

(12) 특허협력조약에 의하여 공개된 국제출원

(19) 세계지식재산권기구
국제사무국

(43) 국제공개일
2012년 12월 20일 (20.12.2012)



(10) 국제공개번호
WO 2012/173391 A2

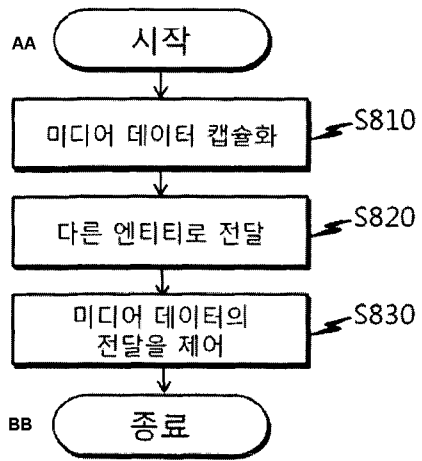
- (51) 국제특허분류: H04N 7/24 (2011.01) H04L 12/56 (2006.01)
- (21) 국제출원번호: PCT/KR2012/004670
- (22) 국제출원일: 2012년 6월 13일 (13.06.2012)
- (25) 출원언어: 한국어
- (26) 공개언어: 한국어
- (30) 우선권정보: 10-2011-0056661 2011년 6월 13일 (13.06.2011) KR
10-2012-0063213 2012년 6월 13일 (13.06.2012) KR
- (71) 출원인 (US 을(를) 제외한 모든 지정국에 대하여): 한국 전자통신연구원 (ELECTRONICS AND TELECOMMUNICATIONS RESEARCH INSTITUTE) [KR/KR]; 대전광역시 유성구 가정동 161번지, 305-700 Daejeon (KR).
- (72) 발명자: 곁
- (75) 발명자/출원인 (US 에 한하여): 배성준 (BAE, Seong Jun) [KR/KR]; 대전시 유성구 반석동 반석마을 7단지 702-1406, 305-750 Daejeon (KR).
- (74) 대리인: 양문옥 (YANG, Moon Ock); 서울 강남구 역삼동 735-10 삼흥역삼빌딩 2층 에센특허법률사무소, 135-080 Seoul (KR).
- (81) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 국내 권리의 보호를 위하여): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 지정국 (별도의 표시가 없는 한, 가능한 모든 종류의 역내 권리의 보호를 위하여): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), 유라시아 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 유럽 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[다음 쪽 계속]

(54) Title: METHOD FOR TRANSMITTING PACKET-BASED MEDIA DATA HAVING HEADER IN WHICH OVERHEAD IS MINIMIZED

(54) 발명의 명칭 : 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법

[Fig. 8]



AA ... Start
 BB ... End
 S810 ... Encapsulate media data
 S820 ... Transfer to another entity
 S830 ... Control transfer of media data

(57) Abstract: Provided is a method for transmitting packet-based media data having a header in which an overhead is minimized. A method for media data transmission may comprise the steps of: encapsulating media data to allow a media fragment unit (MFU) header and an M-unit header to share one format; allowing an MMT payload format to include a header field for an operation which is not supported by an RTP header, and transferring the encapsulated media data from a specific entity to another entity; and controlling the transfer of the media data. Therefore, a transmission overhead is minimized, so that it is possible to efficiently transmit media data.

(57) 요약서: 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법이 제공된다. 미디어 데이터 전송 방법은 미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-unit) 헤더가 하나의 포맷을 공유하도록 미디어 데이터를 캡슐화하고, MMT 페이로드 포맷이 RTP 헤더가 지원하지 않는 동작을 위한 헤더 필드를 포함하도록 하여 상기 캡슐화된 미디어 데이터를 특정 엔티티(entity)로부터 다른 엔티티로 전달하며 상기 미디어 데이터의 전달을 제어하는 단계를 포함할 수 있다. 따라서, 전송 오버헤드를 최소화하여 효율적으로 미디어 데이터를 전송할 수 있다.

WO 2012/173391 A2

공개:

- 국제조사보고서 없이 공개하며 보고서 접수 후 이를 별도 공개함 (규칙 48.2(g))

명세서

발명의 명칭: 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법

기술분야

- [1] 본 발명은 미디어 데이터의 전송 방법에 관한 것으로 보다 구체적으로는 전송 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법에 관한 것이다.

[2]

배경기술

- [3] MPEG-2의 표준화 이후, 비디오 압축 표준(또는 오디오 압축 표준)은 과거 10년간 MPEG-4, H.264/AVC, HEVC(High Efficiency Video Coding) 등으로 꾸준히 새로운 표준이 개발되었다. 또한 각각의 새로운 표준들은 새로운 시장을 형성하면서 MPEG 표준의 활용 영역을 넓혀왔으나, MPEG-2 TS(Transport System)와 같은 전송 기술의 경우 20년 가까운 세월이 흐르는 동안 변함없이 시장에서 디지털방송, 모바일 방송(T-DMB, DVB-H등)등에 널리 사용되고 있으며, 심지어 표준 제정 당시 고려하지 않았던 인터넷을 통한 멀티미디어 전송, 즉 IPTV 서비스에도 널리 활용되고 있는 상황이다.
- [4] 그러나, MPEG-2 TS가 개발될 때의 멀티미디어 전송환경과 오늘날의 멀티미디어 전송환경은 큰 변화를 겪고 있다. 예컨대, MPEG-2 TS 표준은 제정 당시 ATM 망을 통해 멀티미디어 데이터를 전송하는 것을 고려하여 개발되었으나, 오늘날 이러한 목적으로 이용되는 사례는 거의 찾아보기 힘들어졌다. 또한, MPEG-2 TS 표준 제정 당시 인터넷을 이용한 멀티미디어 전송 등의 요구사항(requirement)이 고려되지 않아 최근의 인터넷을 통한 멀티미디어 전송에 효율적이지 못한 요소들이 존재한다. 따라서, MPEG에서는 변화하는 멀티미디어 환경에 걸맞는 인터넷에서의 멀티미디어 서비스를 고려한 새로운 멀티미디어 전송 표준인 MMT(MPEG Multimedia Transport Layer)의 제정이 매우 중요한 과제로 인식되고 있다.
- [5] 이와 같이, MMT 표준화가 진행되는 중요한 이유는 20년 전에 만들어진 MPEG2-TS 표준이 최근 IPTV 방송 서비스, 인터넷 환경 등에 최적화되어 있지 않기 때문에 최근 다양한 이종망(Heterogeneous Network)에서의 멀티미디어 전송 환경에 최적화된 멀티미디어 전송 국제 표준의 시급한 필요에 의해 MPEG에서 MMT를 새로운 전송 기술 표준으로서 표준화를 진행하고 있는 것이다.
- [6] MPEG 미디어 전송 시 캡슐화 계층(Encapsulation layer)과, 전달 계층(Delivery layer)이 존재하며, 캡슐화 계층은 (1) 미디어 프래그먼트 유닛(Media Fragment Unit, MFU), (2) M-유닛(M-Unit), (3) MMT 애셋(MMT Asset), (4) MMT 패키지(MMT package)의 네 부분, 전달 계층은 (1) MMT 페이로드 포맷(MMT

