



**(19) 대한민국특허청(KR)**  
**(12) 등록특허공보(B1)**

(45) 공고일자 2019년09월02일  
 (11) 등록번호 10-2017408  
 (24) 등록일자 2019년08월27일

- (51) 국제특허분류(Int. Cl.)  
 HO4L 29/08 (2006.01) HO4W 4/00 (2018.01)
- (52) CPC특허분류  
 HO4L 67/16 (2013.01)  
 HO4L 67/322 (2013.01)
- (21) 출원번호 10-2017-7027810
- (22) 출원일자(국제) 2016년03월02일  
 심사청구일자 2017년09월28일
- (85) 번역문제출일자 2017년09월28일
- (65) 공개번호 10-2017-0123676
- (43) 공개일자 2017년11월08일
- (86) 국제출원번호 PCT/US2016/020407
- (87) 국제공개번호 WO 2016/141037  
 국제공개일자 2016년09월09일
- (30) 우선권주장  
 62/126,935 2015년03월02일 미국(US)
- (56) 선행기술조사문헌  
 KR1020140111407 A\*  
 WO2015013559 A1  
 US20120005276 A1  
 \*는 심사관에 의하여 인용된 문헌

- (73) 특허권자  
 콘비다 와이어리스, 엘엘씨  
 미국 19809-3727 델라웨어주 월밍턴 스위트 300  
 벨레뷰 파크웨이 200
- (72) 발명자  
 루, 광  
 캐나다 엘4제이 9비6 온타리오 손힐 멘델 크레스  
 트 1  
 스타시닉, 마이클, 에프.  
 미국 18940 펜실베이니아주 뉴타운 앤드류 드라이  
 브 190  
 (뒷면에 계속)
- (74) 대리인  
 양영준, 김연송, 백만기

전체 청구항 수 : 총 16 항

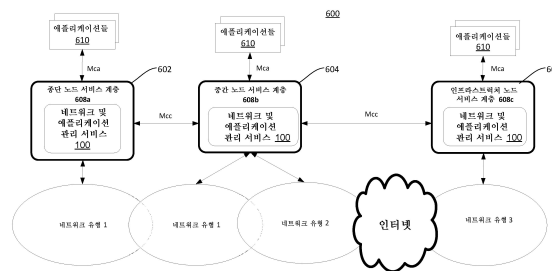
심사관 : 장상배

(54) 발명의 명칭 서비스 계층 능력을 사용한 네트워크 및 애플리케이션 관리

**(57) 요약**

서비스 계층(SL)에 대한 새로운 서비스인 네트워크 및 애플리케이션 관리 서비스(NAMS)가 본원에서 기술된다. 예시적인 실시예에 따르면, NAMS는 애플리케이션들, 기반 네트워크들, 및 다른 서비스들로부터의 정보를 수집, 통합, 및 처리한다.

**대표도**



(52) CPC특허분류

*HO4L 67/327* (2013.01)

*HO4L 67/34* (2013.01)

*HO4W 4/70* (2018.02)

(72) 발명자

**시드, 데일, 엠.**

미국 18104 펜실베이니아주 앨런타운 노스 36번 스트리트 229

**등, 리쥘**

미국 92130 캘리포니아주 샌 디에고 아프리카 홀리 트레일 6085

**플라딘, 카탈리나, 엠.**

미국 19040 펜실베이니아주 해트버러 롤러 로드 137

**플린, 윌리엄, 로버트, 4세**

미국 19473 펜실베이니아주 슈웬크스빌 메이베리 로드 451

## 명세서

### 청구범위

#### 청구항 1

프로세서, 메모리, 및 통신 회로부를 포함하는 장치로서, 상기 장치는 그의 통신 회로부를 통해 통신 네트워크에 연결되고, 상기 장치는 상기 장치의 상기 메모리에 저장된 컴퓨터 실행가능 명령어들을 추가로 포함하고, 상기 컴퓨터 실행가능 명령어들은 상기 장치의 상기 프로세서에 의해 실행될 때, 상기 장치로 하여금

상기 네트워크 상의 애플리케이션 또는 서비스로부터 상기 네트워크와 연관된 정보에 관한 제1 질의를 수신하게 하고;

상기 통신 네트워크를 통해 상기 네트워크 상의 제1 노드로부터, 상기 네트워크와 연관된 적어도 하나의 상태 업데이트를 표시하는 제1 메시지를 수신하게 하고 - 상기 제1 메시지는 상기 제1 질의에 기초함 - ;

상기 네트워크와 연관된 상기 적어도 하나의 상태 업데이트를 저장하게 하며;

상기 적어도 하나의 상태 업데이트가 상기 애플리케이션 또는 상기 서비스에 영향을 미치는지를 결정하게 하고;

상기 적어도 하나의 상태 업데이트가 상기 애플리케이션 또는 상기 서비스에 영향을 미치면, 상기 애플리케이션 또는 서비스가 상기 적어도 하나의 상태 업데이트에 기초하여 조치를 취할 수 있도록, 상기 적어도 하나의 상태 업데이트를 표시하는 제2 메시지를, 상기 애플리케이션 또는 상기 서비스를 호스팅하는 제2 노드에게 송신하게 하는, 장치.

#### 청구항 2

제1항에 있어서, 상기 제2 메시지는 상기 애플리케이션 또는 서비스의 전송 속도의 변화, 상기 애플리케이션 또는 서비스의 전송 지속시간의 변화, 상기 애플리케이션 또는 서비스와 연관된 데이터 품질의 변화, 상기 애플리케이션 또는 서비스와 연관된 해상도의 변화, 미리 결정된 양의 시간 후에 전송하라는 명령어, 네트워크들을 전환하라는 명령어, 또는 상기 애플리케이션 또는 서비스가 전환할 수 있는 제2 네트워크 중 적어도 하나를 추가로 표시하는, 장치.

#### 청구항 3

제1항에 있어서, 상기 적어도 하나의 상태 업데이트는 상기 네트워크와 연관된 데이터 속도(data rate), 상기 네트워크에 의해 제공되는 서비스 품질, 상기 네트워크와 연관된 혼잡 상태, 상기 네트워크와 연관된 패킷 크기, 네트워크 전송 간격, 시간 단위당 최대 데이터, 상기 네트워크에 의해 지원되는 하나 이상의 디바이스들, 상기 네트워크에 의해 지원되는 하나 이상의 서비스들, 또는 상기 네트워크의 지리적 커버리지 중 적어도 하나를 표시하는, 장치.

#### 청구항 4

삭제

#### 청구항 5

제1항에 있어서, 상기 컴퓨터 실행가능 명령어들은 추가로 상기 장치로 하여금:

상기 애플리케이션 또는 서비스로부터 상기 제1 질의를 수신한 것에 응답하여 제2 질의를 제1 노드에게 송신하게 하고, 상기 제2 질의는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 장치.

#### 청구항 6

제1항에 있어서, 상기 컴퓨터 실행가능 명령어들은 추가로 상기 장치로 하여금:

상기 제1 메시지가 제2 질의에 응답하여 수신되도록 상기 제2 질의를 상기 제1 노드에게 송신하게 하고, 상기 제2 질의는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 장치.

**청구항 7**

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 컴퓨터 실행가능 명령어들은 추가로 상기 장치로 하여금:

상기 제1 메시지가 질의에 응답하여 수신되도록 상기 질의를 상기 제1 노드에게 송신하게 하고, 상기 질의는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 장치.

**청구항 8**

제7항에 있어서, 상기 장치는 서비스 계층을 포함하고, 상기 질의는 하나 이상의 서비스들이 상기 서비스 계층에 추가되거나 그로부터 삭제된 것에 응답하여 송신되는, 장치.

**청구항 9**

제7항에 있어서, 상기 질의는 상기 네트워크에 연결된 하나 이상의 애플리케이션들의 변화에 기초하여 송신되는, 장치.

**청구항 10**

삭제

**청구항 11**

삭제

**청구항 12**

삭제

**청구항 13**

삭제

**청구항 14**

삭제

**청구항 15**

삭제

**청구항 16**

삭제

**청구항 17**

삭제

**청구항 18**

통신 네트워크에서 노드들을 관리하는 방법으로서, 상기 방법은 상기 네트워크를 통해 통신하는 서비스 계층을 호스팅하는 장치에 의해 수행되고, 상기 방법은:

상기 네트워크 상의 애플리케이션 또는 서비스로부터 상기 네트워크와 연관된 정보에 관한 제1 질의 메시지를 수신하는 단계;

상기 네트워크를 통해 상기 네트워크 상의 제1 노드로부터, 상기 네트워크와 연관된 적어도 하나의 상태 업데이트를 표시하는 제1 메시지를 수신하는 단계 - 상기 제1 메시지는 상기 제1 질의 메시지에 기초함 - ;

상기 네트워크와 연관된 상기 적어도 하나의 상태 업데이트를 저장하는 단계;

상기 적어도 하나의 상태 업데이트가 상기 애플리케이션 또는 상기 서비스에 영향을 미치는지를 결정하는 단계;

및

상기 적어도 하나의 상태 업데이트가 상기 애플리케이션 또는 상기 서비스에 영향을 미치면, 상기 애플리케이션 또는 서비스가 상기 적어도 하나의 상태 업데이트에 기초하여 조치를 취할 수 있도록, 상기 적어도 하나의 상태 업데이트를 표시하는 제2 메시지를, 상기 애플리케이션 또는 상기 서비스를 호스팅하는 상기 네트워크 상의 제2 노드에게 송신하는 단계를 포함하는, 방법.

**청구항 19**

제18항에 있어서, 상기 제2 메시지는 상기 애플리케이션 또는 서비스의 전송 속도의 변화, 상기 애플리케이션 또는 서비스의 전송 지속시간의 변화, 상기 애플리케이션 또는 서비스와 연관된 데이터 품질의 변화, 상기 애플리케이션 또는 서비스와 연관된 해상도의 변화, 미리 결정된 양의 시간 후에 전송하라는 명령어, 네트워크들을 전환하라는 명령어, 또는 상기 애플리케이션 또는 서비스가 전환할 수 있는 제2 네트워크 중 적어도 하나를 추가로 표시되는, 방법.

**청구항 20**

제18항에 있어서, 상기 적어도 하나의 상태 업데이트는 상기 네트워크와 연관된 데이터 속도(data rate), 상기 네트워크에 의해 제공되는 서비스 품질, 상기 네트워크와 연관된 혼잡 상태, 상기 네트워크와 연관된 패킷 크기, 네트워크 전송 간격, 시간 단위당 최대 데이터, 상기 네트워크에 의해 지원되는 하나 이상의 디바이스들, 상기 네트워크에 의해 지원되는 하나 이상의 서비스들, 또는 상기 네트워크의 지리적 커버리지 중 적어도 하나를 표시하는, 방법.

**청구항 21**

삭제

**청구항 22**

제18항에 있어서, 상기 방법은:

상기 애플리케이션 또는 서비스로부터 상기 제1 질의 메시지를 수신한 것에 응답하여 제2 질의 메시지를 상기 제1 노드에게 송신하는 단계를 추가로 포함하고, 상기 제2 질의 메시지는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 방법.

**청구항 23**

제18항에 있어서, 상기 방법은:

상기 제1 메시지가 제2 질의 메시지에 응답하여 수신되도록 상기 제2 질의 메시지를 상기 제1 노드에게 송신하는 단계를 추가로 포함하고, 상기 제2 질의 메시지는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 방법.

**청구항 24**

제18항 내지 제20항 중 어느 한 항에 있어서, 상기 방법은:

상기 제1 메시지가 질의에 응답하여 수신되도록 상기 질의를 상기 제1 노드에게 송신하는 단계를 추가로 포함하고, 상기 질의는 상기 네트워크와 연관된 정보에 대한 요청을 포함하는, 방법.

**청구항 25**

제24항에 있어서, 상기 질의는 하나 이상의 서비스들이 상기 서비스 계층에 추가되거나 그로부터 삭제된 것에 응답하여 송신되는, 방법.

**청구항 26**

제24항에 있어서, 상기 질의는 상기 네트워크에 연결된 하나 이상의 애플리케이션들의 변화에 기초하여 송신되는, 방법.

**발명의 설명**

**기술 분야**

[0001] 관련 출원들의 상호 참조

[0002] 본 출원은 2015년 3월 2일자로 출원된 미국 가특허출원 제62/126,935호 - 그 개시 내용은 이로써 참조에 의해 그 전체가 본원에 기재된 것처럼 원용됨 - 의 이익을 주장한다.

**배경 기술**

[0003] 개발 중인 oneM2M 표준은 도 1에 도시된 "공통 서비스 엔티티(Common Services Entity, CSE)"라고 불리는 서비스 계층을 정의한다. CSE는 oneM2M-TS-0001, oneM2M Functional Architecture (V0.4.2) - 참조에 의해 그의 내용이 본원에 제시되어 있는 것처럼 원용됨 - 에 추가로 기술되어 있다. 서비스 계층의 목적은, 예를 들어, e-헬스(e-Health), 차량군 관리(fleet management), 및 스마트 홈들에 관련된 시스템들 및 애플리케이션들과 같은, 상이한 "수직(vertical)" M2M(machine-to-machine) 사일로(silo) 시스템들 및 애플리케이션들에 의해 이용될 수 있는 "수평적" 서비스들을 제공하는 것이다. 도 1을 참조하면, CSE는 4개의 참조점들을 지원한다. Mca 참조점은 애플리케이션 엔티티(Application Entity, AE)와 인터페이싱한다. Mcc 참조점은 동일한 서비스 제공자 도메인(service provider domain) 내의 다른 CSE와 인터페이싱하고, Mcc' 참조점은 상이한 서비스 제공자 도메인 내의 다른 CSE와 인터페이싱한다. Mcn은 CSE가 기반 네트워크(underlying network)(들)에서의 기능들에 액세스할 수 있게 하면서 또한 기반 네트워크를 CSE에 대체로 투명하게 유지하는 표준화된 인터페이스이다. Mcn 참조점은 기반 네트워크 서비스 엔티티(underlying network service entity(NSE))와 인터페이싱한다. NSE는, 예를 들어, 디바이스 관리, 위치 서비스들, 및 디바이스 트리거링과 같은, 기반 네트워크 서비스들을 CSE에 제공한다. 기반 네트워크에 대한 NSE의 인터페이스들은 oneM2M에 의해 정의되지 않는다. 이러한 인터페이스들은 전형적으로 기반 네트워크에 의해 정의되지만 그들은 네트워크 운영자(network operator)에 의해 정의되는 독점 인터페이스(proprietary interface)들일 수 있다.

[0004] 또한 도 2를 참조하면, 주어진 CSE는 "공통 서비스 기능(Common Service Function, CSF)들"이라고 불리는 다수의 논리적 기능(logical function)들을 포함할 수 있다. 예시적인 CSF는 "발견(Discovery)" 및 "데이터 관리 및 리포지토리(Data Management & Repository)"를 포함하지만, 이들로 제한되지 않는다. 도 2는 oneM2M에서 개발 중인 CSF들을 나타내고 있다. 예시된 네트워크 서비스 노출(Network Service Exposure), 서비스 실행 및 트리거링(Service Execution and Triggering, NSSE) 기능은 기반 네트워크(들)와 인터페이싱한다. NSSE에 의해 지원되는 현재 기능들은 디바이스 트리거링 및 위치 관리이다. 도 2에 도시된 애플리케이션 및 서비스 계층 관리(Application and Service Layer Management, ASM) CSF는, 예를 들어, 서비스 계층에서 소프트웨어를 설치, 활성화(activate) 및 비활성화(de-activate)하기 위해, CSE 또는 애플리케이션 엔티티(AE)의 소프트웨어 모듈에 대한 소프트웨어 구성 및 관리를 제공한다.

[0005] oneM2M 이전에, ETSI M2M은, ETSI TS 102.690, Machine-to-Machine (M2M) Functional architecture (V2.1.1) - 참조에 의해 그의 내용이 본원에 기재된 것처럼 원용됨 - 에 정의되는, 그 자신의 서비스 계층 지향 M2M 표준을 발표하였다. oneM2M의 아키텍처는 ETSI M2M과 유사하다. ETSI M2M은 서비스 능력들이라고 불리는 다양한 서비스 모듈들로 이루어진 서비스 능력 계층 엔티티(Service Capability Layer entity)를 정의하였다. 서비스들은 네트워크 서비스 노출 능력(Network Exposure Service Capability)에 의해 기반 네트워크에 노출될 수 있다.

[0006] 이제 도 3을 참조하면, 3GPP는 3GPP TS 23.682, Architecture enhancements to facilitate communications with packet data networks and applications, Release 11, V12.0.0 - 참조에 의해 그의 내용이 본원에 기재된 것처럼 원용됨 - 에서 MTC 아키텍처를 정의하였다. 외부 네트워크에서의 MTC 애플리케이션은 전형적으로 애플리케이션 서버(Application Server, AS)에 의해 호스팅되고, 추가의 부가 가치 서비스들을 위해 SCS를 사용할 수 있다. 3GPP 시스템은 전송(transport), 가입자 관리, 및, MTC 디바이스들에 대한 다양한 아키텍처 향상(architectural enhancement)들을 비롯한, 다른 통신 서비스들(예컨대, 제어 평면 디바이스 트리거링)을 제공한다. 3GPP는 MTC 디바이스들에 대한 3개의 통신 모델들: 직접 모델(direct model), 간접 모델(indirect model), 및 하이브리드 모델(hybrid model)을 정의하였다. 직접 모델에서, AS는 어떤 외부 SCS도 사용하지 않고 사용자 장비(user equipment, UE)와의 직접적인 사용자 평면 통신을 수행하기 위해 운영자 네트워크(operator network)에 직접적으로 연결한다. 외부 네트워크에서의 애플리케이션은 3GPP 시스템에 의해 제공되

는 서비스들을 사용할 수 있다. 간접 모델에서, AS는, MTC에 대한 추가의 부가 가치 서비스들(예컨대, 제어 평면 디바이스 트리거링)을 이용하기 위해, SCS의 서비스들을 통해 운영자 네트워크에 간접적으로 연결한다. 하이브리드 모델에서, AS는, SCS도 사용하면서, UE와의 직접적인 사용자 평면 통신을 수행하기 위해 운영자의 네트워크에 직접적으로 연결하기 위해, 직접 모델과 간접 모델을 동시에 사용한다. 3GPP 네트워크 관점에서 볼 때, AS로부터의 직접적인 사용자 평면 통신과 SCS로부터의 임의의 부가 가치 제어 평면 관련 통신은 독립적이며, AS에 의해 호스팅되는 동일한 MTC 애플리케이션을 서빙하고 있을지라도, 서로 아무런 상관관계가 없다.

[0007] 도 3은 앞서 참조된 oneM2M 아키텍처 규격(architecture specification)으로부터 복사한 것이다. 도 3은 oneM2M 참조점들이 3GPP MTC 아키텍처에 어떻게 적용될 수 있는지를 도시하고 있다. 애플리케이션 서비스 노드(Application Service Node)는, 하나의 공통 서비스 엔티티(CSE)를 포함하고 적어도 하나의 애플리케이션 엔티티(AE)를 포함하는, 노드이다. 예를 들어, 애플리케이션 서비스 노드는 M2M 디바이스일 수 있다. 인프라스트럭처 노드(Infrastructure Node, IN)는, 하나의 공통 서비스 엔티티를 포함하고 0개 이상의 애플리케이션 엔티티(Application Entity)들을 포함하는, 노드이다. 예를 들어, IN은 네트워크 서버일 수 있다. 도 3에 도시된 Mcc 참조점은 3GPP 네트워크 상에서 IN-CSE와 ASN-CSE 사이에 있다.

[0008] 3GPP는 3GPP TS 32.101 Telecommunication Management: Principles and high level requirements - 참조에 의해 그의 내용이 본원에 기재된 것처럼 원용됨 - 에서 네트워크 관리를 위한 그의 OAM(Operations, Administration, Maintenance) 아키텍처를 정의한다. 기술된 바와 같이, 네트워크 관리자(Network Manager, NM)는, 디바이스들 및 장비와 같은, 네트워크 요소들의 관리를 책임지고 있는 기능들의 패키지를 제공한다. 관리 기능들은 계정 관리, 장애 관리(fault management), 서비스 품질(QoS), 성능 관리 등을 포함한다. 관리 절차들은, 예를 들어, SNMP(Simple Network Management Protocol)와 같은, 개방형 표준(open standard)들에 의해 지원된다.

[0009] RFC 3411에 정의된 바와 같은, SNMP(Simple Network Management Protocol)는 OSI 모델에서의 애플리케이션 계층 프로토콜로 간주된다. SNMP는 네트워크에서의 디바이스들을 관리하는 데 흔히 사용된다. 네트워크 관리를 위해 SNMP를 사용하는 컴퓨터 네트워크 시스템은 하기의 컴포넌트들: SNMP 관리자, SNMP 에이전트(SNMP agent), 및 SNMP MIB(Management Information Base)로 이루어져 있을 수 있다. SNMP 관리자는 네트워크 관리자(network administrator)의 머신 상에서 실행되는 소프트웨어 또는 컴퓨터 네트워크를 관리하는 임의의 인간 관리자(human manager)일 수 있다. SNMP 에이전트는 모니터링되어야 하는 네트워크 노드 상에서 실행되는 소프트웨어를 지칭할 수 있다. 이 노드는 프린터, 라우터 등일 수 있다. RFC 3418에서 정의되는, SNMP MIB는 관리자와 에이전트 간의 데이터 교환이 구조화된 채로 유지되도록 보장하는 컴포넌트이다. MIB는 트리 구조로 구성되고, 구조의 기본 컴포넌트(basic component)는 객체(object)이다. 예를 들어, 시스템 가동 시간(system up time)은 "iso.org.dod.internet.mgmt.mib-2.system.sysUpTime"로서 표현될 수 있다.

[0010] 관리자와 에이전트 사이의 SNMP 통신은 메시지들의 형태로 이루어진다. 통신을 위해 사용되는 예시적인 기본 메시지들은: SNMP GET, SNMP GET-NEXT, SNMP GET-RESPONSE, SNMP SET, 및 SNMP TRAP을 포함한다. GET 및 GET-NEXT 메시지들은 특정의 MIB 객체의 값을 페치(fetch)하는 데 사용된다. GET-RESPONSE 메시지는 GET 또는 GET-NEXT 메시지에 대한 응답을 송신하기 위해 에이전트에 의해 주로 사용된다. SET 메시지는 특정의 MIB 객체의 새로운 값을 에이전트에 설정하기 위해 관리자에 의해 사용된다. TRAP 메시지는 어떤 객체에 대한 어떤 경보 값(alarming value)들에 관한 정보를 관리자에게 송신하기 위해 에이전트에 의해 사용된다.

[0011] SNMP를 통해, 라우터들, 프린터들, 허브들, 또는 일반 컴퓨터들과 같은 네트워크 디바이스들에 관한 정보를 검색할 수 있다. 검색된 정보는 시스템 가동 시간, CPU 사용 레벨, 네트워크 설정 등을 포함할 수 있다. 디바이스들이 또한 SNMP에 의해 구성될 수 있다.

[0012] SNMP는 보통 전반적인 관리 프레임워크를 제공하는 정책들에 따라 작동한다. 예를 들어, 네트워크 관리자는 시스템 로그 메시지들에 기초하여 이벤트들에 대한 트랩(trap)들을 발생(raise)시키는 이벤트 정책을 구성할 수 있다. 임계값 폴링 정책(threshold poll policy)들은, 예를 들어, 메모리 사용량(memory usage)과 같은, 시스템 및 네트워크 정보를 모니터링하기 위해 상이한 임계값들을 설정할 수 있다.

[0013] W3C 네트워크 정보 API는 웹 애플리케이션들이 디바이스의 기반 연결 정보(underlying connection information)에 액세스하기 위한 인터페이스를 제공한다. W3C 규격은, 현재, 사용자 에이전트에 2개의 속성들: bandwidth 및 metered를 노출하라고 요청한다. 인터페이스가 애플리케이션들과 기반 네트워크 사이에 바로 있는 것처럼 보인다.

[0014] 과부하, 혼잡, 및 약한 신호 강도와 같은, 잠재적인 네트워크 연결성(network connectivity) 문제들은, 예를 들어, 연결 중단들(이들로 한정되지는 않음)과 같은, 비-최적화된 중단간 M2M 연결들을 야기할 수 있다. 어떤 경우에, 네트워크들을 통해 실행되는 애플리케이션들이 네트워크 상황을 인식하지 못하거나, 애플리케이션들이 자신이 상주하는 로컬 네트워크의 상태만을 인식할 수 있다. 네트워크 문제들은 애플리케이션 레벨에서 긴 지연들 및 심지어 연결들의 단절(drop)을 야기할 수 있다. 이러한 문제들은 M2M 시스템들뿐만 아니라 비-M2M 시스템에도 적용될 수 있다. 이 문제들은, 예를 들어, 많은 수의 애플리케이션들, 제약된(constrained)(예컨대, 저전력, 손실(lossy)) 디바이스들, 및 로컬 네트워크들에서의 한계들과 같은, 다양한 인자들로 인해 비-M2M 시스템들보다 M2M 네트워크들에서 더 명백할 수 있다. 예를 들어, 제약된 애플리케이션(constrained application)들은 연결 문제들을 핸들링하기 위해 내장형 지능(build-in intelligence)을 지원할 수 없을지도 모른다. 다른 예로서, 제약된 애플리케이션은 네트워크 연결의 변화에 관계없이 그의 트랜잭션을 유지할 수 있다. 또 다른 예로서, 제약된 애플리케이션들은 전형적으로 다른 애플리케이션들을 고려하여 그들의 트랜잭션들을 적응시킬 수 없다. 따라서, 앞서 기술된 바와 같이, 예를 들어, 자원 제약된(예컨대, M2M) 디바이스들을 갖는 네트워크들에서의 네트워크 및 애플리케이션 관리에 대한 현재의 접근법들은 능력 및 효율성이 부족하다.

**발명의 내용**

[0015] 서비스 계층(SL)에 대한 새로운 서비스인 네트워크 및 애플리케이션 관리 서비스(Network and Application Management Service, NAMS)가 본원에서 기술된다. 예시적인 실시예에 따르면, NAMS는 애플리케이션들, 기반 네트워크들, 및 다른 서비스들로부터의 정보를 수집, 통합, 및 처리한다. NAMS는 애플리케이션들 및 기반 네트워크들을 최적화하기 위해 그 자신의 정책들에 기초하여 출력을 생성할 수 있다. 다수의 예들이 본원에서 M2M 시스템의 맥락에서 제시되고 있지만, NAMS가 M2M 시스템들 및 비-M2M 시스템들에서 구현될 수 있다는 것이 이해될 것이다.

[0016] 예시적인 실시예에서, NAMS를 포함하는 장치는 프로세서, 메모리, 및 통신 회로부를 포함한다. 본 장치는 그의 통신 회로부를 통해 통신 네트워크에 연결되고, 본 장치는, 본 장치의 프로세서에 의해 실행될 때, 본 장치로 하여금, 통신 네트워크를 통해 네트워크 상의 제1 노드로부터, 네트워크와 연관된 적어도 하나의 상태 업데이트를 표시하는 제1 메시지를 수신하게 하는, 본 장치의 메모리에 저장된, 컴퓨터 실행가능 명령어들을 추가로 포함한다. 본 장치는 네트워크와 연관된 적어도 하나의 상태 업데이트를 저장하고, 적어도 하나의 상태 업데이트가 애플리케이션 또는 서비스에 영향을 미치는지를 결정할 수 있다. 적어도 하나의 상태 업데이트가 애플리케이션 또는 서비스에 영향을 미치면, 본 장치는 애플리케이션 또는 서비스가 적어도 하나의 상태 업데이트에 기초하여 조치를 취할 수 있도록 적어도 하나의 상태 업데이트를 표시하는 제2 메시지를, 애플리케이션 또는 서비스를 호스팅하는 제2 노드에게 송신할 수 있다.

