



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2007-0107588  
(43) 공개일자 2007년11월07일

(51) Int. Cl.

H04L 12/26(2006.01)

(21) 출원번호 10-2007-0041796

(22) 출원일자 2007년04월30일  
심사청구일자 2007년04월30일

(30) 우선권주장  
06 113 384.9 2006년05월02일  
유럽특허청(EPO)(EP)

(71) 출원인

리서치 인 모션 리미티드

캐나다 온타리오 워털루 필립 스트리트 295 (우편  
번호 엔2엘 3더블유8)

(72) 발명자

센필드 마이클

캐나다 캐나다주 엘4씨 3에쓰9 리치몬드 힐 스톡  
데일 크레슨트 38

스미스 크리스토퍼

캐나다 캐나다주 엘7알 2피3 버링톤 에머랄드 스트리트 601

(74) 대리인

김태홍, 송승필

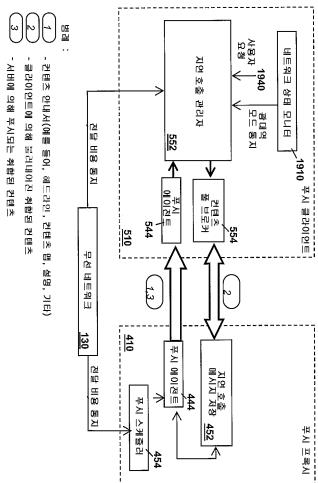
전체 청구항 수 : 총 21 항

#### (54) 동적 신디케이트된 컨텐츠 전달 시스템 및 방법

(57) 요약

동적 신디케이트된 컨텐츠 전달 시스템 및 방법이 제공되며, 이 시스템은, 푸시 프록시와 푸시 클라이언트를 포함하며, 상기 푸시 프록시는, 미래의 전달을 위해 지연된 컨텐츠를 저장하도록 구성되어 있는 지연 호출 메시지 저장 장치, 컨텐츠를 푸시하도록 구성되어 있는 푸시 에이전트, 및 컨텐츠의 푸싱을 스케줄링하기 위해 상기 푸시 에이전트와 통신하도록 구성되어 있고 또한 네트워크 조건에 대하여 무선 네트워크를 모니터링하도록 구성되어 있는 푸시 스케줄러를 포함하고, 상기 푸시 클라이언트는 상기 푸시 프록시의 상기 푸시 에이전트와 통신하도록 구성되어 있는 클라이언트 푸시 에이전트, 상기 푸시 프록시의 상기 지연 호출 메시지 저장 장치와 통신하도록 구성되어 있는 컨텐츠 풀 브로커, 컨텐츠를 풀링하기 위해 상기 컨텐츠 풀 브로커 및 상기 클라이언트 푸시 에이전트와 통신하도록 구성되어 있고 또한 네트워크를 모니터링하고, 상기 네트워크 조건이 상기 지연된 컨텐츠를 수신하기에 적합한 경우 상기 컨텐츠를 풀링하도록 상기 컨텐츠 풀 브로커에 지시하도록 추가로 구성되어 있는 지연 호출 관리자, 상기 네트워크의 상태를 모니터링하도록 구성되어 있는 네트워크 상태 모니터, 및 상기 무선 네트워크를 포함한다.

대표도 - 도19



## 특허청구의 범위

### 청구항 1

동적 신디케이트된 컨텐츠 전달 시스템으로서,

푸시 프록시와;

푸시 클라이언트

를 포함하며,

상기 푸시 프록시는,

미래의 전달을 위해 지연된 컨텐츠를 저장하도록 구성되어 있는 지연 호출 메시지 저장 장치와;

컨텐츠를 푸시하도록 구성되어 있는 푸시 에이전트와;

컨텐츠의 푸싱을 스케줄링하기 위해 상기 푸시 에이전트와 통신하도록 구성되어 있고 또한 네트워크 조건에 대하여 무선 네트워크를 모니터링하도록 구성되어 있는 푸시 스케줄러

를 포함하고,

상기 푸시 클라이언트는,

상기 푸시 프록시의 상기 푸시 에이전트와 통신하도록 구성되어 있는 클라이언트 푸시 에이전트와;

상기 푸시 프록시의 상기 지연 호출 메시지 저장 장치와 통신하도록 구성되어 있는 컨텐츠 풀 브로커와;

컨텐츠를 풀링하기 위해 상기 컨텐츠 풀 브로커 및 상기 클라이언트 푸시 에이전트와 통신하도록 구성되어 있고, 또한 네트워크를 모니터링하며, 상기 네트워크 조건이 상기 지연된 컨텐츠를 수신하기에 적합한 경우 상기 컨텐츠를 풀링하도록 상기 컨텐츠 풀 브로커에 지시하도록 추가로 구성되어 있는 지연 호출 관리자와;

상기 네트워크의 상태를 모니터링하도록 구성되어 있는 네트워크 상태 모니터와;

상기 무선 네트워크

를 포함하는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

### 청구항 2

제1항에 있어서, 상기 네트워크 조건은 전달 비용을 포함하는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

### 청구항 3

제2항에 있어서, 상기 지연된 데이터는 허용가능한 전달 비용을 나타내는 메타데이터를 포함하고,

상기 전달 비용이 상기 허용가능한 전달 비용보다 작은 경우에 상기 네트워크 조건이 적합한 것인 동적 신디케이트된 컨텐츠 전달 시스템.

### 청구항 4

제1항 내지 제3항 중 어느 한 항에 있어서, 상기 네트워크 상태 모니터는 상기 푸시 클라이언트의 상기 네트워크 연결을 모니터링하도록 구성되어 있는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

### 청구항 5

제4항에 있어서, 상기 네트워크 상태는 상기 푸시 클라이언트가 광대역 데이터 연결에 연결되어 있는지 여부를 포함하는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

### 청구항 6

제5항에 있어서, 상기 광대역 데이터 연결은 WiFi 연결, WiMAX 연결, 3G 연결 및 4G 연결로 구성된 그룹으로부터 선택되는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

**청구항 7**

제1항 내지 제6항 중 어느 한 항에 있어서, 상기 지연 호출 관리자는 지연된 컨텐츠를 불러내기 위해 사용자로부터 명령을 수락하도록 추가로 구성되어 있는 것인 동적 신디케이트된 컨텐츠 전달 시스템.

**청구항 8**

보다 적합한 네트워크 조건이 존재할 때까지 푸시 프록시로부터 푸시 클라이언트로의 컨텐츠 전달을 지연시키는 방법으로서,

미래의 전달을 위해 지연된 컨텐츠를 푸시 프록시에 저장하는 단계와;

조건이 전달에 적합한지를 판정하기 위해 상기 푸시 프록시에서 무선 네트워크를 모니터링하는 단계와;

조건이 상기 전달에 적합한 경우, 상기 지연된 컨텐츠를 전달하는 단계를 포함하는 컨텐츠 전달 지연 방법.

**청구항 9**

제8항에 있어서, 상기 조건은 전달 비용을 포함하는 것인 컨텐츠 전달 지연 방법.

**청구항 10**

제9항에 있어서, 상기 지연된 컨텐츠는 허용가능한 전달 비용을 나타내는 메타데이터를 포함하고,

상기 전달 비용이 상기 허용가능한 전달 비용보다 작은 경우에 상기 네트워크 조건이 적합한 것인 컨텐츠 전달 지연 방법.

**청구항 11**

푸시 클라이언트에 의해 푸시 프록시로부터 컨텐츠를 불러내되, 네트워크 조건이 전달에 더 적합할 때까지 상기 컨텐츠가 지연되는 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법에 있어서,

조건이 상기 지연된 컨텐츠를 불러내는데 적합한지 여부를 판정하기 위해 무선 네트워크를 모니터링하는 단계, 및

상기 네트워크 조건이 적합한 경우 상기 지연된 컨텐츠를 상기 푸시 클라이언트로 풀링하는 단계를 포함하는 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

**청구항 12**

제11항에 있어서, 상기 네트워크 조건은 전달 비용을 포함하는 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

**청구항 13**

제12항에 있어서, 상기 지연된 데이터는 허용가능한 전달 비용을 나타내는 메타데이터를 포함하고,

상기 전달 비용이 상기 허용가능한 전달 비용보다 작은 경우에 상기 네트워크 조건이 적합한 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

**청구항 14**

제11항 내지 제13항 중 어느 한 항에 있어서, 상기 푸시 클라이언트의 네트워크 연결을 모니터링하는 단계를 더 포함하는 푸시 프록시로부터 컨텐츠를 불러내는 방법.