Payload format)과 (2) MMT 전송 패킷(MMT transport packet)의 두 부분으로 나뉜다. 본 발명은 MMT를 통하여 미디어를 전송할 때 전송 오버헤드 최소화를 고려한 MMT 전송 오버헤드 구조에 대해 개시한다.

[7]

발명의 상세한 설명 기술적 과제

[8]

MPEG 미디어를 패킷 기반으로 전송할 때 가장 중요한 요소는 전송 오버헤드를 최소화하는 것이다. 여기서 전송 오버헤드는 예를 들면 M2TS, RTP 기반과 같이 현재 사용되고 있는 기존 전송 프레임 워크(delivery framework)에 비해 합리적인 관점에서 상당히 크지 않은 범위 내에서(혹은 더 적은 크기로) 설계되어야 한다는 의미로 해석할 수 있다. 적어도 기존 프레임 워크에서와 동일한 작동(operation) 및/또는 정도(precision)를 제공하는 헤더 필드(header field)에 대해서 새롭게 제안하는 방식은 작거나 같은 헤더 필드 크기를 가져야 한다.

[9]

MPEG 미디어 전송 시 캡슐화 계층과 전달 계층을 상기 기술한 대로 캡슐화의 4부분, 전달의 2부분으로 나누는 경우, 이들 여섯 가지 항목에 대한 헤더를 단순히 순차적으로 각각의 헤더로 중첩/추가하여 패킷을 구성하는 경우, 이 패킷은 실제 패킷 전송 시 필수적으로 필요한 정보 외에 필요없는 정보까지 포함하게 되어 상당한 오버헤드를 형성할 수 있다. 따라서 스트리밍 시에 필요한 오버헤드만을 축약하여 새롭게 정의할 필요가 있다.

[10]

따라서, 본 발명의 목적은 미디어 데이터를 패킷화하여 전송하되 미디어 프래그먼트 유닛 헤더와 M-유닛 헤더가 하나의 포맷을 공유하고 RTP 헤더로 지원되지 않는 동작들을 위해 필요한 헤더 필드들이 모두 MMT 페이로드 포맷에 위치하도록 함으로써, 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법을 제공하는 것이다.

[11]

또한, 본 발명의 다른 목적은 미디어 데이터를 패킷화하여 전송하되 미디어 프래그먼트 유닛 헤더와 M-유닛 헤더가 하나의 포맷을 공유하고 RTP 헤더로 지원되지 않는 동작들을 위해 필요한 헤더 필드들이 모두 MMT 페이로드 포맷에 위치하도록 함으로써, 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송을 위한 인터페이스를 제공하는 것이다.

[12]

과제 해결 수단

[13]

전술한 목적을 달성하기 위한 본 발명의 일 실시예에 따른 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법은 미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-unit) 헤더가 하나의 포맷을 공유하도록 미디어 데이터를 캡슐화하는 단계; MMT 페이로드 포맷이 RTP 헤더가 지원하지 않는 동작을 위한 헤더 필드를 포함하도록 하여

상기 캡슐화된 미디어 데이터를 특정 엔티티(entity)로부터 다른 엔티티로 전달하는 단계; 및 상기 미디어 데이터의 전달을 제어하는 단계를 포함할 수 있다.

[14]

발명의 효과

[15] 전술한 본 발명의 일 실시예에 따른 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법에 따르면 미디어 프래그먼트 유닛 헤더와 M-유닛 헤더가 하나의 포맷을 공유하도록 함으로써 별도의 포맷을 제공하는 경우 인코더가 둘 중 어느 포맷까지 포함해야 하는지 여부를 판단하기 위한 작업을 수행하는 것을 방지할 수 있다. 또한, RTP 헤더로 지원되지 않는 동작들을 위해 필요한 헤더 필드들이 모두 MMT 페이로드 포맷에 위치하도록 함으로써 RTP를 통하여 전송되는 경우와 MMTP(MMT Packet)을 통하여 전송되는 경우의 양쪽 모두 동작할 수 있다. 따라서, 전송 오버헤드를 최소화하여 효율적으로 미디어 데이터를 전송할 수 있다.

[16]

도면의 간단한 설명

[17] 도 1은 MMT 계층 구조를 나타낸 개념도이다.
 [18] 도 2는 M-Unit 헤더 구조를 개략적으로 도시한 것이다.
 [19] 도 3은 인코더에 대한 M-Unit 헤더의 구조를 설명하는 도면이다.
 [20] 도 4는 시스템에 대한 M-Unit 헤더의 구조를 설명하는 도면이다.
 [21] 도 5는 MMT 페이로드 포맷 구조를 설명하는 도면이다.
 [22] 도 6은 MMT 패킷 헤더 구조를 설명하는 도면이다.
 [23] 도 7은 인코더로부터 출력된 미디어데이터(coded data)가 전송을 위해 패킷화되는 과정을 도시한 도면이다.
 [24] 도 8은 본 발명의 일 실시예에 따른 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법의 흐름도이다.