[0017] 이 발명의 내용은 이하에서 발명을 실시하기 위한 구체적인 내용에 추가로 기술되는 일련의 개념들을 간략화된 형태로 소개하기 위해 제공되어 있다. 이 발명의 내용은 청구된 발명 요지의 핵심적인 특징들 또는 필수적인 특징들을 확인해주는 것으로 의도되어 있지도 않고, 청구된 발명 요지의 범주를 제한하는 데 사용되는 것으로 의도되어 있지도 않다. 게다가, 청구된 발명 요지는 본 개시내용의 임의의 부분에서 살펴본 임의의 또는 모든 단점들을 해결하는 제한들로 제한되지 않는다.

**도면의 간단한 설명**

- [0018] 도 1은 oneM2M 아키텍처의 블록도;
- 도 2는 하나의 M2M 아키텍처의 공통 서비스 기능(CSF)들을 나타낸 도면;
- 도 3은 MTC(Machine-Type Communication)를 위한 3GPP 아키텍처의 블록도;
- 도 4는 네트워크 및 관리 서비스(NAMS)를 포함하지 않는 예시적인 M2M(machine-to-machine) 시스템을 도시한 도면;
- 도 5는 예시적인 실시예에 따른 NAMS를 또한 포함하는, 도 4에 도시된 예시적인 M2M 시스템을 도시한 도면;
- 도 6은 예시적인 실시예에 따른, 네트워크 및 애플리케이션 관리 서비스(NAMS)가 어떻게 서비스 계층(SL) 아키텍처에 적합할 수 있는지를 나타낸 도면;
- 도 7은 예시적인 실시예에 따른 NAMS가 어떻게 서비스 계층 기능 아키텍처에 적합할 수 있는지를 나타낸 도면;

- 도 8은 예시적인 실시예에 따른 NAMS의 예시적인 서브기능(subfunction)들을 나타낸 NAMS의 블록도;
- 도 9는 다른 예시적인 실시예에 따른 NAMS와 상호작용하는 도 8의 서브기능들을 독립형 적응 서비스(standalone adaptation service)들로서 나타낸 도면;
- 도 10은 NAMS 및 그와 연관된 적응 서비스들에 의해 수행될 수 있는 예시적인 동작들을 보여주는, 애플리케이션들로부터 네트워크들로의 흐름을 도시한 흐름도;
- 도 11은 NAMS 및 그와 연관된 적응 서비스들에 의해 수행될 수 있는 예시적인 동작들을 보여주는, 네트워크들로부터 애플리케이션들로의 흐름을 도시한 흐름도;
- 도 12는 예시적인 실시예에 따른, NAMS가 애플리케이션들, 네트워크들, 및 다른 서비스들을 구성하는 것을 보여주는 호 흐름(call flow)을 나타낸 도면;
- 도 13은 예시적인 실시예에 따른, NAMS가 애플리케이션들, 네트워크들, 및 다른 서비스들로부터 다양한 정보를 수집하는 것을 보여주는 호 흐름을 나타낸 도면;
- 도 14는 예시적인 실시예에 따른, NAMS가 수집된 정보를 처리하고 애플리케이션들, 네트워크들, 및 다른 서비스들에 대한 입력을 생성하는 것을 보여주는 호 흐름을 나타낸 도면;
- 도 15는 예시적인 실시예에 따른 서비스 계층 지원 네트워크 최적화(service layer assisted network optimization)의 일 예를 나타내는 흐름을 나타낸 도면;
- 도 16은 예시적인 실시예에 따른 서비스 계층 지원 애플리케이션 최적화(service layer assisted application optimization)의 일 예를 나타내는 흐름을 나타낸 도면;
- 도 17은 서비스 계층에서의 NAMS를 보여주는 예시적인 프로토콜 스택을 나타낸 도면;
- 도 18은 NAMS가 어떻게 oneM2M RoA(resource-oriented architecture)에 적합할 수 있는지의 일 예를 나타낸 도면;
- 도 19a 및 도 19b 각각은 NAMS에 의해 사용될 수 있는 예시적인 자원 구조를 나타낸 도면;
- 도 20은 HTTP 프로토콜 및 도 19a에 도시된 자원 구조를 사용하여 NAMS와 애플리케이션 엔티티 사이에서 교환되는 예시적인 메시지들을 보여주는 호 흐름을 나타낸 도면;
- 도 21은 NAMS 및 그에 관련된 기능들이 어떻게 oneM2M SoA(service-oriented architecture)에 적합할 수 있는지의 일 예를 나타낸 도면;
- 도 22는 예시적인 실시예에 따른, MQTT 프로토콜을 사용하여 NAMS, 서버, 및 클라이언트 사이에서 교환되는 예시적인 메시지들을 보여주는 호 흐름을 나타낸 도면;
- 도 23은 예시적인 실시예에 따른 NAMS에 의해 사용될 수 있는 예시적인 네트워크 서비스 엔티티(NSE) 인터페이스들의 블록도;
- 도 24는 도 23에 도시된 NSE 인터페이스들 및 예시적인 실시예에 따른 NAMS를 포함하는 3GPP 간접 및 하이브리드 모델의 블록도;
- 도 25a는 하나 이상의 개시된 실시예들이 구현될 수 있는 예시적인 M2M(machine-to-machine) 또는 IoT(Internet of Things) 통신 시스템의 시스템도.
- 도 25b는 도 25a에 예시된 M2M/IoT 통신 시스템 내에서 사용될 수 있는 예시적인 아키텍처의 시스템도.
- 도 25c는 도 25a에 예시된 통신 시스템 내에서 사용될 수 있는 예시적인 M2M/IoT 단말 또는 게이트웨이 디바이스의 시스템도.
- 도 25d는 도 25a의 통신 시스템의 양태들이 구현될 수 있는 예시적인 컴퓨팅 시스템의 블록도.

**발명을 실시하기 위한 구체적인 내용**

[0019] 본원에서 사용되는 바와 같이, 달리 명시되지 않는 한, "서비스 계층"이라는 용어는 논리적 엔티티를 지칭한다. 서비스 계층은, 예를 들어, M2M(machine-to-machine) 애플리케이션들 또는 IoT(Internet of Things) 애플리케이션들과 같은, 서비스 계층 상에서 애플리케이션들에 의해 공유할 수 있는 서비스 기능들로 이루어져 있을 수

있다. 달리 명시되지 않는 한, M2M 및 IoT라는 용어들은, 제한 없이, 서로 바꾸어 사용될 수 있다. "중단 노드(end node)"라는 용어는 서비스 계층 엔티티를 포함할 수 있거나 그렇지 않을 수 있는 디바이스들(예컨대, 계약된 디바이스)를 지칭한다. 인프라스트럭처 노드는, 본원에서 사용되는 바와 같이, 서비스 계층 엔티티를 포함하고, 애플리케이션들을 포함할 수 있거나 그렇지 않을 수 있는 네트워크 서버를 지칭할 수 있다. 중간 노드(middle node)는, 본원에서 사용되는 바와 같이, 서비스 계층 엔티티를 포함하고 애플리케이션들을 포함할 수 있거나 그렇지 않을 수 있는 게이트웨이를 지칭한다. 본원에서 사용되는 예시적인 참조점들을 이제부터 참조하면, Mca 참조점은 애플리케이션들과 서비스 계층 간의 통신을 지정한다. Mcn 참조점은 서비스 계층과 기반 네트워크 간의 통신을 지정한다. Mcc 참조점은 서로 동일한 서비스 제공자 도메인 내에 있는 하나의 서비스 계층과 다른 서비스 계층 간의 통신을 지정한다. Mcc' 참조점은 서로 상이한 서비스 제공자 도메인들에 있는 하나의 서비스 계층과 다른 서비스 계층 간의 통신을 지정한다. Mff 참조점은 서로 동일한 서비스 계층 내에 있는 상이한 서비스 기능들 간의 통신을 지정한다.

[0020] 앞서 기술된 바와 같이, 다양한 네트워크 문제들은 비-최적화된 M2M 연결들을 야기할 수 있다. 예로서, 도 4는 M2M 시스템과 연관될 수 있는 문제점들의 일 예를 도시하고 있다. 도 4를 참조하면, 사용자(Alice)는 그녀의 전화기에 2개의 스마트 홈 애플리케이션들(앱들)을 가지고 있다. 하나의 애플리케이션은 에너지 제어를 위한 것이고, 하나의 애플리케이션은 홈 보안을 위한 것이다. 그녀는 때때로 감시 카메라 및 에너지 제어 시스템으로부터 보고들을 수신하고, 시스템들을 원격으로 구성할 수 있다. 집으로 오는 길에, Alice는 한 시간 동안 지하철을 타고, 그 시간 동안 그녀의 스마트폰은 셀룰러 신호를 상실한다. 이 기간 동안, 그녀의 홈 에너지 제어 및 보안 시스템들로부터의 어떤 보고도 전달되지 않을 것이다. 그렇지만, 앱들과 디바이스들의 제약된 속성으로 인해, 그들은 전화를 통한 연결의 변화를 인식하지 못하고, 전달가능하지 않은 메시지들을 송신하는 것을 방지할 수 없다.

[0021] 앞서 언급된 기존의 메커니즘들이, 그 중에서도 특히, 이상의 예에서 예시된 문제들을 적절히 해결하지 못한다는 것이 본원에서 인식된다. 예를 들어, SNMP와 같은 네트워크 관리 프로토콜들을 사용하여, 네트워크 관리자는 네트워크 연결의 정보를 받을 수 있고, 디바이스들을 구성할 수 있다. 기존의 SNMP 정책들이 정보들의 트리거링을 구성하는 데 사용될 수 있다는 것이 본원에서 인식된다. 그렇지만, SNMP는 애플리케이션 정보를 갖지 않고, 애플리케이션들에 대한 중단들을 방지하기 위해 애플리케이션들에 통보할 수 없다. 3GPP 네트워크 관리 기능이 셀룰러 연결을 관리할 수 있지만, 모세관 네트워크(capillary network) 및 로컬 네트워크를 관리하지도 않고, 모세관 네트워크 및 로컬 네트워크에 관한 정보를 갖지도 않는다는 것이 본원에서 추가로 인식된다. 교차 계층 네트워크 관리 메커니즘(cross layer network management mechanism)들은, 노드에서의 MAC, IP, 및 애플리케이션 계층 프로토콜들과 같은, 상이한 계층들을 포함한다. 이러한 메커니즘들은 동일한 노드에서의 상위 계층들에 대한 기반 네트워크의 영향을 완화시킬 수 있지만, 중단간 시나리오들을 관리하는 데 한계가 있다. 예를 들어, 연결들이 TCP에 기초하면, TCP 혼잡 제어(congestion control) 및 느린 시작(slow start)은 송신되는 데이터의 양을 감소시킬 수 있다.

[0022] 따라서, 기존의 메커니즘들이, 예를 들어, 중단간 시스템(end-to-end system)에서 그들이 갖는 제한된 역할 및 정보로 인해, 문제점들을 완전히 해결하지 못한다는 것이 본원에서 인식된다. 기존의 메커니즘들은 또한 본질적으로 사후대응적(reactive)이고, 따라서 다양한 문제들이 발생하는 것을 방지하도록 설계되어 있지 않다.

[0023] 서비스 계층이 애플리케이션들, 서비스들, 및 기반 네트워크들로부터의 정보를 알고 있기 때문에, 서비스 계층이, 기존의 메커니즘들과 비교하여, 이상의 예들에 의해 예시된 문제점과 같은, 다양한 문제점들을 해결하기 위한 후보라는 것이 본원에서 인식된다. 게다가, 서비스 계층은 네트워크 및 애플리케이션 관리를 위해 이러한 정보를 집계할 수 있다. 현재, 상이한 기반 네트워크들을 감독 및 관리할 수 있고 적절한 정보를 애플리케이션들에게 제공하는(또는 그 반대인) 서비스들이 없다. 그에 부가하여, 서비스 계층이 현재 이용가능하지 않은 중단간 통신에 대한 진보된 최적화(advanced optimization)를 제공할 수 있다는 것이 본원에서 인식된다. 본원에 기술되는 바와 같이, 서비스 계층에서 다양한 기능들을 구현하는 것은 또한 애플리케이션들을 기반 네트워크들과의 특정 상호작용들로부터 벗어나게 해줄 수 있다. 이것은, 예를 들어, M2M 디바이스와 같은, 자원 제약된 디바이스들에 특히 유용할 수 있다.

[0024] 본원에 기술되는 실시예들은, 이상의 예에 의해 예시된 중단간 통신 문제점과 같은, 다양한 통신 문제들을 해결한다. 이하에서 기술되는 바와 같이, 예시적인 실시예에서, 중단간 통신에 관련된 다양한 기반 네트워크들의 상태들이 서로 간에 교환될 수 있다. 이 네트워크들은 상이한 운영자들에 속할 수 있다. 이와 유사하게, 중단간 통신에 관련된 애플리케이션들의 특성들이 교환되고 기반 네트워크들에 알려질 수 있다. 이러한 정보 교환은 애플리케이션들 간에, 네트워크들 간에, 그리고 지역들 간에 있을 수 있다. 서비스 계층(SL)이 애플리케이션

선들과 기반 네트워크들 사이에 존재하기 때문에, SL이 다양한 통신 목표들을 달성하기에 좋은 후보라는 것이 본원에서 인식된다. SL은 애플리케이션 및 네트워크 관련 정보를 다른 SL과 수평적으로 교환할 수 있다. 본원에 기술되는 바와 같이, SL은 또한 그와 연관된 정책들을 가질 수 있고, 집계된 정보를 지능적으로 처리할 수 있다.

[0025] 이하에서 추가로 기술되는 예시적인 실시예에 따르면, 아키텍처 프레임워크는 SL이 애플리케이션들, 기반 네트워크들, 및 다른 서비스들에 대한 귀중한 입력을 제공할 수 있게 한다. 따라서, SL은 기존의 문제들을 해결하는 최적의 해결책들을 제공할 수 있을 뿐만 아니라, 현재 이용가능하지 않은 새로운 통신 모델들을 사용할 수 있게도 할 수 있다. 본 개시내용은, SL에 대한 새로운 서비스로서, 네트워크 및 애플리케이션 관리 서비스(NAMS)를 기술한다. 예시적인 실시예에 따르면, NAMS는 애플리케이션들, 기반 네트워크들, 및 다른 서비스들로부터의 정보를 수집, 통합, 및 처리한다. NAMS는 애플리케이션들 및 기반 네트워크들을 최적화하기 위해 그 자신의 정책들에 기초하여 출력을 생성할 수 있다. 다수의 예들이 본원에서 M2M 시스템의 맥락에서 제시되고 있지만, NAMS가 M2M 시스템들 및 비-M2M 시스템들에서 구현될 수 있다는 것이 이해될 것이다.

[0026] 서론으로서, 예시적인 실시예에 따르면, SL 아키텍처는 네트워크 및 애플리케이션 관리, 애플리케이션 적응(Application Adaptation), 네트워크 적응(Network Adaptation), 및 서비스 적응(Service Adaptation)과 같은 다양한 서브기능들을 포함하는 NAMS를 포함한다. 다양한 참조점들(예컨대, Mca, Mcc, Mcc', Mcn, Mff)을 통해 송신되는 다양한 동작들 및 메시지들은 이하에서 기술된다. NAMS에 의해 수행되는 다양한 동작들은 이하에서 기술된다. 게다가, NAMS에 의해 집계 및 생성될 수 있는 구체적인 정보는 본원에서 정의된다.

[0027] 도 5 내지 도 16, 도 18 내지 도 22, 및 도 24(이후에 기술됨)는 네트워크들 및 애플리케이션들을 관리하는 방법들 및 장치들의 다양한 실시예들을 예시하고 있다. 이 도면들에서, 다양한 단계들 또는 동작들이 하나 이상의 클라이언트들, 서버들, 및/또는 프록시들에 의해 수행되는 것으로 도시되어 있다. 이 도면들에 예시된 클라이언트들, 서버들, 및 프록시들이 통신 네트워크에서의 논리적 엔티티들을 표현할 수 있고, 이하에서 기술되는 도 25a 또는 도 25b에 예시된 일반 아키텍처들 중 하나를 포함할 수 있는, 이러한 네트워크의 노드의 메모리에 저장되고 그 노드의 프로세서 상에서 실행되는 소프트웨어(즉, 컴퓨터 실행가능 명령어들)의 형태로 구현될 수 있다는 것을 잘 알 것이다. 즉, 도 5 내지 도 16, 도 18 내지 도 22, 및 도 24에 예시된 방법들은, 예를 들어, 도 25c 또는 도 25d에 예시된 노드 또는 컴퓨터 시스템과 같은, 네트워크 노드의 메모리에 저장된 소프트웨어(즉, 컴퓨터 실행가능 명령어들)의 형태로 구현될 수 있고, 이 컴퓨터 실행가능 명령어들은, 노드의 프로세서에 의해 실행될 때, 이 도면들에 예시된 단계들을 수행한다. 이 도면들에 예시된 임의의 전송 및 수신 단계들이 노드의 프로세서의 제어 하에서 노드의 통신 회로부(예컨대, 도 25c의 회로부(34) 또는 도 25d의 회로부(97))에 의해 그리고 프로세서가 실행하는 컴퓨터 실행가능 명령어들(예컨대, 소프트웨어)에 의해 수행될 수 있다는 것을 또한 잘 알 것이다.

[0028] 이제 도 5를 참조하면, 도 4에 예시된 문제가 어떻게 NAMS(100) 및 그에 관련된 메시지들을 사용해 해결되는지에 대한 예시적인 사용 사례가 예시적인 실시예에 따라 예시된다. 원들은 예시된 사용 사례에서 SL이 어디에 존재하는지를 표시한다. 동일하거나 유사한 특징들을 표시하기 위해 다양한 도면들에서 참조 번호들이 반복되는 것을 잘 알 것이다.

[0029] 도 5를 여전히 참조하면, 예시된 예에 따르면, 0에서, 에너지 제어 앱(App1) 및 홈 보안 앱(App2)이 사용자(Alice)의 스마트폰(506) 상의 SL에 등록된다. 스마트 서모스탯(smart thermostat)(502) 및 보안 카메라(504) 둘 다는 홈 게이트웨이(GW)(508)에 있는 SL에 등록된다. 1에서, 스마트폰(506)은 (일반적으로 인터넷(510) 또는 네트워크(510) - 이들로 제한되지 않음 - 라고 지칭될 수 있는, 클라우드(510)에 도시된) 인프라스트럭처 노드에서의 SL에 있는 위치 서비스를 사용한다. 예시된 예에 따르면, 전화기(506)는 그의 위치를 클라우드(510)에 있는 SL에 주기적으로 보고한다. 이 예에서, 전화기(506)는 Alice가 지하철에 들어갈 때 그의 업데이트를 보고한다. 2에서, 인프라스트럭처 노드에 있는 NAMS(100)는 네트워크 커버리지 정보를 기반 셀룰러 네트워크(underlying cellular network)(512)로부터 이전에 획득하였다. 어떤 경우에, NAMS(100)는 네트워크(512)의 지리적 커버리지(geographic coverage)를 수신한다. 인프라스트럭처 노드에 있는 NAMS(100)는, 예를 들어, 전화기(506)와 같은, 디바이스가 커버리지를 상실하고 있는지를 그의 위치 업데이트에 기초하여 평가하기 위해 NAMS(100)를 트리거하는 정책을 갖는다. 디바이스가 커버리지를 상실하고 있으면, NAMS(100)는 커버리지를 상실하고 있는 디바이스 상의 애플리케이션들을 목표로 하는 임의의 트랜잭션들을 중단시킬 수 있다. 예시된 예에 따르면, NAMS(100)는 전화기(506)가 그 네트워크 연결을 상실하고 있다고 결정한다.

[0030] 3에서, 예시된 예에 따르면, NAMS(100)는 App1 및 App2와 통신하는 애플리케이션들을 포함하는 다른 SL들에게

SL 교환 메시지를 송신한다. 예시적인 교환 메시지들은 이하의 표 1에 열거되어 있으며, 이하에서 추가로 기술된다. SL 교환 메시지들은 디바이스(전화기(506)) SL 상의 App1 및 App2에 대한 메시지들이 중단되어야만 한다는 것을 표시한다. NAMS(100)는, 애플리케이션 등록 및 데이터 교환 서비스들과 같은, SL에 있는 다른 서비스들로부터 이러한 정보를 획득할 수 있다. 4에서, SL 교환 메시지를 수신할 시에, 홈 GW(508)에 있는 SL은 그 자신의 정책에 기초하여 결정할 수 있다. 예를 들어, 홈 GW(508)에 있는 SL은 App1 및 App2에 대한 메시지들을 버퍼링할 수 있다. 대안적으로 또는 그에 부가하여, 홈 GW(508)에 있는 SL은 메시지들을 에너지 제어 디바이스(502) 및 홈 보안 디바이스(504)에게 송신하여 그들이 메시지들을 App1 및 App2에게 송신하는 것을 방지한다.

[0031] 따라서, 도 5의 예시적인 사용 사례에 의해 도시된 바와 같이, NAMS는 통신 문제들이 발생하는 것을 방지할 수 있고, 따라서 또한 네트워크 트래픽을 감소시킬 수 있으며, 사용자들의 애플리케이션 통신 경험을 개선시킬 수 있고, 기타를 할 수 있다. 게다가, NAMS(100)는 제약된 디바이스들 및 애플리케이션들을, 이상의 예시적인 사용 사례에서 기술된 문제들과 같은, 다양한 문제들을 해결하기 위해 진보된 특징들을 지원하는 부담으로부터 벗어나게 할 수 있다. 예를 들어, NAMS(100)는 제약된 디바이스들 및 애플리케이션들이 간단한 논리(logic) 및 경량 프로토콜(light protocol)들을 지원하기만 하면 될 수 있게 할 수 있다. NAMS(100)는 상이한 노드들로부터의 SL에서 이용가능한 정보를 연관시키고, 다수의 디바이스들, 애플리케이션들, 또는 기반 네트워크들에 대해, 통합된 결정들 및 행동들을 수행할 수 있다.

[0032] 이제 도 6을 참조하면, 예시적인 서비스 계층 아키텍처(600)가 도시되어 있다. 도 6은 네트워크 및 애플리케이션 관리 서비스(NAMS)(100)가 어떻게 SL 아키텍처, 예를 들어, SL 아키텍처(600)에 적합할 수 있는지의 상위 레벨 도면을 제공한다. SL 아키텍처는 종단 노드(602), 중간 노드(604), 및 인프라스트럭처 노드(606)를 포함한다. 노드들 각각은 서비스 계층(608)을 포함한다. 예를 들어, 도시된 바와 같이, 종단 노드(602)는 제1 SL(608a)을 포함하고, 중간 노드(604)는 제2 SL(608b)을 포함하며, 인프라스트럭처 노드(606)는 제3 SL(608c)을 포함한다. 따라서, 도시된 바와 같이, SL 아키텍처(600)는 예시된 노드들 각각에 서비스 계층 미들웨어(Service Layer middleware)를 포함하고, 서비스 계층들 각각은 네트워크 및 애플리케이션 관리 서비스 기능(100)을 포함한다. 각각의 노드는 기반 액세스 네트워크(underlying access network)에 연결된다. 예로서, 종단 노드(602)는 802.15.4 네트워크에 연결될 수 있다. 추가의 예로서, 도시된 바와 같이, 중간 노드(604)는 2개의 무선부들(예컨대, 일군의 종단 노드들에 연결할 수 있도록 802.15.4를 위한 것, 및 3GPP 셀룰러 네트워크가 백본 네트워크에 연결하기 위한 것)을 가질 수 있다. 인프라스트럭처 노드는 유선 네트워크를 통해 인터넷에 연결할 수 있다. 상이한 노드들에 있는 서비스 계층들(608)은 서로 통신할 수 있다. 노드들 각각은 하나 이상의 애플리케이션들(610)을 호스팅할 수 있다. 기반 네트워크들이 원하는 바에 따라 달라질 수 있다는 것이 이해될 것이다.

[0033] 이제 도 7을 참조하면, 예시적인 서비스 계층 기능 아키텍처(700)가 도시되어 있다. 도 7은 NAMS(100)가 어떻게 서비스 계층 기능 아키텍처, 예를 들어, SL 기능 아키텍처(700)에 적합할 수 있는지를 도시하고 있다. 본원에 기술되는 바와 같이, 다양한 실시예들에 따르면, NAMS(100)는 서비스 계층, 예를 들어, 도 7에 도시된 서비스 계층(608)에 존재하는 서비스이다. NAMS(100)는 애플리케이션들(610)로부터 정보를 수신 및 처리하고, 기반 네트워크들, 예를 들어, 도 7에 도시된 기반 네트워크들(702)에 대한 요구사항들을 생성하며, 결과들을 기반 네트워크들(702)에 전달할 수 있다. NAMS(700)는 또한 기반 네트워크들로부터 정보를 수신하고, 정보를 처리하며, 정보를 애플리케이션들에 전달할 수 있다. 참조점들에 대해 수행될 수 있는 논리 및 동작들에 관한 추가 상세는 이하에서 제시된다. 예시되는 참조점들이 oneM2M 참조점들이지만, oneM2M 참조점들이 예를 위해 제시된 것이고, NAMS(100)가 원하는 바에 따라 하나의 oneM2M에 부가하여 다른 참조점들을 사용하여 구현될 수 있다는 것이 이해될 것이다.

[0034] 예시된 실시예에 따르면, NAMS(100)를 용이하게 하는 적응 서비스들이 제안된다. 예시적인 적응 서비스들은 애플리케이션 적응 서비스(706), 네트워크 적응 서비스(708), 서비스 적응(710)을 포함한다. 애플리케이션 적응 서비스(706)는 애플리케이션들(610)에 대한 접촉 및 집계점(contact and aggregation point)으로서 역할할 수 있다. 네트워크 적응 서비스(708)는 기반 네트워크들(702)에 대한 접촉 및 집계점으로서 역할할 수 있다. 서비스 적응(710)은 2개 이상의 서비스 계층 엔티티들 간의 접촉 및 집계점으로서 역할할 수 있다.

[0035] 예시적인 실시예에 따르면, 적응 서비스들의 사용은 NAMS(100)에 대해 임의적이다. 적응 서비스들은 NAMS(100)와 다른 엔티티들 간의 통신의 효율을 증가시킬 수 있다. 대안적으로, 예를 들어, 서비스 계층(608)이 앞서 기술된 적응 서비스들을 포함하지 않으면, NAMS(100)는 다양한 애플리케이션들(610), 기반 네트워크들(702), 및 다른 서비스 계층들과 직접적으로 인터페이스한다. 예를 들어, 도 8에 도시된 예에서, 적응 서비스들은

NAMS(100)의 서브기능들이다. 도 9에 도시된 대안의 예에서는, 적응 서비스들이 NAMS(100)와 분리되어 있다.