**청구항 15**

제14항에 있어서, 상기 네트워크 조건은 상기 푸시 클라이언트가 광대역 데이터 연결에 연결되어 있는지 여부를 포함하는 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

**청구항 16**

제15항에 있어서, 상기 푸시 클라이언트가 상기 광대역 데이터 연결에 연결되어 있는 경우 상기 네트워크 조건

은 상기 모니터링 단계에 적합한 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

### 청구항 17

제14항 내지 제16항 중 어느 한 항에 있어서, 상기 지연된 데이터는 허용가능한 속도 조건을 나타내는 메타데이터를 포함하고,

상기 네트워크의 속도가 상기 허용가능한 속도 조건과 같거나 높은 경우에 상기 네트워크 조건이 적합한 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

### 청구항 18

제15항 또는 제16항에 있어서, 상기 광대역 데이터 연결은 WiFi 연결, WiMAX 연결, 3G 연결 및 4G 연결로 구성된 그룹으로부터 선택되는 것인 푸시 프록시로부터 컨텐츠를 불러내는 방법.

### 청구항 19

제11항 내지 제18항 중 어느 한 항에 있어서, 사용자가 지연된 컨텐츠를 요청하는지 여부를 모니터링하는 단계와;

사용자가 지연된 컨텐츠를 요청한 경우, 상기 지연된 컨텐츠를 폴링하는 단계를 더 포함하는 푸시 프록시로부터 컨텐츠를 불러내는 방법.

### 청구항 20

보다 적합한 네트워크 조건이 존재할 때까지 푸시 프록시로부터 푸시 클라이언트로의 컨텐츠 전달을 지연시키는 컴퓨터 프로그램 제품으로서,

제8항 내지 제10항 중 어느 한 항에 기재된 방법을 구현하는 컴퓨팅 기기, 시스템 또는 장치에 의해 실행가능한 프로그램 코드 수단을 구현하는 컴퓨터 판독가능 매체를 포함하는, 보다 적합한 네트워크 조건이 존재할 때까지 푸시 프록시로부터 푸시 클라이언트로의 컨텐츠 전달을 지연시키는 컴퓨터 프로그램 제품.

### 청구항 21

푸시 클라이언트에 의해 푸시 프록시로부터 컨텐츠를 검색하는 컴퓨터 프로그램 제품으로서,

상기 컨텐츠는 네트워크 조건이 전달에 더 적합할 때까지 지연되고,

상기 컴퓨터 프로그램 제품은, 제11항 내지 제19항 중 어느 한 항에 기재된 방법을 구현하는 컴퓨팅 기기, 시스템 또는 장치에 의해 실행가능한 프로그램 코드 수단을 구현하는 컴퓨터 판독가능 매체를 포함하는, 푸시 클라이언트에 의해 푸시 프록시로부터 컨텐츠를 검색하는 컴퓨터 프로그램 제품.

## 명세서

### 발명의 상세한 설명

#### 발명의 목적

#### 발명이 속하는 기술 및 그 분야의 종래기술

&lt;31&gt;

본 발명의 방법 및 시스템은 일반적으로 모바일 환경에서의 동적 컨텐츠 전달에 관한 것으로서, 상세하게는 아키텍처를 변경함이 없이 애플리케이션 및 컨텐츠 제공자가 추가될 수 있는 일반적인 동적 컨텐츠 전달 아키텍처에 관한 것이다.

&lt;32&gt;

모바일 장치 또는 모바일 사용자 장비(user equipment, UE)의 사용자는 사용자가 그의 모바일 장치에 요구하는 기능 및 사용자가 모바일 장치로부터 데이터에 액세스하는 방식과 관련하여 점점 더 세련되고 있다.

&lt;33&gt;

동적 컨텐츠 전달은 사용자로 하여금, 가서 데이터를 찾아내야만 하기 보다는, 정보 또는 데이터가 사용자에게 푸시되도록 할 수 있게 해준다. 데이터의 예는 주식 시세, 최신 날씨 정보, 최신 교통 정보, 동적 바탕 화면(dynamic wallpaper), 광고, 애플리케이션 또는 사용자에게 바람직한 기타 데이터를 포함할 수 있다.

<34> WAP(wireless application protocol, 무선 애플리케이션 프로토콜) 등의 모바일 장치에 대한 현재의 기술은 컨텐츠를 푸시할 수 있는 기능을 가지고 있지만, WAP은 무선 애플리케이션 프로토콜을 만족시키도록 웹 사이트가 재작성될 것을 요구하며 사이트를 보는 사용자의 능력을 배려하기 위해 변화하지 않는 균일한 사이트를 사용자에게 제공할 것을 요구한다.

<35> 다른 대안들은 SMS 기반 푸시 및 브로드캐스트(SMS based push and broadcast) 또는 셀 브로드캐스트(cell broadcast)를 포함한다. 브로드캐스트의 경우, 전달이 특정의 사용자의 요구 또는 특정의 장치의 성능에 맞춰 커스터마이즈될 수 없다. 따라서, 이들 시스템은 그와 연관된 지능이 없다. 모바일 장치에 보다 나은 해결 방안이 필요하다.

### 발명이 이루고자 하는 기술적 과제

<36> 본 발명의 시스템 및 방법은 양호하게는, 아키텍처를 수정할 필요없이, 일반적인 애플리케이션 및 컨텐츠 제공자가 시스템에 추가될 수 있게 해주는 동적 컨텐츠 전달 아키텍처 및 시스템을 제공한다. 상세하게는, 본 발명의 시스템 및 방법은 모바일 장치가, 애플리케이션이 모바일 장치에 추가될 수 있고 또 컨텐츠가 모바일 장치로 제공될 수 있는 동적 애플리케이션 플랫폼이 될 수 있게 해주며, 이 때 동적 컨텐츠 전달 시스템의 아키텍처는 장치 상에 설치될 수 있는 애플리케이션의 유형이나 장치가 수신하는 컨텐츠의 유형 어느 것도 제한하지 않는다.

<37> 본 발명의 한 측면에서, 양호하게는, 메타데이터가 제공되고 동적 컨텐츠 전달 아키텍처 내의 여러가지 처리 요소를 위해 컨텐츠에 지능(intelligence)을 부가하도록 컨텐츠와 연관되어 있다. 이 아키텍처는 컨텐츠 제공(content provision)과, 푸시 프록시, 무선 네트워크, 푸시 클라이언트 및 클라이언트 애플리케이션을 포함한 서비스 제공(service provision)을 제공하는 논리적 구성요소를 포함한다.

<38> 본 발명의 다른 측면에서, 양호하게는 푸시 컨텐츠 메타데이터에 대한 계층화된 "엔벨로프" 모델(layered enveloped model)에서 메타데이터가 제공된다. 컨텐츠는 푸시 프레임워크 내의 각각의 요소에서의 프로세싱을 위해 사용될 수 있는 메타데이터로 래핑되어 있다. 각각의 연속적인 요소에 대한 메타데이터는 계층화되어 있으며, 그에 의해 처리 요소가 그 요소에 대한 메타데이터만을 추출할 수 있게 해준다. 예를 들어, 푸시 프록시 및 클라이언트 애플리케이션에 보내지는 메타데이터를 포함하는 컨텐츠 패키지는 클라이언트 애플리케이션에 대한 제1 레벨의 메타데이터 및 푸시 프록시에 대한 제2 계층의 메타데이터를 갖는 컨텐츠를 포함할 수 있다. 그에 의해, 엔벨로프가 푸시 프록시에 도달할 때, 푸시 프록시에 대한 메타데이터가 추출되고 컨텐츠에 적용되며, 클라이언트 애플리케이션에 대한 수정된 컨텐츠 및 메타데이터가 추가의 처리 요소로 전달된다.

