 방송 제작에 참여하는 다수의 제작장비들을 쉽게 식별하고 제어할 수 있는 인터페이스를 정의한다. 이를 위하여 다음 항목을 포함하여 인터페이스를 정의한다.

- 원격기기 발견/등록(Remote Discovery/Registration): 제작장비의 기기명, 타입, 용도, 성능, 네트워크 정보 등 속성값을 서버에 제공
- 원격기기 연결 제어(Remote Connection): 카메라의 영상 송출 개시/중지 수행, 조명 혹은 마이크의 온/오프 수행 명령
- 원격기기 제어(Remote Control): 기기 고유의 동작 수행 명령
 - 카메라: 팬/틸트/줌(PTZ) 제어, 색상/화이트밸런스 등 제어
 - 조명: 색온도/밝기레벨 등 파라미터 제어
- 원격기기 탈리 알림(Remote Tally): PRV/PGM 비디오 스위칭 제어에 따른 탈리 신호 전송

3 인용 표준과의 비교

해당 사항 없음

Preface

1 Purpose

The purpose of this standard is to define the system architecture, functionalities, and interfaces for device control and management between local broadcast production equipment (e.g., cameras, microphones, lighting) and a remote video switching server (or cloud-based video switcher). This enables participants involved in live remote production and broadcasting to conveniently access and control production devices.

2 Summary

Broadcast program producers (e.g., producers, individual creators) can remotely identify and control multiple production devices through the standardized control interfaces provided between devices and servers.

For example, a producer may control camera pan/tilt/zoom (PTZ), trigger tally signals according to PRV/PGM switching commands, adjust parameters of lighting devices, and monitor captured video streams in real time from a remote location.

The remote device control interface standard for broadcast production includes the following elements between devices and the server:

- **Remote Discovery/Registration:** Devices provide their attributes to the server, such as device name, type, usage, performance, and network information.
- **Remote Connection Control:** Execution of commands such as starting/stopping video output from cameras, or turning on/off lighting and microphones.
- **Remote Device Control:** Execution of device-specific operational commands:
 - *Camera:* Pan/Tilt/Zoom (PTZ) control
 - *Lighting:* Control of parameters such as color temperature and brightness level
- **Remote Tally Event:** Issuance of tally signals in accordance with PRV/PGM video switching commands

3 Relationship to Reference Standards

목 차

1 적용 범위	1
2 인용 표준	1
3 용어 정의	1
4 약어	2
5 원격 제작 시스템 개요	3
5.1 원격 제작 서비스 개요	3
5.2 NMOS 리소스와 MQTT 프로토콜 개요	3
6 원격기기 제어 인터페이스	6
6.1 원격기기 제어 동작 절차	6
6.2 MQTT 기반 메시지 전송을 위한 구조	7
6.2.1 공통 MQTT 토픽 정의	8
6.3 장치 발견 등록	9
6.4 장치 연결 제어	10
6.5 제작 장치 제어	12
6.5.1 카메라 제어	12
6.5.2 조명 제어	15
6.6 탈리 알림	17
부록 1-1 지식재산권 요약서 정보	19
1-2 시험인증 관련 사항	20
1-3 본 표준의 연계(family) 표준	21
1-4 참고 문헌	22
1-5 영문표준 해설서	23
1-6 표준의 이력	24

방송 제작을 위한 원격기기 제어 인터페이스 (Remote Device Control Interface in Broadcast Production)

1 적용 범위

본 표준은 로컬의 방송 제작 기기들(예: 카메라, 마이크, 조명 등)과 원격의 비디오 스위칭 서버(혹은 클라우드 비디오 스위처)간 장치제어 및 관리를 위한 시스템 구성요소들 간 인터페이스를 정의한다. 이를 통해, 원격으로 라이브 방송 제작과 송출을 수행하는 참여자들 간의 제작 장비 사용 접근과 제어를 용이하도록 지원한다.

본 표준에서 정의하는 인터페이스는 인용 표준인 NMOS의 리소스 모델(Node, Device, Source, Flow, Sender, Receiver 등)을 활용하여 장치 식별 및 관리 체계를 유지하여, 기존 NMOS 기반 시스템과의 호환성을 지원한다. 따라서, 로컬 NMOS 레지스트리(RDS) 및 컨트롤러 시스템과의 통합 운용이 가능하다. 또한, 인용 표준인 NMOS가 HTTP/REST 기반의 제어 방식을 정의하고 있지만, 본 표준에서는 원격 제작 환경에서의 효율적인 메시지 전달을 위해 MQTT 기반 발행/구독 메시지 구조를 정의한다.

본 표준은 다음과 같은 적용 효과를 기대할 수 있다.

- 상호운용성 향상: 다양한 제조사와 유형의 제작장비 간 제어 및 관리 명령의 호환성을 확보함으로써, 다양한 장치 구성의 제작 환경에서도 일관된 운영이 가능하다.
- 원격 운용 효율성 향상: 원격지에서도 장비 식별, 연결, 제어가 가능하여 제작 현장의 인력 및 장비 운영 효율을 높인다.
- 범용 표준 활용 제고: NMOS, MQTT 등 공통 프로토콜 기반의 통신 구조를 사용함으로써 장치 등록, 제어, 탈리 관리 등의 기능을 손쉽게 적용 가능하다.
- 다중 참여자 협업 향상 기대: 다중 참여자(프로듀서, 엔지니어, 크리에이터 등)가 협업 가능한 인프라 구축에 기여한다.