- [0036] 도 8을 참조하면, 예시된 실시예에 따르면, NAMS(100)는 적응 서브기능들, 예를 들어, 애플리케이션 적응 서브기능(102), 네트워크 및 애플리케이션 관리 서브기능(104), 네트워크 적응 서브기능(106), 및 서비스 적응 서브기능(108)을 포함한다. 따라서, NAMS(100)는 (Mca를 통해) 애플리케이션들(610), (Mcn을 통해) 기반 네트워크 기능들, 그리고 (Mff를 통해) 동일한 서비스 계층 또는 (Mcc 또는 Mcc'을 통해) 상이한 서비스 계층들과 직접적으로 인터페이스한다.
- [0037] 도 9에 도시된 대안의 실시예를 참조하면, 적응 기능들이라고도 지칭될 수 있는 적응 서비스들은 NAMS(100)에 부가하여 다른 서비스들(예컨대, 다른 CSF들)을 제공할 수 있는 일반 기능들이다. 따라서, 적응 서비스들은 NAMS(100)의 외부에 있다. 도 9에 도시된 바와 같이, NAMS(100)는 Mff를 통해 적응 기능들과 인터페이스할 수 있다. 따라서, 적응 기능들은 NAMS(100)에 관련된 특징들을 포함할 뿐만 아니라, 적응 기능들은 다른 서비스들, 애플리케이션들, 및/또는 기반 네트워크들에 의해 사용될 수 있는 특징들(예컨대, 서비스 디스패칭 및 통합(service dispatching and consolidating))도 포함할 수 있다.
- [0038] NAMS(100)의 예시적인 서브기능들이 이제부터 기술된다. 앞서 기술된 바와 같이, 서브기능들이 또한 NAMS(100)와 분리되어 있는 서비스들로서 구현될 수 있다. 따라서, 이하의 설명이 편의상 NAMS(100)의 서브기능들을 언급할 수 있지만, 서브기능들이 독립형 서비스들로서 구현될 수 있다는 것이 이해될 것이다. 이와 유사하게, 이하의 설명이 편의상 NAMS(100)와 상호작용하는 서비스들을 언급할 수 있지만, 서비스들이 NAMS(100) 내의 서브기능들로서 구현될 수 있다는 것이 이해될 것이다.
- [0039] 예시적인 실시예에서, 네트워크 및 애플리케이션 관리 서브기능(104)은 NAMS(100)에 대한 처리 센터이다. 네트워크 및 애플리케이션 서브기능(104)은 (애플리케이션 적응 기능(102)을 통해) 애플리케이션들로부터 수집된 정보, (네트워크 적응 기능(106)을 통해) 기반 네트워크들로부터의 정보, 및 (서비스 적응 기능(108)을 통해) 다른 서비스들로부터의 정보를 가질 수 있다. 어떤 경우에, NAMS(100)의 주요 기능은 통신을 최적화하기 위해 애플리케이션들, 서비스들, 및 기반 네트워크들로부터의 정보를 분석하는 것이다.
- [0040] NAMS(100)는, 수집된 정보를 분석할 수 있도록, 그가 수집한 정보를 연관시킬 수 있다. 예를 들어, NAMS(100)는 애플리케이션을 애플리케이션이 실행 중인 기반 네트워크와 연관시킬 수 있다. 종종, 이러한 정보는 애플리케이션 등록 정보에 포함되어 있지 않다. 따라서, 예를 들어, 애플리케이션을 관여된 기반 네트워크들과 연관시키기 위해, NAMS(100)는 (예컨대, "등록" 서비스로부터) 애플리케이션이 등록한 서비스 계층을 결정할 수 있고, NAMS(100)는 이어서 결정된 서비스 계층에 대한 기반 네트워크를 결정할 수 있다. 도 8에 예시된 바와 같이, NAMS(100)는 동일한 서비스 계층을 갖는 다른 서비스들에 대한 인터페이스(Mff)를 갖는다. 상이한 NAMS 서비스(NAMS)들은 그들의 출력을 Mcc 참조점을 통해 서로 교환할 수 있다.
- [0041] NAMS(100)는 그의 정책들에 기초하여 결정 또는 추천을 하고 출력을 생성할 수 있다. 어떤 경우에, 정책들은 구성가능하고, 각각의 NAMS는 서로 상이한 다수의 정책들을 보유할 수 있다. 예를 들어, 비즈니스 관계에 따라, 정책들은 서비스 제공자들에 의해, 기반 네트워크들의 운영자들에 의해, 또는 애플리케이션들에 의해 제공될 수 있다. 상이한 정책들이 상이한 범위(scope)들에 적용될 수 있다. 예로서, 주어진 서비스 제공자의 정책은 주어진 서비스 제공자에 의한 서비스 계층에 등록된 모든 애플리케이션들에 적용될 수 있다. 다른 예로서, 주어진 애플리케이션의 정책은 주어진 애플리케이션과의 트랜잭션들을 갖는 다른 애플리케이션들에만 영향을 미칠 수 있다. 정책은 정책이 적용되는 노드들과, 특정 결정을 하는 기준들 및 논리를 정의할 수 있다. 예를 들어, 예시적인 상황 또는 시나리오가 주어지면, 상이한 정책들은 그 자신의 논리에 기초하여 상이한 해결책들을 산출할 수 있다. 정책 자원 구조의 예시적인 실시예가 이하에서 기술된다.
- [0042] NAMS(100)에서의 예시적인 정책들 및 관련 동작들은 이하에서 제시된다. 원하는 바에 따라 다른 정책들이 NAMS(100)에서 구현될 수 있다는 것이 이해될 것이다.
- [0043] 예를 들어, 애플리케이션 대역폭 조절 정책(Applications Bandwidth Regulation Policy)은 애플리케이션이 SL에 등록할 때 대역폭 요구사항을 SL에 제공할 수 있다. 어떤 경우에, SL이 존재하는 노드가 기반 네트워크에 연결할 때, 기반 네트워크는 그가 지원하는 대역폭을 SL에 제공한다. 예시적인 실시예에서, NAMS(100)는 대역폭 요구사항 및 지원되는 대역폭 둘 다에 액세스할 수 있다. 게다가, NAMS(100)는 실행 중인 애플리케이션들에 의해 요구되는 총 대역폭(aggregate bandwidth)이 기반 네트워크에 의해 지원되는 대역폭을 초과한다고 결정할 수 있고, 정책에 기초하여, NAMS(100)는 조치를 취할 수 있다. 예를 들어, NAMS(100)는 선택된 애플리케이션들에 속도(rate)를 감소시키라고 명령할 수 있거나, 기반 네트워크에 전송 속도(transmission rate)를 증가시키라

고 명령할 수 있다. 그렇게 함으로써, NAMS(100)는 애플리케이션(들)이 네트워크 대역폭 한계로 인해 차선의 성능으로 실행되는 상황들을 방지할 수 있다.

[0044] 총 네트워크 부하 표시(Aggregated Network Load Indication) 정책은 상이한 노드들 상의 SL에 있는 NAMS(100)에서 구현될 수 있다. NAMS는 네트워크들, 예를 들어, WiFi, ZigBee, 및 셀룰러 네트워크들의 상이한 세그먼트들로부터 기반 네트워크 부하 조건(underlying network load condition)을 획득할 수 있다. NAMS는 상이한 애플리케이션들에 대한 네트워크 부하 조건들과 연관된 상이한 임계값들을 가질 수 있다. 예로서, 80%의 부하가 스마트 미터링 애플리케이션(smart metering application)에 대해서는 중간이지만, 홈 보안 애플리케이션에 대해서는 높을 수 있다. NAMS(100)는 이러한 정보를 애플리케이션들에 전달할 수 있고, 따라서 애플리케이션들은 그의 동작들에 대한 보다 스마트한 결정을 할 수 있다. 예를 들어, NAMS(100)가 네트워크의 한 세그먼트에 높은 부하가 걸려 있다고, 예를 들어, 과부하가 걸려 있다고 결정한 후에, 애플리케이션은 트랜잭션들을 지연시킬 수 있다. 본원에서 사용되는 바와 같이, 네트워크와 연관된 부하 조건 또는 상태는 네트워크의 혼잡 상태라고도 지칭될 수 있다.

[0045] 일반적으로, 적응 기능들은 상이한 노드들 또는 엔티티들 간의 인터페이스들의 복잡성을 감소시킨다. 예를 들어, NAMS(100)가 그의 정책들에 기초하여 입력을 생성할 때, 입력은 다수의 애플리케이션들, 네트워크들, 또는 다른 서비스들에 적용될 수 있다. 따라서, 적응 기능들은 입력을 각자의 노드들에게 디스패치(dispatch)할 수 있다. 어떤 경우에, NAMS(100)에의 다수의 입력들이 있을 때, 적응 기능들은 입력을 통합하고 나서 입력을 NAMS(100)에게 송신할 수 있다. 대안적으로, 입력이, 구성에 기초하여, NAMS(100)에게 곧바로 전달될 수 있다. 예를 들어, 인프라스트럭처 노드에 있는 NAMS는 중간 노드들에 있는 NAMS들로부터 입력을 수신할 수 있다. 인프라스트럭처 노드에 있는 NAMS는 입력이 일정 기간에 걸쳐 통합되어 있기를 원한다는 것을 그의 구성 메시지에 표시할 수 있다.

[0046] 도 7 및 도 8에 도시된 바와 같이, 서비스 적응 기능(108) 및 서비스 적응 서비스(710)는 NAMS(100)와 서비스 계층에 있는 다른 서비스들 사이에서 상호작용한다. 다른 서비스들은 동일한 노드에 있는 동일한 서비스 계층 엔티티 내에 또는 상이한 노드에 있는 상이한 서비스 계층들에 존재할 수 있다. NAMS(100)는 하나 또는 다수의 서비스들에 대한 입력을 송신할 수 있고, 서비스 적응 기능(108) 또는 서비스 적응 서비스(710)는 NAMS(100)로부터의 입력을 상이한 서비스들에게 디스패치할 수 있다.

[0047] 애플리케이션 적응 서비스(706)는 서비스 적응(710)과 유사한 역할을 맡을 수 있다. 애플리케이션 적응 서비스(706)는 애플리케이션들 및 서비스 계층에 있는 NAMS와 상호작용할 수 있다. 하나의 노드에 있는 NAMS는 다수의 애플리케이션들과 상호작용할 수 있다. NAMS(100)는 다수의 애플리케이션들에 적용되는 입력을 생성할 수 있고, 애플리케이션 적응 서비스(706)는 NAMS(100)로부터의 입력을 상이한 애플리케이션들에게 디스패치할 수 있다. 어떤 경우에, 상이한 애플리케이션들로부터의 다수의 보고들이 NAMS(100)에게 있을 때, 애플리케이션 적응 서비스(706)는 보고들을 통합하고 나서 통합된 보고들을 NAMS(100)에게 송신할 수 있다. 네트워크 적응 서비스(708)는 기반 네트워크들 및 서비스 계층에 있는 NAMS(100)와 상호작용한다. 네트워크 적응 서비스(708)는 하나 또는 다수의 기반 네트워크들로부터 NAMS(100)에의 입력을 통합하고, NAMS(100)로부터의 입력을 상이한 기반 네트워크들에게 디스패치할 수 있다.

[0048] 또한 도 10 및 도 11을 참조하면, 앞서 언급된 동작들이 추가로 예시되어 있다. 명확함 및 편의를 위해, 도 10은 애플리케이션들(610)로부터 네트워크들(702)로(제1 방향)의 동작들을 도시하고 있으며, 도 11은 네트워크들(702)로부터 애플리케이션들(610)로(제2 방향)의 동작들을 도시하고 있지만, 제1 방향 및 제2 방향에서의 동작들이 동시에 일어날 수 있고, 서로 병렬로 동작할 수 있다는 것이 이해될 것이다. 도시된 바와 같이, NAMS(100)는 제1 방향 및 제2 방향 둘 다에서 출력을 수신 및 송신한다. 기반 네트워크 정보는 다양한 네트워크들 상에 존재하는 상이한 서비스 계층들에 의해 교환될 수 있다. 도시된 바와 같이, 이러한 정보는 네트워크 관리 및 서비스 적응 기능들 사이에서 서비스 계층간 참조점(inter Service Layer referent point)들을 통해 교환된다. 게다가, 서비스 적응 기능의 동작들이 "입력"과 "출력"으로 나누어지는데, 그 이유는 기능이 도 10 및 도 11에서 양방향으로 동작하기 때문이다. "입력"은 서비스 적응으로부터 네트워크 관리로의 전송을 지칭하고, "출력"은 서비스 적응으로부터 다른 SL들의 전송을 지칭한다.

[0049] 이하의 표 1은 앞서 기술된 아키텍처 및 NAMS(100) 기능들에 기초하여 상이한 참조점들을 통해 교환될 수 있는 정보 요소들을 열거하고 있다. 메시지 이름들이 편의상 제시되어 있고, 따라서 메시지들이 원하는 바에 따라 임의의 다른 이름들에 의해 식별될 수 있다는 것에 유의해야 한다. 이하에서 정의되는 메시지들은 임의의 특정 프로토콜들의 임의의 신택스(syntax)로 한정(specific)되지 않고, 따라서 메시지가 다양한 프로토콜들에 의해

실현될 수 있다. 예를 들어, 메시지들이 RESTful 메서드(RESTful method)(예컨대, HTTP 또는 CoAP)를 사용하여 요청/응답(Request/Response)에 의해, MQTT를 사용하여 구독/발행(Subscribe/Publish)에 의해, Diameter를 사용하여 요청/응답(Request/Answer)에 의해, 기타에 의해 실현될 수 있다.

[0050] 메시지들이 이벤트들에 의해 트리거될 수 있고 그리고/또는 메시지가 주기적으로 트리거될 수 있다. 트리거링 이벤트들의 예들이 표 1에 제시되어 있지만, 원하는 바에 따라 다른 이벤트들이 메시지들을 트리거할 수 있다는 것이 이해될 것이다.

표 1

참조점들을 통한 예시적인 NAMS 메시지들 및 정보

메시지 이름	참조점	방향	정보 요소들(IE)	메시지의 트리거링	수신자의 동작들
NW Status Update	Mcn	네트워크로부터 NAMS로	<p>네트워크가 SL에 공개하기로 합의하는 정보를 포함한다. NW 상태 업데이트는, 제한 없이, 하기를 포함할 수 있다:</p> <ul style="list-style-type: none"> <li>- 네트워크 이름, 유형 및 ID</li> <li>- 네트워크 데이터 속도</li> <li>- 네트워크에 의해 제공되는 QoS</li> <li>- 네트워크 부하 상태(혼잡 상태)</li> <li>- 네트워크 패킷 크기</li> <li>- 네트워크 전송 간격</li> <li>- 시간 단위당 최대 데이터</li> <li>- 네트워크의 M2M 디바이스들에 대한 지원</li> <li>- 이하에서 추가로 논의되는 바와 같은, 지원되는 기반 네트워크 서비스들</li> <li>- 네트워크 커버리지(예컨대, 지리적 커버리지 구역)</li> </ul>	<p>정적 네트워크 정보가 SL과 네트워크 간의 초기 통신 중에 SL에게 전달될 수 있고; 네트워크가 이러한 정보가 SL에게 전달되어야만 한다고 결정할 때 동적 네트워크 정보가 트리거될 수 있으며; 또는 그것이 이하에서 기술되는 "NW Status Query" 메시지를 사용하여 SL에 의해 풀링(pull)될 수 있다.</p>	<p>일 예에서, NW Status Update에서 정보를 수신할 시에, NAMS는 각각의 기반 네트워크에 대한 정보를 업데이트 및 저장하고; NAMS는 업데이트들이 임의의 애플리케이션 및/또는 서비스들에 영향을 미치는지를 체크하고, 그렇다면, NAMS는 영향을 받은 애플리케이션들에게로의 "SL to App Input" 메시지 및/또는 다른 SL들에게로의 "Service Layer Exchange" 메시지를 게시할 수 있다.</p>

[0051]

			- "SL to NW Input" 메시지에 기초한 네트워크의 행동들		
NW Status Query	McN	NAMS로부터 네트워크로	NAMS는, 예를 들어, 위의 행에서 언급된 정보의 전부 또는 서브셋(subset)에 대해, 기반 네트워크에게 질의할 수 있다.	SL은 초기 통신시에 기반 네트워크에게 질의를 개시할 수 있고; 후속 질의들은, 서비스들의 추가 및 삭제와 같은, SL의 변화에 의해; 또는 기반 네트워크에 영향을 줄 수 있는 애플리케이션들의 변화에 기초하여 트리거될 수 있다. NAMS는 질의를 송신하기로 결정한다.	일 예에서, 기반 네트워크는 요청된 정보를 "NW Status Update" 메시지에 채운다.
SL to NW Input	McN	NAMS로부터 네트워크로	서비스 계층이 기반 네트워크에게 전달하고자 하는 정보를 포함한다. SL to NW input은, 제한 없이, 하기를 포함할 수 있다:  - 애플리케이션들로부터의 개별적 또는 결합된 대역폭 요구사항  - 애플리케이션들에 의해 요구된 QoS  - 지속시간(duration), 연속적/버스티(continuous/bursty) 등과 같은, 애플리케이션들로부터의 트랜잭션 특성들	NAMS는 통합된 정보를 처리하고 언제 입력을 네트워크에게 송신할지를 결정한다. 트리거들은, 기반 네트워크에 영향을 미칠 수 있는 요청들과 같은, 서비스들 및 애플리케이션들에 대한 변경들일 수 있다.	일 예에서, 기반 네트워크는 서비스 계층으로부터 입력을 받고, 그의 연결들을 최적화하기 위해 조치들을 취할 수 있다. 서비스 계층과 기반 네트워크 간의 비즈니스 협약/관계에 따라, 기반 네트워크는 서비스 계층으로부터의 입력에 기초하여 자율적인 결정을 할 수 있거나, 서비스 계층이 요청한 대로 동작할 수 있다. 기반 네트워크는 취해진 조치들을 "NW Status Update"를 사용하여 SL에게 송신할 수 있다.

[0052]

			<p>- 네트워크 관리 명령들</p> <p>- 예컨대, 다른 기반 네트워크의 상태와 같은, Mcn을 통해 전달될 다른 서비스들로부터의 정보.</p>		
NW to SL Query	Mcn	네트워크로부터 NAMS로	<p>기반 네트워크는, 예를 들어, 위의 행에서 언급된 정보의 전부 또는 서브셋에 대해, 서비스 계층에게 질의할 수 있다.</p>	<p>질의는, 네트워크 구성들을 최적화하기 위해 상위 계층 정보를 획득할 필요성과 같은, 네트워크 자체의 결정에 기초하여 트리거될 수 있다.</p>	<p>일 예에서, NAMS는 요청된 정보를 채우고 답신(reply)을 "SL to NW Input"에서 송신한다. 질의는 관련 정보를 획득하기 위해 애플리케이션들 및 다른 서비스들 또는 SL들을 폴링하라고 NAMS를 트리거할 수 있다.</p>
SL to NW Config	Mcn	NAMS로부터 네트워크로	<p>일 예에서, 서비스 계층에 있는 NAMS는 SL이 정보를 요청했다는 것을 알려주기 위해 구성 정보를 기반 네트워크에게 송신한다. 실제 정보 요소들은 "NW Status Update" 메시지에 열거된 IE들의 서브셋일 수 있다.</p> <p>그에 부가하여, 이 메시지는 또한, 주기적 보고 (periodical reporting), 이벤트 트리거 방식 보고 (event triggered reporting), 질의 기반 보고(query based reporting) 등과 같은, 보고 구성을 포함할 수 있다.</p>	<p>이 메시지의 트리거는 SL의 인스턴스화 및 SL과 기반 네트워크 사이의 초기 설정일 수 있거나, SL과 기반 네트워크들 사이의 임의의 계약 또는 협약에 기초할 수 있다. 일 예에서, 임의의 후속 변화들에 대해, NAMS가 재구성을 개시하기 위해 메시지가 사용될 수 있다.</p>	<p>일 예에서, 네트워크가 구성을 수락할 수 있거나, 네트워크가 "NW to SL Config" 메시지를 사용하여 구성을 협상할 수 있다.</p>

[0053]

NW to SL Config	Mcn	네트워크로부터 NAMS로	이 메시지는 네트워크(들)가 어떤 네트워크 특정 정보(network specific information)를 서비스 계층에게 제공할 수 있는지를 SL에게 알려주기 위해 사용될 수 있다. NAMS와 네트워크들이 어떤 정보가 교환될 수 있는지 및 SL과 기반 네트워크 사이에서 얼마나 자주 정보가 교환될 수 있는지를 협상하기 위해, 이 메시지와 "SL to NW Config" 메시지가 함께 사용될 수 있다.	일 예에서, 이 메시지의 트리거는 SL과 기반 네트워크 사이의 초기 통신일 수 있거나, SL과 기반 네트워크들 사이의 계약 또는 협약에 기초할 수 있다. 일의 후속 변화들에 대해, NW가 재구성을 개시하기 위해 메시지가 사용될 수 있다.	일 예에서, SL은 구성을 수락하거나, "SL to NW Config" 메시지를 사용하여 구성을 협상할 수 있다.
App Status Update	Mca	애플리케이션들로부터 NAMS로	일 예에서, 애플리케이션들은 정보를 서비스 계층에게 푸시(push)한다. 푸시된 정보는, 제한 없이, 하기를 포함할 수 있다: <ul style="list-style-type: none"> <li>- 애플리케이션 유형, ID, 이름</li> <li>- 상태(활성(Active), 휴면(Dormant), ...)</li> </ul> 애플리케이션 전송 속도 <ul style="list-style-type: none"> <li>- 데이터의 양</li> <li>- 전송 간격</li> </ul> - "SL to App Input" 메시지에 기초한 애플리케이션의 행동들.	일 예에서, 애플리케이션들은 언제 메시지를 개시할지 결정한다. 이는 애플리케이션 트랜잭션들의 변화에 기초할 수 있다.	일 예에서, NAMS는 그가 수신하는 정보를 분석한다. NAMS는 기반 네트워크들의 입력을 생성할 수 있다. NAMS는 정보를 통합 및 추가 프로세스를 위해 다른 수집된 정보와 함께 저장할 수 있다.

[0054]

App Status Query	Mca	NAMS로부터 애플리케이션들로	일 예에서, NAMS는 위의 행에 언급된 정보의 전부 또는 서브셋에 대해 애플리케이션들에게 질의할 수 있다.	일 예에서, NAMS는 질의를 트리거해야 할 때, 예를 들어, 긴급한 네트워크 연결성 문제들이 있고 NAMS가 애플리케이션들로부터의 보다 최신의 상태를 필요로 할 때를 결정한다.	질의를 받은 애플리케이션(들)은 요구된 정보를 "App Status Update" 메시지에 채울 수 있다.
SL to App Input	Mca	NAMS로부터 애플리케이션들로	<p>일 예에서, 이 메시지는 서비스 계층이 애플리케이션들에게 전달하고자 하는 정보를 포함한다. 메시지는, 예를 들어 그리고 제한없이, 하기를 포함할 수 있다:</p> <ul style="list-style-type: none"> <li>- 전송 속도를 증가/감소</li> <li>- 지속시간을 증가/감소</li> <li>- 총 데이터 양을 증가/감소</li> <li>- 해상도를 증가/감소</li> <li>- X 시간 후에 또는 특정 시간 후에 전송</li> <li>- 애플리케이션 관리 명령들</li> <li>- 스위치 네트워크 (switch network); 후보 네트워크들을 나타낼 수 있다.</li> </ul>	<p>예를 들어, NAMS에 의해 생성된 입력(NAMS generated input)이 이러한 애플리케이션들에 영향을 미칠 수 있을 때, 예컨대, 입력이 애플리케이션들이 어떻게 데이터를 전송할 수 있는지에 영향을 미치는 경우 등에, SL은 개개의 애플리케이션 또는 일군의 애플리케이션들에게 입력을 푸시할 수 있다.</p>	<p>일 예에서, 애플리케이션은 SL에 있는 NAMS로부터의 입력에 기초하여 조치들을 취할 수 있다. 애플리케이션들 및 서비스 계층의 소유권 및 비즈니스 관계에 따라, 애플리케이션들은 자율적인 결정을 할 수 있거나 서비스 계층에 의해 요청된 대로 동작할 수 있다.</p>

[0055]

App to SL Query	Mca	애플리케이션들로부터 NAMS로	일 예에서, 애플리케이션들은 위의 행에 언급된 정보의 전부 또는 서브셋에 대해 서비스 계층에게 질의할 수 있다.	애플리케이션들은, 예를 들어, 통신의 품질이 변했거나 애플리케이션이 그의 통신 패킷(예컨대, 데이터의 양, 속도의 요청 (request of rate) 등)을 변경할 예정일 때, 독단으로 질의하기로 결정할 수 있다.	일 예에서, 서비스 계층(NAMS)은 요청된 정보를 "SL to App Input"에 채워서 다시 송신한다.
SL to App Config	Mca	NAMS로부터 애플리케이션들로	NAMS는 NAMS가 애플리케이션들로부터 어떤 정보를 획득하고자 하는지에 대해 애플리케이션들을 구성할 수 있다. 이 메시지는 "App Status Update"로부터의 IE들의 서브셋을 포함할 수 있다.	일 예에서, 이 메시지는 NAMS 개시 또는 보고 정책들의 후속 변화들에 의해 트리거된다.	애플리케이션은 구성을 수락하고 SL이 무엇을 요청했는지를 보고할 수 있거나, 애플리케이션들이 "App to SL Config" 메시지를 사용하여 협상할 수 있다.
App to SL Config	Mca	애플리케이션들로부터 NAMS로	애플리케이션들은 애플리케이션들이 NAMS로부터 어떤 정보를 획득하고자 하는지에 대해 NAMS를 구성할 수 있다. 이 메시지는 "SL to App Input"으로부터의 IE들의 서브셋을 포함할 수 있다.	일 예에서, 이 메시지는 애플리케이션이 SL에 처음으로 등록할 때 또는 후속 변화들로 인해 트리거된다.	일 예에서, NAMS는 구성을 수락하거나 "SL to App Config" 메시지를 사용하여 협상할 수 있다.
SL Exchange	Mcc, Mcc', Mff	2개의 서비스들 또는 2개의 서비스 계층들 사이에서	My는 2개의 서비스들 또는 서비스 계층들이 교환할 수 있는 네트워크 관리 관련 정보를 포함한다. 메시지는, 제한 없이, 하기를 포함할 수 있다:  - 서비스 계층이 존재하는 네트워크 ID, 유형, 및 이	일 예에서, SL은 기반 네트워크들, 애플리케이션들, 또는 서비스들에 관련된 정보의 변화에 기초하여 정보를 다른 SL에게 푸시할 때를 결정한다.	수신측 서비스 계층에 있는 NAMS는 정보를 처리하고, 정보를 애플리케이션들 또는 그의 기반 네트워크들에게 전달하기로 결정할 수 있다. 일 예에서, 교환된 정보가 NAMS의 정책이면, 수신측 서비스 계층은 그 자신의 NAMS에서 정책을 구현할 수 있다. 교환된

[0056]

			<p>름</p> <ul style="list-style-type: none"> <li>- "Network Status Update"에 대한 행에서 언급된 바와 같은 그 자신의 기반 네트워크의 상태</li> <li>- 그 자신의 애플리케이션 정보</li> <li>- 정책들 또는 NAMS로부터의 출력</li> </ul>		<p>정보가 NAMS의 출력이면, 수신측 서비스 계층은 그 자신의 결정에 기초하여 조치들을 취할 수 있다. 예를 들어, 수신측 서비스 계층은 대역폭 요구사항들을 그의 기반 네트워크에게 전달할 수 있다</p>
SL Query	Mcc, Mcc', Mff	2개의 서비스들 또는 2개의 서비스 계층들 사이에서	<p>일 예에서, 이 메시지는 동일한 서비스 계층의 다른 서비스들에게 또는 상이한 서비스 계층들에 있는 다른 서비스들에게 질의하기 위해 하나의 서비스 계층에 있는 NAMS에 대해 사용된다.</p>	<p>일 예에서, 이 메시지는 NAMS가 동일한 서비스 계층의 다른 서비스들로부터 또는 상이한 서비스 계층의 다른 NAMS로부터 추가 정보를 얻어야 할 필요가 있을 때 트리거된다.</p>	<p>수신자는 요청된 정보를 제공하고 이를 "SL Exchange" 메시지에서 다시 송신할 수 있다.</p>
SL to SL Config	Mcc, Mcc', Mff	2개의 서비스들 또는 2개의 서비스 계층들 사이에서	<p>이 메시지는, (Mff를 통해) 2개의 서비스들 사이에서 또는 (Mcc 또는 Mcc를 통해) 2개의 서비스 계층들 사이에서, 어떤 정보가 수집되고 그들 사이에서 교환될 수 있는지를 구성 및 협상하기 위해 사용될 수 있다. 정보는 "Service Layer Exchange" 메시지의 서브셋일 수 있다. 그에 부가하여, 메시지는 보고 구성을 포함할 수 있다.</p>	<p>일 예에서, 메시지는, 개시 또는 보고 정책들의 후속 변화 시에, 임의의 서비스 또는 서비스 계층에 의해 개시될 수 있다.</p>	<p>수신자는 구성을 수락할 수 있다. 또는, 동일한 유형의 메시지를 사용하여, 수신자가 구성을 협상할 수 있다.</p>

[0057]

[0058]

표 1에 제시된 메시지들은 3개의 동작 국면(phase of operation)들로 분류될 수 있다. 일부 메시지들은 "구성" 국면("configuration" phase)으로 분류될 수 있다. 예시적인 구성 국면 동안, NAMS는 네트워크들, 서비스들, 및 애플리케이션들이 NAMS가 어떤 정보를 수집하고자 하는지를 통보받도록 기반 네트워크들, 다른 서비스들, 및 애플리케이션들을 구성할 수 있다. 예시적인 구성 국면이 도 12에 도시되어 있다. 게다가, 기반 네트워크들, 서비스들, 및 애플리케이션들은 그들 각각이 NAMS로부터 어떤 입력을 수신하고자 하는지를 NAMS에게 표시할 수 있다. 도 12를 참조하면, 예시적인 구성 메시지들이 명확함을 위해 A, B, 및 C로 그룹화된다. 예시된 실시예에 따르면, 'A' 그룹 내의 단계(step)들은 NAMS와 기반 네트워크들 사이의 구성을 위한 것이고, 'B' 그룹 내의 단계들은 NAMS와 다른 서비스들 사이의 구성을 위한 것이며, 'C' 그룹 내의 단계들은 NAMS와 애플리케이션들 사이의 구성을 위한 것이다. 반복을 줄이기 위해 그리고 명확함을 위해, 단계들(A7, B7, 및 C7)이 절차들로서 나타내어져 있다.