<39> 본 발명의 다른 측면에서 메타데이터는 정적 메타데이터(본 명세서에서, 채널 메타데이터라고도 함) 및 동적 메타데이터(본 명세서에서 컨텐츠 메타데이터라고도 함)로 분할될 수 있다. 정적 메타데이터는 양호하게는 애플리케이션 및 컨텐츠 제공자 둘다의 등록 시에 설정된다. 그렇지만, 채널 메타데이터는 나중에 설정될 수 있다. 채널 메타데이터는 전달되고 있는 컨텐츠의 유형에 고유한 프로세싱 규칙 및 컨텐츠 유형에 대한 애플리케이션 요구사항을 규정한다.

<40> 동적 메타데이터는, 이와 반대로, 전달되고 있는 특정의 컨텐츠와 연관되어 있다.

<41> 본 발명의 다른 측면에서, 플러그-인 등록 모델(plug-in registration model)은 양호하게는 푸시 프레임워크 내에 제공된다. 일반적인 푸시 클라이언트 및 푸시 프록시가 식별되며, 그 각각은 이들 요소가 컨텐츠 및 메타데이터 둘다를 처리할 수 있게 해주는 여러가지 처리 블록 또는 모듈을 갖는다. 이들 블록은 전달되고 있는 컨텐츠, 전달되고 있는 메타데이터, 또는 전달되고 있는 컨텐츠 및 메타데이터 둘다를 처리하도록 보내질 수 있다.

<42> 플러그-인 등록은 더욱 양호하게는, 컨텐츠 제공자와 애플리케이션 간의 채널 메타데이터의 설정을 가능하게 해주기 위해, 서비스 매니페스트(service manifest) 및 애플리케이션 매니페스트(application manifest)의 전달을 제공한다. 구체적으로는, 서비스 매니페스트는 컨텐츠 제공자를 푸시 프레임워크에 등록시키는데 사용될 수 있으며, 애플리케이션 매니페스트는 애플리케이션을 푸시 프레임워크에 등록시키는데 사용될 수 있다.

<43> 본 발명의 다른 측면에서, 양호하게는, 그의 우선순위에 기초하여 그리고 데이터 전송 비용, 연결된 네트워크의 유형, 또는 사용자의 선호도를 포함한 네트워크 인자에 기초하여 데이터를 처리하는 것을 가능하게 해주는 신디케이트된 컨텐츠(syndicated content)를 푸시하는 방법이 제공된다. 신디케이트된 컨텐츠에 대한 선택적인 혼합형 푸시/풀 모델(mixed push/pull model)은, 네트워크 상황이 유리하게 될 때 푸시 프록시가 컨텐츠를 푸시할 수 있게 해주거나, 네트워크 상황이 유리하게 될 때 또는 사용자가 컨텐츠를 요구할 때 클라이언트가 컨텐츠를

푸시할 수 있게 해준다.

- <44> 여러가지 모바일 장치를 수용하기 위해, 본 발명의 다른 측면은 양호하게는, 비선형 컨텐츠 프래그먼트화를 포함한, 컨텐츠에 대한 컨텐츠 프래그먼트화를 제공한다. 비선형 컨텐츠 프래그먼트화는 컨텐츠를 메타데이터로 보강하여, 데이터가 클라이언트로 전달된 경우, 그 데이터가 재구성될 수 있게 해주는 것을 포함한다.
- <45> 이들 및 다른 측면은 도면과 관련하여 보다 자세히 설명될 것이다.
- <46> 본 발명은 따라서 양호하게는 동적 신디케이트된 컨텐츠 전달 시스템을 제공하며, 이 시스템은 푸시 프록시와 푸시 클라이언트를 포함하며, 상기 푸시 프록시는, 미래의 전달을 위해 지연된 컨텐츠를 저장하도록 구성되어 있는 지연 호출 메시지 저장 장치, 컨텐츠를 푸시하도록 구성되어 있는 푸시 에이전트, 및 컨텐츠의 푸싱을 스케줄링하기 위해 상기 푸시 에이전트와 통신하도록 구성되어 있고 또한 네트워크 조건에 대하여 무선 네트워크를 모니터링하도록 구성되어 있는 푸시 스케줄러를 포함하고, 상기 푸시 클라이언트는 상기 푸시 프록시의 상기 푸시 에이전트와 통신하도록 구성되어 있는 클라이언트 푸시 에이전트, 상기 푸시 프록시의 상기 지연 호출 메시지 저장 장치와 통신하도록 구성되어 있는 컨텐츠 폴 브로커, 컨텐츠를 폴링하기 위해 상기 컨텐츠 폴 브로커 및 상기 클라이언트 푸시 에이전트와 통신하도록 구성되어 있고, 또한 네트워크를 모니터링하고, 상기 네트워크 조건이 상기 지연된 컨텐츠를 수신하기에 적합한 경우 상기 컨텐츠를 폴링하도록 상기 컨텐츠 폴 브로커에 지시하도록 추가로 구성되어 있는 지연 호출 관리자, 상기 네트워크의 상태를 모니터링하도록 구성되어 있는 네트워크 상태 모니터, 및 상기 무선 네트워크를 포함한다.
- <47> 본 발명은 또한 양호하게는 푸시 클라이언트에 의해 푸시 프록시로부터 컨텐츠를 불러내는 방법을 제공하며, 상기 컨텐츠는 네트워크 조건이 전달에 더 적합할 때까지 지연되고, 상기 방법은, 조건이 상기 지연된 컨텐츠를 불러내는데 적합한지 여부를 판정하기 위해 무선 네트워크를 모니터링하는 단계, 및 상기 네트워크 조건이 적합한 경우 상기 지연된 컨텐츠를 상기 푸시 클라이언트로 폴링하는 단계를 포함한다.
- <48> 본 발명은 또한 양호하게는 보다 적합한 네트워크 조건이 존재할 때까지 푸시 프록시로부터 푸시 클라이언트로의 컨텐츠 전달을 지연시키는 방법을 제공하며, 이 방법은, 미래의 전달을 위해 지연된 컨텐츠를 푸시 프록시에 저장하는 단계, 조건이 전달에 적합한지를 판정하기 위해 상기 푸시 프록시에서 무선 네트워크를 모니터링하는 단계, 및 조건이 상기 전달에 적합한 경우, 상기 지연된 컨텐츠를 전달하는 단계를 포함한다.

### 발명의 구성 및 작용

- <49> 본 발명은 첨부 도면을 참조하면 보다 잘 이해될 것이다.
- <50> 이제 도 1을 참조한다. 동적 컨텐츠를 클라이언트 애플리케이션으로 전달하는 일반적인 푸시 시스템이 예시되어 있다. 도 1의 시스템은 간략화된 시스템이며, 동적 컨텐츠 전달 아키텍처에 있어야 하는 논리적 구성요소를 나타내고 있지만, 당업자라면 다른 구성요소가 존재할 수 있거나 여러가지 구성요소가 함께 그룹화되어 있을 수 있음을 잘 알 것이다.
- <51> 아키텍처(100)는 컨텐츠 제공자(110)를 포함한다. 컨텐츠 제공자(110)는 컨텐츠 제공자(110)에 가입되어 있는 사용자에게 동적 컨텐츠를 제공하도록 구성되어 있다. 예로는, 예를 들어, 책을 판매하는 웹 사이트를 포함할 수 있다. 사용자는 지정된 분야 내의 새로 출간된 서적의 리스트를 얻기 위해 컨텐츠 제공자(110)에 등록할 수 있다. 다른 예로는, 무엇보다도 특히, 주기적으로 사용자에게 헤드라인을 제공할 수 있는 뉴스 사이트(news site), 하루 중 어떤 기간 동안에 사용자에게 최신의 교통 정보를 제공할 수 있는 교통 사이트, 생신된 주식 시세 또는 통화 환율을 사용자에게 제공할 수 있는 주식 시장 사이트가 있을 수 있다.
- <52> 이하에 보다 상세히 기술하는 바와 같이, 컨텐츠 제공자(110)는 서비스 제공자의 클라이언트가 컨텐츠 제공자(110)로부터 컨텐츠를 수신할 수 있도록 해주기 위해, 서비스 제공자(120)에 등록을 한다. 서비스 제공자(120)는 클라이언트 또는 클라이언트 애플리케이션에 대해 프록시로서 기능하고 또 컨텐츠 제공자(110)가 컨텐츠를 전송할 목적지를 제공하는 푸시 프록시(122)를 포함한다.
- <53> 서비스 제공자(120)는 무선 네트워크(130)를 통해 모바일 장치 상에 위치하는 푸시 클라이언트(140)와 통신을 한다. 푸시 클라이언트(140)에 대해서는 이하에서 보다 상세히 기술한다. 푸시 클라이언트(140)는 컨텐츠 제공자(110)로부터 전달되고 있는 컨텐츠를 수신하고 이 컨텐츠를, 궁극적으로 컨텐츠를 소비하는 클라이언트 애플리케이션(150)으로 전달할 수 있다.
- <54> 본 명세서에서, 컨텐츠 제공자(110), 서비스 제공자(120), 푸시 프록시(122), 무선 네트워크(130), 푸시 클라이