2 인용 표준

- [1] AMWA IS-04 NMOS Discovery and Registration Specification, v1.3.3
- [2] AMWA IS-05 NMOS Device Connection Management Specification, v1.1.2
- [3] AMWA IS-07 NMOS Event & Tally Specification, v1.0.1

3 용어 정의

3.1 원격기기 제어(Remote Device Control)

물리적으로 먼 거리에 위치한 제작참여자가 원격의 제작장비(카메라, 마이크, 조명 등)를 동작·운영하는 기술 및 방법

3.2 IP 미디어 게이트웨이(IP Media Gateway)

SDI와 같은 기존 방송 인터페이스와 IP 기반 미디어 네트워크 간의 미디어 신호를 변환하거나 중계하는 장치로서, 미디어 스트림의 인코딩, 패킷화, 전송 프로토콜 변환 등의 기능을 포함

3.3 비디오 스위처(Video switcher)

여러 개의 영상 입력 신호를 실시간으로 선택/전환(Program, Preview)하고 합성하는 방송 제작 장비

3.4 등록/발견 서버(RDS)

네트워크에 연결된 미디어 장치의 리소스 정보(Node, Device, Source 등)를 등록하고 관리하는 장치 서버

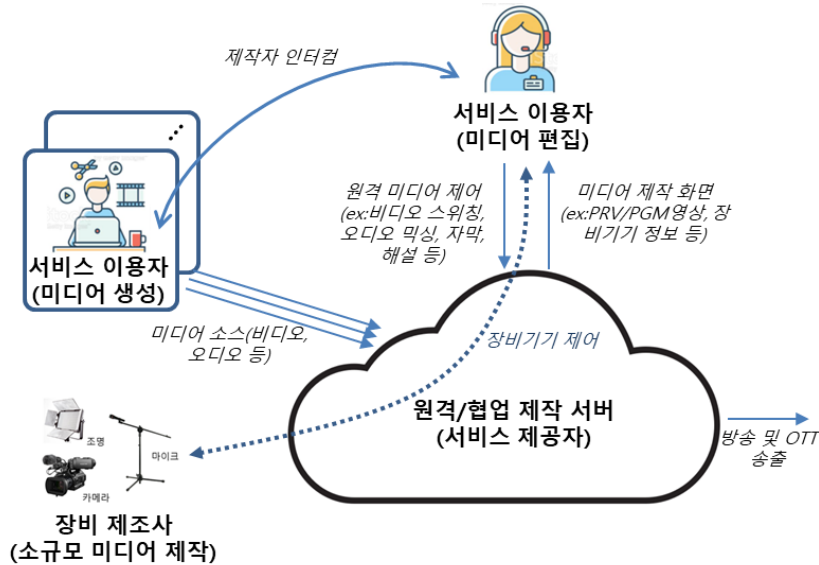
4 약어

AMWA	Advanced Media Workflow Association
MQTT	Message Queuing Telemetry Transport
NMOS	Networked Media Open Specifications
PGM	PROGRAM
PRV	PREVIEW
RDS	Registration & Discovery System
RTMP	Real-Time Messaging Protocol
RTP	Real-Time Transport Protocol
SDI	Serial Digital Interface
SMPTE	Society of Motion Picture and Television Engineers
SRT	Secure Reliable Transport
UDP	User Datagram Protocol

5. 원격 제작 시스템 개요

5.1 원격 제작 서비스 개요

방송 제작 장비의 원격 관리 및 제어 기능은 제작자가 서버를 통해 카메라, 마이크, 조명과 같은 원격지의 제작 장치를 연결 관리할 수 있도록 하는 기능이다. <그림 1>은 원격 제작 시스템에서의 제작 서비스 구성을 보여준다. 인터넷 환경에서 미디어 제작 장치를 원격으로 관리하기 위해 NMOS 개방형 표준과 MQTT 프로토콜을 기반으로 발견, 등록, 원격 연결, 리소스 관리, 탈리 기능을 가능하게 한다.

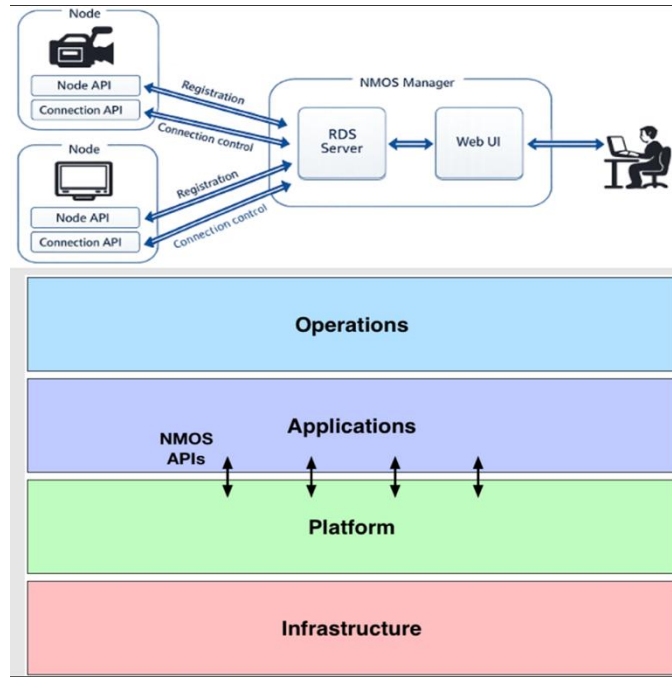


<그림 1> 원격 제작 시스템 서비스 구성도

5.2 NMOS 리소스와 MQTT 프로토콜 개요

NMOS는 AMWA(Advanced Media Workflow Association)에서 제정한 IP 기반 전문 미디어 시스템의 상호운용성 확보를 위한 개방형 규격이다. <그림 2>에 나타낸 바와 같이, NMOS는 SMPTE ST 2110 실시간 비압축 IP 미디어 전송 환경에서 로컬 IP 네트워크 상의 미디어 장비들을 자동으로 발견 등록하고, 연결을 설정하며, 이벤트, 상태 정보의 교환을 수행할 수 있도록 설계되었다. NMOS는 다음과 같은 주요 기능을 제공함으로써 방송 제작 장비 간의 상호운용성(Interoperability)을 보장하며 제조사 종속성이 없는 개방형 생태계를 구축한다.