[0059]

예시적인 실시예에서, 정보 보고 및 수집 국면(information reporting and collecting phase)이 구성 국면에 뒤따른다. 도 13을 참조하면, 예시된 실시예에 따르면, 구성 후에, NAMS(100)는 원하는 바에 따라 기반 네트워크들, 다른 서비스들, 및 애플리케이션들로부터 정보를 수집할 수 있다. 도시된 바와 같이, 메시지들은 네트워크들, 서비스들, 및 애플리케이션들에 대해, 각각, A 그룹, B 그룹, 및 C 그룹으로 그룹화된다. 도 13이, 보고를 받기 위해, NAMS(100)로부터 송신되는 질의 메시지를 나타내고 있지만, 네트워크들(702), 서비스들, 또는 애플리케이션들(610)이, 그에 부가하여 또는 대안적으로, 예를 들어, 구성에 기초하여, 그들의 보고들을 NAMS(100)에게 푸시할 수 있다는 것이 이해될 것이다.

[0060]

예를 들어, 표 1 및 도 12 내지 도 14의 요약을 통해, NAMS는 장치 또는 노드 상에 구현될 수 있으며, 본원에

기술되는 다양한 기법들이 하드웨어, 펌웨어, 소프트웨어, 또는, 적절한 경우, 이들의 조합들과 관련하여 구현될 수 있다. 이러한 하드웨어, 펌웨어, 및 소프트웨어는 통신 네트워크의 다양한 노드들에 위치한 장치들에 존재할 수 있다. 장치들은 본원에 기술되는 방법들을 수행하기 위해 단독으로 또는 서로 결합하여 동작할 수 있다. 본원에서 사용되는 바와 같이, "장치", "네트워크 장치", "노드", "디바이스", 및 "네트워크 노드"라는 용어들은, 달리 명시되지 않는 한, 서로 바꾸어 사용될 수 있다.

[0061] 따라서, NAMS는 프로세서, 메모리, 및 통신 회로부를 포함하는 장치 또는 노드 상에 구현될 수 있다. 장치는 그의 통신 회로부를 통해 통신 네트워크에 연결될 수 있고, 장치는, 장치의 프로세서에 의해 실행될 때, 장치로 하여금 조치를 취하게 하는 장치의 메모리에 저장된 컴퓨터 실행가능 명령어들을 추가로 포함할 수 있다. 예를 들어, 장치는, 통신 네트워크를 통해 네트워크 상의 제1 노드로부터, 네트워크와 연관된 적어도 하나의 상태 업데이트를 표시하는 제1 메시지(예를 들어, "NW Status Update" 메시지)를 수신할 수 있다. 장치는 네트워크와 연관된 적어도 하나의 상태 업데이트를 저장할 수 있고, 적어도 하나의 업데이트가 애플리케이션 또는 서비스에 영향을 미치는지를 결정할 수 있다. 적어도 하나의 업데이트가 애플리케이션 또는 서비스에 영향을 미치면, 장치는 애플리케이션 또는 서비스가 적어도 하나의 상태 업데이트에 기초하여 조치를 취할 수 있도록 적어도 하나의 상태 업데이트를 표시하는 제2 메시지(예컨대, "SL to App Input")를, 애플리케이션 또는 서비스를 호스팅하는 제2 노드에게 송신할 수 있다. 표 1에 나타난 바와 같이, 제2 메시지는, 예를 들어 그리고 제한 없이, 애플리케이션 또는 서비스의 전송 속도의 변화, 애플리케이션 또는 서비스의 전송 지속시간의 변화, 애플리케이션 또는 서비스와 연관된 데이터 품질의 변화, 애플리케이션 또는 서비스와 연관된 해상도의 변화, 미리 결정된 양의 시간 후에 전송하라는 명령어, 네트워크들을 전환하라는 명령어, 또는 애플리케이션 또는 서비스가 전환할 수 있는 제2 네트워크 중 적어도 하나를 추가로 표시할 수 있다. 표 1에 추가로 나타난 바와 같이, 적어도 하나의 상태 업데이트는, 예를 들어 그리고 제한 없이, 네트워크와 연관된 데이터 속도(data rate), 네트워크에 의해 제공되는 서비스 품질, 네트워크와 연관된 혼잡 상태, 네트워크와 연관된 패킷 크기, 네트워크 전송 간격, 시간 단위당 최대 데이터, 네트워크에 의해 지원되는 하나 이상의 디바이스들, 기반 네트워크에 의해 지원되는 하나 이상의 서비스들, 또는 네트워크의 지리적 커버리지 중 적어도 하나를 표시할 수 있다.

[0062] 예시적인 실시예에서, 장치(예컨대, NAMS)는 애플리케이션 또는 서비스로부터 제1 질의(예컨대, "App to SL Query" 메시지)를 수신할 수 있고, 제1 메시지는 제1 질의에 기초할 수 있다. 장치는, 애플리케이션 또는 서비스로부터 제1 질의 메시지를 수신한 것에 응답하여, 제2 질의 메시지(예컨대, "NW Status Query" 메시지)를 제1 노드에게 송신할 수 있고, 여기서 제2 질의 메시지는 네트워크와 연관된 정보에 대한 요청을 포함할 수 있다. 대안적으로 또는 그에 부가하여, 장치는, 제2 질의 메시지에 응답하여 제1 메시지가 수신되도록, 제2 질의 메시지를 제1 노드에게 송신할 수 있다. 따라서, 장치(예컨대, NAMS)는 네트워크 관련 정보에 대해 원하는 바에 따라 언제든지 네트워크에게 질의할 수 있다. 예를 들어, NAMS는 네트워크와 연관된 정보에 대한 요청을 포함하는 질의를 송신할 수 있고, 질의는 하나 이상의 서비스들이 서비스 계층에 추가되거나 그로부터 삭제된 것에 응답하여 송신될 수 있다. 다른 예에서, 질의가 네트워크에 연결된 하나 이상의 애플리케이션들의 변화에 기초하여 송신될 수 있다.

[0063] 표 1 및 도 12 내지 도 14의 추가 요약을 통해, NAMS는 장치 상에 구현될 수 있고, 장치는, 통신 네트워크를 통해 네트워크 상의 애플리케이션으로부터, 애플리케이션과 연관된 적어도 하나의 업데이트를 표시하는 제1 메시지(예컨대, "App Status Update" 메시지)를 수신할 수 있다. 장치는 애플리케이션과 연관된 적어도 하나의 업데이트를 저장할 수 있다. 장치는, 제1 노드가 적어도 하나의 업데이트에 기초하여 조치를 취할 수 있도록, 적어도 하나의 업데이트를 표시하는 제2 메시지(예컨대, "SL to NW Input" 메시지)를 네트워크 상의 제1 노드에게 송신할 수 있다. 게다가, 장치는, 예를 들어 그리고 제한 없이, 애플리케이션의 대역폭 요구사항, 애플리케이션의 서비스 품질 요구사항, 또는 애플리케이션의 트랜잭션 특성 중 적어도 하나를 표시하는 제2 메시지(예컨대, "SL to NW Input" 메시지)를 네트워크 상의 제1 노드에게 송신할 수 있다. 적어도 하나의 업데이트는, 예를 들어 그리고 제한 없이, 애플리케이션과 연관된 상태, 애플리케이션과 연관된 전송 속도, 애플리케이션과 연관된 데이터의 양, 애플리케이션과 연관된 전송 간격, 애플리케이션에 의해 취해진 조치 중 적어도 하나를 표시할 수 있다. 장치는 제1 노드로부터 제1 질의 메시지(예컨대, "NW to SL Query" 메시지)를 수신할 수 있고, 여기서 제1 메시지는 제1 질의 메시지에 기초한다. 예에서, 장치는, 제1 노드로부터 제1 질의 메시지를 수신한 것에 응답하여, 제2 질의 메시지(예컨대, "App Status Query" 메시지)를 애플리케이션에게 송신한다. 제2 질의 메시지는 애플리케이션과 연관된 정보에 대한 요청을 포함할 수 있다. 다른 예에서, 장치는, 제2 질의 메시지에 응답하여 제1 메시지가 수신되도록, 제2 질의 메시지(예컨대, "App Status Query" 메시지)를 애플리케이션에게 송신한다. 또 다른 예에서, 장치는 서비스 계층을 포함하고, 제1 질의 메시지는 네트워크와 연관

된 연결성 문제에 기초하여 송신된다.

[0064] 이제 도 14 및 표 1을 참조하면, NAMS(100)가 정보를 처리하고 입력을 생성하는 제3 국면이 나타내어져 있다. 도 14는 NAMS(100)가 그의 입력으로 응답하기 전에 질의 메시지가 NAMS(100)에게 송신될 수 있다는 것을 나타내지만, NAMS(100)가 질의 메시지 없이 그의 입력을 무시할 수 있다는 것이 이해될 것이다. 예시된 예에 따르면, 1에서, NAMS(100)는 수집된 정보를 분석하고(예컨대, 도 13을 참조), 정책들에 기초할 수 있는 결과들을 생성한다. A1에서, 제1 네트워크의, NSE(network service entity)라고 지칭될 수 있는, 제1 네트워크 노드는 NW to SL Query 메시지를 네트워크 적응 서비스(708)에게 송신한다. A2에서, 다른 네트워크의, NSE라고도 지칭될 수 있는, 다른 네트워크 노드는 NW to SL Query 메시지를 네트워크 적응 서비스(708)에게 송신한다. A3에서, 예시된 예에 따르면, 네트워크 적응 서비스(708)는 질의 메시지들을 통합한다. A4에서, 네트워크 적응 서비스(708)는 통합된 질의(NW to SL Query 메시지)를 NAMS(100)에게 송신한다. A5에서, NAMS(100)는 SL to NW input 메시지를 네트워크 적응 서비스(708)에게 송신한다. A6에서, NAMS(100)는 입력들을 디스패치한다. 예를 들어, A7에서, NAMS(100)는 SL to NW Input 메시지를 제1 네트워크에게 송신하고, A8에서, NAMS(100)는 SL to NW Input 메시지를 다른 네트워크에게 송신한다.

[0065] 이와 유사하게, 도 14 및 표 1을 계속 참조하면, 예시된 예에 따르면, B1에서, 제1 서비스를 포함하는 제1 노드는 SL Query 메시지를 서비스 적응 서비스(710)에게 송신한다. B2에서, 다른 서비스를 포함하는 다른 노드는 SL Query 메시지를 서비스 적응 서비스(710)에게 송신한다. B3에서, 예시된 예에 따르면, 서비스 적응 서비스(710)는 SL query 메시지들을 통합한다. B4에서, 서비스 적응 서비스(710)는 통합된 질의(SL Query 메시지)를 NAMS(100)에게 송신한다. B5에서, NAMS(100)는 SL to SL Exchange 메시지를 서비스 적응 서비스(710)에게 송신한다. B6에서, NAMS(100)는 입력들을 디스패치한다. 예를 들어, B7에서, NAMS(100)는 SL to SL Exchange 메시지를 제1 서비스에게 송신하고, B8에서, NAMS(100)는 SL to SL Exchange 메시지를 다른 서비스에게 송신한다.

[0066] 표 1 및 도 14의 예시된 예를 여전히 참조하면, C1에서, 제1 애플리케이션은 App to SL Query 메시지를 애플리케이션 적응 서비스(706)에게 송신한다. C2에서, 다른 애플리케이션은 App to SL Query 메시지를 애플리케이션 적응 서비스(706)에게 송신한다. C3에서, 예시된 예에 따르면, 애플리케이션 적응 서비스(706)는 App to SL query 메시지들을 통합한다. C4에서, 애플리케이션 적응 서비스(706)는 통합된 질의(App to SL Query 메시지)를 NAMS(100)에게 송신한다. C5에서, NAMS(100)는 SL to App input 메시지를 애플리케이션 적응 서비스(706)에게 송신한다. C6에서, NAMS(100)는 입력들을 디스패치한다. 예를 들어, C7에서, NAMS(100)는 SL to App Input 메시지를 제1 애플리케이션에게 송신하고, C8에서, NAMS(100)는 SL to App Input 메시지를 다른 애플리케이션에게 송신한다.

[0067] 이제부터 NAMS를 사용하는 예시적인 사용 사례 시나리오들을 살펴보면, NAMS(100)는 네트워크 최적화 및 애플리케이션 최적화를 가능하게 할 수 있다. 제1 예시적인 사용 사례가, 예시적인 서비스 계층 지원 네트워크 최적화를 예시하는, 도 15에 도시되어 있다.

[0068] 도 15를 참조하면, 예시된 예에 따르면, 2개의 애플리케이션들, 디바이스 1 상에서 실행 중인 애플리케이션 1 및 디바이스 2 상에서 실행 중인 애플리케이션 2가 네트워크 노드에 있는 서비스 계층에 등록된다. 도시된 바와 같이, 네트워크 노드 상의 서비스 계층에 의해 제공되는, 서비스 계층 기능들이라고도 지칭될 수 있는, 다른 서비스들은 등록 서비스(1502), 위치 서비스(1504), 네트워크 노출 서비스(1506), 그리고 데이터 관리 및 리포지토리(1508)를 포함한다. 네트워크 노드에 있는 NAMS(100)는 애플리케이션들 및 다른 서비스들로부터 정보를 수집하고, 예를 들어, 통신을 최적화하기 위해, 그의 정책에 기초하여 디바이스 1 및 디바이스 2를, P2P(peer-to-peer) 통신 모드라고도 지칭될 수 있는, D2D(device-to-device)로 전환하라고 추천한다.

[0069] 도 15를 여전히 참조하면, 예시된 사용 사례에 따르면, 1에서, 애플리케이션 2는 등록 서비스(1502)를 사용하여 네트워크 노드에 있는 서비스 계층에 등록한다. 2에서, 애플리케이션 1은 네트워크 노드에 있는 서비스 계층에 등록한다. 따라서, 애플리케이션들은, 현재 서비스 계층에 이미 정의되어 있는 바와 같은 위치 서비스(1504) 그리고 데이터 관리 및 리포지토리 서비스(1508)를 비롯한 예시된 서비스들과 같은, 서비스 계층에 있는 다른 기존의 서비스들을 사용할 수 있다. 3에서, NAMS(100)는 "App Status Update" 메시지를 사용하여 애플리케이션 1과 관련된 정보를 검색한다. 4에서, NAMS(100)는 "App Status Update" 메시지를 사용하여 애플리케이션 2와 관련된 정보를 검색한다. 검색된 정보는, 예를 들어, 애플리케이션의 유형, 트랜잭션의 지속시간 등과 같은, 다양한 애플리케이션 관련 정보를 포함할 수 있다. NAMS(100)가, 간략함을 위해 도 15에 도시되지 않은, "App Status Query" 메시지를 사용하여 이러한 정보를 풀링할 수 있다는 것이 이해될 것이다. 예시적인 App Status

Update 및 App Status Query 메시지들은 전술한 표 1에 기술되어 있다. 애플리케이션들과 연관된 정보에 기초하여, 예시된 예에 따르면, NAMS(100)는 애플리케이션 1과 애플리케이션 2 사이의 트랜잭션들이 긴 지속시간들을 가지며 대량의 데이터를 포함한다고 결정한다. 6 및 7에서, NAMS(100)는 등록 서비스(1502)로부터 서비스 계층 등록 정보를 수신한다. 예를 들어, 도시된 바와 같이, NAMS(100)는 애플리케이션 1이 디바이스 1 상에 있고, 애플리케이션 2가 디바이스 2 상에 있으며, 애플리케이션 1과 애플리케이션 2가 동일한 서비스 계층에 등록되어 있다는 것을 표시하는 등록 정보를 수신한다. 8 및 9에서, NAMS(100)는, 네트워크 노출 서비스(1506)로부터, 디바이스 1 및 디바이스 2와 연결되어 있는 네트워크가 트래픽 과부하 문제를 겪고 있다고 결정한다. 예시된 예에 따르면, 10 및 11에서, NAMS(100)는, 서비스 계층에 있는 위치 서비스(1504)로부터 정보를 획득하는 것에 의해, 디바이스 1 및 디바이스 2의 위치, 그리고 따라서 애플리케이션 1 및 애플리케이션 2의 위치들을 결정할 수 있다. 따라서, NAMS(100)는 2개 이상의 디바이스들, 예를 들어 디바이스 1과 디바이스 2가 서로 근접해 있다고 결정할 수 있다.

[0070] 도 15를 계속하여 참조하면, 12 및 13에서, 예시된 예에 따르면, NAMS(100)는 데이터 관리 및 리포지토리 서비스(1508)로부터 트랜잭션 기록(transaction record)들을 검색할 수 있다. 예를 들어, 기록들에 기초하여, NAMS(100)는 애플리케이션 1과 애플리케이션 2가 서로 통신한다고 결정할 수 있다. 14에서, 예를 들어, 앞서 기술된 검색된 정보에 기초하여, NAMS(100)는 그의 내부 정책에 기초하여 그 자신의 의사결정 논리를 트리거할 수 있다. 따라서, NAMS(100)는 디바이스 1 및 디바이스 2가 D2D 연결 모드로 전환할 수 있다고 결정할 수 있다. 15에서, 예시된 예에 따르면, NAMS(100)는 그가 생성하는 결과들을 네트워크 노출 서비스(1506)에게 송신할 수 있다. 네트워크 결과들은 D2D가, 이전 연결 모드, 예를 들어 디바이스 1과 디바이스 2가 기반 네트워크를 통해 서로 연결되어 있는 것과 비교하여, 디바이스 1과 디바이스 2에 대해 보다 최적화된 연결 모드라는 것을 표시할 수 있다. 서비스 계층은 또한 의사 결정에 관련된 정보를, 예를 들어, 이러한 정보가 노출되도록 허용되어 있는 경우, 기반 네트워크에게 전달하기로 선택할 수 있다. 네트워크 노출 서비스(1506)는 그 정보를 Mcn 참조점을 통해 기반 네트워크에게 전달할 수 있다. 16에서, 예시된 예에 따르면, 기반 네트워크는 서비스 계층으로부터 추천을 수신한다. 네트워크는 또한, 디바이스 1 및 디바이스 2의 가입자 계약들, 네트워크 통계 등과 같은, 그 자신의 정보를 체크할 수 있다. 추천에 기초하여 그리고/또는 네트워크 자신의 정보에 기초하여, 네트워크는 디바이스들(예컨대, 디바이스 1 및 디바이스 2)을 D2D 모드로 전환할지를 결정할 수 있다. 기반 네트워크는 디바이스들의 모드를 전환하기 위해 그 자신의 절차들을 트리거할 수 있다.

[0071] 서비스 계층 지원 네트워크 최적화의 다른 예시적인 사용 사례 시나리오에서, 서비스 계층에 있는 NAMS(100)로부터 획득되는 애플리케이션 관련 정보를 사용하여, 주어진 네트워크는 애플리케이션들을 상이한 액세스 네트워크들로 오프로드(offload)할 수 있다. 예로서, NAMS(100)는 동일한 기반 네트워크 상에서 실행 중인 많은 수의 애플리케이션들이 있다고 그리고 애플리케이션들이 많은 양의 데이터를 전송한다고 결정할 수 있다. NAMS(100)는 또한 이용가능한 기반 네트워크들 및 그들과 연관된 네트워크 부하(예컨대, 혼잡) 정보를 다른 서비스(예컨대, 네트워크 노출 서비스(1506))로부터 결정할 수 있다. 어떤 경우에, NAMS(100)는 하나 이상의 애플리케이션들이, 하나 이상의 애플리케이션들이 현재 실행되는 네트워크와 동일한 네트워크 운영자를 가질 수 있는, 상이한 기반 네트워크 상에서 실행되어야만 한다는 추천을 한다. NAMS(100)는 입력(추천)을 하나 이상의 애플리케이션들 및/또는 기반 네트워크에게 송신할 수 있다. 애플리케이션 또는 네트워크 중 어느 하나는 애플리케이션이 실행되는 네트워크를 이전(transfer)하는 조치를 개시할 수 있다.

[0072] 서비스 계층 지원 네트워크 최적화의 또 다른 예시적인 사용 사례 시나리오에서, 서비스 계층을 통해 수신되는 다른 네트워크들로부터의 정보를 사용하여, 주어진 네트워크는, 예를 들어, 전력 제어를 위해 그의 커버리지를 조정할 수 있다. 예로서, 2개의 중간 노드들(예컨대, 무선 게이트웨이들)이 2개의 상이한 영역 네트워크(area network)들 상에 존재한다고 가정한다. 각각의 중간 노드 내의 서비스 계층으로부터 획득된 전력 또는 신호 강도 정보를 사용하여, NAMS(100)는 중간 노드가 그의 송신 전력을 감소시키기를 원할 수 있다는 것을 표시하기 위해 출력을 생성할 수 있다. NAMS(100)는 이러한 정보를 Mcn을 통해 네트워크 스택(network stack)에게 전달할 수 있다. NAMS(100)로부터 수신된 정보에 적어도 부분적으로 기초하여, 주어진 네트워크는 그의 전력을 조정하기로 선택할 수 있다.

[0073] 이제 도 16을 참조하면, 예시적인 서비스 계층 지원 애플리케이션 최적화를 나타내는 예시적인 사용 사례가 도시되어 있다. 예시된 예에서, 1에서, 제1 네트워크(NW1)에 연결된 제1 디바이스(디바이스 1) 상의 제1 애플리케이션(애플리케이션 1)은 제1 서비스 계층(서비스 계층 1)에 등록한다. 3에서, 제2 네트워크(NW2)에 연결된 제2 디바이스(디바이스 2) 상의 제2 애플리케이션(애플리케이션 2)은 제2 서비스 계층(서비스 계층 2)에 등록한다. 2 및 4에서, 각자의 디바이스들 상에, 각각, 있는 제1 및 제2 서비스 계층들은 또한 인프라스트럭처 노드

(1602)에 있는 서비스 계층에 등록한다. 서비스 계층들 각각은 NAMS(100)를 포함한다. 예시된 예에 따르면, 5 및 6에서, 애플리케이션 1 및 애플리케이션 2는 그들의 정보를 "App Status Update" 메시지를 사용하여, 각각, 서비스 계층 1 및 서비스 계층 2에 보고한다. 따라서, 애플리케이션들은 그들이 등록되어 있는 서비스 계층들에게 정보를 보고할 수 있다. 보고된 정보는, 예를 들어 그리고 제한 없이, 애플리케이션의 유형, 애플리케이션이 전송하는 데이터의 양 등을 포함할 수 있다. SL, 상세하게는 NAMS(100)가, 간략함을 위해 도 16에 도시되지 않은, "App Status Query" 메시지를 사용하여 이러한 정보를 폴링할 수 있다는 것이 이해될 것이다. 7 및 8에서, 디바이스 1에 있는 서비스 계층 1 및 디바이스 2에 있는 서비스 계층 2는 이러한 정보를, 예를 들어, "SL Exchange" 메시지를 사용하여, 인프라스트럭처 노드(1602)에 있는 SL에게 추가로 보고할 수 있다. 따라서, 9에서, 인프라스트럭처 노드(1602)에 있는 서비스 계층은 애플리케이션 1 및 애플리케이션 2와 연관된 정보를 획득한다.

[0074] 도 16을 여전히 참조하면, 10에서, 서비스 계층 1, 상세하게는 서비스 계층 1(SL1)에 있는 NAMS(100)는 네트워크 노출 서비스(1506)를 통해 기반 네트워크 1(NW1)과 연관된 통계를 획득한다. 이와 유사하게, 11에서, 서비스 계층 2, 상세하게는 서비스 계층 2(SL2)에 있는 NAMS(100)는 네트워크 노출 서비스(1506)를 통해 기반 네트워크 2(NW2)와 연관된 통계를 획득한다. 12 및 13에서, 디바이스 1 및 디바이스 2 상의 서비스 계층들이 인프라스트럭처 노드(1602)에 있는 서비스 계층에 등록되기 때문에, 인프라스트럭처 노드(1602)에 있는 서비스 계층은 "SL Exchange" 메시지들을 통해 통계를 수신한다. 따라서, 14에서, NW1 및 NW2와 연관된 통계는 인프라스트럭처 노드(1602)에 있는 SL에 의해 획득된다. 15에서, 예시된 예에 따르면, 인프라스트럭처 노드(1602)에 있는 NAMS(100)는 수집된 정보를 분석하고, NAMS(100)의 정책들에 기초하여 결정을 한다. 예로서, 인프라스트럭처 노드(1602)에 있는 NAMS(100)는 애플리케이션 1 및 애플리케이션 2가 보다 낮은 대역폭을 할당받아야 한다는 표시를 제공할 수 있다. 다른 예로서, 인프라스트럭처 노드(1602)에 있는 NAMS(100)는, 예를 들어, 주어진 네트워크 연결성 문제가 일시적이면, 애플리케이션들이 전송들을 지연시켜야 한다는 것을 표시할 수 있다. 도시된 바와 같이, 이러한 표시들은 16 및 17에서 "SL Exchange" 메시지를 통해 디바이스 1 및 디바이스 2에 있는 NAMS(100)에게 송신될 수 있고, 디바이스들에 있는 NAMS는 표시들을 18 및 19에서 "SL to App Input" 메시지를 통해 각자의 애플리케이션들에게 송신할 수 있다. 예시적인 SL Exchange 메시지들 및 SL to App Input 메시지들은 표 1에서 앞서 기술되어 있다.

[0075] 서비스 계층 지원 애플리케이션 최적화의 다른 예시적인 사용 사례 시나리오에서, NAMS(100)는 2개 이상의 애플리케이션들 간의 관리를 용이하게 할 수 있다. 예를 들어, 애플리케이션(관리자 애플리케이션)은 하나 이상의 다른 애플리케이션들을 관리할 수 있다. 관리자 애플리케이션은, 예를 들어, 미리 결정된 임계값(조건)이 충족될 때 하나 이상의 애플리케이션들을 특정의 기반 네트워크로 전환하도록 트리거하기 위해, 그의 정책을 NAMS(100)에게 전달할 수 있다. NAMS(100)는 요구된 입력을 수집할 수 있고, 그 조건이 충족될 때, NAMS(100)는, 관리자 애플리케이션을 대신하여, 트리거들을 하나 이상의 애플리케이션들에게 송신할 수 있다.