[25]

발명의 실시를 위한 형태

[26] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다.
 [27] 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.
 [28] 제1, 제2 등의 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성요소로 명명될

수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다. 및/또는이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

- [29] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [30] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [31] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가진 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [32] 이하, 첨부한 도면들을 참조하여, 본 발명의 바람직한 실시예를 보다 상세하게 설명하고자 한다. 본 발명을 설명함에 있어 전체적인 이해를 용이하게 하기 위하여 도면상의 동일한 구성요소에 대해서는 동일한 참조부호를 사용하고 동일한 구성요소에 대해서 중복된 설명은 생략한다.
- [33]
- [34] 도 1은 MMT 계층 구조를 나타낸 개념도이다.
- [35] 도 1을 참조하면, MMT 계층은 캡슐화 계층(Encapsulation layer), 전달 계층(Delivery layer) 및 제어 계층(Control layer)의 기능 영역(functional area)을 포함한다. MMT 계층은 전송 계층(Transport layer) 위에서 동작한다.
- [36] 캡슐화 계층(Encapsulation layer; E-layer)은 예를 들어 전송되는 미디어의 패킷화(packetization), 프래그멘테이션(Fragmentation), 동기화(Synchronization), 멀티플렉싱(Multiplexing)등의 기능을 담당할 수 있다.
- [37] 캡슐화 계층(E-layer)은, 도 1에 도시된 바와 같이, MMT E.1 계층(MMT E.1 Layer), MMT E.2 계층(MMT E.2 Layer) 및 MMT E.3 계층(MMT E.3 Layer)으로 구성될 수 있다.
- [38] E.3 계층은 미디어 코덱(A) 계층으로부터 제공된 미디어 프래그먼트

- 유닛(Media Fragment Unit; MFU)를 캡슐화하여 M-유닛(M-Unit)을 생성한다.
- [39] MFU는 미디어 디코더에서 독립적으로 소비될 수 있는 데이터 유닛을 신을 수 있는, 임의의 특정 코덱(codec)에 독립적인, 포맷을 가질 수 있다. MFU는 예를 들어 비디오의 픽처(picture) 또는 슬라이스(slice)가 될 수 있다.
- [40] M-유닛은 하나 또는 복수의 MFU 로 구성될 수 있으며, 하나 또는 복수의 액세스 유닛(Access Unit)을 신을 수 있는, 특정 코덱(codec)에 독립적인, 포맷을 가질 수 있다.
- [41] E.2 계층은 E.3 계층에서 생성된 M-유닛을 캡슐화하여 MMT 애셋(MMT Asset)을 생성한다.
- [42] MMT 애셋은 단일의 데이터 소스로부터의 하나 또는 복수의 M-유닛으로 이루어진 데이터 엔티티(data entity)로서, 컴포지션 정보(composition information) 및 전송 특성(transport characteristics)이 정의된 데이터 유닛이다. MMT 애셋은 PES(packetized elementary streams)에 대응될 수 있으며, 예를 들어 비디오, 오디오, 프로그램 정보(program information), MPEG-U 위젯(widget), JPEG 이미지, MPEG 4 파일 포맷(File Format), M2TS(MPEG transport stream) 등에 대응될 수 있다.
- [43] E.1 계층(E.1 Layer)은 E.2 계층에서 생성된 MMT 애셋을 캡슐화하여 MMT 패키지(MMT Package)를 생성한다.
- [44] MMT 패키지는 컴포지션 정보(composition information) 및 전송 특성(transport characteristics)과 같은 부가 정보와 함께 하나 또는 복수의 MMT 애셋으로 구성될 수 있다. 컴포지션 정보(composition information)는 MMT 애셋들 사이의 관계(relationship)에 대한 정보를 포함하며, 하나의 콘텐츠(content)가 복수개의 MMT 패키지로 이루어질 경우 복수의 MMT 패키지간의 관계(relationship)를 나타내기 위한 정보를 더 포함할 수 있다. 전송 특성(transport characteristics)은 MMT 애셋 또는 MMT 패킷의 전송 조건(delivery condition)을 결정하기 위해 필요한 전송 특성 정보를 포함할 수 있으며, 예를 들어 트래픽 기술 파라미터(traffic description parameter) 및 QoS 기술자(QoS descriptor)를 포함할 수 있다. MMT 패키지는 MPEG-2 TS의 프로그램(Program)에 대응될 수 있다.
- [45]
- [46] 전달 계층(Delivery layer)은 예를 들어 네트워크를 통해 전송되는 미디어의 네트워크 플로우 멀티플렉싱(Network flow multiplexing), 네트워크 패킷화(Network packetization), QoS 제어 등을 수행할 수 있다.
- [47] 전달 계층(D-layer)은, 도 1에 도시된 바와 같이, MMT D.1 계층(MMT D.1 Layer), MMT D.2 계층(MMT D.2 Layer) 및 MMT D.3 계층(MMT D.3 Layer)으로 구성될 수 있다.
- [48] D.1 계층(D.1-layer)은 E.1 계층에서 생성된 MMT 패키지를 받아서 MMT 페이로드 포맷(MMT Payload format)을 생성한다. MMT 페이로드 포맷은 MMT 애셋을 전송하고, 그리고 MMT 애플리케이션 프로토콜 또는 RTP와 같은 다른

기존의 애플리케이션 전송 프로토콜에 의한 소비를 위한 정보를 전송하기 위한 페이로드 포맷이다. MMT 페이로드는 AL-FEC와 같은 정보와 함께 MFU의 프래그먼트를 포함할 수 있다.