- IS-04 Discovery & Registration: 네트워크 내 장치 및 서비스의 자동 탐지 및 등록
- IS-05 Connection Management: 엔드포인트 간 미디어 스트림 연결 설정 및 관리
- IS-07 Event & Tally: 이벤트 기반 신호 전달 및 제작 시스템의 탈리 상태 제어



<그림 2> NMOS API 기반 로컬 제작 기기의 등록 및 제어 관리

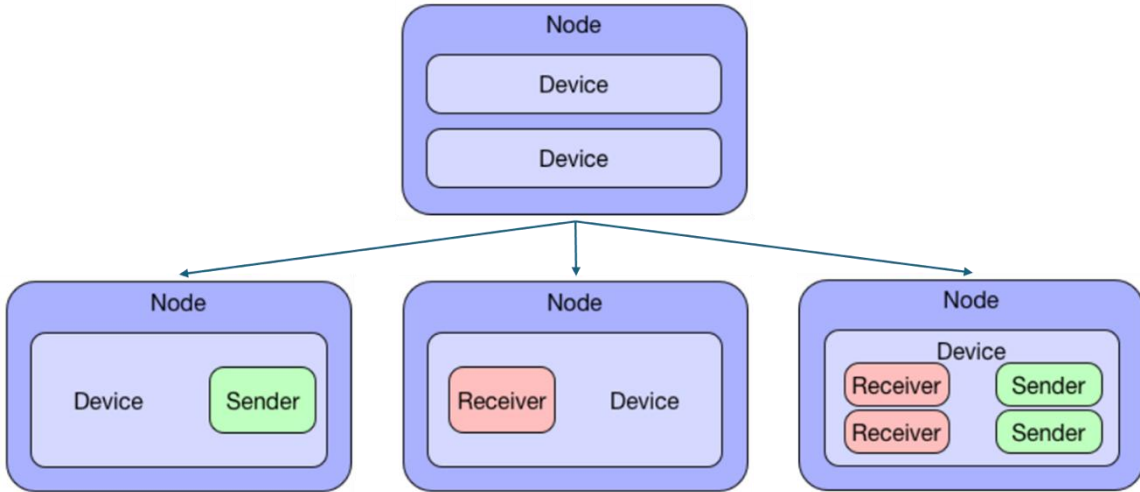
NMOS 데이터 모델은 물리적인 장비를 6개의 논리적인 리소스(Node, Device, Sender, Receiver, Source, Flow)들이 계층적으로 중첩된 구조를 가지며, 각 리소스는 고유한 UUID를 갖는다. Node는 네트워크상의 물리적 또는 가상 리소스이며, IP 기반 방송 환경에서 다른 시스템이 이를 탐색하고 연결할 수 있도록 하는 상위 계층 단위이다. <그림 3>은 NMOS Node를 구성하는 핵심 요소를 Device 수준에서 도시한 것으로, NMOS Node는 최소 하나 이상의 Sender 또는 Receiver를 포함한다.

● Sender

- Node 가 외부로 제공하는 미디어 데이터 스트림
- 예: 카메라 → 1080p60 ST 2110-20 Video Sender
- 속성:
 - ✓ Format (video/audio/data)
 - ✓ Transport Params (source IP, multicast, port)
 - ✓ Flow ID 와 연결

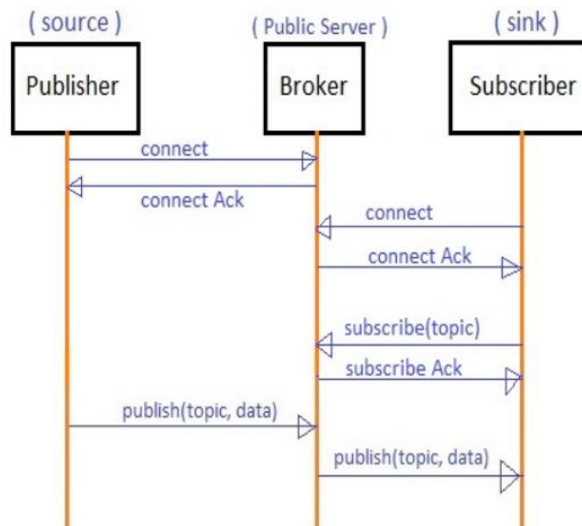
● Receiver

- Node 가 외부에서 미디어를 수신할 수 있는 자원
- 예: 모니터 · 스위치 입력 포트
- 속성:
 - ✓ 지원 Format
 - ✓ Transport Params (수신 IP, join/leave 명령 지원)



<그림 3> NMOS Node 의 다양한 구성 예시

MQTT는 IoT, 산업제어, 원격장비 관리 분야에서 널리 사용되는 경량 메시지 통신 프로토콜로, 네트워크 대역폭 사용과 시스템 자원 소모가 매우 적도록 설계되었다. 특히, 불안정한 네트워크 환경에서도 안정적인 메시지 전송이 가능하여 원격 방송 제작 환경에 적합한 제어·상태 모니터링 채널로 활용된다. MQTT는 <그림 4>에 나타낸 바와 같이, 중앙에 위치한 Broker를 통해 Publisher와 Subscriber가 메시지를 주고받는 Publish/Subscribe 구조로 동작한다. 클라이언트는 Broker에 연결한 뒤 특정 Topic을 구독하거나 해당 Topic으로 메시지를 발행하며, Broker는 같은 Topic을 구독 중인 모든 Subscriber에게 메시지를 전달하는 방식으로 송신자와 수신자가 직접 연결되지 않아 구조가 단순하고 확장성이 높다.



<그림 4> MQTT 기본 동작 개요

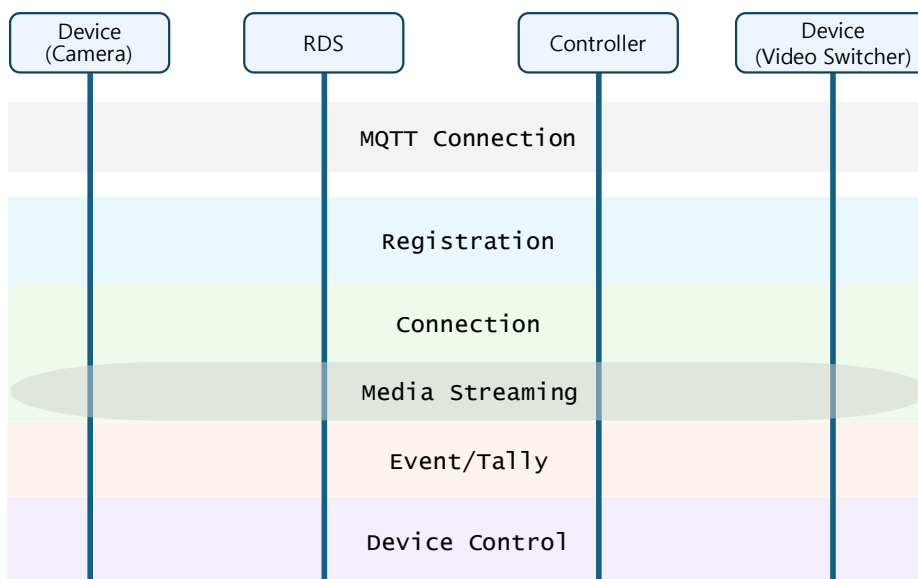
6 원격기기 제어 동작 절차 및 인터페이스

6.1 원격기기 제어 동작 절차

원격기기 제어 인터페이스는 IP 네트워크 환경에서 동작하는 미디어 장치의 상호 운용성을 확보하기 위해 정의된다. 본 표준의 인터페이스는 서로 다른 제조사 및 기능을 갖는 장치들에 대해 공통된 방식의 발견(Discovery), 등록(Registration), 상태 조회(Monitoring), 연결 관리(Connection Management), 및 동작 제어(Control)를 가능하게 하며, 네트워크 기반 제작 시스템에서 장치 간 통합 운용을 지원한다.