[0076] 이제 도 17을 참조하면, 앞서 기술된 바와 같이, 서비스 계층에 있는 NAMS(100)는 독립적으로 동작할 수 있거나, 이러한 메커니즘들을 보완하기 위해 기존의 네트워크 관리 메커니즘들과 함께 동작할 수 있다. 도 17은 예시적인 계층들의 프로토콜 스택(1700)을 나타낸다. SNMP는 애플리케이션 프로토콜 계층(1702)에서 사용되는 예시적인 프로토콜이다. 애플리케이션 프로토콜 계층(1702) 위쪽에, 네트워크들은, 일반적으로 네트워크 관리 계층(1704)이라고도 지칭될 수 있는, 네트워크 관리 기능들을 가질 수 있다. 도시된 바와 같이, 서비스 계층(1706)은 애플리케이션 프로토콜 계층(1702) 및 네트워크 계층(1704) 위에, 애플리케이션들(1708) 바로 아래에 존재한다. 서비스 계층(1706)은, 앞서 기술된 Mcn을 통한 메시지들 및 정보 교환들에 포함된 정보와 같은, 기반 네트워크 정보를 획득하기 위해 네트워크 관리 계층(1704)과 상호작용할 수 있다. 예를 들어, TCP와 같은, 일부 전송 프로토콜들은 그들의 전송 속도들을 적응시킬 수 있는 능력을 갖는다. 이러한 프로토콜들은 사후대응적인 반면, 본원에서 기술되는 NAMS(100)는 다양한 상황들을 방지할 수 있고, 따라서 기존의 프로토콜들과 비교하여 전체적인 성능을 향상시킨다.

[0077] 서비스 계층(1706)은 SNMP로부터 정보를 수신할 수 있고, SNMP는 Mcn 참조점이 서비스 계층(1706)과 상호작용하기 위해 사용될 수 있다. SNMP는, 예를 들어, 네트워크 디바이스 관리와 같은, 그의 기존의 기능들을 여전히 수행할 수 있다.

[0078] 앞서 언급된 바와 같이, NAMS(100)는 그 자신의 집계된 정보 및 정책들에 기초하여 기반 네트워크에의 입력을 생성할 수 있다. 따라서, 기반 네트워크들은, 예를 들어, NAMS(100) 및 기존의 네트워크 관리 기능들과 같은, 상이한 입력 소스들에 기초하여 조치들을 취할 수 있다.

[0079] 예시적인 실시예에 따르면, 도 18을 참조하면, NAMS(100) 및 앞서 기술된 적응 기능들은 oneM2M RoA 아키텍처에 적합하다. 예시된 실시예에 따르면, NAMS(100)는 새로운 공통 서비스 기능(CSF)이다. 도시된 바와 같이, 네트워크 적응 서비스(708)는 NSSE CSF의 일부일 수 있다. 게다가 도시된 바와 같이, 애플리케이션 적응 서비스(706) 및 서비스 적응(710)은 임의의 CSF들의 외부에 있다. 대안의 실시예에서, NAMS(100)는 NSSE의 일부일 수 있다.

[0080] 이제 도 19a 및 도 19b를 참조하면, 2개의 예시적인 대안의 자원 구조들이 본원에 기술되는 NAMS 기능들을 지원하는 것으로 도시되어 있다. 자원 구조들은 애플리케이션들, 기반 네트워크들, 및 다른 서비스 계층들로부터의 입력을 포함한다. NAMS(100)는 정보를 처리하고 출력을 생성하기 위해 입력을 사용할 수 있다. 앞서 표 1에서 기술된 예시적인 정보 교환(IE) 메시지들은 도 19a 및 도 19b에 예시된 자원들에 저장될 수 있다. 예를 들어, NAMS(100)는 로컬 정책들 또는 외부 정책들에 대한 링크들을 <policy> 자원에 저장할 수 있다. <policy> 자원을 운영하는 것에 의해 정책들이 추가, 삭제, 또는 업데이트될 수 있다. 이하의 표 2는, 예시적인 실시예에 따른, <policy> 구조가 어떻게 정의될 수 있는지를 나타내고 있다.

[0081] 표 2와 도 19a 및 도 19b를 참조하면, SL에 있는 NAMS(100)은 그가 어떤 정보를 애플리케이션들, 네트워크들, 및 다른 서비스들에게 송신할 수 있는지를 결정할 수 있다. 일 예에서, 표 1에 기술된 정보의 서브셋은 <toApp>, <toNw> 및 <toSL>에 저장될 수 있다. 예를 들어, "SL to NW Input" 메시지에서의 정보 교환(IE)들은 <toNw>에 저장될 수 있다.

[0082] 일 예에서, 애플리케이션들, 네트워크들, 및 다른 서비스들로부터의 입력은 <fromApp>, <fromNw>, 및 <fromSL>에 저장된다. 각각의 자원에 어느 정보가 저장될 수 있는지는 NAMS(100)의 구성에 기초하고, 예를 들어, 표 1에 기술된 "config" 메시지들을 사용하는, NAMS(100)와 다른 노드들 사이의 협상들에 기초할 수 있다. 예를 들어, "NW Status Update" 메시지에서 교환되는 정보는 <fromNw>에 저장될 수 있다. 각각의 네트워크는 각각의 네트워크와 연관된 고유 ID에 의해 서로 구별될 수 있다.

**표 2**

"policy"자원의 예시적인 속성들

속성 이름	설명
policyID	이것은 각각의 정책의 고유 ID이다
policyName	이것은 정책의 이름이다.
description	이것은 정책을 기술하는 텍스트이다.
label	이것은 정책에 대해 탐색가능한 키워드이다 정책들은 상이한 그룹들로 분류되고 그룹 키워드들에 의해 탐색가능할 수 있다 예를 들어, "QoS", "대역폭".
externalPolicy	이것은 NAMS의 외부에 있는 외부 정책에 대한 링크, 예를 들어, 외부 정책에 대한 URI이다. 하기의 속성들은 외부 정책에 대해 필요하지 않을 수 있는데, 그 이유는 정책의 상세들이 거기에 정의되어 있을 수 있기 때문이다.
namsPolicy	이것은 정책의 기능에 대한 포인터이다. 이것은 SL 소프트웨어에 의해 지원되는 기능이다.
input	이것은 이 정책의 입력 파라미터들의 리스트이다
output	이것은 이 정책의 출력이다
creationTime	정책이 작성된 때를 타임스탬프.
expirationTime	정책이 만료되어야만 하는 때를 나타내는 타임스탬프. 정책이 영구적일 수 있고, 삭제되지 않는 한, 만료되지 않을 것이다.
applicability	이 정책이 적용되는 엔티티들의 리스트를 나타낸다. 각각의 엔티티는, 애플리케이션 ID, CSE-ID 및 네트워크 ID와 같은, 고유 ID에 의해 식별된다.

[0083]

[0084] 이제 도 20을 참조하면, HTTP 프로토콜 및 도 19a에 도시된 자원 구조를 사용하는 예시적인 호 흐름이 도시되어 있다. 이하에서 추가로 기술되는 바와 같이, 1 내지 3에서, 도 20은 표 1에 정의된 "App Status Update"가 어떻게 HTTP 프로토콜을 사용하여 수행될 수 있는지의 일 예를 나타내고 있다. 단계 4 내지 단계 8은 예시적인 "App Status Query" 동작을 나타내고 있다.

- [0085] 도 20을 여전히 참조하면, 예시된 실시예에 따르면, 1에서, 애플리케이션 엔티티(AE1)는 HTTP POST 요청을 개시하는 것에 의해, 제한 없이 공통 서비스 엔티티(CSE)라고도 지칭될 수 있는, 서비스 계층에 있는 NAMS(100)에게 "App Status Update" 메시지를 송신한다. 예시된 CSE는 CSE1이라고도 지칭될 수 있다. 요청에서, AE1은 그의 자원이 생성되어야만 하는 목적지 URI를 표시한다. 이 예에서, 이는 `http://cse1.m2m.spl.com/cse1/nams/appInfo/fromApp/`에 저장된다. AE1은 또한 자원 유형이 "appInfo" 자원이어야만 한다는 것과 자원 이름이 `ae1Status`라는 것을 표시한다. 예시된 예가 자원을 구조화하는 한 방법을 나타내지만, 자원이 원하는 바에 따라 대안적으로 구조화될 수 있다는 것이 이해될 것이다. 예시된 예에서, 상이한 AE 상태들에 대한 자원들은 <appInfo>에 바로 저장된다. 대안의 예에서, CSE1은 <fromApp>을 컬렉션(collection)으로서 사용하고 모든 AE 상태 자원들을 /appInfo/fromApp에 저장할 수 있다. 도시된 바와 같이, AE1은 요청 ID 0001을 메시지에 포함시키고, 메시지에서의 내용 포맷(content format) 및 내용 길이(content length)를 표시한다. 페이로드(payload)에, AE1은 AE 상태 업데이트에 대한 자원 표현을 포함시킬 수 있다. 2에서, 요청을 수신할 시에, CSE1은 생성자(creator)의 액세스 권한(access right)을 검증하고, 자원을 생성하도록 허용된 경우, CSE1은 AE1 상태에 대한 자원을 생성한다. 3에서, 예시된 예에 따르면, 자원을 성공적으로 생성한 후에, CSE1은, 자원의 URI 및 동작의 상태를 포함하는, HTTP 응답 메시지(HTTP response message)를 송신한다.
- [0086] 도 20을 계속하여 참조하면, 4에서, CSE1에 있는 NAMS(100)는 애플리케이션 상태를 질의하기로 결정한다. 이것이 이전 단계들 직후에 일어날 필요는 없지만, 예시된 동작들 동안 특정 때에 일어날 수 있다. 5에서, CSE1은 "App Status Query" 메시지를 HTTP GET 요청을 사용하여 AE1에게 송신한다. 이 동작이 AE가 HTTP 서버일 수 있다는 것을 표시하고, AE에게 검색을 위해 그 자신의 상태를 로컬적으로 저장하라고 요청한다는 것에 유의한다. 6에서, AE1은 그의 상태를 CSE1에게 반환(return)하기로 결정한다. 7에서, AE1은 그의 상태를 <appInfo>의 자원 표현으로 반환한다. 8에서, CSE1은 검색된 AE1 상태를 그의 자원 구조 내의 위치에 저장한다.
- [0087] 이제 도 21을 참조하면, 본원에 기술되는 NAMS(100)의 기능들이 어떻게 oneM2M SoA 아키텍처에 적용될 수 있는지의 일 예가 도시되어 있다. 일 예에서, NAMS(100)는, 도 21에 도시된 바와 같이, 독립적인 새로운 서비스로서 정의될 수 있다. 상자 1, 상자 2, 및 상자 3으로 도시된, 적응 기능들은 기존의 서비스 노출 컴포넌트들의 일부일 수 있다. Mf' 참조점들을 통한 서비스들 간의 상호작용들은 Msc 참조점에 의해 달성될 수 있다. 앞서 기술된 메시지들 및 정보 교환들이 SoA 아키텍처에도 적용될 수 있다는 것이 이해될 것이다.
- [0088] 도 22를 참조하면, 앞서 기술된 메시지들을 달성하기 위해 MQTT 프로토콜을 사용하는 예시적인 호 흐름이 도시되어 있다. 예시된 호 흐름은 "App Status Update" 및 "App Status Query"를 지원하기 위해 MQTT가 어떻게 사용될 수 있는지의 일 예를 나타내고 있다. 도 22는, 외부 서버일 수 있거나 서비스 계층 엔티티에 내장될 수 있는, MQTT 서버(2202)를 나타내고 있다. 예시된 예에서, MQTT 서버(2202)는 명확함을 위해 외부 엔티티로서 도시되어 있다. 예시된 AE(2204) 및 CSE(2206)는 MQTT 클라이언트들이다. 1a 내지 1d를 참조하면, 서비스 계층 관련 동작들 이전에, AE(2204) 및 CSE(2206)는 MQTT CONNECT 메시지를 사용하여 MQTT 서버(2202)에 연결한다. MQTT 서버(2202)는 그에 연결된 AE(들) 및 CSE(들)의 식별자 및 주소를 획득한다. 2a 및 2b에서, 예시된 예에 따르면, 서비스 계층에 있는 NAMS(100)는 MQTT SUBSCRIBE 메시지를 트리거한다. 이 메시지에, NAMS(100)은 그가 "App Status Update"의 MQTT 토픽(MQTT Topic)에 관심이 있다는 것을 포함시킨다. NAMS(100)은 그가 관심이 있는 AE들의 특정 식별자를 표시할 수 있다. 이것은 표 1에 기술된 "App Status Query"와 동등할 수 있다. 3a 내지 3c에서, 예시된 예에 따르면, AE(2204)는 MQTT PUBLISH 메시지를 사용하여 그 자신의 상태 업데이트들을 발행(publish)한다. 메시지는 메시지 페이로드에 상태 업데이트 정보를 포함하고 있다. 메시지를 수신할 시에, MQTT 서버(2202)는 정보를 그의 토픽 리스트에 저장할 수 있다. 애플리케이션 상태와 연관된 정보는 표 1에 기술되어 있다. 4a 내지 4b에서, NAMS(100)가 AE1 상태 업데이트의 토픽을 구독했기 때문에, MQTT 서버(2202)는 통지들을 MQTT PUBLISH 메시지를 사용하여 NAMS(100)에게 송신할 것이다. MQTT 서버(2202)는 그가 CSE(2206)에 있는 NAMS(100)에게 송신하는 PUBLISH 메시지의 페이로드에 AE(2204) 상태 업데이트를 넣을 수 있다.
- [0089] 이제부터 기반 네트워크들과 인터페이싱하는 것을 살펴보면, Mcn 참조점을 통해 앞서 기술된 NAMS 메시지들을 포함시키는 예시적인 실시예들이 이제부터 기술된다. 도 23을 참조하면, NSE의 블록도, CSE에 대한 그의 인터페이스, 및 기반 네트워크(들) 쪽으로 가능한 몇몇 예시적인 인터페이스들이 도시되어 있다. Mcn 참조점은 서비스들, 예를 들어, 이하의 표 3에 열거된 서비스들에 액세스하기 위한 oneM2M 정의 API(oneM2M defined API)들을 CSE에게 제공한다. 지원되는 서비스들은 각각의 디바이스에 의해 사용되고 있는 기반 액세스 네트워크(들),

CSE 소유자와 네트워크 운영자 간의 비즈니스 협약, 및 NSE의 능력들에 의해 지원되는 것에 의존할 수 있다.

[0090] 일 예에서, NSE들은 다양한 목적으로 CSE 및 NSE에 의해 사용하는 기본 API(base API) 세트를 제공한다. 예를 들어, CSE는, CSE 식별자들, 애플리케이션 식별자들, 및 액세스 네트워크 식별자들 간의 매핑을 NSE에게 제공하기 위해, 기본 API 세트를 사용할 수 있다. 어떤 경우에, 기본 API들이 이 정보를 NSE에 프로비저닝하는 데 사용되지 않으면, NSE는 CSE 또는 애플리케이션 식별자들을 인식하지 못할 것이다. NSE는 NSE에서 프로비저닝되지 않는 액세스 네트워크 식별자들을 인식할 수 있다. 애플리케이션 식별자 또는 CSE 식별자가 액세스 네트워크 식별자에 대한 매핑을 갖지 않으면, 그 식별자가 NSE에게 아무런 가치(의미)도 없을 수 있다는 것에 유의한다. CSE는, 각각의 디바이스, CSE, 및 애플리케이션에 대해 어느 서비스들이 이용가능한지를 체크하기 위해, 기본 API 세트를 사용할 수 있다. CSE는 또한, 디바이스들, CSE, 및 애플리케이션들에 대한 서비스들을 인에이블(enable)시키기 위해, 기본 API 세트를 사용할 수 있다.

**표 3**

Mcn을 통해 이용가능한 예시적인 서비스들

구독 관리
오프라인 과금 이벤트 보고
온라인 과금 이벤트보고
온라인 과금 신용 관리
트리거링
QoS 관리
소규모 데이터 교환(Small Data Exchange)
스케줄 조정(Schedule Coordination)
디바이스 관리
SW 설치
디바이스 모니터링
인증
권한부여(Authorization)
키 설정(Key Establishment)
프레즌스(Presence)

[0091]

[0092] 일 예에서, 음성, SMS, IP 메시징, 파일 공유, 비디오 스트리밍, 고객 인증, 위치, 과금, 및 프레즌스와 같은 서비스들을 노출시키기 위해, oneAPI 게이트웨이에 대한 인터페이스가 사용될 수 있다. 이 서비스들은 표 3에 열거된 서비스들을 지원하기 위해 NSE에 의해 이용가능하다.

[0093] 다른 예시적인 실시예에서, 3GPP 인터페이스들이 사용된다. 예를 들어, NSE는 Mcn 인터페이스를 통한 명령어들을 사우스바운드 인터페이스(southbound interface)를 통한 동작들에 매핑하는 내부 논리를 가질 수 있다. NSE가 3GPP 네트워크에서 어떻게 지원될 수 있는지의 예들이 이하에서 기술된다. 일 예에서, NSE가 배포되고, 3GPP 네트워크 내의 상이한 엔티티들에서 정보가 생성되는 곳이면 어디든지에서 정보가 수집된다. 정보는, 예를 들어, Tsp, Rx, Mh와 같은, 다수의 참조점들에 의해 송신된다.

[0094] 표 4(이하)는 표 3에 열거된 서비스들을 수용하기 위해 필요할 수 있는 3GPP 인터페이스들을 열거한다. Tsp 및 Rx 참조점들이 3GPP에 의해 정의되지만, 표 3에 열거된 M2M 서비스들을 지원하기 위해 새로운 명령어들(또는 파라미터들)이 이 인터페이스들에 추가될 수 있다는 것에 유의한다.

[0095] Mh, Mf, 및 Mo 참조점들이 3GPP에 의해 완전히 정의되지 않는다는 것이 본원에서 인식된다. 이 참조점들은 MOSAP 작업 항목(MOSAP work item) 하에 작성된 3GPP TR에서 도입된다. 3GPP TR 23.862, "EPC Enhancements to Support Interworking with Data Application Providers (MOSAP)" - 그의 개시내용은 참조에 의해 그 전체가 본원에 기재된 것처럼 원용됨 - 은 Mh, Mf, 및 Mo 참조점들이 Sh, Ro, 및 Rf 참조점들에, 각각, 기초할 수 있다고 명시하고 있다. 표 3에 열거된 M2M 서비스들을 지원하기 위해 새로운 명령어들(또는 파라미터들)이 이 인터페이스들에 추가될 수 있다는 것이 본원에서 인식된다.

표 4

3GPP 기반 네트워크 인터페이스들

참조점	목적	프로토콜
Tsp	MTC 등	Diameter
Rx	QoS	Diameter
Mh	구독 관리	Diameter
Mf	온라인 과금	Diameter
Mo	오프라인 과금	Diameter
Le	위치 관리	OMA MLP(Mobile Location Protocol) 또는 OSA-API(Open Service Access Application Programming Interface)
Zn	보안	Diameter

[0096]

[0097]

이제 도 24를 참조하면, 도 24는 NSE가 3GPP 네트워크에 의해 지원될 수 있는 중앙 집중식 모델을 도시하고 있다. NAMS(100)는 3GPP 간접 및 하이브리드 모델에서 SCS의 일부이다. 일 예에서, MTC-IWF는, 예를 들어, HSS, MSC, MME 및 SGSN과 같은, 다른 3GPP 노드들로부터 네트워크 관련 제어 평면 정보를 수집할 수 있는, NSE 기능을 갖는다. 다른 예에서, 트래픽 및 사용자 평면 정보가 수집될 수 있는 GGSN/P-GW에 있는 NSE도 있을 수 있다. 따라서, Mcn은 Tsp 인터페이스와 Gi/SGi 인터페이스 둘 다에 의해 실현될 수 있다. SCS 내의 네트워크 적응 서비스는 네트워크 정보를 통합할 수 있다. SCS로부터 3GPP 네트워크로의 메시지는 제어 평면에 의해 MTC-IWF로 그리고 사용자 평면에 의해 GGSN/P-GW로 분할된다. 도 24는 본원에 기술되는 NAMS(100) 및 참조점들을 통한 메시지들이 어떻게 3GPP MTC 연동 모델(3GPP MTC interworking model)에서 지원될 수 있는지의 일 예를 도시하고 있다. 게다가, NAMS(100) 기능을 포함하는 UE 상의 서비스 계층이 있을 수 있다.

[0098]

이제 도 26을 참조하면, 네트워크 및 애플리케이션/서비스 관리에 관련된 파라미터들을 디스플레이 및/또는 구성하기 위해, 디바이스, 게이트웨이들, 및/또는 서버들과 같은, M2M/WoT/IoT 엔티티에 의해 지원될 수 있는 예시적인 NAMS 그래픽 사용자 인터페이스(GUI) 화면(2600)이 도시되어 있다. 도 26의 예에서, 설정(2602)은 예시적인 애플리케이션과 연관된 정보에 관한 것이고, 설정(2604)은 NAMS(100)가 애플리케이션에게 제공할 수 있는 통지들에 대응한다. 도시된 바와 같이, 예시적인 통지들은 기반 네트워크와 연관된 이벤트들(상태 업데이트들)에 대응한다. 사용자가, 다양한 네트워크 이벤트들에 기초하여, 애플리케이션이 취할 적절한 조치를 선택할 수 있도록, 설정(2604)이 사용자에게 디스플레이될 수 있다. 이와 유사하게, 사용자가 애플리케이션의 요구사항들을 입력할 수 있도록, 설정(2602)이 사용자에게 제시될 수 있다. 실제로는, 원하는 바에 따라, 다른 설정들 및 파라미터들이 선택 및 디스플레이될 수 있다. 본원에 기술되는 네트워크 및 애플리케이션 관리 서비스들이 GUI들을 통해 다수의 방식으로 사용자에게 의해 관측되고 제어될 수 있다는 것이 이해될 것이다.

[0099]

본원에 기술되는 다양한 기법들이 하드웨어, 펌웨어, 소프트웨어, 또는 적절한 경우, 이들의 조합들과 관련하여 구현될 수 있다. 이러한 하드웨어, 펌웨어, 및 소프트웨어는 통신 네트워크의 다양한 노드들에 위치한 장치들에 존재할 수 있다. 장치들은 본원에 기술되는 방법들을 수행하기 위해 단독으로 또는 서로 결합하여 동작할 수 있다. 본원에서 사용되는 바와 같이, "장치", "네트워크 장치", "노드", "디바이스", 및 "네트워크 노드"라는 용어들은 서로 바꾸어 사용될 수 있다.

[0100]

도 25a는 하나 이상의 개시된 실시예들이 구현될 수 있는 예시적인 M2M(machine to machine), IoT(Internet of Things), 또는 WoT(Web of Things) 통신 시스템(10)의 다이어그램이다. 일반적으로, M2M 기술들은 IoT/WoT에 대한 구성 블록들을 제공하고, 임의의 M2M 디바이스, M2M 게이트웨이 또는 M2M 서비스 플랫폼은 IoT/WoT는 물론 IoT/WoT 서비스 계층 등의 컴포넌트일 수 있다. 도 5 내지 도 16, 도 18 내지 도 22, 또는 도 24에 예시된 클라이언트, 프록시, 또는 서버 디바이스들 중 임의의 것은 도 25a 내지 도 25d에 예시된 것과 같은 통신 시스템의 노드를 포함할 수 있다.

[0101]

도 25a에 도시된 바와 같이, M2M/IoT/WoT 통신 시스템(10)은 통신 네트워크(12)를 포함한다. 통신 네트워크(12)는 고정 네트워크(예컨대, 이더넷, 파이버(Fiber), ISDN, PLC 등) 또는 무선 네트워크(예컨대, WLAN, 셀룰러 등) 또는 이종 네트워크들의 네트워크일 수 있다. 예를 들어, 통신 네트워크(12)는 음성, 데이터, 비디오,

메시징, 방송 등과 같은 콘텐츠를 다수의 사용자들에게 제공하는 다수의 액세스 네트워크들을 포함할 수 있다. 예를 들어, 통신 네트워크(12)는 CDMA(code division multiple access), TDMA(time division multiple access), FDMA(frequency division multiple access), OFDMA(orthogonal FDMA), SC-FDMA(single-carrier FDMA) 등과 같은 하나 이상의 채널 액세스 방법들을 이용할 수 있다. 게다가, 통신 네트워크(12)는, 예를 들어, 코어 네트워크, 인터넷, 센서 네트워크, 산업 제어 네트워크, 개인 영역 네트워크, 융합 개인 네트워크(fused personal network), 위성 네트워크, 홈 네트워크, 또는 엔터프라이즈 네트워크와 같은 다른 네트워크들을 포함할 수 있다.

[0102] 도 25a에 도시된 바와 같이, M2M/IoT/WoT 통신 시스템(10)은 인프라스트럭처 도메인(Infrastructure Domain) 및 필드 도메인(Field Domain)을 포함할 수 있다. 인프라스트럭처 도메인은 종단간 M2M 배치(end-to-end M2M deployment)의 네트워크층을 지칭하고, 필드 도메인은, 보통 M2M 게이트웨이 후방에 있는, 영역 네트워크(area network)들을 지칭한다. 필드 도메인 및 인프라스트럭처 도메인 둘 다는 네트워크의 각종의 상이한 노드들(예컨대, 서버들, 게이트웨이들, 디바이스들 등)을 포함할 수 있다. 예를 들어, 필드 도메인은 M2M 게이트웨이들(14) 및 단말 디바이스들(18)을 포함할 수 있다. 원하는 바에 따라, 임의의 수의 M2M 게이트웨이 디바이스들(14) 및 M2M 단말 디바이스들(18)이 M2M/IoT/WoT 통신 시스템(10)에 포함될 수 있다는 것을 잘 알 것이다. M2M 게이트웨이 디바이스들(14) 및 M2M 단말 디바이스들(18) 각각은 통신 네트워크(12) 또는 직접 무선 링크(direct radio link)를 통해 신호들을 전송 및 수신하도록 구성되어 있다. M2M 게이트웨이 디바이스(14)는 무선 M2M 디바이스들(예컨대, 셀룰러 및 비-셀룰러)은 물론 고정 네트워크 M2M 디바이스들(예컨대, PLC)이, 통신 네트워크(12)와 같은, 운영자 네트워크들 또는 직접 무선 링크를 통해 통신할 수 있게 한다. 예를 들어, M2M 디바이스들(18)은 데이터를 수집하고, 데이터를 통신 네트워크(12) 또는 직접 무선 링크를 통해 M2M 애플리케이션(20) 또는 M2M 디바이스들(18)에게 송신한다. M2M 디바이스들(18)은 또한 M2M 애플리케이션(20) 또는 M2M 디바이스(18)로부터 데이터를 수신할 수 있다. 게다가, 이하에서 기술되는 바와 같이, 데이터 및 신호들이 M2M 서비스 계층(22)을 통해 M2M 애플리케이션(20)에게 송신되고 그로부터 수신될 수 있다. M2M 디바이스들(18) 및 게이트웨이들(14)은, 예를 들어, 셀룰러, WLAN, WPAN(예컨대, Zigbee, 6LoWPAN, Bluetooth), 직접 무선 링크, 및 유선(wireline)을 비롯한 다양한 네트워크들을 통해 통신할 수 있다. 예시적인 M2M 디바이스들은 태블릿, 스마트폰, 의료 디바이스, 온도 및 기상 모니터, 커넥티드 카, 스마트 미터, 게임 콘솔, PDA(personal digital assistant), 건강 및 피트니스 모니터, 전등, 서모스탯, 가전기기, 차고문 및 다른 액추에이터 기반 디바이스, 보안 디바이스, 및 스마트 콘센트를 포함하지만, 이들로 제한되지 않는다.