- [49] D.2 계층(D.2-layer)은 D.1 계층에서 생성된 MMT 페이로드 포맷을 받아서 MMT 전송 패킷(MMT Transport Packet) 또는 MMT 패킷(MMT Packet)를 생성한다. MMT 전송 패킷 또는 MMT 패킷은 MMT를 위한 애플리케이션 전송 프로토콜에 사용되는 데이터 포맷이다.
- [50] D.3 계층(D.3-layer)은 교차 계층 설계(cross-layer Design)에 의해 계층간에 정보를 교환할 수 있는 기능을 제공하여 QoS를 지원한다. 예를 들어, D.3 계층은 MAC/PHY 계층의 QoS 파라미터를 이용하여 QoS 제어를 수행할 수 있다.
- [51]
- [52] 제어 계층(Control layer)은 예를 들어 전송되는 미디어의 세션 초기화/제어/관리(session initialization/control/management), 서버 기반 및/또는 클라이언트 기반의 트릭 모드, 서비스 디스커버리(Service discovery), 동기화(Synchronization) 기능 등을 수행할 수 있다.
- [53] 제어 계층 (C-layer)은, 도 1에 도시된 바와 같이, MMT C.1 계층(MMT C.1 Layer) 및 MMT C.2 계층(MMT D.2 Layer)으로 구성될 수 있다.
- [54] C.1 계층은 서비스 디스커버리(Service discovery), 미디어의 세션 초기화/종료(media session initialization/termination), 미디어의 세션 표현/제어(media session presentation/control), 전달(D) 계층 및 캡슐화(E) 계층과의 인터페이스 기능 등을 수행할 수 있다. C.1 계층은 미디어 표현 세션 관리(presentation session management)를 위한 애플리케이션들간의 제어 메시지들의 포맷을 정의할 수 있다.
- [55] C.2 계층은 흐름 제어(flow control), 전달 세션 관리(delivery session management), 전달 세션 모니터링(delivery session monitoring), 에러 제어(error control), 하이브리드망 동기화 제어(Hybrid network synchronization control)에 관한 전달 계층(D-layer)의 전달 엔드-포인트들(delivery end-points)간에 교환되는 제어 메시지의 포맷을 정의할 수 있다.
- [56] C.2 계층은 전달 계층의 동작을 지원하기 위하여 전달 세션 설정 및 해제(delivery session establishment and release), 전달 세션 모니터링, 흐름 제어, 에러 제어, 설정된 전달 세션에 대한 리소스 예약, 복합 전달 환경하에서의 동기화를 위한 시그널링, 적응적 전달(adaptive delivery)를 위한 시그널링을 포함할 수 있다. 송신측(sender)와 수신측(receiver)간에 필요한 시그널링을 제공할 수 있다. 즉, C.2 계층은 전송한 바와 같은 전달 계층의 동작을 지원하기 위하여 송신측(sender)와 수신측(receiver)간에 필요한 시그널링을 제공할 수 있다. 또한, C.2 계층은 전달 계층 및 캡슐화 계층과의 인터페이스 기능을 담당할 수 있다.

[57]

[58] 본 발명의 일 실시예에 따른 패킷 기반의 미디어 데이터 전송을 위한 인터페이스는 미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-unit) 헤더가 하나의 포맷을 공유하도록 미디어 데이터를 캡슐화하는 캡슐화 계층, MMT 페이로드 포맷이 RTP 헤더가 지원하지 않는 동작을 위한 헤더 필드를 포함하도록 하여 상기 캡슐화된 미디어 데이터를 특정 엔티티(entity)로부터 다른 엔티티로 전달하는 전달 계층 및 상기 미디어 데이터의 전달을 제어하는 제어 계층을 포함할 수 있다. 하기에서 상기 미디어 데이터 전송을 위한 인터페이스 및 미디어 데이터 전송 방법을 상세히 설명한다.

[59]

[60] **1 전달 오버헤드(Proposed Delivery Overhead)**

[61]

[62] 본 발명에서는 전달 타임(네트워크 스트리밍을 수행하는 어플리케이션의 경우)과 캡슐화 타임(파일 등으로 저장장치에 저장(archiving)하는 어플리케이션의 경우)에 사용되는 오버헤드의 구조를 구분하여 정의할 필요가 있다고 보며, 전달시(네트워크 스트리밍 시)의 오버헤드 구조에 대해 서술한다.

[63] 이러한 관점에서 본 출원은 전달 시에 필요로 하는 오버헤드에 대해 하기와 같은 추가적인 가이드라인을 제시한다.

[64]

[65] 1. 네트워크 스트리밍 어플리케이션의 경우 인코더로부터 만들어진 미디어 데이터를 패킷화 하여 송신하는 경우 필요로 하는 헤더 단계를 (1) AU 레벨의 헤더와, (2) 패킷 레벨의 헤더의 2단계 구조로 둘 수 있다.

[66] 2. 인코더가 그 내용을 작성하는 미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-Unit, MU) 헤더는 하나의 포맷을 공유하며, 단일 포맷 내에서 MFU의 경우 분할화(fragmentation)와 연관된 정보들을 담기 위해 옵션으로 확장될 수 있다.

[67] 3. 전달 계층에 존재하는 MMT 페이로드 포맷의 소계층과, MMT 전송 패킷의 소계층에서 기존의 RTP 헤더로 지원되지 않는 동작(Operation)들을 위해 필요한 헤더 필드는 모두 MMT 페이로드 포맷(MMT Payload format, MMTPL)에 위치할 수 있다.

[68]

[69] 1.1 전달 오버헤드의 2단계 구조 : 패킷 레벨 헤더, AU 레벨 헤더

[70]

[71] 미디어 데이터를 네트워크를 통해 스트리밍 하는 경우, 인코더에서 작성하는 미디어 유닛(Media Unit) 이후의 헤더는 (1) AU 레벨의 헤더와, (2) 패킷 레벨의 헤더의 2단계 구조를 둔다. 미디어 데이터를 스트리밍하는 경우 패킷 전송을 통해 한쪽에서 다른쪽으로 전달되어야 하는 정보는 크게 다음 세 가지로 나눌 수 있다.