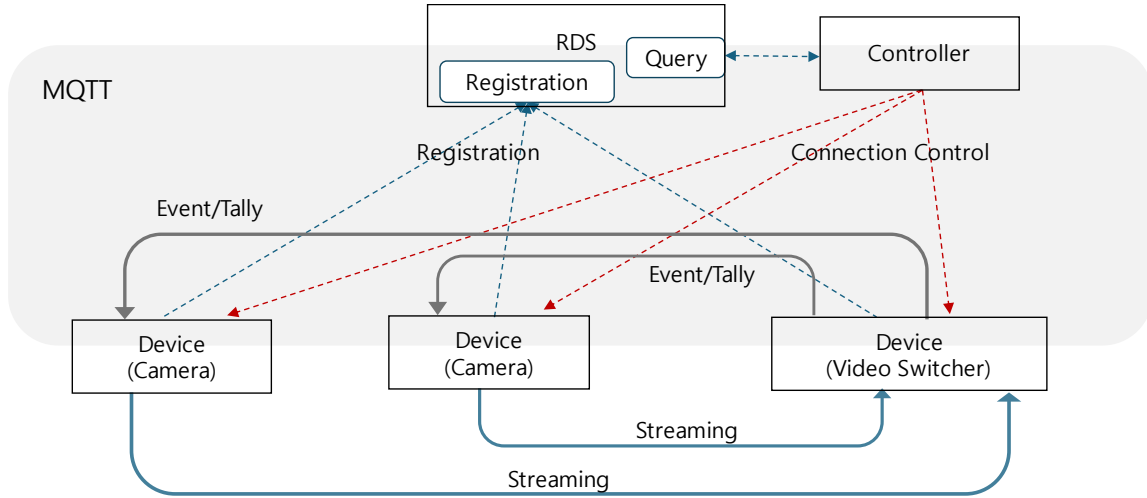
방송 제작 환경에서의 원격기기 제어는 <그림 5>에 나타난 동작 절차를 따른다. 장치가 네트워크에 접속하면 사전에 정의된 방식으로 레지스트리의 위치를 발견하고 해당 주소를 이용하여 자신을 등록한다. 등록 정보는 계층적 리소스 모델로 구성되며, 장비 자체를 나타내는 Node, 기능 단위를 의미하는 Device, 오디오/비디오 신호의 논리적 출처 Source, 스트림 속성을 기술하는 Flow, 그리고 송신 및 수신 엔드포인트 Sender/Receiver를 포함한다. 컨트롤러는 레지스트리에 등록된 정보를 조회하여 네트워크에 존재하는 장치의 구성, 상태 및 제공 기능을 파악한다. 이를 기반으로 특정 Sender와 Receiver 간의 미디어 연결을 설정하거나 변경한다. 이 과정에서 전송에 필요한 네트워크 파라미터와 연결의 활성 상태가 함께 제어된다. 연결 설정이 완료되면 실제 미디어 스트림이 네트워크를 통해 전송된다.

원격기기 제어 기능은 스트림 연결 관리에 한정되지 않으며, 장치의 상태 제어 및 이벤트 기반 동작 관리로 확장될 수 있다. 컨트롤러는 제어 인터페이스를 통해 장치의 동작 상태를 변경할 수 있으며, 장치에서 발생하는 이벤트 및 상태 변화를 수신하여 시스템 전체의 동작을 동기화한다.



<그림 5> 원격기기 제어 동작 절차

원격기기 제어 인터페이스는 NMOS 미디어 관리 체계를 기반으로 하며, 여기에 MQTT 기반 메시징 시스템을 결합하여 운영된다. 이러한 구조는 원격 제작 환경에서 요구되는 실시간 이벤트 전달, 제어 명령 전송, 상태 정보 동기화를 효율적으로 수행하기 위한 것이다. 실시간 이벤트, 제어, 상태 동기화를 위한 MQTT 기반의 원격기기 제어 인터페이스 구조는 <그림 6>과 같다.



<그림 6> 원격기기 제어 인터페이스 구조

6.2 MQTT 기반 메시지 전송을 위한 구조

본 표준은 NMOS의 리소스 모델(Node, Device, Source, Flow, Sender, Receiver 등)을 그대로 활용하여 장치 식별 및 관리 체계를 유지하며, 기존 NMOS 기반 시스템과의 연동을 고려한다. 따라서, 로컬 NMOS 레지스트리(RDS) 및 컨트롤러 시스템과의 통합 운용이 가능하며, 기존 NMOS가 HTTP/REST 기반의 제어 방식을 사용하는 것과 달리, 본 표준에서는 원격 제작 환경에서의 효율적인 메시지 전달을 위해 MQTT 기반 Publish/Subscribe 메시지 구조를 추가로 정의한다. MQTT payload에 적재되는 NMOS 데이터는 <표 1>에 나타낸 바와 같이 JSON 포맷으로 구성한다.