[0103] 도 25b를 참조하면, 필드 도메인에서의 예시된 M2M 서비스 계층(22)은 M2M 애플리케이션(20), M2M 게이트웨이 디바이스들(14), 및 M2M 단말 디바이스들(18) 그리고 통신 네트워크(12)에 대한 서비스들을 제공한다. M2M 서비스 계층(22)이 원하는 바에 따라 임의의 수의 M2M 애플리케이션들, M2M 게이트웨이 디바이스들(14), M2M 단말 디바이스들(18), 및 통신 네트워크들(12)과 통신할 수 있다는 것이 이해될 것이다. M2M 서비스 계층(22)은 하나 이상의 서버들, 컴퓨터들 등에 의해 구현될 수 있다. M2M 서비스 계층(22)은 M2M 단말 디바이스들(18), M2M 게이트웨이 디바이스들(14) 및 M2M 애플리케이션들(20)에 적용되는 서비스 능력들을 제공한다. M2M 서비스 계층(22)의 기능들은 각종의 방식으로, 예를 들어, 웹 서버로서, 셀룰러 코어 네트워크에, 클라우드에, 기타로 구현될 수 있다.

[0104] 예시된 M2M 서비스 계층(22)과 유사하게, 인프라스트럭처 도메인에도 M2M 서비스 계층(22')이 있다. M2M 서비스 계층(22')은 인프라스트럭처 도메인 내의 M2M 애플리케이션(20') 및 기반 통신 네트워크(12')에 대한 서비스들을 제공한다. M2M 서비스 계층(22')은 또한 필드 도메인 내의 M2M 게이트웨이 디바이스들(14) 및 M2M 단말 디바이스들(18)에 대한 서비스들을 제공한다. M2M 서비스 계층(22')이 임의의 수의 M2M 애플리케이션들, M2M 게이트웨이 디바이스들 및 M2M 단말 디바이스들과 통신할 수 있다는 것이 이해될 것이다. M2M 서비스 계층(22')은 상이한 서비스 제공자에 의한 서비스 계층과 상호작용할 수 있다. M2M 서비스 계층(22')은 하나 이상의 서버들, 컴퓨터들, 가상 머신들(예컨대, 클라우드/컴퓨팅/저장 팜 등) 등에 의해 구현될 수 있다.

[0105] 도 25b를 여전히 참조하면, M2M 서비스 계층(22 및 22')은 다양한 애플리케이션들 및 버티컬(vertical)들이 이용할 수 있는 핵심 서비스 전달 능력들의 세트(core set of service delivery capabilities)를 제공한다. 이 서비스 능력들은 M2M 애플리케이션들(20 및 20')이 디바이스들과 상호작용하고 데이터 수집, 데이터 분석, 디바이스 관리, 보안, 청구서 작성(billing), 서비스/디바이스 발견 등과 같은 기능들을 수행할 수 있게 한다. 기본적으로, 이 서비스 능력들은 애플리케이션들로부터 이 기능들을 구현하는 부담을 덜어주고, 따라서 애플리케이션 개발을 단순화하고 출시까지의 비용 및 시간을 감소시킨다. 서비스 계층(22 및 22')은 또한 M2M 애플리케이션들(20 및 20')이 서비스 계층(22 및 22')이 제공하는 서비스들과 관련하여 다양한 네트워크들(12 및 12')을

통해 통신할 수 있게 한다.

- [0106] M2M 애플리케이션들(20 및 20')은 운송, 건강 및 웰빙(health and wellness), 커넥티드 홈(connected home), 에너지 관리, 자산 추적, 그리고 보안 및 감시(이들로 제한되지 않음)와 같은 다양한 산업들에서의 애플리케이션들을 포함할 수 있다. 앞서 언급된 바와 같이, 시스템의 디바이스들, 게이트웨이들, 및 다른 서버들에 걸쳐 동작하는, M2M 서비스 계층은, 예를 들어, 데이터 수집, 디바이스 관리, 보안, 청구서 작성, 위치 추적/지오펜싱(geofencing), 디바이스/서비스 발견, 및 레거시 시스템 통합과 같은 기능들을 지원하고, 이 기능들을 서비스들로서 M2M 애플리케이션들(20 및 20')에게 제공한다.
- [0107] 일반적으로, 도 25a 및 도 25b에 예시된 서비스 계층들(22 및 22')과 같은, 서비스 계층(SL)은 API(application programming interface) 및 기반 네트워킹 인터페이스들의 세트를 통해 부가 가치 서비스 능력들을 지원하는 소프트웨어 미들웨어 계층을 정의한다. ETSI M2M 아키텍처와 oneM2M 아키텍처 둘 다는 서비스 계층을 정의한다. ETSI M2M의 서비스 계층은 서비스 능력 계층(SCL)이라고 지칭된다. SCL은 ETSI M2M 아키텍처의 각종의 상이한 노드들에 구현될 수 있다. 예를 들어, 서비스 계층의 인스턴스는 M2M 디바이스(여기서 디바이스 SCL(device SCL, DSCL)이라고 지칭됨), 게이트웨이(여기서 게이트웨이 SCL(gateway SCL, GSCL)이라고 지칭됨), 및/또는 네트워크 노드(여기서 네트워크 SCL(network SCL, NSCL)이라고 지칭됨) 내에 구현될 수 있다. oneM2M 서비스 계층은 공통 서비스 기능(CSF)들(즉, 서비스 능력들)의 세트를 지원한다. 하나 이상의 특정 유형의 CSF들의 세트의 인스턴스화는, 상이한 유형의 네트워크 노드들(예컨대, 인프라스트럭처 노드, 중간 노드, 애플리케이션 특정 노드(application-specific node)) 상에서 호스팅될 수 있는, 공통 서비스 엔티티(CSE)라고 지칭된다. 3GPP(Third Generation Partnership Project)는 또한 MTC(machine-type communications)에 대한 아키텍처도 정의하였다. 그 아키텍처에서, 서비스 계층과, 서비스 계층이 제공하는 서비스 능력들이 서비스 능력 서버(Service Capability Server, SCS)의 일부로서 구현된다. ETSI M2M 아키텍처의 DSCL, GSCL, 또는 NSCL에, 3GPP MTC 아키텍처의 서비스 능력 서버(SCS)에, oneM2M 아키텍처의 CSF 또는 CSE에, 또는 네트워크의 어떤 다른 노드에 구현되는 간에, 서비스 계층의 인스턴스는, 서버들, 컴퓨터들, 및 다른 컴퓨팅 디바이스들 또는 노드들을 비롯한, 네트워크 내의 하나 이상의 독립형 노드들 상에서 또는 하나 이상의 기존의 노드들의 일부로서 실행되는 논리적 엔티티(예컨대, 소프트웨어, 컴퓨터 실행가능 명령어 등)에 구현될 수 있다. 일 예로서, 서비스 계층 또는 그의 컴포넌트(예컨대, NAMS(100))의 인스턴스는 이하에서 기술되는 도 25c 또는 도 25d에 예시된 일반적인 아키텍처를 갖는 네트워크 노드(예컨대, 서버, 컴퓨터, 게이트웨이, 디바이스 등) 상에서 실행되는 소프트웨어의 형태로 구현될 수 있다.
- [0108] 게다가, 본원에 기술되는 방법들 및 기능들이, 예를 들어, 앞서 기술된 네트워크 및 애플리케이션 관리 서비스와 같은, 서비스들에 액세스하기 위해 SOA(Service Oriented Architecture) 및/또는 ROA(resource-oriented architecture)를 사용하는 M2M 네트워크의 일부로서 구현할 수 있다.
- [0109] 도 25c는, 도 25a 및 도 25b에 예시된 것과 같은 M2M 네트워크 내의 M2M 서버, 게이트웨이, 디바이스, 또는 다른 노드로서 동작할 수 있는, 도 5 내지 도 16, 도 18 내지 도 22, 또는 도 24에 예시된 클라이언트들, 서버들, 또는 프로세스들 중 하나와 같은, 네트워크의 노드의 예시적인 하드웨어/소프트웨어 아키텍처의 블록도이다. 도 25c에 도시된 바와 같이, 노드(30)는 프로세서(32), 송수신기(34), 송신/수신 요소(36), 스피커/마이크로폰(38), 키패드(40), 디스플레이/터치패드(42), 비동식 메모리(44), 이동식 메모리(46), 전원(48), GPS(global positioning system) 칩셋(50), 및 다른 주변기기들(52)을 포함할 수 있다. 노드(30)는 또한, 송수신기(34) 및 송신/수신 요소(36)와 같은, 통신 회로부를 포함할 수 있다. 노드(30)가 일 실시예와 부합한 채로 있으면서 전술한 요소들의 임의의 서브컴비네이션(sub-combination)을 포함할 수 있다는 것을 잘 알 것이다. 이 노드는 본원에 기술되는 NAMS 기능을 구현하는 노드일 수 있다.
- [0110] 프로세서(32)는 범용 프로세서, 특수 목적 프로세서, 종래의 프로세서, DSP(digital signal processor), 복수의 마이크로프로세서들, DSP 코어와 연관된 하나 이상의 마이크로프로세서들, 제어기, 마이크로컨트롤러, ASIC(Application Specific Integrated Circuit)들, FPGA(Field Programmable Gate Array) 회로들, 임의의 다른 유형의 IC(integrated circuit), 상태 머신 등일 수 있다. 프로세서(32)는 신호 코딩, 데이터 처리, 전력 제어, 입출력 처리, 및/또는 노드(30)가 무선 환경에서 동작할 수 있게 하는 임의의 다른 기능을 수행할 수 있다. 프로세서(32)는 송수신기(34)에 결합될 수 있고, 송수신기(34)는 송신/수신 요소(36)에 결합될 수 있다. 도 25c가 프로세서(32)와 송수신기(34)를 개별적인 컴포넌트들로서 도시하고 있지만, 프로세서(32)와 송수신기(34)가 전자 패키지 또는 칩에 하나로 통합될 수 있다는 것을 잘 알 것이다. 프로세서(32)는 애플리케이션 계층 프로그램들(예컨대, 브라우저들) 및/또는 무선 액세스 계층(RAN) 프로그램들 및/또는 통신을 수행할 수 있다. 프로세서(32)는, 예를 들어, 액세스 계층 및/또는 애플리케이션 계층 등에서, 인증, 보안 키 합의, 및/

또는 암호화 동작(cryptographic operation)과 같은 보안 동작들을 수행할 수 있다.

- [0111] 도 25c에 도시된 바와 같이, 프로세서(32)는 그의 통신 회로부(예컨대, 송수신기(34) 및 송신/수신 요소(36))에 결합되어 있다. 프로세서(32)는, 컴퓨터 실행가능 명령어들의 실행을 통해, 노드(30)로 하여금 그에 연결되어 있는 네트워크를 통해 다른 노드들과 통신하게 하기 위해 통신 회로부를 제어할 수 있다. 상세하게는, 프로세서(32)는 본원에(예컨대, 도 5 내지 도 16, 도 18 내지 도 22, 및 도 24에) 그리고 청구항들에 기술된 전송 및 수신 단계들을 수행하기 위해 통신 회로부를 제어할 수 있다. 도 25c가 프로세서(32)와 송수신기(34)를 개별적인 컴포넌트들로서 도시하고 있지만, 프로세서(32)와 송수신기(34)가 전자 패키지 또는 칩에 하나로 통합될 수 있다는 것을 잘 알 것이다.
- [0112] 송신/수신 요소(36)는 M2M 서버들, 게이트웨이들, 디바이스들 등을 비롯한, 다른 노드들에게 신호들을 전송하거나 그들로부터 신호들을 수신하도록 구성될 수 있다. 예를 들어, 일 실시예에서, 송신/수신 요소(36)는 RF 신호들을 전송 및/또는 수신하도록 구성된 안테나일 수 있다. 송신/수신 요소(36)는 WLAN, WPAN, 셀룰러 등과 같은 다양한 네트워크들 및 무선 인터페이스(air interface)들을 지원할 수 있다. 일 실시예에서, 송신/수신 요소(36)는, 예를 들어, IR, UV 또는 가시 광 신호들을 전송 및/또는 수신하도록 구성된 방출기/검출기일 수 있다. 또 다른 실시예에서, 송신/수신 요소(36)는 RF 신호 및 광 신호 둘 다를 전송 및/또는 수신하도록 구성될 수 있다. 송신/수신 요소(36)가 무선 또는 유선 신호들의 임의의 조합을 전송 및/또는 수신하도록 구성될 수 있다는 것을 잘 알 것이다.
- [0113] 그에 부가하여, 송신/수신 요소(36)가 도 25c에서 단일의 요소로서 도시되어 있지만, 노드(30)가 임의의 수의 송신/수신 요소들(36)을 포함할 수 있다. 보다 구체적으로는, 노드(30)는 MIMO 기술을 이용할 수 있다. 이와 같이, 일 실시예에서, 노드(30)는 무선 신호들을 전송 및 수신하기 위한 2개 이상의 송신/수신 요소들(36)(예컨대, 다수의 안테나들)을 포함할 수 있다.
- [0114] 송수신기(34)는 송신/수신 요소(36)에 의해 전송되어야 하는 신호들을 변조하도록 그리고 송신/수신 요소(36)에 의해 수신되는 신호들을 복조하도록 구성될 수 있다. 앞서 살펴본 바와 같이, 노드(30)는 다중 모드 능력들을 가질 수 있다. 이와 같이, 송수신기(34)는 노드(30)가, 예를 들어, UTRA 및 IEEE 802.11과 같은 다수의 RAT들을 통해 통신할 수 있게 하기 위한 다수의 송수신기들을 포함할 수 있다.
- [0115] 프로세서(32)는 비이동식 메모리(44) 및/또는 이동식 메모리(46)와 같은 임의의 유형의 적당한 메모리로부터의 정보에 액세스하고 그에 데이터를 저장할 수 있다. 비이동식 메모리(44)는 RAM(random-access memory), ROM(read-only memory), 하드 디스크, 또는 임의의 다른 유형의 메모리 저장 디바이스를 포함할 수 있다. 이동식 메모리(46)는 SIM(subscriber identity module) 카드, 메모리 스틱, SD(secure digital) 메모리 카드 등을 포함할 수 있다. 다른 실시예들에서, 프로세서(32)는, 서버 또는 홈 컴퓨터 상에와 같이, 노드(30) 상에 물리적으로 위치되지 않은 메모리로부터 정보에 액세스하고 데이터를 그에 저장할 수 있다. 프로세서(32)는 NAMS, 상세하게는 NAMS와 통신하는 기반 네트워크들, 애플리케이션들, 또는 다른 서비스들의 상태를 반영하기 위해 디스플레이 또는 표시자들(42) 상의 조명 패턴들, 영상들, 또는 색상들을 제어하도록 구성될 수 있다. 프로세서(32)는 전원(48)으로부터 전력을 받을 수 있고, 노드(30) 내의 다른 컴포넌트들에게 전력을 분배하고 그리고/또는 전력을 제어하도록 구성될 수 있다. 전원(48)은 노드(30)에게 전력을 공급하기 위한 임의의 적당한 디바이스일 수 있다. 예를 들어, 전원(48)은 하나 이상의 건전지 배터리들(예컨대, 니켈-카드뮴(NiCd), 니켈-아연(NiZn), 니켈 금속 수소화물(NiMH), 리튬 이온(Li 이온) 등), 태양 전지들, 연료 전지들 등을 포함할 수 있다.
- [0116] 프로세서(32)는 또한 노드(30)의 현재 위치에 관한 위치 정보(예컨대, 경도 및 위도)를 제공하도록 구성되는 GPS 칩셋(50)에 결합될 수 있다. 노드(30)가 일 실시예와 부합한 채로 있으면서 임의의 적당한 위치 결정 방법을 통해 위치 정보를 취득할 수 있다는 것을 잘 알 것이다.
- [0117] 프로세서(32)는, 부가의 특징들, 기능 및/또는 유선 또는 무선 연결성을 제공하는 하나 이상의 소프트웨어 및/또는 하드웨어 모듈들을 포함할 수 있는, 다른 주변기기들(52)에 추가로 결합될 수 있다. 예를 들어, 주변기기들(52)은 가속도계, e-나침반(e-compass), 위성 송수신기, 센서, 디지털 카메라(사진 또는 비디오용), USB(universal serial bus) 포트, 진동 디바이스, 텔레비전 송수신기, 핸즈프리 헤드셋, Bluetooth® 모듈, FM(frequency modulated) 라디오 유닛, 디지털 음악 플레이어, 미디어 플레이어, 비디오 게임 플레이어 모듈, 인터넷 브라우저 등을 포함할 수 있다.
- [0118] 도 25d는, 도 25a 및 도 25b에 예시된 것과 같은 M2M 네트워크 내의 M2M 서버, 게이트웨이, 디바이스, 또는 다른 노드로서 동작할 수 있는, 도 5 내지 도 16, 도 18 내지 도 22, 또는 도 24에 예시된 클라이언트들, 서버들,

또는 프록시들과 같은, 네트워크의 하나 이상의 노드들을 구현하는 데 역시 사용될 수 있는 예시적인 컴퓨팅 시스템(90)의 블록도이다. 컴퓨팅 시스템(90)은 컴퓨터 또는 서버를 포함할 수 있고, 주로 컴퓨터 판독가능 명령어들 - 소프트웨어의 형태로 되어 있을 수 있고, 이러한 소프트웨어는 어느 곳에든 또는 어떤 수단에 의해서든 저장되고 액세스됨 - 에 의해 제어될 수 있다. 이러한 컴퓨터 판독 가능 명령어들은 컴퓨팅 시스템(90)으로 하여금 일을 하게 하기 위해 중앙 처리 유닛(central processing unit, CPU)(91) 내에서 실행될 수 있다. 많은 공지된 워크스테이션들, 서버들, 및 개인용 컴퓨터들에서, 중앙 처리 유닛(91)은 마이크로프로세서라고 불리는 단일 칩 CPU에 의해 구현된다. 다른 머신들에서, 중앙 처리 유닛(91)은 다수의 프로세서들을 포함할 수 있다. 코프로세서(81)는 부가의 기능들을 수행하거나 CPU(91)를 보조하는, 주 CPU(91)와 별개인, 임의적인 프로세서이다. CPU(91) 및/또는 코프로세서(81)는, 세션 자격증명들을 수신하는 것 또는 세션 자격증명들에 기초하여 인증하는 것과 같은, E2E M2M 서비스 계층 세션들에 대한 개시된 시스템들 및 방법들에 관련된 데이터를 수신, 생성, 및 처리할 수 있다.

[0119] 동작을 설명하면, CPU(91)는 명령어들을 페치, 디코딩, 및 실행하고, 정보를 컴퓨터의 주 데이터 전송 경로인 시스템 버스(80)를 통해 다른 자원들에게 그리고 그로부터 전송한다. 이러한 시스템 버스는 컴퓨팅 시스템(90) 내의 컴포넌트들을 연결시키고, 데이터 교환을 위한 매체를 정의한다. 시스템 버스(80)는 전형적으로 데이터를 송신하기 위한 데이터 라인들, 주소들을 송신하기 위한 주소 라인들, 및 인터럽트들을 송신하고 시스템 버스를 작동시키기 위한 제어 라인들을 포함한다. 이러한 시스템 버스(80)의 일 예는 PCI(Peripheral Component Interconnect) 버스이다.

[0120] 시스템 버스(80)에 결합된 메모리 디바이스들은 RAM(random access memory)(82) 및 ROM(read only memory)(93)을 포함한다. 이러한 메모리들은 정보가 저장 및 검색될 수 있게 하는 회로부를 포함한다. ROM(93)은 일반적으로 용이하게 수정될 수 없는 저장된 데이터를 포함한다. RAM(82)에 저장된 데이터는 CPU(91) 또는 다른 하드웨어 디바이스들에 의해 판독 또는 변경될 수 있다. RAM(82) 및/또는 ROM(93)에의 액세스는 메모리 제어기(92)에 의해 제어될 수 있다. 메모리 제어기(92)는, 명령어들이 실행될 때, 가상 주소들을 물리 주소들로 변환하는 주소 변환 기능을 제공할 수 있다. 메모리 제어기(92)는 또한 시스템 내에서 프로세스들을 격리시키고 시스템 프로세스들을 사용자 프로세스들로부터 격리시키는 메모리 보호 기능을 제공할 수 있다. 따라서, 제1 모드에서 실행되는 프로그램은 그 자신의 프로세스 가상 주소 공간(process virtual address space)에 의해 매핑된 메모리에만 액세스할 수 있고; 프로세스들 간의 메모리 공유가 설정되어 있지 않으면 다른 프로세스의 가상 주소 공간에 있는 메모리에 액세스할 수 없다.

[0121] 그에 추가하여, 컴퓨팅 시스템(90)은 명령어들을 CPU(91)로부터, 프린터(94), 키보드(84), 마우스(95), 및 디스크 드라이브(85)와 같은, 주변기기들에게 전달하는 일을 책임지고 있는 주변기기 제어기(83)를 포함할 수 있다.

[0122] 디스플레이 제어기(96)에 의해 제어되는, 디스플레이(86)는 컴퓨팅 시스템(90)에 의해 생성된 시각적 출력을 디스플레이하는 데 사용된다. 이러한 시각적 출력은 텍스트, 그래픽, 애니메이션화된 그래픽(animated graphics), 및 비디오를 포함할 수 있다. 디스플레이(86)는 CRT 기반 비디오 디스플레이, LCD 기반 평판 디스플레이, 가스 플라즈마 기반 평판 디스플레이, 또는 터치 패널로 구현될 수 있다. 디스플레이 제어기(96)는 디스플레이(86)에게 송신되는 비디오 신호를 생성하는 데 필요한 전자 컴포넌트들을 포함한다.

[0123] 게다가, 컴퓨팅 시스템(90)은, 컴퓨팅 시스템(90)이 네트워크의 다른 노드들과 통신할 수 있게 하기 위해, 컴퓨팅 시스템(90)을, 도 25a 및 도 25b의 네트워크(12)와 같은, 외부 통신 네트워크에 연결시키는 데 사용될 수 있는, 예를 들어, 네트워크 어댑터(97)와 같은, 통신 회로부를 포함할 수 있다. 통신 회로부는, 단독으로 또는 CPU(91)와 조합하여, 본원에(예컨대, 도 5 내지 도 16, 도 18 내지 도 22, 및 도 24에) 그리고 청구항들에 기술되는 전송 및 수신 단계를 수행하는 데 사용될 수 있다.

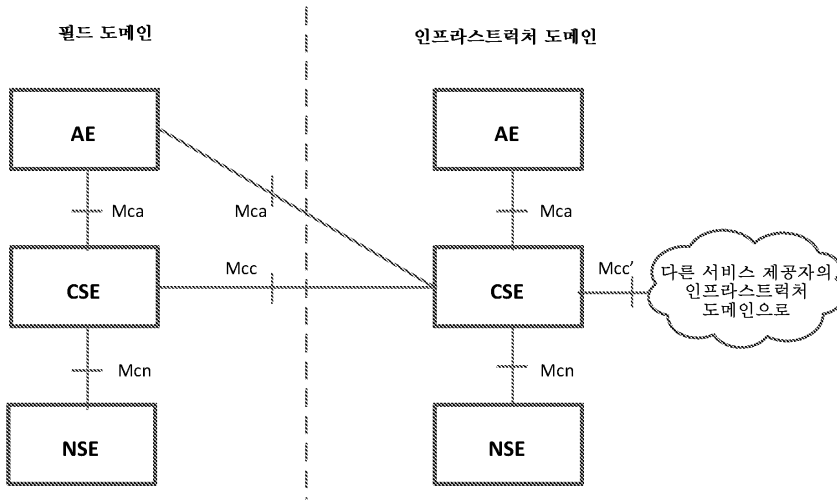
[0124] 본원에 기술되는 방법들 및 프로세스들 중 임의의 것이 컴퓨터 판독가능 저장 매체 상에 저장된 컴퓨터 실행가능 명령어들(즉, 프로그램 코드)의 형태로 구현될 수 있고, 이 명령어들이, 컴퓨터, 서버, M2M 단말 디바이스, M2M 게이트웨이 디바이스 등과 같은 머신에 의해 실행될 때, 본원에 기술되는 시스템들, 방법들 및 프로세스들을 수행 및/또는 구현한다는 것이 이해될 것이다. 구체적으로는, 앞서 기술된 단계들, 동작들 또는 기능들 중 임의의 것이 이러한 컴퓨터 실행가능 명령어들의 형태로 구현될 수 있다. 컴퓨터 판독가능 저장 매체는 정보의 저장을 위해 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 이동식 및 비이동식 매체 모두를 포함하지만, 이러한 컴퓨터 판독가능 저장 매체가 신호들은 포함하지 않는다. 컴퓨터 판독가능 저장 매체는 RAM, ROM, EEPROM, 플래시 메모리 또는 다른 메모리 기술, CD-ROM, DVD(digital versatile disk) 또는 다른 광학 디스크 저장소, 자기 카세트, 자기 테이프, 자기 디스크 저장소 또는 다른 자기 저장 디바이스, 또는 원하는 정보를 저

장하는 데 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 물리 매체를 포함하지만, 이들로 제한되지 않는다.

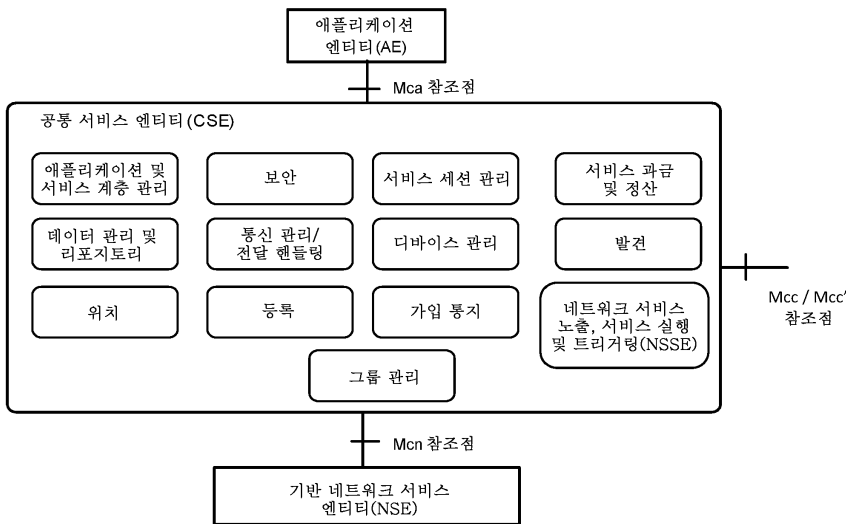
- [0125] 본 개시내용의 발명 요지의 바람직한 실시예들을 기술함에 있어서, 도면들에 예시된 바와 같이, 명확함을 위해 특정 용어가 이용된다. 그렇지만, 청구된 발명 요지는 그렇게 선택된 특정 용어로 제한되는 것으로 의도되어 있지 않으며, 각각의 특정 요소가 유사한 목적을 달성하기 위해 유사한 방식으로 동작하는 모든 기술적 등가물들을 포함한다는 것을 잘 알 것이다.
- [0126] 하기는 이상의 설명에서 나올 수 있는 서비스 레벨 기술들에 관련된 두문자어들의 리스트이다.
- [0127] AE                    Application Entity
- [0128] AS                    Application Server
- [0129] ASN                  Application Service Node
- [0130] CSE                  Common Service Entity
- [0131] CSF                  Common Service Function
- [0132] D2D                  Device to Device communication
- [0133] IE                    Information Element
- [0134] IN-CSE    Infrastructure CSE
- [0135] MIB                  Management Information Base
- [0136] MTC-IWF    Machine Type Communications-InterWorking Function
- [0137] NAMS                  Network and Application Management Service
- [0138] NSE                  (underlying) Network Service Entity
- [0139] NSSE                Network Service Exposure, Service Execution and Triggering
- [0140] NW                    (underlying) Network
- [0141] OSI                  Open Systems Interconnection model
- [0142] RoA                  Resource Oriented Architecture
- [0143] SCS                  Services Capability Server
- [0144] SL                    Service Layer
- [0145] SNMP                Simple Network Management Protocol
- [0146] SoA                  Service Oriented Architecture
- [0147] 이러한 서면 설명은 최상의 실시 형태(best mode)를 비롯한 본 발명을 개시하기 위해 그리고 또한 본 기술분야의 통상의 기술자가, 임의의 디바이스들 또는 시스템들을 제조 및 사용하는 것 그리고 임의의 포함된 방법들을 수행하는 것을 비롯하여, 본 발명을 실시할 수 있게 하기 위해 예들을 사용한다. 본 발명의 특허가능 범주는 청구항들에 의해 정의되고, 본 기술분야의 통상의 기술자에게 안출되는 다른 예들을 포함할 수 있다. 이러한 다른 예들은, 청구항들의 문언적 표현과 상이하지 않은 구조적 요소들을 가지는 경우, 또는 청구항들의 문언적 표현과 그다지 차이를 갖지 않는 등가의 구조적 요소들을 포함하는 경우, 청구항들의 범주 내에 속하는 것으로 의도된다.