[72] (1) 세션 레벨의 정보 (예: 세션에 속해 있는 미디어의 개수, 코덱 타입, 화면

- 구성 정보 등과 같은 세션 구성 정보), (2) AU-레벨의 정보(PTS-Presentation Timestamp : 프리젠테이션 타임스탬프, AU레벨로 붙어야 하는 정보), (3) 패킷 레벨의 정보 (한 패킷내의 다중유닛수용 : aggregation, 한 유닛의 여러 패킷으로의 분할 : fragmentation, 패킷레벨의 멀티플렉싱, 패킷에러처리 등)
- [73] 보통의 경우, 세션 레벨의 정보는 별도의 전송 채널을 통해 아웃 오브 밴드(out-of-band)로 보내질 수 있으며, 패킷 헤더에 본 정보를 포함하고 있을 필요는 없다. 따라서 본 발명에서는 전송시의 헤더에 대해 AU 레벨 헤더와 패킷 레벨 헤더를 가지는 2 계층이면 충분하며 또한 최적이라고 판단한다. AU 레벨 헤더는 캡슐화 계층에 포함된 소계층들에서 스트리밍 시 필요로 하는 모든 필드를 포함할 수 있으며 각 소계층별로 명시적인 필드 구분을 하지 않을 수 있다. 다만, AU 레벨 헤더는 인코더에 의해 내용이 채워지는 필드인지, 시스템에 의해 내용이 채워지는 필드인지의 여부에 따라 한 가지의 필드 구분만을 둘 수 있다.
- [74]
- [75] 1.2 미디어 프래그먼트 및 M-유닛을 위한 통일된 필드
- [76]
- [77] 인코더가 그 내용을 작성하는 미디어 프래그먼트 유닛(MFU) 헤더와 M-유닛(MU) 헤더는 하나의 포맷을 공유하며, 단일 포맷 내에서 MFU의 경우 분할화(fragmentation)와 연관된 정보들을 담기 위해 옵션으로 확장될 수 있다.
- [78] 만약 MFU와 MU에 대해 서로 다른 포맷을 제공하게 된다면, 인코더는 둘 중 어느 헤더까지를 적어주어야 하는지 구분하기 위하여 추가적인 작업을 필요로 하는 경우가 발생한다. 예를 들어, 프래그먼트(Fragment) 단위로 출력하는 인코더의 경우 프래그먼트(Fragment)를 출력함과 동시에 AU레벨로 MU헤더를 붙여주기 위하여 자신이 출력하는 스트림의 AU 시작과 끝지점을 추가적으로 체크해야 하는 기능이 필요하다. 수많은 종류의 코덱과 해당하는 인코더가 존재하는 현재의 상황을 고려하면 모든 인코더에게 이러한 구분기능을 강요하는 것은 불합리하다.
- [79]
- [80] 1.3 MMT 패킷(MPEG Media Transport Packet, MMTP) 동작의 최소화
- [81] 전달 헤더의 경우 기존의 RTP 헤더로 지원되지 않는 동작(Operation)들은 MMT 페이로드 포맷(MMT Payload format, MMTPL)에 위치시킬 수 있다.
- [82] MMTPL의 역할은, 새롭게 정의하게 될 본 MMT 표준을 사용하는 다양한 미디어를 RTP 등 기존 어플리케이션 표준에서도 공통으로 사용할 수 있는 공통 계층이다. 이러한 관점에서 RTP 등 기존 어플리케이션 전송포맷을 재활용하는 어플리케이션에서 본 MMT 전달 표준에서 새롭게 제안하는 동작을 활용할 수 있기 위해서는 RTP 등 기존의 프로토콜에서 지원하지 않는 새로운 기능의 경우 MMT 페이로드 포맷에 위치해야만 RTP를 통하여 전송되는 경우와 MMTP를 통하여 전송되는 경우의 양쪽 모두 동작할 수 있다.

[83]

[84] **2헤더 정의(Header Definitions) : 특성(Characteristics) 및 동작들(Operations)**

[85]

[86] 본 절에서는 상기에서 설명한 디자인 규칙에 따라 헤더 구조를 도시한다. 두 레벨 계층의 헤더는 AU 레벨의 헤더로 M-Unit 헤더를 포함할 수 있으며, 패킷 레벨의 헤더로 MMT 페이로드 헤더와 MMT 패킷 헤더를 포함할 수 있다.

[87]

[88] 2.1 M-Unit 헤더 (AU 레벨 헤더)

[89] 도 2는 M-Unit 헤더 구조를 개략적으로 도시한 것이다.

[90]

도 2에 도시된 바와 같이 M-Unit 헤더는 AU 레벨의 헤더로 하나의 M-Unit은 하나의 M-Unit 헤더를 가진다. M-Unit 헤더는 계속해서 인코더를 위한 M-Unit 헤더(MUH-E)와, 시스템을 위한 M-Unit 헤더(MUH-S)의 두 파트로 나누어질 수 있다. MUH-E는 인코더에 의해 필드가 채워지도록 목적하고 있으며 인코더가 시스템에 좀 더 효율적인 전송을 할 수 있게 하기 위해 필요한 정보들을 포함한다. 반면, MUH-S는 시스템에 의해 필드가 채워지도록 목적하고 있으며 해당 MU(AU) 관점에서 시스템이 처리해야 할 정보들을 포함한다.

[91]

MUH-E는 기본적으로 공통 필드(Common Field)로 구성되어 있으며, 옵션으로 프래그먼트 필드(Fragment Field)를 가지도록 확장될 수 있다. 공통 필드는 기본 필드로 현재 MU와 관련된 일반적인 정보를 포함하며, AU 단위로 출력하는 인코더와 프래그먼트(Fragment) 단위로 출력하는 인코더 양쪽 모두에 의해 공통적으로 사용된다. 프래그먼트 필드는 프래그먼트 분할과 관련된 정보들을 포함하기 위해 확장될 수 있는 필드로, 현재 유닛이 AU의 부분(fragment)을 포함하고 있을 때만 사용된다. 따라서, 프래그먼트 필드는 프래그먼트 단위로 출력하는 인코더의 경우에만 사용될 수 있다.

[92]

[93] 2.1.1. 인코더를 위한 M-Unit 헤더(MUH-E)

[94]

[95] 도 3은 인코더에 대한 M-Unit 헤더의 구조를 설명하는 도면이다.

[96]

MUH-E는 인코더에 의해 내용이 채워지며 전송시스템의 전송 품질을 향상하기 위해 인코더가 시스템에 제공할 수 있는 정보들을 포함할 수 있다.

[97]

T : Timed Flag - 본 유닛이 Timed MU인지 Non-timed MU 인지 구분한다. (T=1 : Timed MU, T=0 : Non-timed MU) - Timed MU는 실제 재생 시 화면에 출력될 시간이 명확하게 정해져 있는 미디어데이터를 의미하며, Non-timed MU는 화면에 재생될 시간이 명확하게 정해져 있지 않은 일반적인 데이터를 의미한다.

[98]

A : Anchor Flag - 1로 표시된 경우 본 유닛의 시작지점이 앵커 포인트(Anchor Point)를 포함하고 있음을 의미한다. 앵커 포인트는 랜덤 액세스 지점과 같이 스트림 가운데에 접근이 가능해야 하는 지점을 의미한다. 만약 본 유닛을 패킷화 하는 하부 전송시스템이 이 플래그=1을 발견하면 시스템은 패킷화 시 패킷의

시작부분에 본 유닛의 시작부분을 반드시 위치시켜야 한다.

[99] L : Lossless - 1로 표시된 경우 본 유닛이 포함하고 있는 데이터는 패킷손실을 겪지 않도록 송신되어야 함을 표시한다.

[100] F : Fragmented - 1로 표시된 경우 본 유닛이 포함하고 있는 데이터는 MU의 프래그먼트(fragment)로 MFU를 포함하고 있으며, 본 헤더 필드의 다음으로 반드시 Fragment Field를 추가적으로 가진다.

[101] S : Start of MU - 1로 표시된 경우 본 유닛은 한 MU 내에서 분할된 프래그먼트 중 첫번째 프래그먼트를 포함한다.