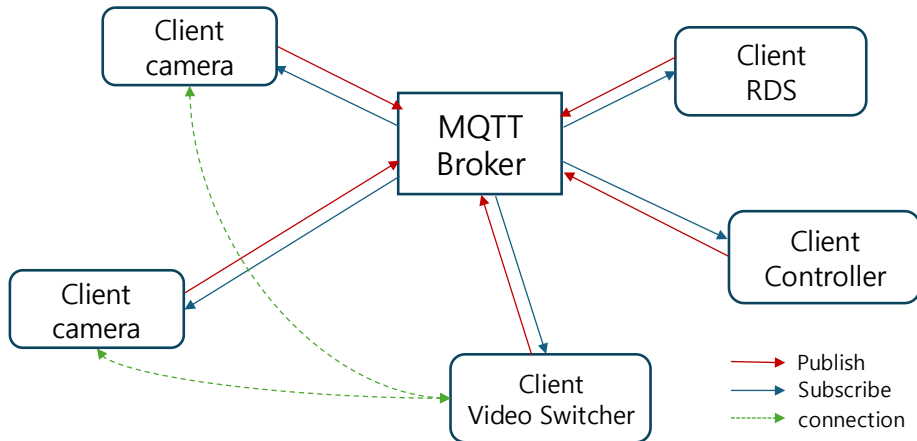
<표 1> MQTT 헤더 및 페이로드 구조

상위 필드	하위 필드	설명
MQTT Header	Fixed	
	Control Packet Type	CONNECT, PUBLISH, SUBSCRIBE 등
	DUP Flag	재전송 여부
	QoS Level	0,1,2
	RETAIN Flag	Retain 여부
	Remaining Length	전체 메시지 길이

MQTT Variable Header		
	Topic Name	메시지를 전달할 토픽 - 등록/소스/연결 토픽: 6.2
	Packet Identifier	QoS 1/2에서 사용
	Properties (MQTT v5)	Content-Type, User Property 등
MQTT Payload		JSON 포맷

6.2.1 공통 MQTT 토픽 정의

카메라, 조명, 마이크의 미디어 장치뿐만 아니라 RDS 및 컨트롤러를 포함하여 제작에 사용되는 모든 장치는 MQTT 클라이언트로 동작한다. 각 장치는 MQTT 브로커에 정의된 토픽에 대해 발행 및 구독 방식으로 메시지를 송수신한다. <그림 7>은 각 장치들 사이의 MQTT 기반 메시지 연결 구조를 나타낸다.



<그림 7> MQTT 기반 발행/구독 메시지 구조

본 표준에서는 원격 프로그램 제작 환경에서 요구되는 실시간성, 확장성, 상호 운용성을 확보하기 위하여 공통의 MQTT 토픽 체계를 정의한다. 제작에 사용되는 MQTT 토픽은 계층형 구조로 설계하며, 일관된 자원 식별과 기능 분류가 가능하도록 표준화된 형식으로 정의한다. 또한 해당 토픽을 통해 전달되는 메시지는 NMOS API 스키마 구조를 준용하여 구성함으로써 기존 NMOS 자원 모델과의 정합성과 시스템 간 연동성을 유지한다.

- MQTT 공통 토픽 기본 구성요소 정의
 - x-nmos/{facility}/{programId}/events/
 - {facility}: 제작 시설 또는 물리적 위치
 - {programId}: 제작 프로그램 구분을 위한 논리적 그룹

장치 발견 등록, 장치 연결 제어, 장치 기기 제어, 탈리 알림을 위해 사용되는 공

통 토픽은 다음과 같다.

- 등록토픽: $x-nmos/{facility}/{programId}/events/{version}/common$
- 연결토픽: $x-nmos/{facility}/{programId}/events/{version}/connections/{node\ id}$
- 소스토픽: $x-nmos/{facility}/{programId}/events/{version}/sources/{source\ id}$

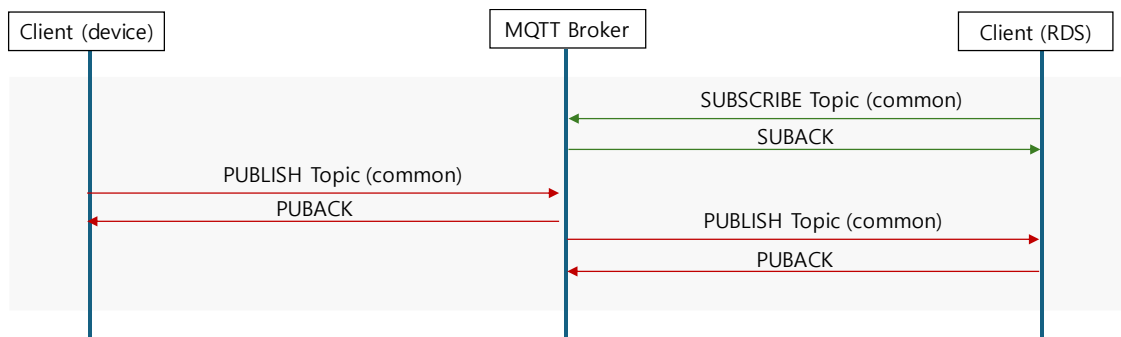
6.3 장치 발견 등록

미디어 제작에 사용되는 실제 하드웨어 또는 소프트웨어 장치는 최상위 개체로 Node를 정의하며, Node 하위에 Device, Source 등 해당 장치가 보유한 리소스를 계층적으로 구성하여 레지스트리에 등록한다. 이러한 구조는 장치 단위 식별과 기능 단위를 분리하여 시스템 내 자원 관리의 일관성과 확장성을 확보하기 위한 것이다.

● 장치 발견 등록 절차

1. Node 부팅: Node는 초기화 과정에서 내부 리소스를 식별하고, /nodes, /devices, /sources 등 표준 자원 구조에 따라 관리 정보를 구성한다.
2. RDS 발견: Node는 DNS-SD 기반 서비스 탐색을 통해 RDS의 위치를 확인한다.
3. 리소스 등록: Node는 공통 등록 토픽을 이용하여 자신의 리소스 정보를 발행한다. 등록 대상에는 Node, Device, Sender, Receiver, Source, Flow 정보가 포함된다.

미디어 단말은 장치 등록을 위해 등록 토픽을 이용하여 node, device, source와 같은 리소스 등록 메시지를 발행한다. RDS는 해당 등록 토픽을 구독하여 네트워크에 접속한 장치들의 등록 정보를 수신하고, 이를 서버에서 통합 관리한다. 이 과정을 통해 네트워크 상의 장치 존재 여부, 구성 정보, 기능 상태를 중앙에서 일관되게 파악할 수 있다. 등록 토픽을 이용한 장치 등록 절차는 <그림 8>과 같다.