언트(140) 또는 클라이언트 애플리케이션(150)에 대한 참조는 도 1의 아키텍처를 다시 참조한다.

<55> 도 2를 참조하면, 당업자라면 도 1의 구성요소가 단지 논리적 구성요소이며 반드시 개별적인 물리적 구성요소일 필요는 없다는 것을 잘 알 것이다. 도 1은 하나의 컨텐츠 제공자(110), 하나의 푸시 프록시(122), 하나의 푸시 클라이언트(140) 및 하나의 클라이언트 애플리케이션(150)이 존재하는 일반적인 아키텍처를 나타낸 것이다. 대안의 예가 도 2에 도시되어 있다.

<56> 구체적으로는, 제1 대안의 아키텍처(210)는 푸시 프록시(122)와 통신하는 다수의 컨텐츠 제공자(110)를 포함한다. 도 1의 아키텍처에서와 같이, 푸시 프록시(122)는 무선 네트워크(130)를 통해 푸시 클라이언트(140)와 통신을 한다. 게다가, 아키텍처(210)에 다수의 클라이언트 애플리케이션(150)이 존재한다. 따라서, 이것은 다수의 컨텐츠 제공자(110) 및 다수의 클라이언트 애플리케이션(150)을 갖는 N-1-1-N 시스템이다.

<57> 도 2의 아키텍처(220)는 푸시 프록시(122)와 통신을 하고 또 그에 등록되어 있는 하나의 컨텐츠 제공자(110)를 포함한다. 게다가, 푸시 프록시(122)는 무선 네트워크(130)를 통해 다수의 푸시 클라이언트(140)와 통신을 한다. 각각의 푸시 클라이언트(140)는 클라이언트 애플리케이션(150)과 통신을 한다. 따라서, 아키텍처(220)는 클라이언트 애플리케이션(150) 및 푸시 클라이언트(140)의 논리적 구성요소를 그룹화하며 N(1-1)-1-1 시스템이다.

<58> 도 2의 아키텍처(230)는 각각이 컨텐츠 제공자(110)와 통신하는 다수의 푸시 프록시(122)를 갖고 있다. 각각의 푸시 프록시 및 컨텐츠 제공자 조합(232)은 무선 네트워크(130)를 통해 일반적인 푸시 클라이언트(140)와 통신을 하고, 이 푸시 클라이언트는 이어서 클라이언트 애플리케이션(150)과 통신을 한다. 이것은 1-1-N(1-1) 시스템이다.

<59> 도 2의 아키텍처(240)에서, 컨텐츠 제공자(110) 및 푸시 프록시(122) 그룹(232)은 무선 네트워크(130)를 통해 일반적인 푸시 클라이언트(140) 및 클라이언트 애플리케이션(150) 조합과 통신을 한다. 따라서, 이것은 N(1-1)-N(1-1) 시스템이다.

<60> 당업자라면 잘 알고 있는 바와 같이, 다른 대안들이 가능하다. 이상에서, 개별적인 물리적 구성요소일 수 있거나 함께 그룹화되어 있을 수 있는 여러가지 논리적 구성요소를 나타내었다. 예를 들어, 푸시 클라이언트가 애플리케이션에 내장될 수 있거나, 공통의 공유 클라이언트가 다수의 애플리케이션에 의해 사용될 수 있거나, 다른 대안이 있을 수 있다.

<61> 이제부터, 도 3을 참조한다. 시스템에 지능을 부가하기 위해, 컨텐츠는 메타데이터와 연관된다. 이 경우에, 메타데이터는 컨텐츠를 처리하기 위해 처리 요소에 의해 사용될 수 있는 데이터로서 정의된다. 잘 알고 있는 바와 같이, 일반적인 푸시 시스템은 여러가지 컨텐츠 제공자 및 애플리케이션이 그 시스템 내에 존재할 수 있도록 해주기 위해 메타데이터를 필요로 한다. 이 메타데이터는, 프로세싱 파라미터 또는 규칙, 또는 직접 제공되는 프로세싱 핸들러, 코드 또는 레퍼런스, 또는 다른 장소에 있는 프로세싱 핸들러, 코드 또는 규칙에 대한 링크를 포함한, 여러가지 형태로 되어 있을 수 있다.

<62> 도 3에서 알 수 있는 바와 같이, 컨텐츠는 컨텐츠 제공자(110)로부터 클라이언트 애플리케이션(150)으로 전달되며, 화살표(310)로 나타내어져 있다. 아키텍처(100) 내의 여러가지 구성요소에 명령어를 제공하는 메타데이터는 통상 컨텐츠와 함께, 아키텍처(100) 내의 구성요소들 간에 또한 전달될 수 있다. 예를 들어, 화살표(320)는 컨텐츠 제공자에서 발생하는 메타데이터를 나타낸 것이며, 그 메타데이터는 클라이언트 애플리케이션(150)에 도달할 때까지 전달 시스템에 대하여 투명하다.

<63> 화살표(330)는 컨텐츠 제공자(110)에 의해 생성되어 푸시 클라이언트(140)로 보내지는 메타데이터를 나타낸 것으로서, 따라서 일반적인 푸시 클라이언트(140)로만 플로우한다.

<64> 화살표(340)는 서비스 제공자(120)에 의해 생성되어 푸시 클라이언트(140)로 보내지는 메타데이터를 나타낸 것으로서, 따라서 푸시 프록시(122)에서 컨텐츠 먼저 연관되고 일반적인 푸시 클라이언트(140)에서 컨텐츠로부터 제거된다. 이것이 일어날 수 있는 경우의 예는 과금 계획(billing plan) 및 제공될 서비스의 레벨에 관한 사용자와 서비스 제공자 간의 합의를 포함하며, 여기서 서비스 제공자는 이용가능 서비스를 제한하거나 향상된 서비스를 제공하기 위해 메타데이터를 사용할 수 있다.

<65> 메타데이터의 흐름 및 메타데이터의 역할에 대해 이하에서 보다 상세히 기술한다.

<66> 이하, 도 4를 참조한다. 도 4는 본 시스템 및 방법과 관련하여 사용될 수 있는 예시적인 푸시 프록시(410)를 상세히 나타낸 것이다. 당업자라면 잘 알고 있는 바와 같이, 푸시 프록시(410)는 도 1 및 도 2의 푸시 프록시와

동일할 수 있다.