[102] E : End of MU - 1로 표시된 경우 본 유닛은 한 MU 내에서 분할된 프래그먼트 중 마지막 프래그먼트를 포함한다.

[103] 또는, 상기 S 및 E 두 비트를 통합하여 하기와 같이 start_end_indicator 하나의 필드로 사용할 수 있다. 여기서 각 비트의 조합은 상기 S 및 E와 같으며, MU 뿐만 아니라 AU에 대해서 사용될 수 있다.

[104] 1. Start_end_indicator == 01 : AU의 마지막 프래그먼트(fragment)를 의미

[105] 2. Start_end_indicator == 00 : AU의 가운데 프래그먼트(fragment)를 의미

[106] 3. Start_end_indicator == 10 : AU의 첫 번째 프래그먼트(fragment)를 의미

[107] 4. Start_end_indicator == 11 : MU가 하나의 AU만을 포함하고 있음을 의미

[108] Frg.Order : 프래그먼트 오더 - 한 MU 내에서 분할된 프래그먼트 중 본 유닛이 몇 번째 프래그먼트인지를 나타낸다.

[109]

[110] 2.1.2. 시스템을 위한 M-Unit 헤더(MUH-S)

[111] 도 4는 시스템에 대한 M-Unit 헤더의 구조를 설명하는 도면이다.

[112] MUH-S는 시스템에 의해 내용이 채워지며, 현재 MU(AU)와 관련하여 시스템에 의해 제공되는 정보들을 포함한다.

[113] PTS : 프리젠테이션 타임 스탬프(Presentation Timestamp) - 현재 M-Unit의 화면 디스플레이를 위한 시간정보

[114]

[115] 2.2 MMT 페이로드(MMT Payload) 헤더 및 패킷 헤더(패킷 레벨 헤더)

[116] 도 5는 MMT 페이로드 포맷 구조를 설명하는 도면이다.

[117] MMT 패킷 레벨은 두 단계의 헤더를 가질 수 있다. 하나는 MMTPL(MMT Payload Format) 헤더이며 다른 하나는 MMTP(MMT Packet) 헤더이다. 하나의 MMT 패킷은 하나의 MMT 페이로드 포맷 유닛을 포함한다. MMT에서 새롭게 설계되는 기능들을 RTP를 재사용하는 어플리케이션과 MMTP를 사용하는 어플리케이션에서 동시에 사용될 수 있기 위해서는, 새롭게 설계되는 기능들에 해당하는 헤더필드가 MMT 페이로드 포맷 헤더에 위치해야 하며, MMTP는 기존의 RTP가 지원하는 기능을 승계하되 MMTP이 지원할 수 없는 최소한의 기능만을 수용할 수 있다.

[118] C : Control/Media Flag - 이 패킷이 컨트롤정보를 담고 있는지, 미디어 콘텐츠를

- 담고 있는지 표시한다. (1:Control, 0:Media)
- [119] S : Start of unit - 1로 표시된 경우, 본 패킷이 MU(또는 MFU)의 시작부분을 포함하고 있음을 표시한다. 여기서 유닛(unit)은 MU(또는 MFU)로 한정되지 아니한다.
- [120] E : End of unit - 1로 표시된 경우, 본 패킷이 MU(또는 MFU)의 끝부분을 포함하고 있음을 표시한다. 여기서 유닛(unit)은 MU(또는 MFU)로 한정되지 아니한다.
- [121] A : Anchor - 1로 표시된 경우, 본 패킷의 시작부분이 랜덤액세스지점과 같이 미디어의 Anchor point를 포함하고 있음을 표시한다.
- [122] M : Multiple unit - 1로 표시된 경우, 본 패킷이 여러 개의 MU(또는 MFU)를 포함하고 있음을 표시하며, 헤더의 마지막 부분에 “multiple MU Description” 필드가 추가된다. 일 실시예로서, Multiple MU는 MU 이외의 다른 유닛도 수용하는 Multiple Unit으로 사용될 수 있으며 flag_multiple_unit으로 사용될 수 있다.
- [123] Substream ID - 서로 다른 종류의 애셋(asset) 또는 패키지(package)를 패킷단위로 멀티플렉싱하기 위한 지시자(identifier) 역할을 할 수 있다. M2TS의 PID와 비슷한 역할을 수행한다. 본 ID값의 실제 할당은 세션을 셋업할 때 동적으로 결정되며, 인 밴드(in-band)로 보내지는 매핑 정보나, 아웃 오브 밴드(out-of-band)로 보내지는 별도의 세션 셋업 프로토콜을 이용할 수 있다.
- [124] Substream SeqNo. - 패킷 시퀀스번호로 동일한 Substream ID를 가지고 있는 패킷스트림에 대해 패킷당 1씩 증가한다.
- [125] 도 6은 MMT 패킷 헤더 구조를 설명하는 도면이다.
- [126] Sequence No. - 패킷 스트림의 시퀀스번호이다. 패킷 순서에 따라 1씩 증가시킨다.
- [127] 도 7은 인코더로부터 출력된 미디어데이터(coded data)가 전송을 위해 패킷화되는 과정을 도시한 도면이다.
- [128] 인코더는 AU 단위로 출력하는 인코더와 MFU 단위로 출력하는 인코더가 있으며, 두 인코더 공히 MUH-E의 공통 필드(common field)를 채우며, MFU를 출력하는 인코더의 경우 추가로 MUH-E의 프래그먼트 필드(Fragment Field)를 확장하여 채운다. (도 7에서 Fragment MU Case)
- [129] 이를 넘겨받은 스트리밍 시스템은 MUH-E 뒤에 MUH-S를 붙인 MU(또는 MFU)의 스트림을 형성하며 MMT 애셋 스트림(Asset stream)을 만들 수 있다. 각각의 MMT 애셋 스트림은 “애셋 정보(Asset Info)”를 추가하여 독립 MMT 애셋(standalone MMT Asset)을 형성한다. 애셋 정보(Asset Info)는 현재 애셋에 필요한 부가 정보이나 전송시에는 전송될 필요가 없는 정보를 포함할 수 있다.
- [130] 이렇게 구성된 다양한 코덱 또는 미디어타입 별 독립 MMT 애셋 들은 화면 구성 정보(composition information)와 전송 특성 정보(Transmission Characteristic)과 함께 MMT 패키지(MMT Package)를 구성한다. 화면 구성 정보는

신호 타입(Signal Type)의 MMT 패킷이나 다른 아웃 오브 밴드(out-of-band) 전송 채널을 통해 보내질 수 있다. 또한 각 MMT 애셋 별 전송 특성 정보는 애셋 정보(Asset Info)에 저장하거나, MMT 패키지 내의 별도 필드로 저장할 수 있다.