<그림 8> 장치 등록 절차

- 등록 토픽: x-nmos/{ *facility* }/{ *programId* }/events/{ *version* }/common
- 리소스 등록 메시지 구조
 - Node 등록정보

```
{
  "id": "node_id",
  "label": "UHD Camera #1",
  "version": "1441703336:902850419",    # <seconds>:<nanoseconds>
  "title": "Node Resource"
}
```

- Device 등록정보

```
{
  "id": "device_id",
  "label": "UHD Camera #1 Device",
  "version": "1441703336:902850419",
  "node_id": "node_id",
  "title": "Device Resource",
  "senders": [
    { "id": "sender_id" }
  ],
  "receivers": [
    { "id": "receive_id" }
  ]
}
```

- Source 등록정보

```
{
  "id": "source_id",
  "description": "Source - UHD Camera #01 Video Source",
  "title": "Source Resource",
  "label": "UHD Camera #01 Video Source",
  "version": "1441703336:902850419",
  "device_id": "device_id",
  "format": "urn:x-nmos:format:video"
}
```

6.4 장치 연결 제어

단말은 서버에 자신의 접속 상태를 알리기 위해 이벤트 기반의 연결 상태 토픽을 사용한다. 단말이 서버와 연결을 수립할 때에는 해당 연결 토픽으로 'true' 메시지를 발행하여 활성 상태를 알리고, 단말이 종료되거나 네트워크에서 이탈할 때에는 'false' 메시지를 전송하여 비활성 상태를 통지한다.

각 단말은 고유한 Node ID를 기준으로 연결 상태 토픽을 생성하며, 생성된 토픽에 연결 상태 정보를 MQTT 브로커에 발행한다. 이를 통해 장치 별 개별 접속 상태를 식별 가능하도록 한다.

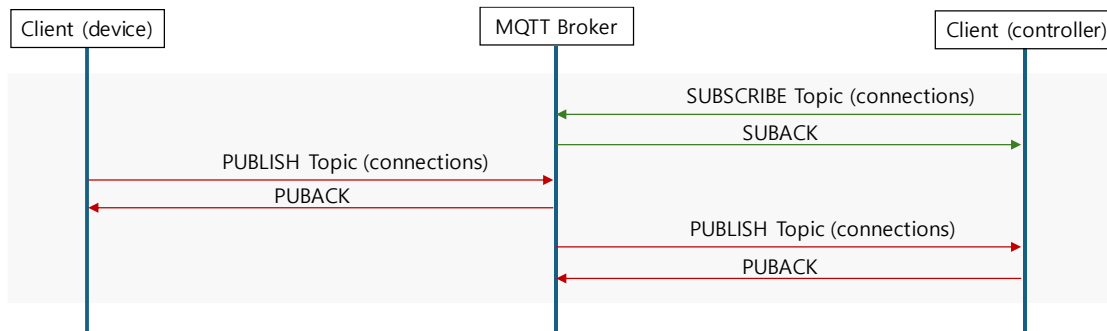
컨트롤러는 MQTT 브로커에서 해당 연결 상태 토픽을 구독하여 단말의 접속 및 이탈 여부를 실시간으로 수신하고, 시스템 내 장치 가용 상태를 관리한다. 이러한 방식을 MQTT 기반 환경에서 장치의 가용성과 운용 상태를 판단하기 위한 구조로 동작한다. 연결 토픽 구조와 세부 메시지 사용 예는 다음과 같다. 연결 토픽을 이용한 단말의 연결 상태 정보 알림 절차는 <그림 9>와 같다.

- 연결 토픽:

$x-nmos/{facility}/{program/d}/events/{version}/connections/{node\ id}$

- 연결 토픽 메시지 구조 (단말 → 컨트롤러)

```
{
  "active": "true",    # 연결: true, 연결해제: false
  "message_type": "connection_status"
}
```



<그림 9> 단말의 연결 상태 정보 알림 절차

컨트롤러는 단말의 스트리밍 목적지 주소, 전송 프로토콜 유형, 콘텐츠 형식 등의 전송 파라미터를 설정하기 위해 스트리밍 설정용 연결 토픽을 사용한다.

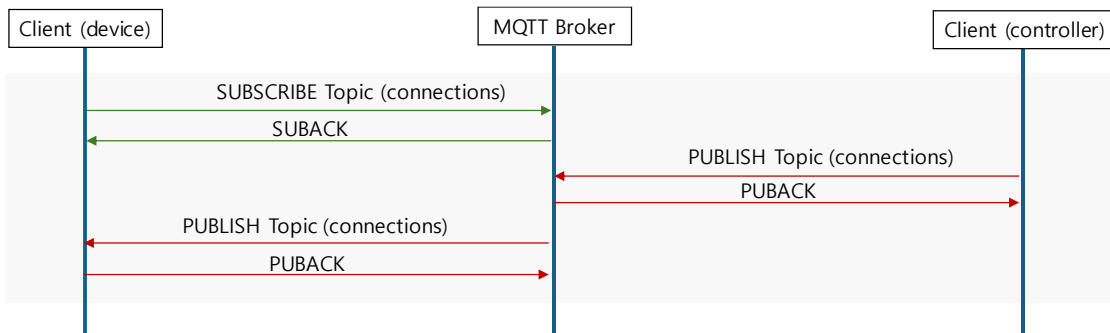
단말은 컨트롤러가 발행하는 설정 정보를 수신하기 위해 해당 연결 토픽을 MQTT 브로커에 구독하며, 컨트롤러는 단말의 스트리밍 구성을 지정하기 위해 동일 토픽으로 설정 메시지를 발행한다. 이를 통해 단말은 동적으로 전송 대상, 전송 방식 및 미디어 속성을 구성할 수 있다. 이 방식은 컨트롤러가 중앙에서 스트리밍 경로와 전송 조건을 제어하고, 단말이 이를 수신하여 즉시 반영하도록 하는 제어 모델을 제공한다.

스트리밍 설정을 위한 연결 토픽에서 사용되는 세부 메시지 형식 및 사용 예는 다음과 같다.

● 연결 토픽 메시지 구조 (컨트롤러 → 단말)

```
{
  "target_url": [
    {"ip address1": "port1"},
    {"ip address2": "port2"}
  ],
  "media_type": "video",
  "transport": "SRT",          #스트리밍 프로토콜: SRT/RTMP/RTP/UDP
  "resolution": "1920x1080",
  "frame_rate": "59.94"
}
```

<그림 10>은 단말에 대한 장치 연결을 위한 설정 정보 알림 절차를 나타낸다.