- <67> 푸시 프록시(410)는 푸시 프록시(410)가 일반적인 푸시 환경에서 동작할 수 있게 해주는 여러가지 구성요소를 포함한다. 이것은 유연성을 촉진시키는데, 그 이유는 푸시 프록시가 특정의 컨텐츠 제공자 또는 푸시 클라이언트와의 상호작용으로 제한되지 않고 그 대신에 동적 환경에 적응될 수 있기 때문이다. 푸시 프록시(410)에 대해 이하에 기술되는 구성요소들은 양호하게는 푸시 프록시(410) 내에 들어 있지만, 이들 구성요소는 전부 망라한 것이 아니며 다른 구성요소들이 있을 수 있다. 게다가, 어떤 구성요소들이 푸시 프록시(410)로부터 생략될 수 있으며, 나머지 구성요소들이 여전히 일반적인 푸시 서비스를 수행할 수 있다.
- <68> 푸시 프록시(410)는 그에 등록되어 있는 컨텐츠 제공자(412)를 포함한다. 컨텐츠 제공자(412)는 컨텐츠 제공자 등록 서비스 제공자 인터페이스(SPI)(420)에 등록한다. 이하에서 보다 상세히 기술하는 바와 같이, 이 등록에서 컨텐츠 제공자(412)가, 본 명세서에서 채널 메타데이터라고 하는, 설정되고 있는 채널에 대한 어떤 정보를 포함하고 있는 것이 바람직하다. 컨텐츠 제공자(412)는 도 1의 컨텐츠 제공자(110)와 동일할 수 있다.
- <69> 푸시 프록시(410)는 또한 푸시 프록시 서비스를 관리하는 서비스 관리 블록(430)을 포함한다.
- <70> 푸시 프록시(410)는 컨텐츠 및 그 컨텐츠와 연관된 메타데이터 둘다를 처리하는 여러가지 모듈을 포함한다. 제1 모듈은 컨텐츠 제공자(412)로부터의 메시지를 소모하여 컨텐츠 전달 큐(content delivery queue)를 관리하는 서브시스템인 메시지 브로커 및 전달 큐(440)이다. 당업자라면 잘 알고 있는 바와 같이, 모든 클라이언트 애플리케이션에 대한 모든 컨텐츠가 한번에 전달될 수 있는 것은 아니며, 큐를 적당한 때에 전달하기 위해 전달 큐가 구축될 필요가 있다. 예를 들어, 장치가 서비스 영역을 벗어나 있을 수 있고 컨텐츠가 저장될 필요가 있을 수 있다.
- <71> 푸시 프록시(410)는 또한 흐름 제어 관리 블록(442)을 포함한다. 흐름 제어 관리 블록(442)은 컨텐츠 흐름의 제어를 가능하게 해준다. 예를 들어, 제한된 공간을 갖는 이동국은 일정량의 정보를 수신하기만 할 수 있다. 이 경우, 모바일 장치는, 도 1에 나타낸 푸시 클라이언트(140)를 통해, 푸시 클라이언트(140)로의 데이터의 흐름을 중단시키도록 푸시 프록시(410)에 요청할 수 있다. 흐름 제어 관리 블록(442)이 이것을 처리한다.
- <72> 다른 대안으로서, 모바일 장치가 오프라인일 수 있다. 흐름 제어 관리 블록(442)은, 푸시 프록시(410)에 의해 수신되는 컨텐츠가 전달될 수 없을 때, 푸시 클라이언트(140)로의 데이터의 흐름을 중단 및 시작할 수 있다.
- <73> 푸시 프록시(410)의 또 다른 구성요소는 푸시 에이전트(push agent)(444)이다. 푸시 에이전트(444)는 데이터를 클라이언트로 전송하는 일을 맡고 있다.
- <74> 당업자라면 잘 알고 있는 바와 같이, 블록(440, 442, 444)은 메시징만을 처리하며, 메타데이터와 관련되어 있지 않다. 환언하면, 이들 블록은 메시지의 내용을 처리하고 메시지와 연관된 메타데이터를 처리하지 않는다.
- <75> 푸시 프록시(410)의 다른 구성요소는 컨텐츠 메타데이터 추출기 및 캐쉬 블록(450)이다. 컨텐츠 메타데이터 추출기 및 캐쉬 블록(450)은 내포된 컨텐츠 메타데이터(enveloped content metadata)에 작용한다. 구체적으로는, 이하에서 보다 상세히 기술하게 되는 메타데이터 시스템의 엔벨로프 모델(envelope model)에서, 시스템 내의 각각의 논리적 구성요소는 컨텐츠 프로세싱과 연관된 메타데이터를 가질 수 있다. 이 메타데이터는 논리적 구성요소로 하여금 컨텐츠에 대해 동작을 수행할 수 있게 해준다. 따라서, 각각의 논리적 구성요소는 그와 연관된 메타데이터를 추출할 수 있어야만 한다.
- <76> 컨텐츠 메타데이터 추출기 및 캐쉬 블록(450)은 푸시 프록시(410)와 연관된 메타데이터를 추출하고 이 메타데이터를 캐싱하는 일을 맡고 있다. 캐싱 기능은 동일한 컨텐츠 제공자로부터의 차후의 컨텐츠 엔벨로프로 동일한 메타데이터를 전달할 필요가 없게 해줌으로써 최적화를 가능하게 해준다. 메타데이터의 추출 및 캐싱에 대해 이하에서 기술한다.
- <77> 컨텐츠 또는 그의 일부분을 클라이언트 애플리케이션으로 전달하는 것이 효과적이 아닐 때, 지연 호출 메시지 저장 블록(deferred retrieval message store block)(452)이 사용된다. 지연 호출 메시지 저장 블록(452)은, 컨텐츠를 전송하는 것이 효과적일 때까지 또는 컨텐츠가 클라이언트에 의해 폴링(pull)될 때까지, 클라이언트에 저장되지 않는 컨텐츠를 저장하는데 사용될 수 있다. 지연 호출 메시지 저장은 또한, 이미 전달된 컨텐츠에 걸친 클라이언트 애플리케이션 내비게이션에 따라, 선택적으로 클라이언트에 전송될 수 있거나 클라이언트에 의해 폴링될 수 있는 보조 컨텐츠를 캐싱하는데 사용될 수 있다.
- <78> 지연 호출 메시지 저장 블록(452)의 목적에 대해 도 19 및 도 21을 참조하여 이하에서 더 설명한다. 예로서, 지연 호출 메시지 저장 블록(452)은 사용자가, 사용자의 위치에 가까운 레스토랑 등의, 위치 정보를 요청한 경우

에 사용될 수 있다. 컨텐츠 제공자는 광고자가 호출 요청에 그의 정보를 추가하기 위해 지불할 수 있는 경우 정보를 제공하는 모델을 가질 수 있다. 따라서, 위치에 대한 레스토랑 정보를 요청하고 있는 사용자는 또한 그의 요청에 수반된, 그의 위치에 가까운 상점, 골프 코스, 체육관 또는 다른 서비스에 관한 정보를 가질 수 있다. 컨텐츠 제공자는 요청된 레스토랑 정보를 부가 정보와 묶어서 이를 푸시 프록시(410)로 전달한다.

<79> 푸시 프록시(410)는, 제공된 메타데이터에 기초하여, 클라이언트로 전송할 컨텐츠 패키지를 생성할 수 있다. 컨텐츠 패키지는 클라이언트에 의해 요청된 정보는 물론 사용자가 관심을 가질 수 있는 관련 정보의 개요 또는 요약을 포함할 수 있다. 이 요약이 사용자에게 전송되지만, 지역 호출 메시지 저장 블록(452)은 컨텐츠 제공자(110)로부터 수신된 실제 데이터를 저장한다. 따라서, 미래에 사용자가 정보에 관한 더 많은 상세 정보를 개요내에서 획득하고자 하는 경우, 이 정보는 이미 푸시 프록시(410)에 저장되어 있다.

<80> 지역 호출 메시지 저장 블록(452)의 또 다른 용도는 사용자가 전체 컨텐츠를 한꺼번에 받을 수 없는 경우이다. 예를 들어, 모든 컨텐츠를 장치로 전송하는 것이 실행가능하거나 경제적이지 않는 경우, 컨텐츠의 일부가 나중에 클라이언트에 의해 풀링될 수 있을 때까지 저장될 수 있거나 미리 정의된 규칙이 만족될 때 푸시될 수 있다. 이들 규칙은 어떤 네트워크 또는 서비스 조건이 충족됨으로써 네트워크 또는 서비스 조건에 의해 규정될 수 있다. 이것에 대해 이하에서 도 19를 참조하여 보다 상세히 기술한다.