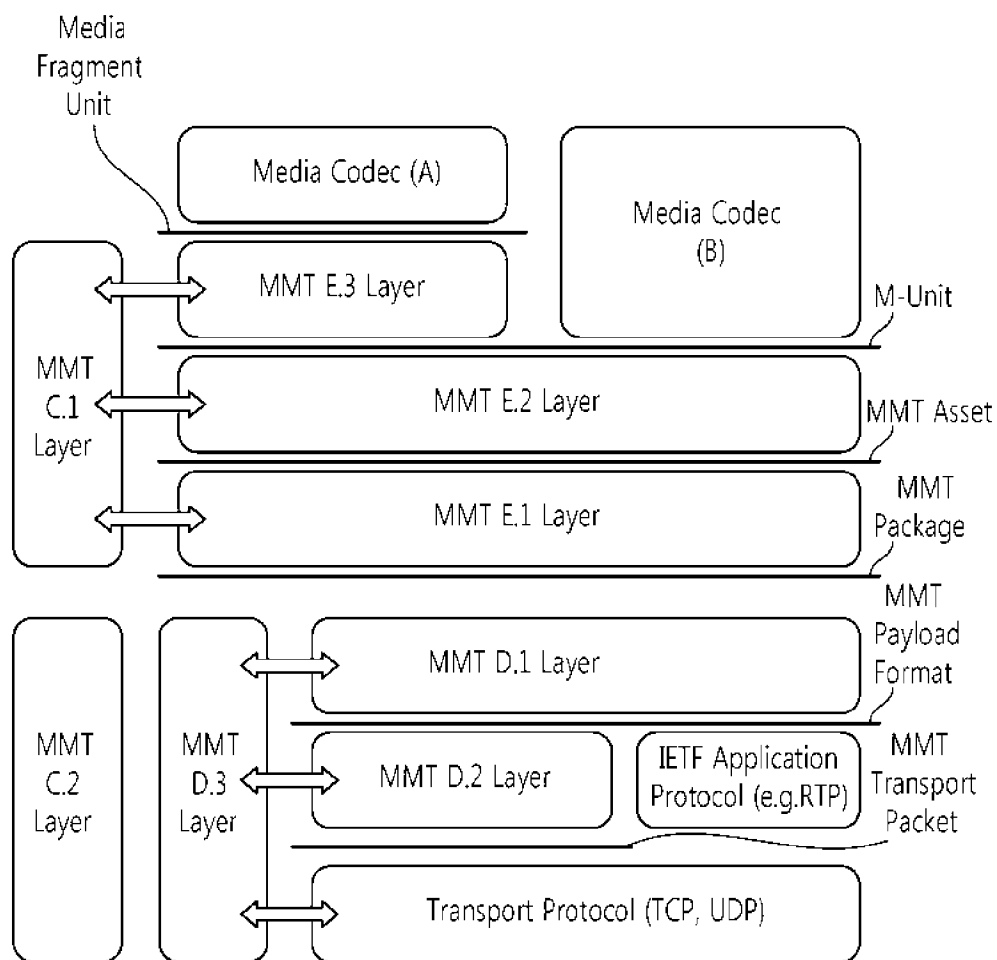
- [131] 스트리밍 시, 시스템은 MU(또는 MFU)를 MMT 패키지 내의 독립 애셋으로부터 추출하여 현재 네트워크에서 가용한 MTU 패킷크기에 따라 적절한 패킷화 방법을 선택할 수 있다(병합 또는 분할). 그리고 시스템은 MMTPL 헤더(도 7의 PLH)와 MMTP헤더를 패킷에 추가한 후 하위 계층의 UDP 또는 TCP 프로토콜로 전송한다. 이 때, 시스템은 각 애셋 별 전송 특성 정보(Transmission Characteristics)에 따라 적절한 프로토콜을 선택하거나(UDP 또는 TCP) 응용계층 FEC(application layer FEC) 등과 같은 에러 처리를 위한 정책을 선택할 수 있다.
- [132] 만약 어플리케이션이 RTP 프로토콜을 사용하는 경우 시스템은 MMTPL 헤더 이후 RTP 헤더를 붙인 후 전송할 수 있다.
- [133] 도 8은 본 발명의 일 실시예에 따른 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법의 흐름도이다.
- [134] 도 8에 도시된 바와 같이, 본 발명의 일 실시예에 따른 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법은 먼저 미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-unit) 헤더가 하나의 포맷을 공유하도록 미디어 데이터를 캡슐화한다(S810). 즉, 미디어 프래그먼트 유닛(MFU) 헤더와 M-유닛(MU) 헤더는 하나의 포맷을 공유하며, 단일 포맷 내에서 MFU의 경우 분할화(fragmentation)와 연관된 정보들을 담기 위해 옵션으로 확장될 수 있다. 만약 MFU와 MU에 대해 서로 다른 포맷을 제공하게 된다면, 인코더는 둘 중 어느 헤더까지를 적어주어야 하는지 구분하기 위하여 추가적인 작업을 필요로 하는 경우가 발생한다.
- [135] 이후, MMT 페이로드 포맷이 RTP 헤더가 지원하지 않는 동작을 위한 헤더 필드를 포함하도록 하여 상기 캡슐화된 미디어 데이터를 특정 엔티티(entity)로부터 다른 엔티티로 전달한다(S820). RTP 등 기존 어플리케이션 전송포맷을 재활용하는 어플리케이션에서 본 MMT 전달 표준에서 새롭게 제안하는 동작을 활용할 수 있기 위해서는 RTP 등 기존의 프로토콜에서 지원하지 않는 새로운 기능의 경우 MMT 페이로드 포맷에 위치해야만 RTP를 통하여 전송되는 경우와 MMTP를 통하여 전송되는 경우의 양쪽 모두 동작할 수 있다.
- [136] 이후, 본 발명의 일 실시예에 따른 미디어 데이터 제공 방법은 상기 미디어 데이터의 전달을 제어한다(S830).
- [137]