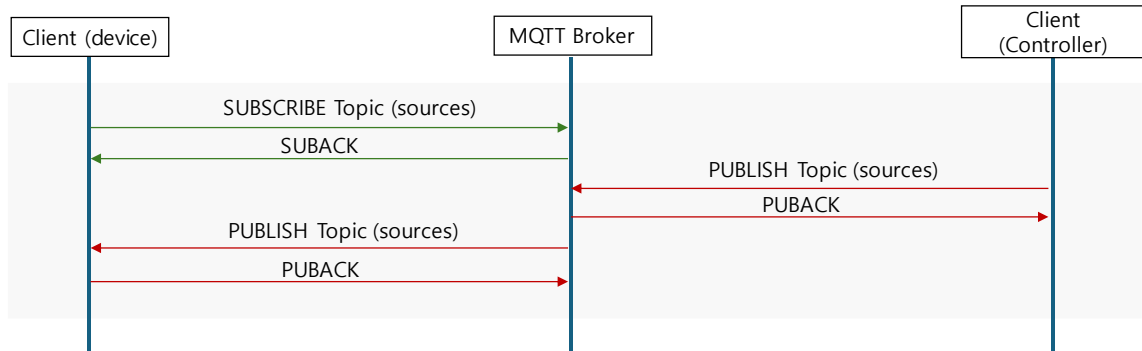
<그림 10> 단말에 대한 연결 설정 정보 알림

6.5 제작 장치 제어

네트워크에 연결된 제작 장치 제어를 위해서는 소스 토픽을 사용한다. 장치 제어는 단말이 보유한 Source ID를 기준으로 수행되며, 이를 통해 카메라, 조명, 마이크 등 각 장치 또는 기능 단위에 대한 제어 대상을 식별한다.

단말은 자신의 Source ID에 대응하는 소스 토픽을 MQTT 브로커에 구독하여, 해당 토픽으로 전달되는 제어 메시지를 수신하고 적용한다. 컨트롤러는 제어 대상 기기의 Source ID를 이용해 제어 메시지를 구성한 뒤, 해당 소스 토픽으로 발행함으로써 기기 동작을 제어한다. <그림 11>은 소스 토픽을 이용하여 제작 장치를 제어하는 절 본 표준에서는 카메라, 조명 장치 등 제작 기기에 대한 제어 절차 및 메시지 구조를 정의한다.

- 소스 토픽 `x-nmos/{ facility}/{ programId}/events/{ version}/sources/{ source id}`



<그림 11> 소스 토픽을 이용한 제작 장치 제어

6.5.1 카메라 제어

단말(카메라)은 자신의 Source ID에 해당하는 소스 제어 토픽을 MQTT 브로커에 구독하여 해당 토픽으로 전달되는 제어 메시지를 수신한다. 이를 통해 컨트롤러가 발행하는 제어 명령을 해당 카메라 단말에 직접 적용할 수 있다. 본 표준에서는 카메라 스트리밍 제어, 카메라 동작 제어대한 메시지 구조를 정의한다.

1) 카메라 스트리밍 제어

컨트롤러는 카메라의 스트리밍 동작을 제어하기 위해 Source ID 기반의 소스 토픽을 사용한다. 해당 토픽을 통해 스트리밍 시작/중지/일시정지 등의 동작을 제어할 수 있다.

스트리밍 제어 사용되는 소스 토픽 사용 예는 다음과 같다.

- 카메라 스트리밍 제어 소스 토픽 메시지 구조

```

{
  "event_type": "string/enum/StreamingControl",
  "identity": {
    "flow_id": "flow id",
    "source_id": "source id"
  },
  "payload": {
    "type": "string"
    "value": "start" # 스트리밍 시작: start, 종료: stop, 일시정지/재개:
    pause/resume
  },
  "timing": {
    "creation_timestamp": "1441703336:902850419"
  }
}
    
```

```

“message_type”: “state”
}

```

2) 카메라 동작 제어

컨트롤러는 연결된 카메라의 팬/틸트/줌 동작 제어를 위해 소스 토픽을 사용한다. 연결된 카메라의 팬/틸트/줌, 포커스, 화이트밸런스과 같은 카메라 동작을 제어할 수 있다.

카메라 동작 제어 사용되는 소스 토픽 사용 예는 다음과 같다.

● 카메라 동작 제어 소스 토픽 메시지 구조

```

{
  "event_type": "string/enum/PTZControl",
  "identity": {
    "flow_id": "flow id",
    "source_id": "source id"
  },
  "payload": {
    "type": "object",
    "value": {
      "pan_factor": 0,
      "pan_speed": 52,
      "tilt_factor": 0,
      "tilt_speed": -32,
      "zoom_factor": 0,
      "zoom_speed": 52,
      "focus_factor": 0,
      "focus_speed": 15,
      "white-balance_auto": "off",
      "white-balance_blue": 15
    }
  },
  "timing": {
    "creation_timestamp": "1441703336:902850419"
  },
  "message_type": "state"
}

```

3) 카메라 종료/재시작 제어

컨트롤러는 소스 토픽을 사용하여 연결된 카메라의 shutdown/reboot를 제어할 수 있다.

단말 장치의 shutdown/reboot 제어에 사용되는 소스 토픽 사용 예는 다음과 같다.

● 카메라 종료/재시작 제어 소스 토픽 메시지 구조

```

{
  "event_type": "string/enum/StatusControl",
  "identity": {
    "flow_id": "flow_id",
    "source_id": "source_id"
  },
  "payload": {
    "type": "string"
    "value": "shutdown"    #연결 제어: shutdown/reboot
  },
  "timing": {
    "creation_timestamp": "1441703336:902850419"
  },
  "message_type": "state"
}
    
```

6.5.2 조명 제어

단말(조명)은 자신의 Source ID에 해당하는 소스 제어 토픽을 MQTT 브로커에 구독하여, 해당 토픽으로 전달되는 제어 메시지를 수신한다. 이를 통해 컨트롤러가 발행하는 제어 명령을 해당 조명 단말에 직접 적용할 수 있다. 본 표준에서는 조명 on/off 제어와 조명설정 제어에 대한 메시지 구조를 정의한다.