<81> 푸시 스케줄러(454)는 클라이언트에 대한 전달 슬롯을 스케줄링한다. 상기한 바와 같이, 어떤 상황에서는, 컨텐츠 전부를 한꺼번에 푸시하는 것이 효율적이지 않을 수 있다. 푸시 스케줄러(454)는 어떤 정보를 즉시 푸시하고 나머지를 미리 정의된 스케줄에 따라 푸시할 것인지를 결정할 수 있다. 또한, 푸시 스케줄러(454)는 컨텐츠가 언제 푸시되어야만 하는지를 결정하기 위해 컨텐츠의 속성을 사용할 수 있다. 구체적으로는, 메타데이터는 어떤 컨텐츠가 높은 우선순위를 가지거나 시간이 제한된 만료를 가짐을 표시할 수 있고, 이 컨텐츠는 즉시 푸시될 수 있는 한편, 낮은 우선순위를 갖거나 만료가 없는 것으로 표시된 컨텐츠는 나중에 데이터를 전달하기 위한 조건이 더 좋은 때에 푸시될 수 있다.

<82> 당업자라면 잘 알고 있는 바와 같이, 블록(450, 452, 454)은 메시지의 내용 및 메시지와 연관된 메타데이터 둘다를 처리한다.

<83> 가입 및 규칙 블록(460)은 서비스를 받기 위해 등록되어 있는 애플리케이션을 추적하고 전달되고 있는 특정의 컨텐츠를 어떻게 처리해야 하는지에 관한 규칙을 모니터링한다. 컨텐츠는 일반적으로 클라이언트에 의한 가입에 기초하여 또는 클라이언트를 위해 전달된다. 예를 들어, 사용자가 특정의 서비스를 원하는 경우에, 사용자는 능동적으로 가입을 요청할 수 있다. 예를 들어, 사용자가 서비스에 대한 혜택을 받기 위해 그의 서비스 제공자(120)와의 계약에 서명한 경우, 사용자를 위해 가입이 이루어질 수 있다. 이것은, 사용자가 매일 일정수의 광고를 반기로 합의하는 한, 사용자가 양호한 등급을 받은 경우를 포함할 수 있다. 이 경우에, 서비스 제공자(120)는 클라이언트를 위해서 광고 제공자에 가입을 할 수 있다.

<84> 애플리케이션이 모바일 장치 상에서 삭제되는 경우 또는 애플리케이션이 가입에 등록되어 있지 않은 경우, 가입 및 규칙 블록(460)은 그 사용자의 가입 등록을 취소시킬 수 있다.

<85> 컨텐츠 의존성 블록(content dependencies block)(462)은 모바일 장치 사용자가 이용할 수 있는 서비스를 광고하기 위해 푸시 블록(410)에 의해 사용된다. 따라서, 모바일 장치 사용자가 서비스에 충분한 스크린 또는 대역폭 또는 메모리를 가지고 있지 않은 경우, 컨텐츠 의존성 블록(462)은 사용자에 대한 그 서비스의 광고를 차단시킬 수 있다.

<86> 컨텐츠 프래그먼트화 블록(content fragmentation block)(464)은 컨텐츠를 프래그먼트화하는데 사용된다. 이것은, 예를 들어, 모바일 장치가 컨텐츠 전부를 한꺼번에 받을 수 없는 경우에 사용될 수 있다. 컨텐츠 프래그먼트화 블록(464)은 컨텐츠를 여러가지 컴포넌트로 쪼개기 위해 사용된다. 이는 아직 전달되지 않은 프래그먼트화된 컨텐츠를 저장하기 위해 지역 호출 및 메시지 저장 블록(452)과 관련하여 사용될 수 있다.

<87> 컨텐츠 만료 및 교체 블록(content expiry and replacement block)(466)은 2가지 목적을 위해 사용된다. 첫째, 이 블록은 가입을 모니터링하기 위해 사용될 수 있다. 각각의 가입은 만료 시간을 가지며, 이 만료 시간이 만족될 때, 가입이 종료될 수 있다.

<88> 또한, 컨텐츠 만료 및 교체 블록(466)은 정보를 모니터링하기 위해 사용될 수 있다. 어떤 컨텐츠는 정보의 유효성에 관한 시간 제한을 갖는다. 예를 들어, 혼잡 시간대 교통(rush hour traffic)을 모니터링하는데 사용되는 교통 애플리케이션은 매우 시간 의존적이다. 어떤 이유로 푸시 프록시(410)가 모바일 장치로 즉시 컨텐츠를 전

달할 수 없는 경우, 이 컨텐츠는 미래의 전달을 위해 컨텐츠 저장 장치(480)에 저장된다. 그렇지만, 컨텐츠가 어떤 지정된 기간 내에 전달되지 않는 경우, 그 컨텐츠는 만료되어 전혀 전달되지 않을 수 있다.

- <89> 이와 유사하게, 컨텐츠 교체는 정보가 갱신되고 있는 상황을 처리한다. 예를 들어, 주식 시세를 수신하고 있는 클라이언트 애플리케이션은 마지막 주식 시세만을 원할 수 있다. 따라서, 푸시 프록시(410)가 푸시 클라이언트(140)로 주식 시세를 전달할 수 있고 또 차후의 주식 시세가 컨텐츠 제공자(110)로부터 수신되는 경우, 차후의 주식 시세 내의 메타데이터는 이 차후의 주식 시세가 이전의 주식 시세를 대체하는데 사용되어야만 함을 나타낼 수 있다. 모든 정보를 전달 큐에 추가하지 않고 저장된 정보를 대체하는 것은 컨텐츠 저장 장치(480) 내의 공간을 비우게 한다.
- <90> 채널 메타데이터 저장소(470)는 채널 메타데이터를 저장하는데 사용되며, 이에 대해서는 이하에서 보다 상세히 기술한다.
- <91> 이상에서 본 명세서에서의 방법 및 시스템에서 사용될 수 있는 예시적인 푸시 프록시(410)에 대해 기술하였다. 푸시 프록시(410)의 블록 및 구성요소는 푸시 프록시(410)가, 컨텐츠의 유형 및 애플리케이션에서의 컨텐츠의 처리가 변할 수 있고 미리 정해져 있지 않은 일반적인 동적 컨텐츠 전달 시스템에서 사용될 수 있게 해준다.
- <92> 이제부터, 도 5를 참조한다. 도 5는 본 명세서에서의 시스템 및 방법과 관련하여 사용될 수 있는 푸시 클라이언트(510)를 나타낸 것이다. 푸시 클라이언트(510)는 도 1 및 도 2의 푸시 클라이언트(140)와 동일할 수 있다.
- <93> 당업자라면 잘 알고 있는 바와 같이, 컨텐츠 및 컨텐츠의 처리가 미리 정해지지 않은 일반적인 시스템에서 사용되어야 하는 푸시 클라이언트(510)는 컨텐츠 및 컨텐츠와 연관된 메타데이터 둘다에 대처하기 위해 사용될 수 있는 블록 또는 모듈을 포함해야만 한다. 도 5와 관련하여 정의된 블록들은 전부 망라한 것이 아니며, 푸시 클라이언트(510) 내에 다른 블록들도 존재할 수 있다. 게다가, 푸시 클라이언트(510) 내의 블록들은, 어떤 경우에, 푸시 클라이언트(510) 내의 나머지 블록들의 기능을 제한함이 없이 생략될 수 있다.
- <94> 푸시 클라이언트(510)는 애플리케이션을 서비스하며, 하나 이상의 애플리케이션(512)이 푸시 클라이언트(510)에 등록할 수 있다. 애플리케이션 등록은 등록을 위한 인터페이스로서 애플리케이션 제공자 인터페이스(514)를 사용하며, 애플리케이션 제공자 인터페이스(514)는 또한, 이하에서 보다 상세히 기술하는 바와 같이, 애플리케이션에 대한 채널 메타데이터를 추출하는데 사용될 수 있다.
- <95> 푸시 클라이언트(510)는 푸시 클라이언트(510)를 관리하는데 사용되는 클라이언트 관리(client administration)(520)를 포함한다.
- <96> 도 4의 푸시 서버(410)에서와 같이, 푸시 클라이언트(510)는 메시징을 처리하는 여러가지 블록, 메타데이터를 처리하는 여러가지 블록, 및 메시징 및 메타데이터 둘다를 처리하는 여러가지 블록을 포함한다.
- <97> 메시지 브로커 및 애플리케이션 큐(message broker and application queue)(540)는 애플리케이션(512)으로 전달하기 위해 푸시 프록시(410)로부터의 메시지를 처리한다. 애플리케이션 큐는 애플리케이션(512)에 대한 메시지의 큐이다.
- <98> 흐름 제어 관리 블록(542)은 컨텐츠를 푸시하는 일을 중단하도록 또는 컨텐츠를 푸시하는 일을 재개하도록 도 4의 푸시 프록시(410)에 통지하는데 사용된다. 이것은, 예를 들어, 푸시 클라이언트(510)가 푸시된 컨텐츠를 받을 수 있는 제한된 양의 메모리를 가질 때 사용될 수 있다. 이 경우에, 푸시 컨텐츠가 소모되기 이전에, 푸시 클라이언트(510)는 푸시 프록시(410)로부터의 컨텐츠의 흐름을 중단시킬 필요가 있다. 컨텐츠가 소모되었으면, 데이터의 흐름을 다시 시작하기 위해 흐름 제어 관리 블록(542)이 사용될 수 있다.
- <99> 푸시 클라이언트(510) 내의 푸시 에이전트(push agent)(544)는 도 4의 푸시 프록시(410)로부터 정보를 수신하는데 사용된다.
- <100> 당업자라면 잘 알고 있는 바와 같이, 메시지 브로커 및 애플리케이션 큐(540), 흐름 제어 관리 블록(542), 및 푸시 에이전트(544)는 메시징만을 처리하며 메타데이터를 처리하지 않는다.
- <101> 컨텐츠 메타데이터 추출기 및 캐쉬 블록(550)은 푸시 클라이언트(510)로 보내지는 동적 메타데이터를 추출하는데 사용된다. 도 4의 푸시 프록시(410)를 참조하여 상기한 바와 같이, 동적 컨텐츠 전달 아키텍처 내의 처리 요소 중 임의의 요소는 그에게로 보내진 메타데이터를 가지고 있을 수 있으며 이 메타데이터가 추출될 필요가 있다. 따라서, 푸시 클라이언트(510)로 보내지는 메타데이터는 컨텐츠 메타데이터 추출기 및 캐쉬 블록(550)에 의해 추출된다.