청구범위

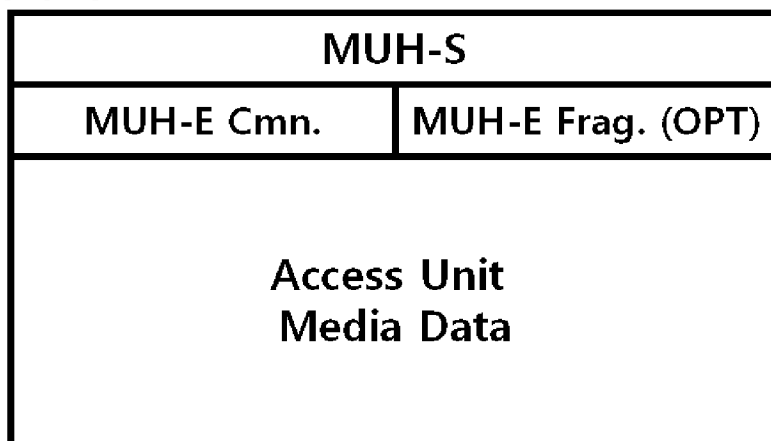
[청구항 1]

미디어 프래그먼트 유닛(Media Fragment Unit, MFU) 헤더와 M-유닛(M-unit) 헤더가 하나의 포맷을 공유하도록 미디어 데이터를 캡슐화하는 단계;
MMT 페이로드 포맷이 RTP 헤더가 지원하지 않는 동작을 위한 헤더 필드를 포함하도록 하여 상기 캡슐화된 미디어 데이터를 특정 엔티티(entity)로부터 다른 엔티티로 전달하는 단계; 및
상기 미디어 데이터의 전달을 제어하는 단계를 포함하는 오버헤드를 최소화한 헤더를 가지는 패킷 기반의 미디어 데이터 전송 방법

[Fig. 1]



[Fig. 2]



[Fig. 3]

```

0
0 1 2 3 4 5 6 7
+==+==+==+==+==+==+
|T|A|L|(TBD.) |F|      : Common Field
+==+==+==+==+==+==+
|S|E| Frg.Order |
+--+--+--+--+--+--+--+ : Fragment Field (Exists if F=1)
|      (TBD.)      |
+==+==+==+==+==+==+

```

T : Timed or Non-timed Flag

A : Anchor (Should be placed at the beginning of the transport Packet)

L : Lossless (Contents to be delivered in lossless manner)

F : Fragmented

S : Start of MU

E : End of MU

[Fig. 4]

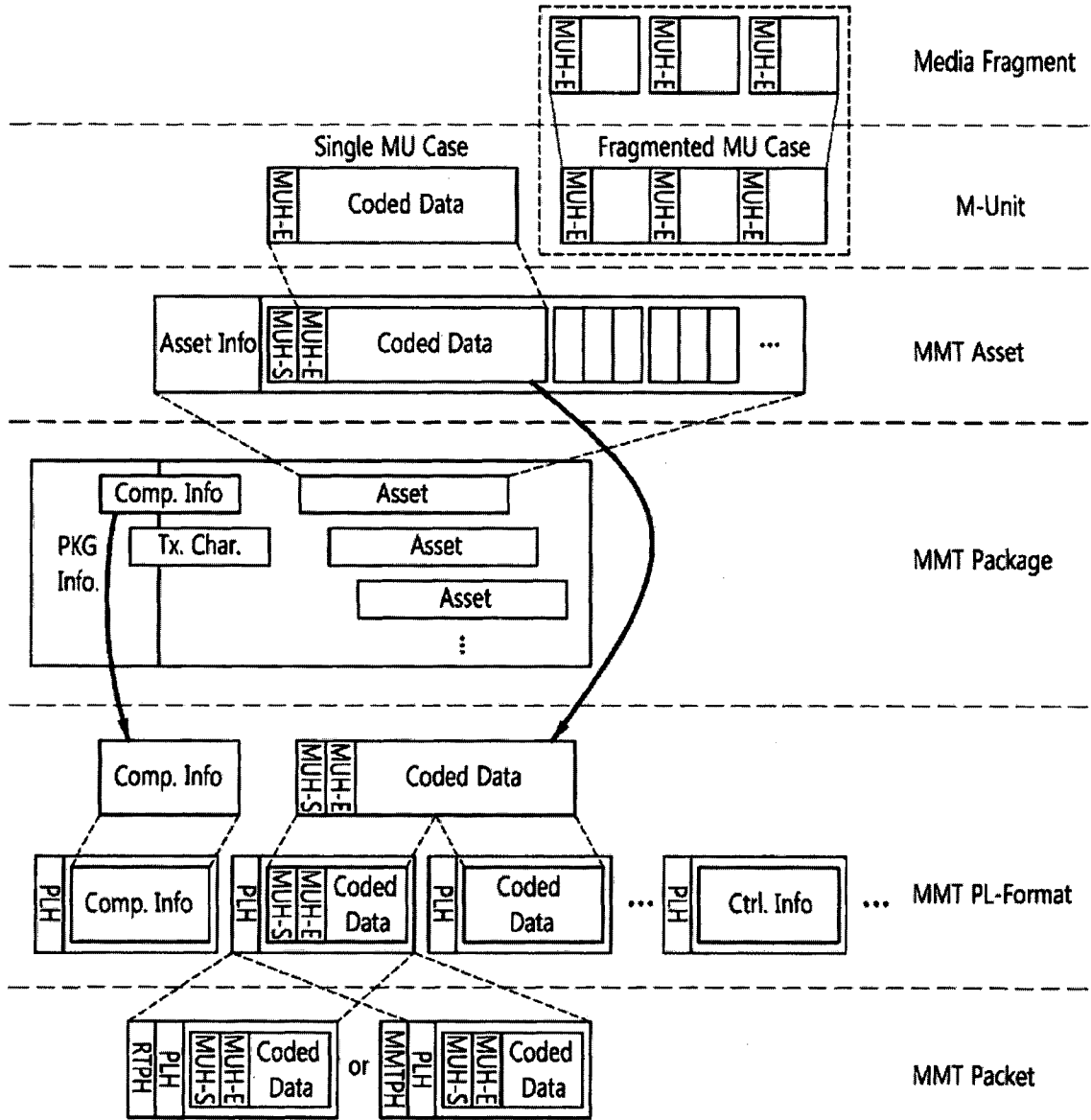
```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+
|                               PTS                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               PTS                               |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                               (TBD.)                           |
+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+==+

```

PTS : Presentation Timestamp (32bit)

[Fig. 7]



[Fig. 8]