1) 조명 On/Off 제어

컨트롤러는 조명 단말의 전원 상태를 제어하기 위해 Source ID 기반의 소스 토픽을 사용한다. 해당 토픽을 통해 조명의 켜짐(On) 및 꺼짐(Off) 상태를 제어할 수 있다. 조명 단말은 해당 토픽을 통해 수신한 제어 메시지에 따라 전원 상태를 변경하며, 필요 시 상태 정보를 별도의 상태 토픽으로 회신할 수 있다.

조명 On/Off 제어에 사용되는 소스 토픽 사용 예는 다음과 같다.

● 조명 On/Off 제어 소스 토픽 메시지 구조

```

{
  "event_type": "string/enum/LighteningOnOff",
  "identity": {
    "source_id": "source id"    #제어 대상 조명 Source ID
  },
  "payload": {
    "type": "string"
    "value": "on"              # on: 조명 켜기, off: 조명 끄기
  }
}
    
```

```

    },
    "timing": {
        "creation_timestamp": "1441703336:902850419"
    },
    "message_type": "state"
}

```

2) 조명 설정 제어

컨트롤러는 조명의 설정 상태를 제어하기 위해 Source ID 기반의 소스 토픽을 사용한다. 해당 토픽을 통해 조명의 밝기, 색 온도, 색상 등을 제어할 수 있다. 또한 조명 장비에 사전 정의된 복합 설정 값을 일괄 적용할 수도 있다.

조명 설정 제어에 사용되는 소스 토픽 사용 예는 다음과 같다.

● 조명 설정 제어 소스 토픽 메시지 구조

```

{
    "event_type": "string/enum/DmxControl",
    "identity": {
        "source_id": "source id"
    },
    "payload": {
        "type": "object",
        "value": {
            "universe": 1,
            "address": [1 ~ 512],           #조명 채널 번호
            "data": "HEXA-BINARY"         #조명 채널 설정 값
        }
    },
    "timing": {
        "creation_timestamp": "1441703336:902850419"
    },
    "message_type": "state"
}

```

● 조명 사전 정의된 설정 소스 토픽 메시지 구조

```

{
    "event_type": "string/enum/PresetControl",
    "identity": {
        "source_id": "source id"
    },
    "payload": {

```

```

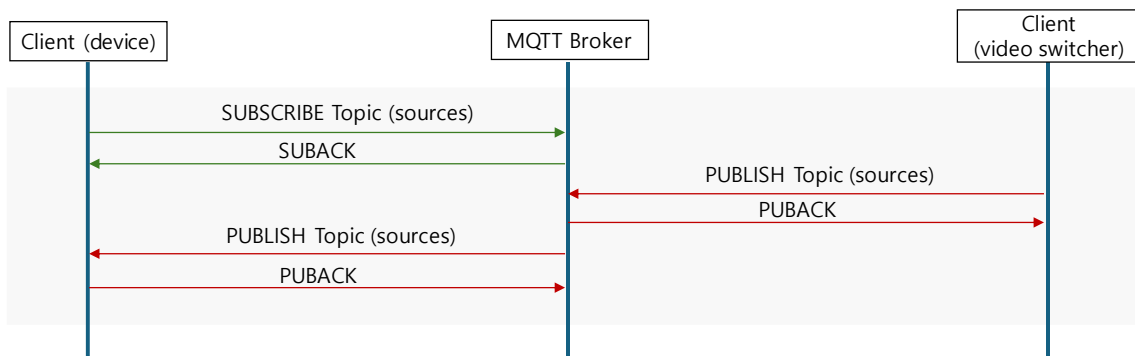
        "type": "string",
        "value": "studio_news_01"      #호출할 사전 정의된 식별자
    },
    "timing": {
        "creation_timestamp": "1441703336:902850419"
    },
    "message_type": "state"
}
    
```

6.6 탈리 알림

카메라 단말을 포함한 미디어 소스는 자신의 Source ID에 해당하는 소스 토픽을 구독함으로써 탈리 신호를 수신할 수 있다. 탈리 신호는 라이브 방송 제작에서 미디어 소스가 현재 프로그램 출력(PGM)에 사용 중인지, 아니면 프리뷰(PRW) 상태인지 알려주는 운용 정보이다. 이 신호는 실시간 영상 송출 상황을 직관적으로 표시해 라이브 방송 제작에 중요한 정보로 활용된다.

비디오 스위처는 프로그램 채널 및 프리뷰 채널의 스위칭 상태에 따라 연결된 카메라를 식별하고, 해당 카메라의 Source ID 기반 소스 토픽으로 탈리 정보를 발행한다. 이를 통해 각 카메라 단말은 자신의 온에어 여부 및 대기 상태를 실시간으로 인지할 수 있다. <그림 12>는 소스 토픽을 이용한 탈리 알림 절차를 나타낸다.

- 소스 토픽 `x-nmos/{facility}/{programId}/events/{version}/sources/{sourceId}`



<그림 12> 소스 토픽을 이용한 탈리 알림

소스 토픽을 이용한 탈리 알림 메시지 사용 예는 다음과 같다.

- 탈리 알림 소스 토픽 메시지 구조

```

{
    "event_type": "string/enum/TallyStatus",
    "identity": {
        "flow_id": "flow_id",
    }
}
    
```

```
"source_id": "source_id"  
},  
"payload": {  
  "type": "string",  
  "value": "PGM"    #탈리 상태: PGM/PRV/OFF  
},  
"timing": {  
  "creation_timestamp": "1441703336:902850419"  
},  
"message_type": "state"  
}
```

부 록 1-1

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

지식재산권 협약서 정보

해당 사항 없음

부 록 1-2

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

시험인증 관련 사항

1-2.1 시험인증 대상 여부

해당 사항 없음

1-2.2 시험표준 제정 현황

해당 사항 없음

부 록 1-3

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

본 표준의 연계(family) 표준

해당 사항 없음

부 록 1-4

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

참고 문헌

해당 사항 없음

부 록 1-5

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

영문표준 해설서

해당 사항 없음

부 록 1-6

(본 부록은 표준을 보충하기 위한 내용으로 표준의 일부는 아님)

표준의 이력

판수	채택일	표준번호	내용	담당 위원회
제1판	2026.04.10.	제정 FBMF-STD-xxx	-	UHD융합기술분과위원회