- <102> 게다가, 컨텐츠 메타데이터 추출기 및 캐쉬 블록(550)은 양호하게는 메타데이터를 캐싱하도록 구성되어 있다. 제1 컨텐츠 패키지와 제2 컨텐츠 패키지 간에 변하지 않는 푸시 클라이언트(510)에 대한 메타데이터가 전달될 필요가 없으며, 이 메타데이터를 추출할 필요가 없어서 푸시 클라이언트(510)에서의 처리 시간을 절감시키며 또한 푸시 클라이언트(510)에 대한 메타데이터가 무선 네트워크(130)를 거쳐 전달될 필요가 없어서 네트워크 자원을 절감시킨다.
- <103> 수신되는 컨텐츠의 프래그먼트를 분석하고 그 컨텐츠를 정확하게 합치기 위해 지역 호출 관리자(552)가 사용된다. 이하에서 보다 상세히 기술되는 바와 같이, 데이터는 선형적이거나 비선형적일 수 있다. 데이터가 비선형적인 경우, 그 데이터를 재구성하기 위해 메타데이터가 필요하며, 이것은 지역 호출 관리자(552)에 의해 행해진다. 지역 호출 관리자(552)는 또한 푸시 프록시(510)의 지역 호출 저장 블록(452)에서 이용가능한 정보의 개요를 분석하도록 구성되어 있고 사용자에 의해 요구될 때 이 정보를 불러내기 위해 컨텐츠 폴 브로커(content pull broker)(554)(이하에 기술됨)를 구동한다. 이것은 컨텐츠 내비게이션(content navigation)이 컨텐츠 구조 그래프의 어떤 브랜치에 들어갈 때 또는 대역폭이나 비용 조건이 만족될 때 예측 호출(predictive retrieval)을 포함한다.
- <104> 컨텐츠 폴 브로커(554)는 푸시 클라이언트(510)도 역시 어떤 상황에서 컨텐츠를 폴링할 수 있는 푸시/풀 모델(push/pull model)에서 사용된다. 이러한 상황에 대해 도 19를 참조하여 이하에서 보다 상세히 설명한다.
- <105> 당업자라면 잘 알고 있는 바와 같이, 컨텐츠 메타데이터 추출기 및 캐쉬(550), 지역 호출 관리자(552), 및 컨텐츠 폴 브로커(554)는 메시징 컨텐츠 및 메타데이터 둘다를 처리한다.
- <106> 가입 관리 블록(560)은 도 4의 가입 및 규칙 블록(460)과 동일하다. 구체적으로는, 가입 관리 블록(560)은 가입을 관리하기 위해 사용된다. 애플리케이션이 등록 취소(de-register)되거나 모바일 장치로부터 삭제되는 경우, 가입 관리 블록(560)은 가입을 종료시킨다. 가입 관리 블록(560)은 또한 가입 채널이 만료할 때 클라이언트 애플리케이션을 대신하여 재가입할 수 있다.
- <107> 갱신 통지 블록(update notification block)(562)은 클라이언트 애플리케이션과 함께 동작하고, 또 새로운 컨텐츠가 애플리케이션을 기다리고 있음을 애플리케이션에 통지하기 위해 사용된다. 이것은 다음의 3가지 방식 중 한 방식으로 행해질 수 있다.
- <108> a. 갱신 통지 블록(562)이 애플리케이션(512)에 통지할 수 있는 제1 방식은 푸시 클라이언트(510)가 컨텐츠를 애플리케이션(512)으로 직접 전송하는 것이다.
- <109> b. 갱신 통지 블록(562)이 애플리케이션(512)에 새로운 컨텐츠를 통지할 수 있는 제2 방식은 컨텐츠를 컨텐츠 저장 장치(580)에 저장하고 또 선택적으로 컨텐츠가 기다리고 있음을 애플리케이션(512)에 통지하는 것이다. 이 경우의 통지는 선택적이다. 구체적으로는, 애플리케이션(512)이 그에게로 보내진 정보가 특정의 메모리 블록에 저장되어 있음을 알고 있는 경우, 애플리케이션이 새로운 데이터를 가지고 있음을 애플리케이션이 발견하기 위한 한 가지 옵션은 메모리 장소에 어떤 것이 기록되어 있는지 여부를 알아보기 위해 메모리 장소를 주기적으로 폴링하는 것이다. 다른 대안으로서, 갱신 통지 블록(562)은, 애플리케이션이 데이터가 저장되어 있는 장소에 아마도 새로운 데이터를 가지고 있음을 나타내는, 메시지를 애플리케이션(512)으로 전송할 수 있다.
- <110> c. 갱신 통지 블록(562)이 애플리케이션(512)에 새로운 컨텐츠를 통지할 수 있는 제3 방식은 컨텐츠를 내부적으로 저장하고 애플리케이션에 통지하는 것이다. 애플리케이션은 이어서 컨텐츠를 불러내도록 푸시 클라이언트에 요청할 수 있다.
- <111> 컨텐츠 의존성 블록(564)은 도 4의 컨텐츠 의존성 블록(462)과 동일하며, 모바일 장치에 서비스를 광고할지 여부를 결정할 수 있다.
- <112> 컨텐츠 만료 및 교체 블록(566)은 도 4의 컨텐츠 교체 및 만료 블록(466)과 동일하다. 컨텐츠의 만료 및 컨텐츠의 교체는 따라서 푸시 서버 또는 푸시 프록시에 부가하여 푸시 클라이언트(510)에서 처리될 수 있다.
- <113> 채널 메타데이터 저장소(570)는 애플리케이션(512)에 대한 채널 메타데이터를 저장하는데 사용된다.
- <114> 백그라운드 갱신 처리 모듈(background update processing module)(575)은 애플리케이션(512)이 이용가능하지 않을 때 갱신을 수행하는데 사용된다. 백그라운드 갱신은, 예를 들어, 애플리케이션 저장 장치 내부에서의 데이터의 더 새로운 데이터로의 교체를 가능하게 해준다. 그 후에, 사용자가 애플리케이션을 기동할 때, 애플리케이션에 의해 디스플레이된 데이터가 정정 및 갱신된다.