도면

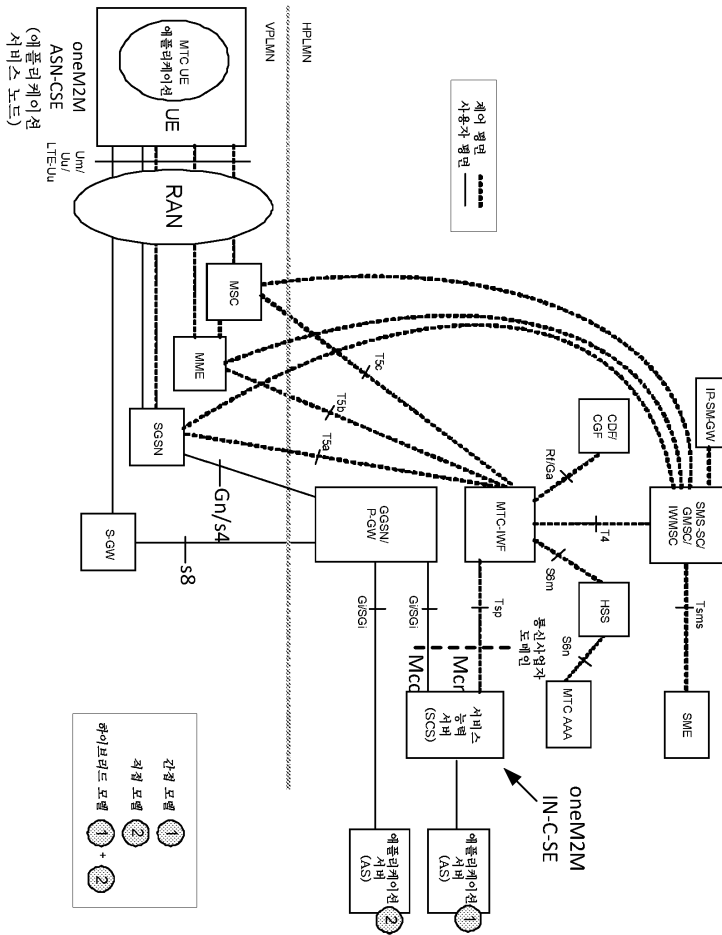
도면1



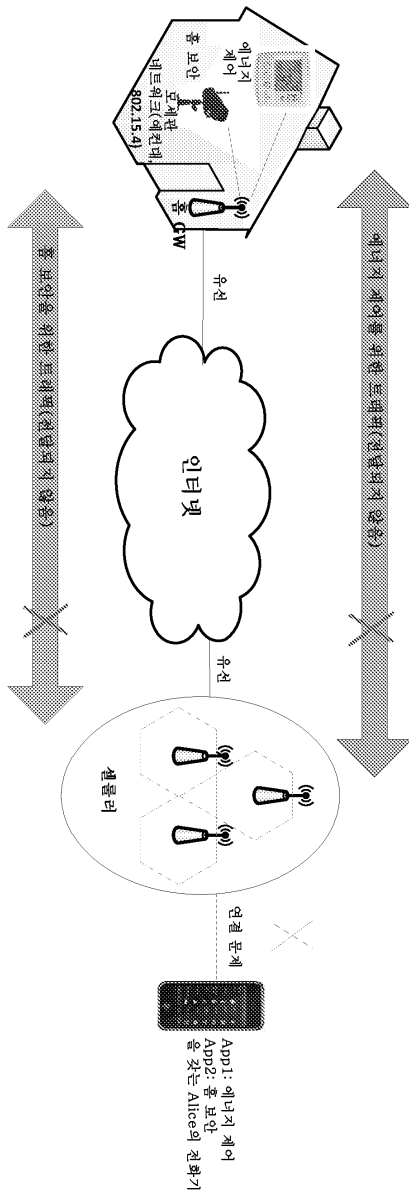
도면2



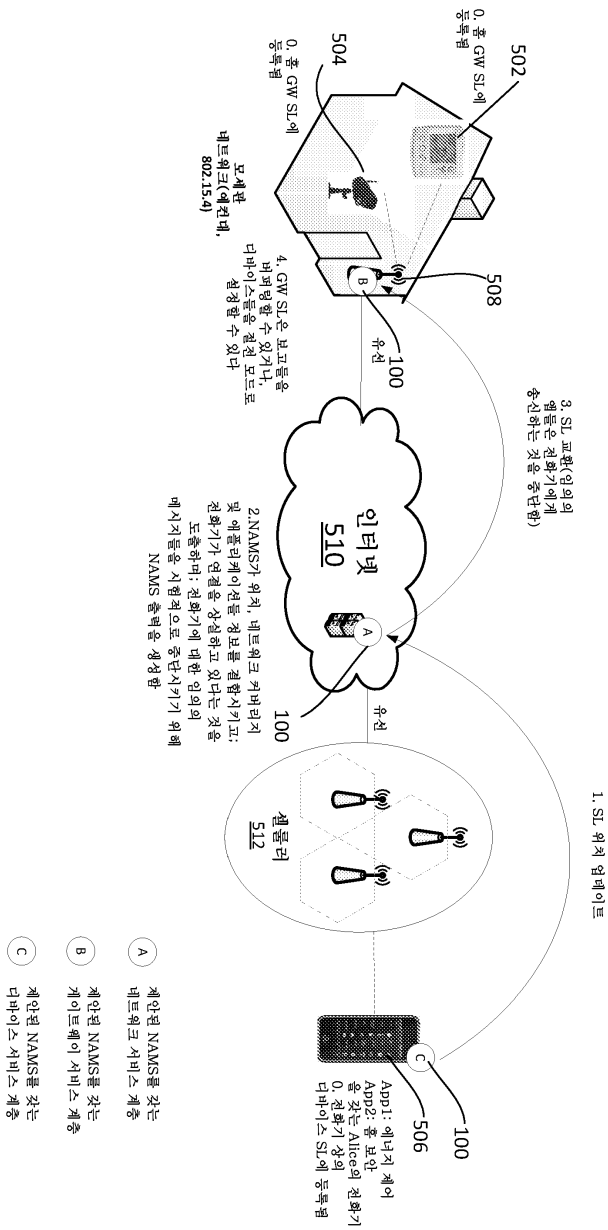
도면3



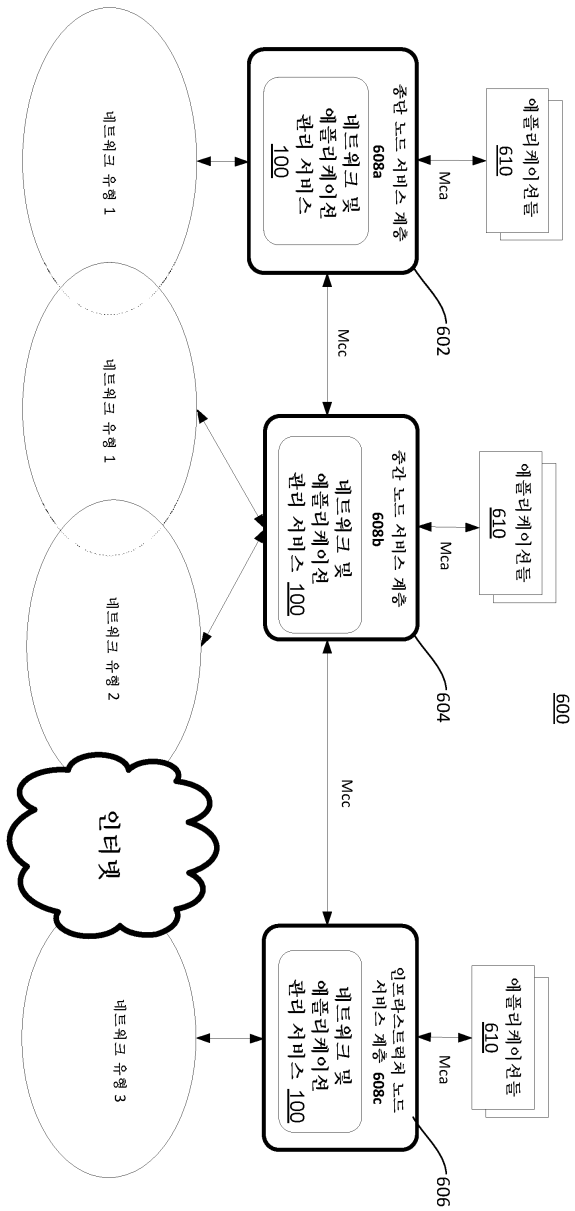
도면4



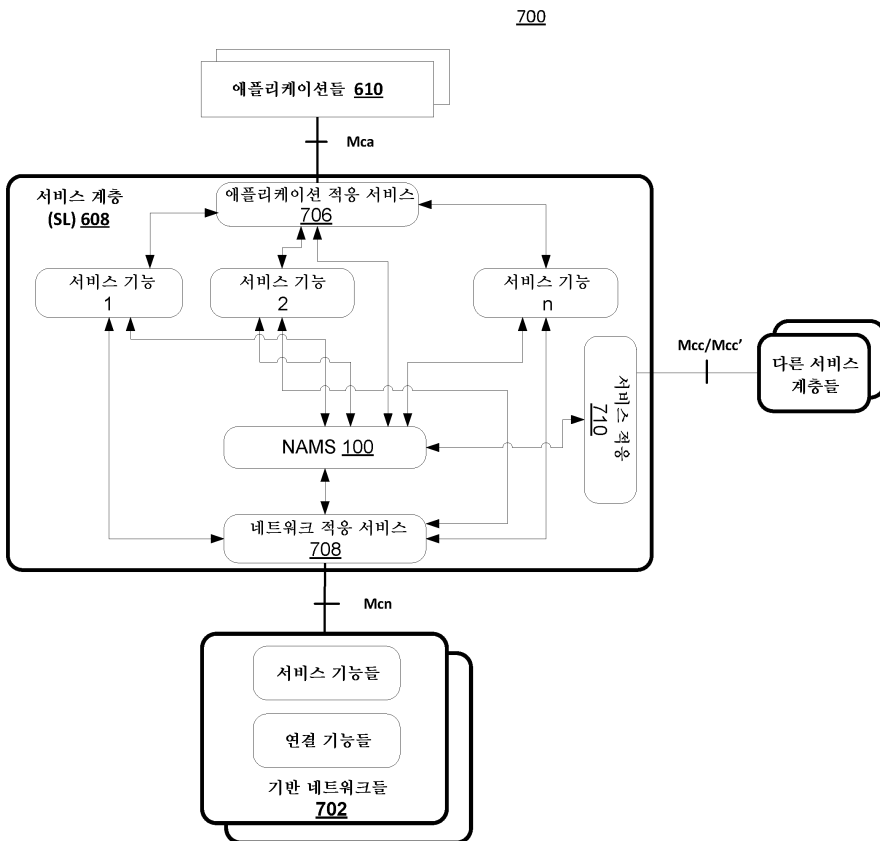
도면5



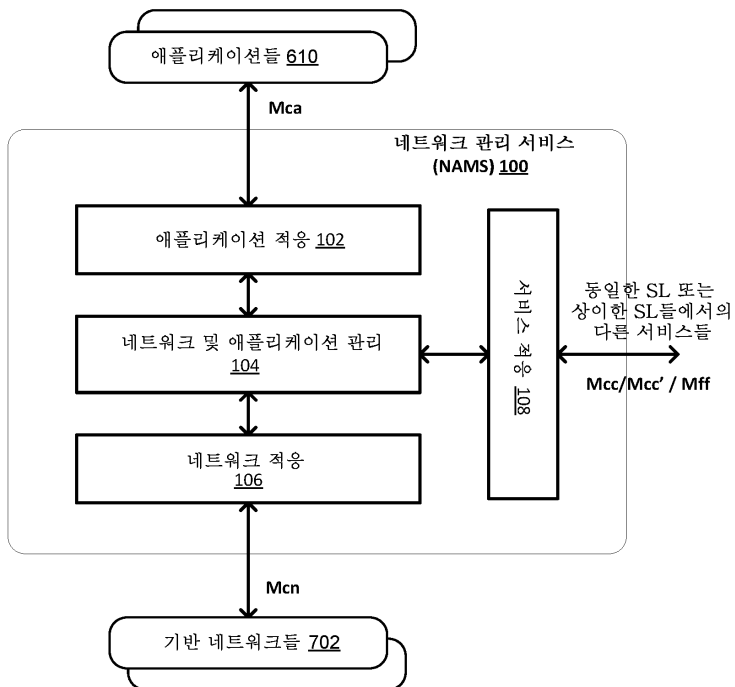
도면6



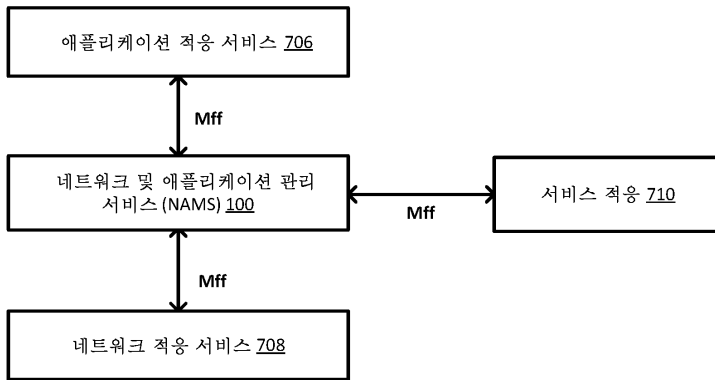
도면7



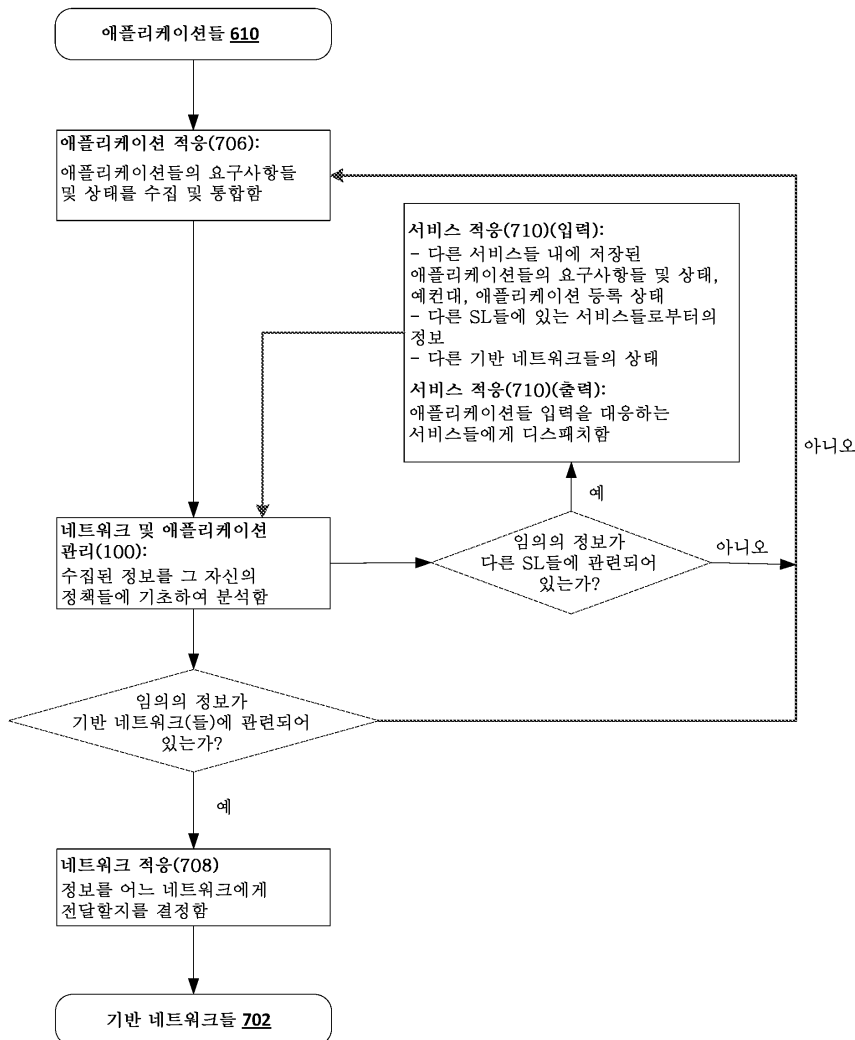
도면8



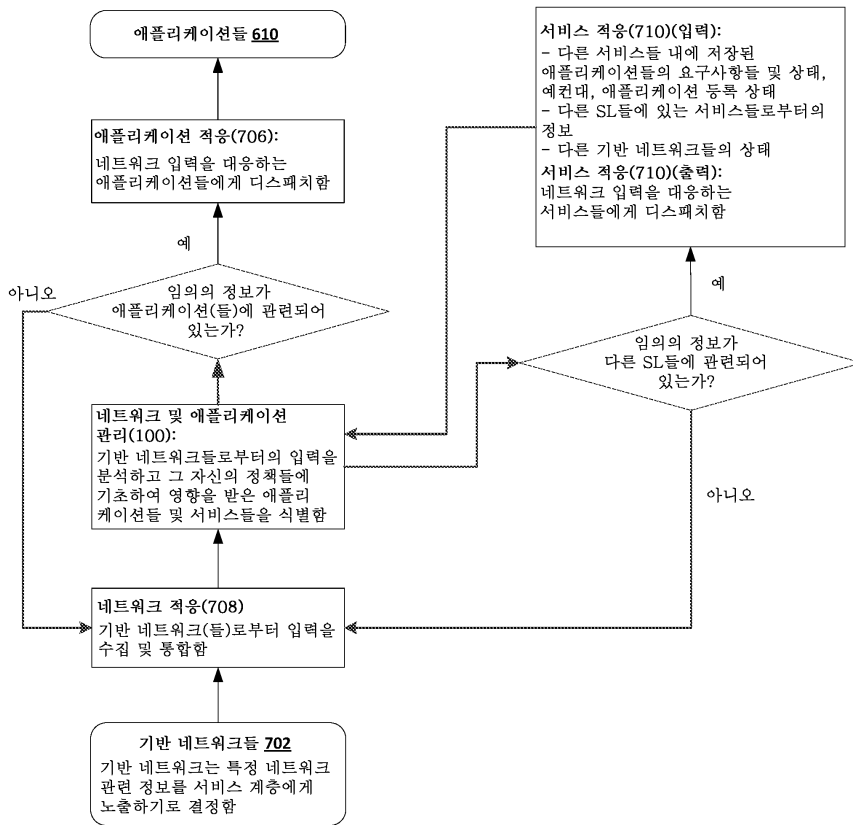
도면9



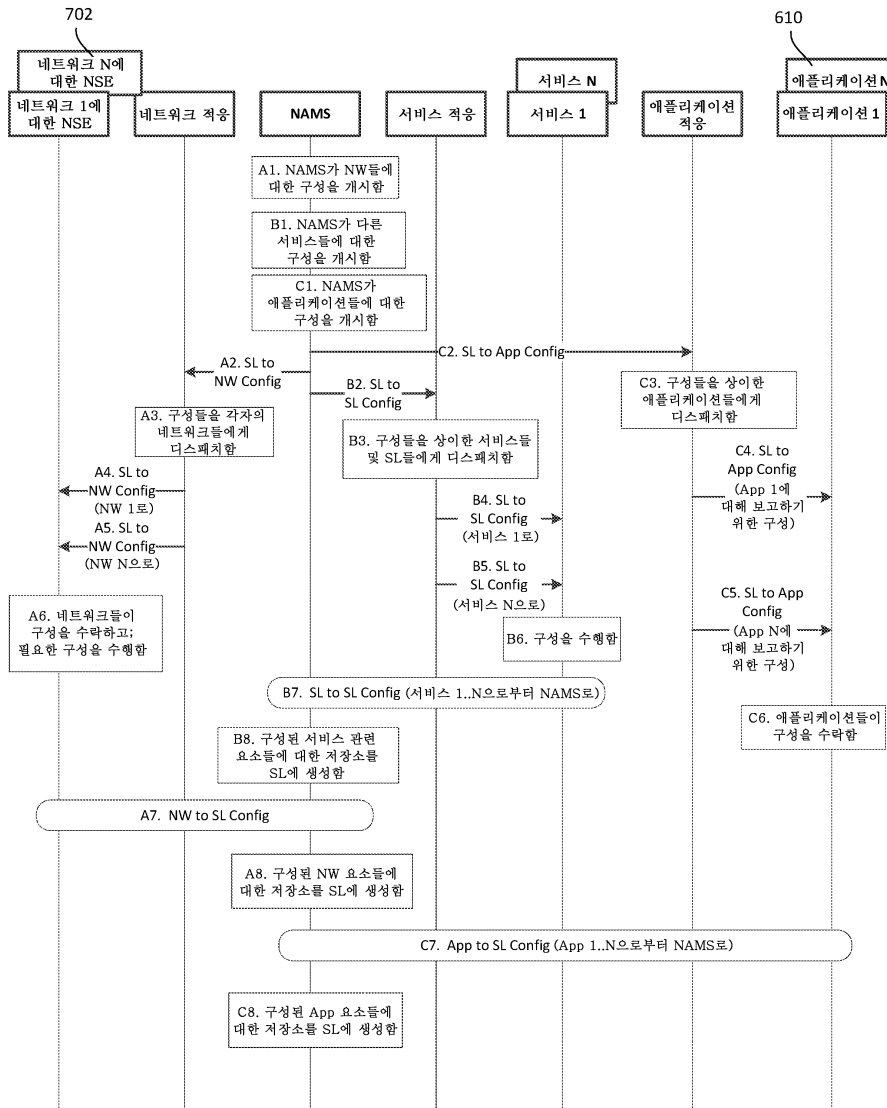
도면10



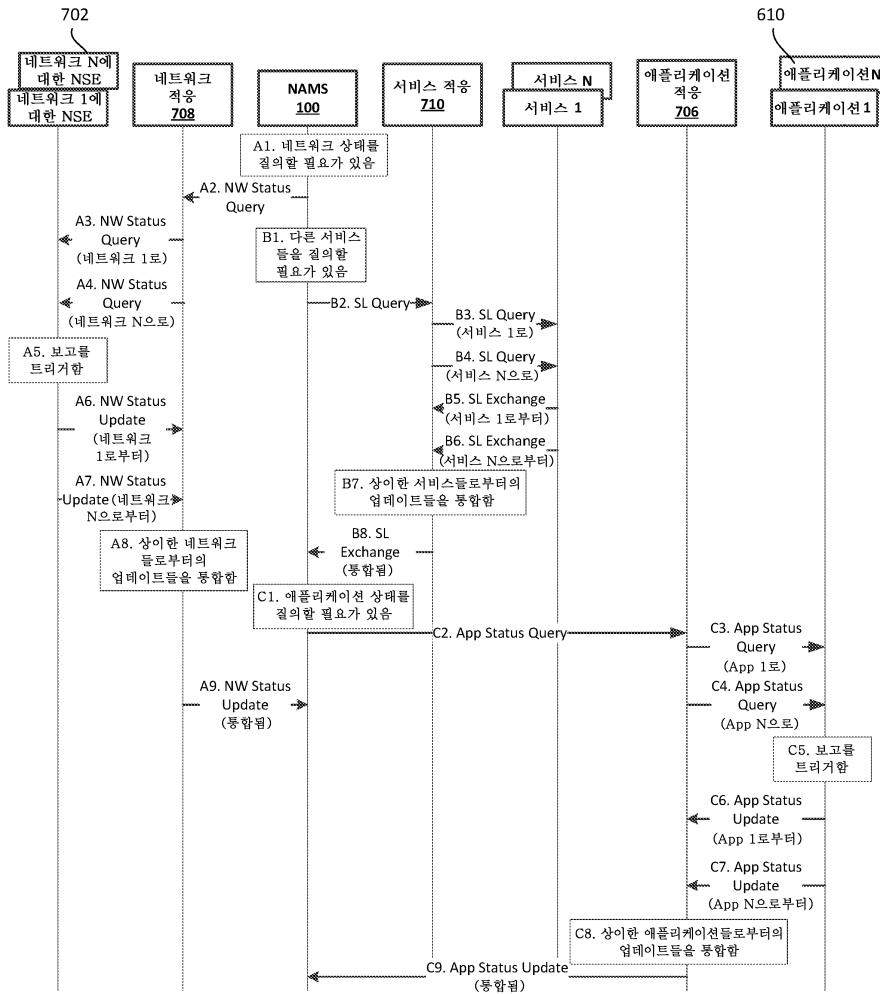
도면11



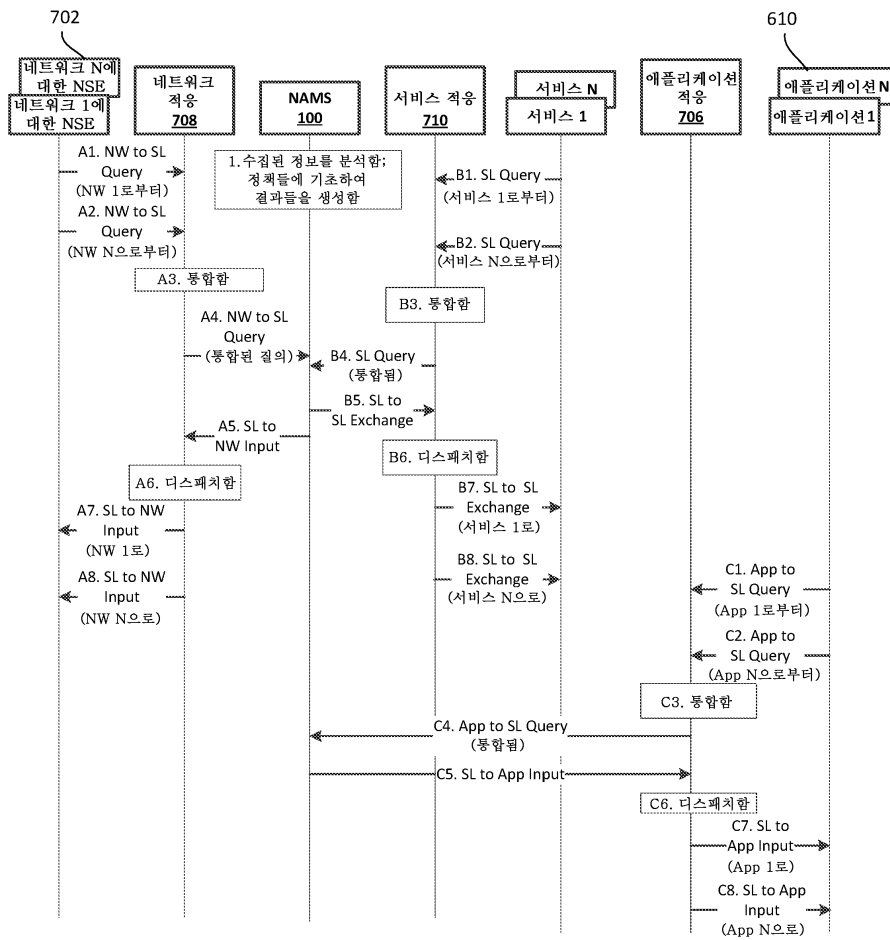
도면12



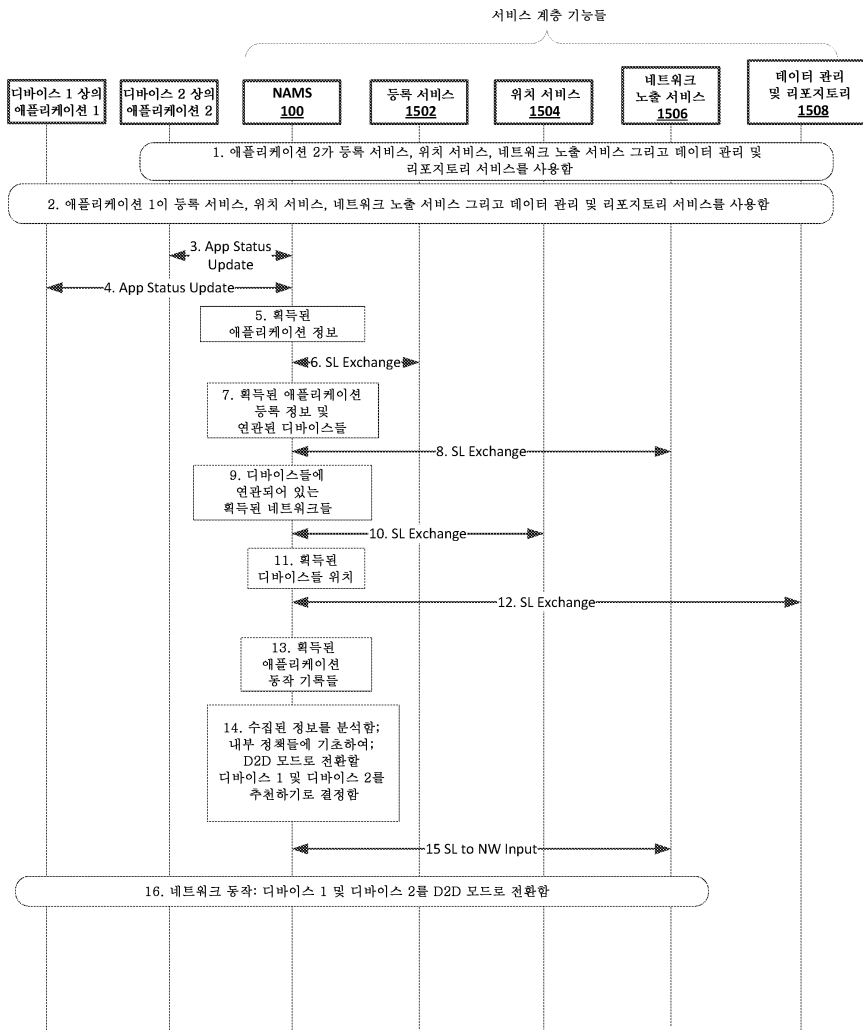
도면13



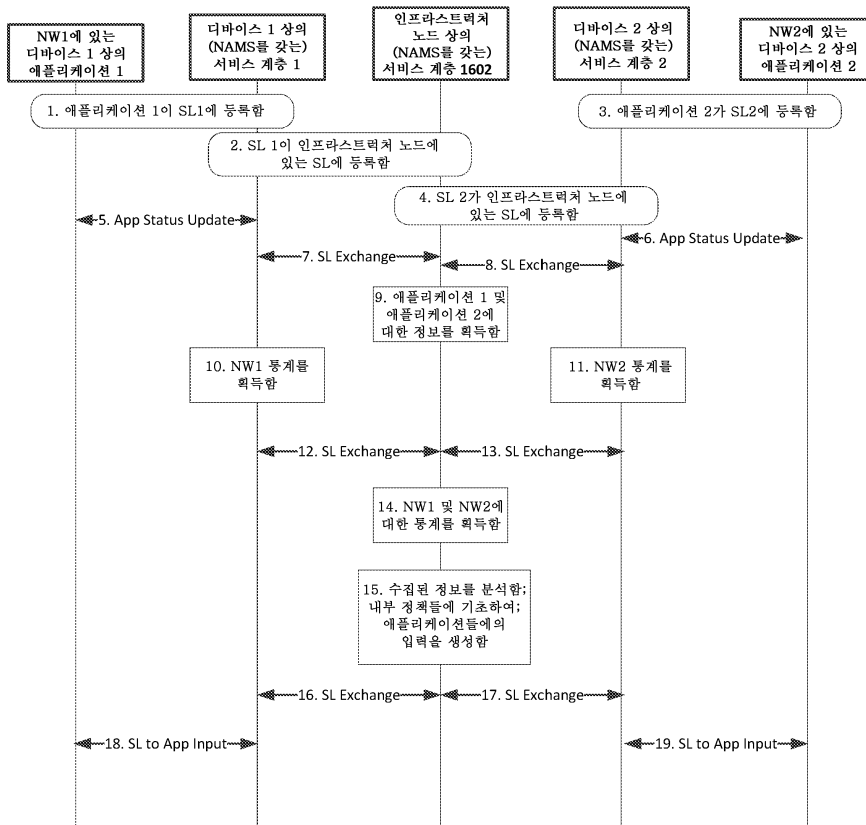
도면14



도면15



도면16

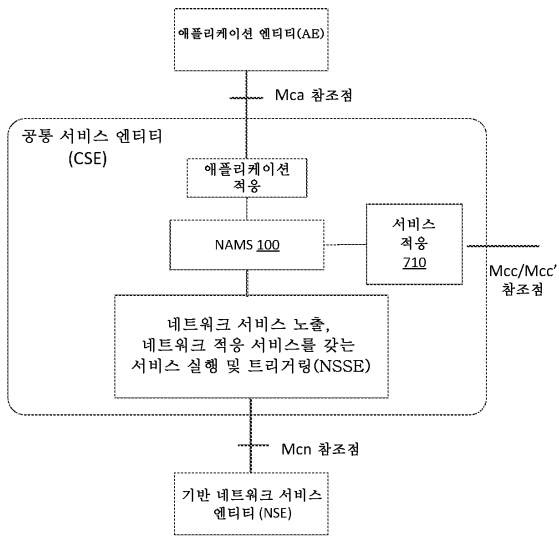


도면17

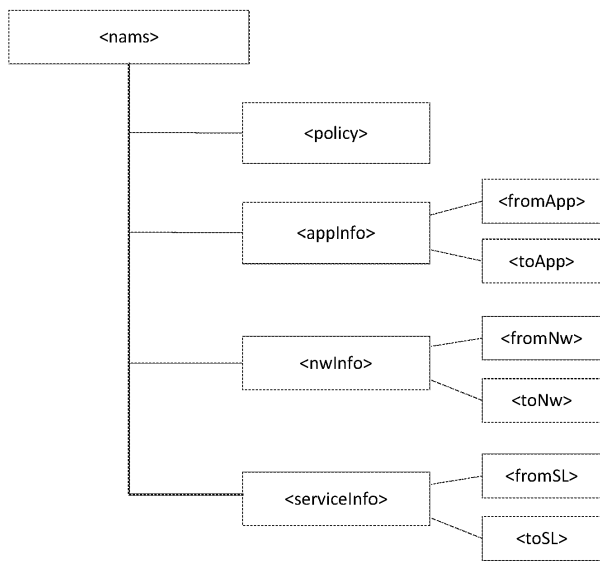
1700

애플리케이션들 1708
서비스 계층 (NAMS ...) 1706
네트워크 관리 1704
애플리케이션 프로토콜 계층 (SNMP ...) 1702
전송 계층 (TCP, UDP ...)
IP
PHY/MAC

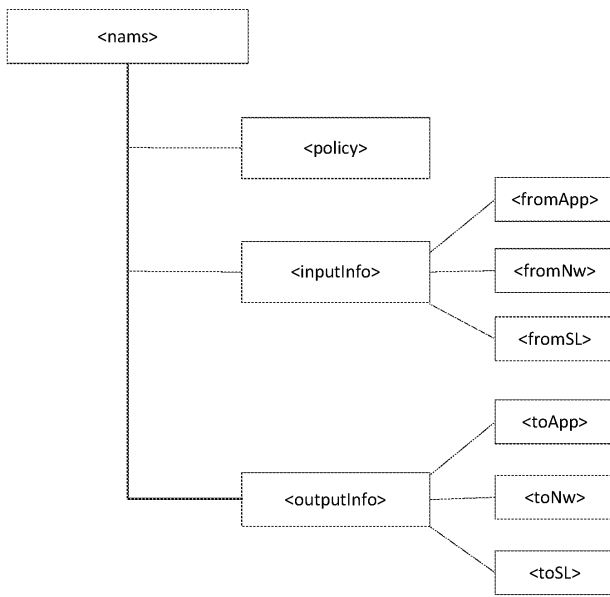
도면18



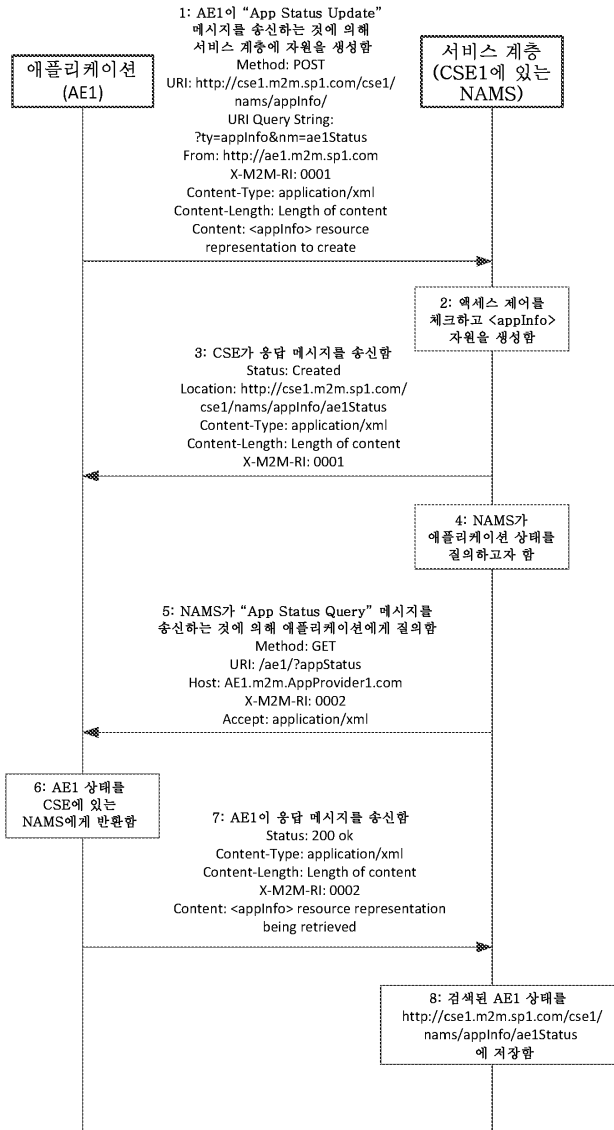
도면19a



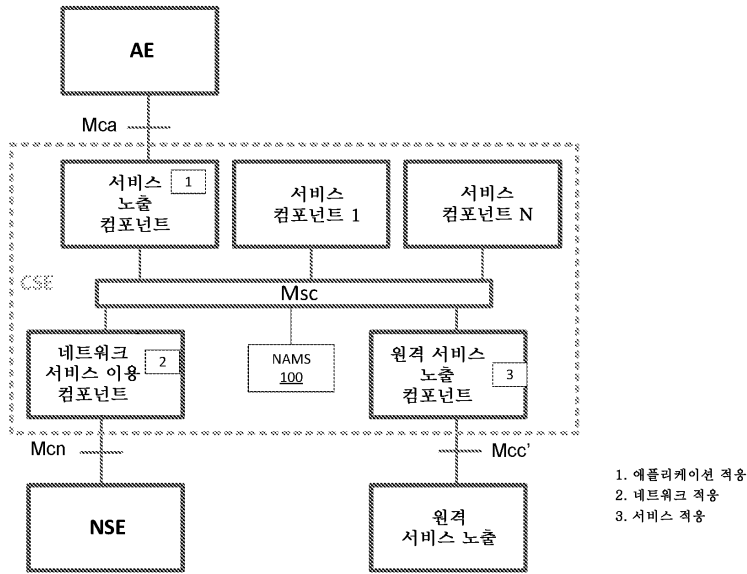
도면19b



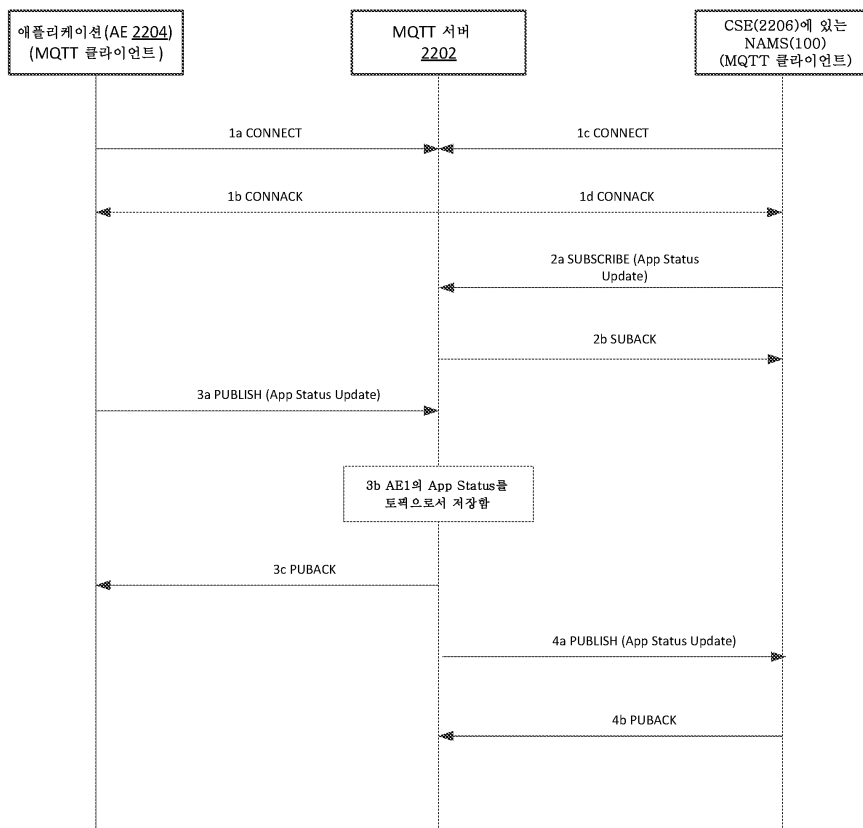
도면20



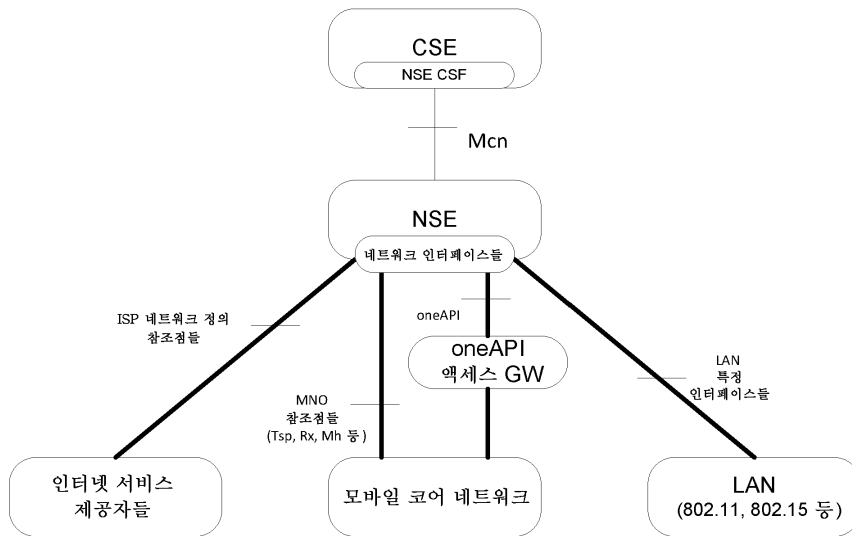
도면21



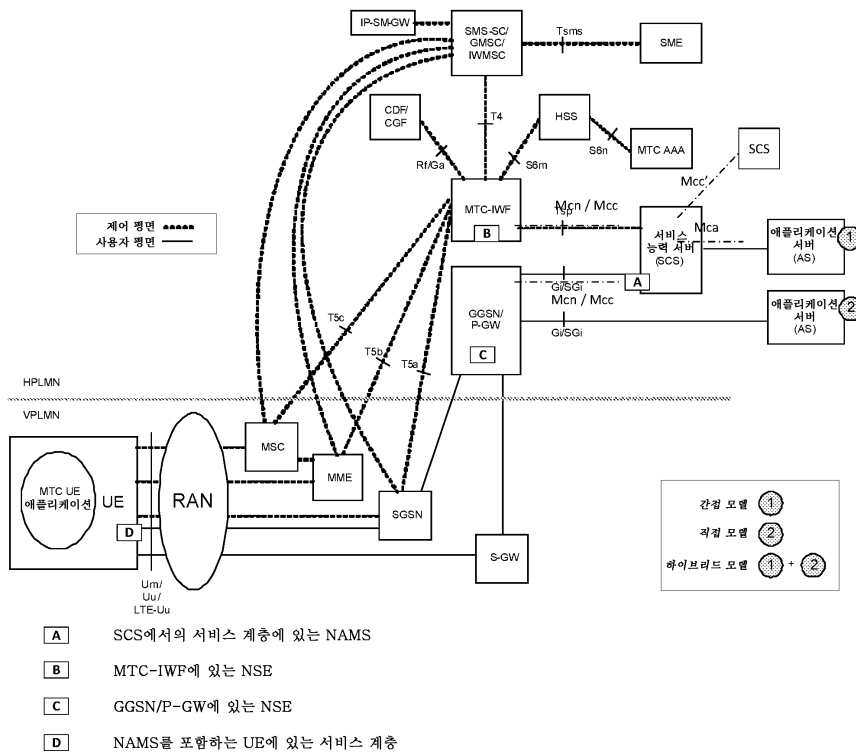
도면22



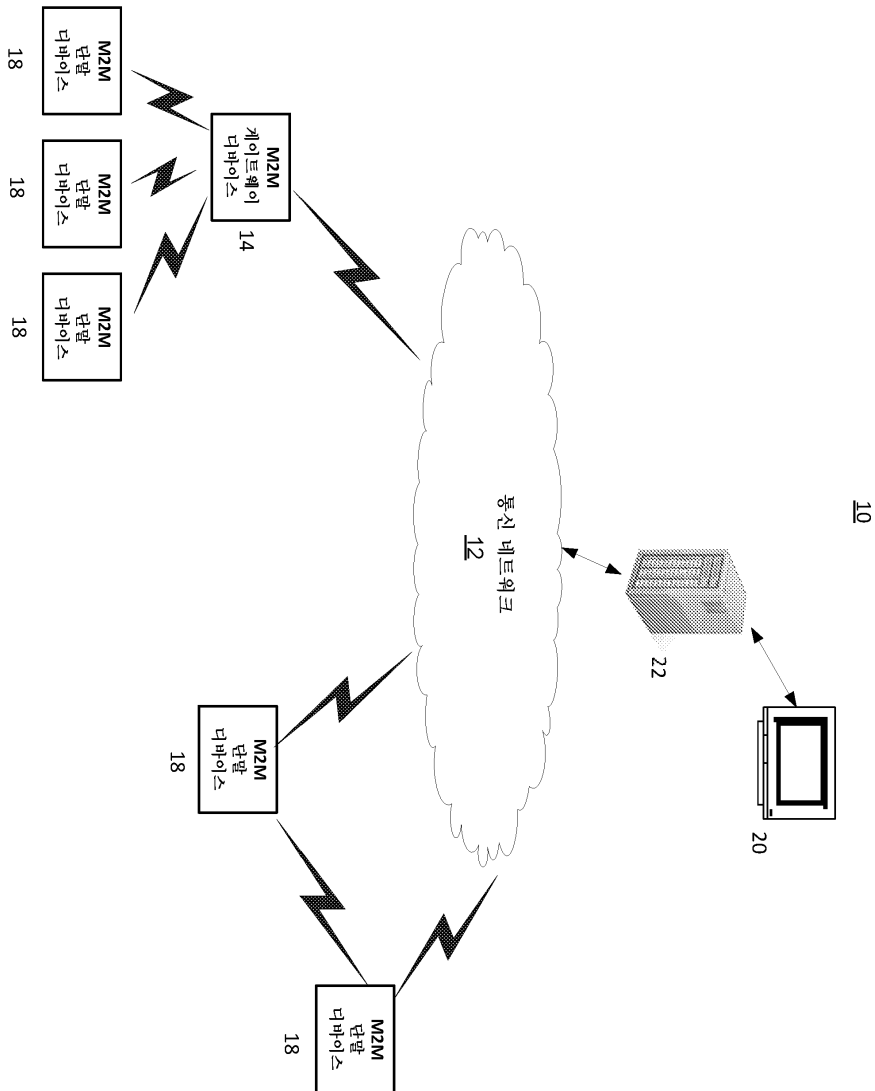
도면23



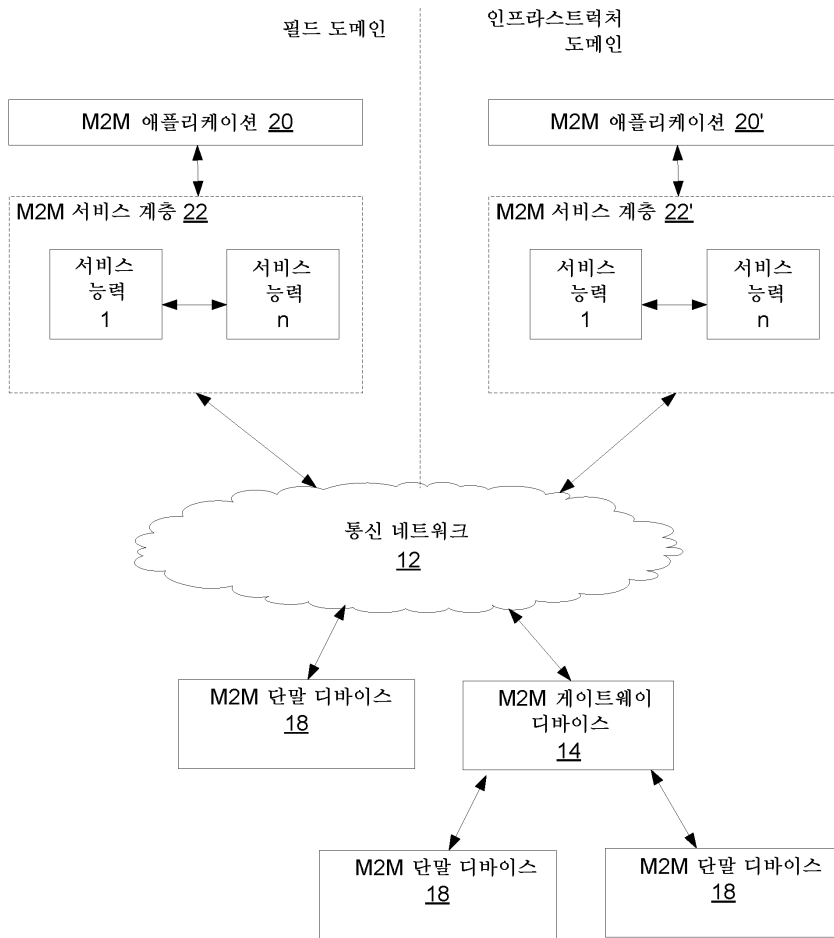
도면24



도면25a



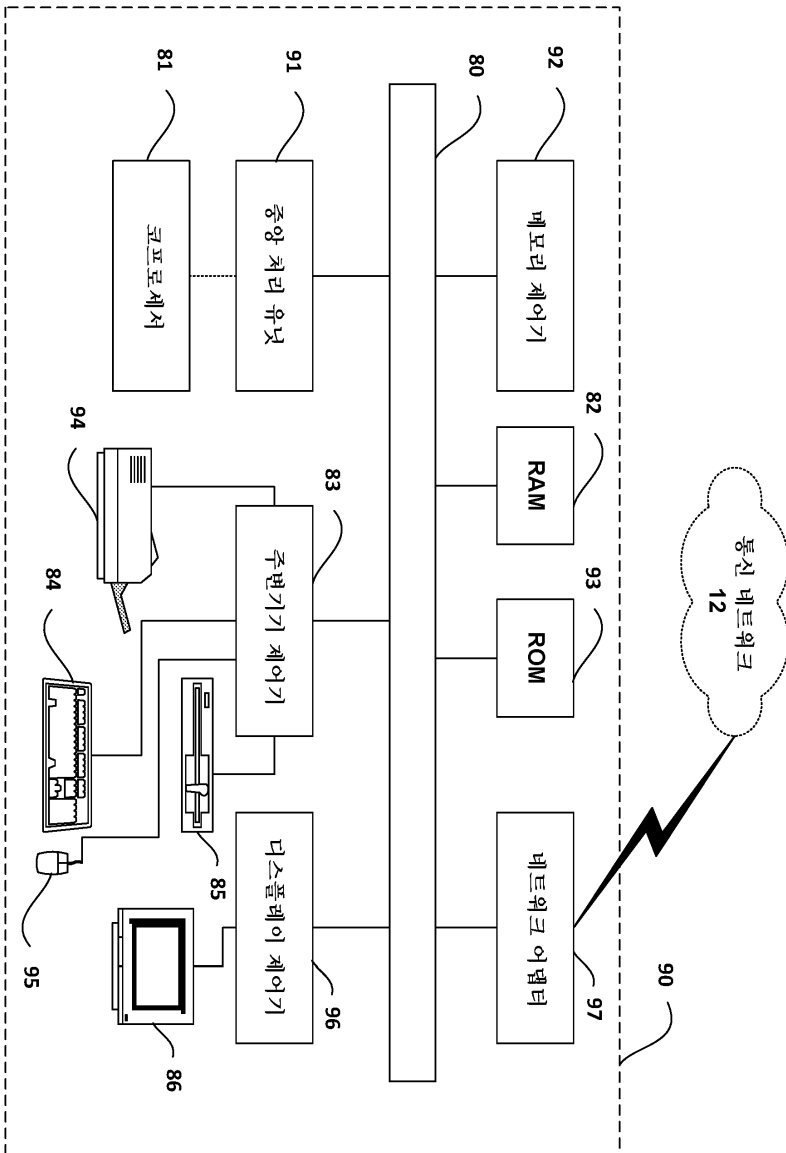
도면25b



도면25c



도면25d



도면26

2600 

아래에서 애플리케이션 특정 정보를 입력하세요: 2602

- 애플리케이션 유형:
- 애플리케이션 ID:
- 애플리케이션 이름:
- 애플리케이션 상태(활성, 휴면):
- 애플리케이션 데이터 유형:
- 애플리케이션 데이터 전송 속도:
- 애플리케이션 데이터 전송 스케줄:
- 서비스 계층 이벤트들에 기초한 애플리케이션 조치들:

**Service Layer Events to Application** 2604

- 보다 나은 네트워크 연결이 이용가능합니다, 전환하겠습니까?
- 네트워크가 혼잡합니다, 애플리케이션 데이터 전송 속도가 감소되어도 괜찮습니까?
- 네트워크가 혼잡합니다, 애플리케이션 데이터가 시각 X로 재스케줄링되어도 괜찮습니까?
- 네트워크가 혼잡합니다, 애플리케이션 데이터 해상도가 감소되어도 괜찮습니까?