- <115> 백그라운드 갱신 처리 모듈(575)은 처리 규칙을 사용하여 컨텐츠를 애플리케이션에 대해 허용가능한 포맷으로 변환한다. 이는 컨텐츠 저장 장치(580) 내의 컨텐츠를 실행 및 처리할 수 있다.
- <116> 예로서, 한밤중에 계약자에 대해 갱신된 작업 리스트가 그 리스트로 푸시된 작업을 가질 수 있다. 이 동안에는 작업 애플리케이션이 시작되지 않으며, 백그라운드 갱신 처리 모듈(575)이 사용될 수 있다. 이것은 XML(extensible mark-up language) 파일을 처리하기 위한 코드로 행해질 수 있으며, 장치 상에서 "handler.exe"라고 하는 파일에 존재할 수 있다. 푸시 클라이언트(510) 상의 백그라운드 갱신 처리 블록(575)은 handler.exe를 실행하여, XML 문서를 파라미터로서 전달할 수 있다. 이어서, 핸들러는 작업을 애플리케이션의 내부 포맷으로 구성한다.
- <117> 푸시 클라이언트(510)의 백그라운드 갱신 처리 블록(575)이 작업을 애플리케이션 내부 포맷으로 구성하였으면, 그 블록(575)은 작업을 컨텐츠 저장 장치(580)로부터 작업 리스트 새로운 작업을 리스트에 첨부 할 수 있다. 이 블록(575)은 이어서, 작업 애플리케이션이 그 다음에 푸시 클라이언트(510)에 연결될 때를 위해, 수정된 것을 다시 컨텐츠 저장 장치(580)에 저장할 수 있다.
- <118> 따라서, 도 5는, 컨텐츠 및 컨텐츠의 처리가 동적이고 미리 정해지지 않은 일반적인 동적 컨텐츠 전달 시스템에서 사용될 수 있는 푸시 클라이언트(510)를 나타낸다. 도 5의 푸시 클라이언트(510)를 참조하여 상기한 블록들은 시스템의 동적 특성에 대처하기 위해 사용된다.
- <119> 도 3을 참조하여 상기한 바와 같이, 컨텐츠의 처리를 위한 지능을 제공하기 위해 컨텐츠는 메타데이터와 연관되어 있다. 본 방법 및 시스템에 따르면, 메타데이터는 2가지 유형의 메타데이터로 분할될 수 있다. 구체적으로는, 정적(채널) 메타데이터 및 동적(컨텐츠) 메타데이터로 분할될 수 있다.
- <120> 컨텐츠 제공자 및 애플리케이션의 유형이 무한하기 때문에, 일반적인 시스템을 구축하기 위해 메타데이터가 중요하다. 특정 유형의 컨텐츠를 처리하는 유일한 방법은 메타데이터를 통하는 것이다.
- <121> 정적 메타데이터는 특정 유형의 컨텐츠를 어떻게 처리해야 하는지에 관한 규칙을 제공하는 메타데이터이다. 정적 메타데이터는 여러가지 추상화 레벨로 분해될 수 있고 또, 예를 들어, 컨텐츠 자체에 관한 구조 정보를 포함 할 수 있다. 예를 들어, RSS(Real-time Simple Syndication) 문서는 RSS 2.0 XSD 구조로 전달될 수 있으며, 그 컨텐츠 제공자로부터의 모든 컨텐츠가 이 구조로 전달된다.
- <122> 정적 메타데이터에 대한 다른 추상화 레벨은 컨텐츠 서브유형에 대한 처리 규칙의 제공을 포함한다. 이것은 애플리케이션에 관련된 것일 수 있다. 따라서, 예를 들어, 금융 뉴스 애플리케이션은 데이터가 금융 뉴스 RSS 스트림으로부터 추출되어, 미리 정의된 장소에 저장되어야만 하며 또 그 애플리케이션이 정보의 도달에 관하여 통지를 받아야만 함을 알려준다. 이 애플리케이션은 그에게로 보내지는 컨텐츠가 이와 같이 처리되도록 항상 요구 한다.
- <123> 정적 메타데이터(본 명세서에서 채널 메타데이터라고도 함)는 애플리케이션과 컨텐츠 제공자 간에 가입 동안 내내 동일한 채로 있고, 따라서 정적 메타데이터는 그 아키텍처 내의 각각의 요소에 대해 또한 각각의 컨텐츠 전달 채널마다 한번 작성될 수 있다. 일 실시예에서, 이것은 애플리케이션 또는 컨텐츠 제공자의 등록 시에 행해 진다.
- <124> 동적 메타데이터는 컨텐츠의 특정 피스와 연관되어 있는 메타데이터이다. 예를 들어, 특정의 데이터와 연관된 만료 정보 또는 특정의 데이터와 연관된 교체 규칙 및 정보이다(예를 들어, 문서 K가 문서 L을 대신한다).
- <125> 도 4 및 도 5를 참조하여 상기한 바와 같이, 각각의 프로세싱 개체는 그 프로세싱 개체로 보내지는 정적 및 동적 메타데이터 둘다를 수신할 수 있다. 따라서, 푸시 프록시(410)는 컨텐츠 메타데이터 추출기 및 캐쉬(450)를 사용하여 동적 메타데이터를 추출하고, 컨텐츠 만료 및 교체 모듈(466)은 전달되지 않은 컨텐츠를 푸시 프록시(410)에 수신된 더 새로운 컨텐츠로 대체하는데 사용된다.
- <126> 이제부터 도 6을 참조한다. 도 6은 컨텐츠 메타데이터에 대한 다층 엔벨로프 모델(multilayer envelope model)이다.
- <127> 푸시 프록시(410)는 프록시 서버(612)에 대한 컨텐츠 프로세싱 메타데이터를 포함하는 푸시 엔벨로프(push envelope)(610) 및 푸시 클라이언트 엔벨로프(614)를 수신한다. 푸시 프록시(410)는 컨텐츠 프로세싱 메타데이터(612)를 추출하고, 이 메타데이터를 사용하여 푸시 클라이언트 엔벨로프(614)를 처리한다. 메타데이터(612)는 푸시 클라이언트 엔벨로프(614)로 무엇을 해야 하는지를 푸시 프록시에 지시한다.

- <128> 푸시 클라이언트 엔벨로프(614)는 푸시 클라이언트(510)로 전달되어, 이곳에서 푸시 클라이언트 엔벨로프(614)는 컨텐츠 엔벨로프(620) 및 컨텐츠 프로세싱 메타데이터(622)로 분해된다. 컨텐츠 프로세싱 메타데이터(622)는 컨텐츠 엔벨로프(620)를 처리하기 위해 푸시 클라이언트(510)에 의해 사용된다. 예를 들어, 클라이언트 애플리케이션(150)이 컨텐츠의 마지막 버전에만 관심이 있는 경우 푸시 클라이언트(510)에 이전에 전달된 컨텐츠 엔벨로프(620)의 마지막 엔벨로프로의 교체를 수행하도록 푸시 클라이언트(510)에 지시하기 위해 이것이 사용될 수 있다.
- <129> 컨텐츠 엔벨로프(620)는 클라이언트 애플리케이션(150)으로 전달된다. 컨텐츠 엔벨로프(620)는 애플리케이션에 대한 컨텐츠 프로세싱 메타데이터(630) 및 클라이언트 애플리케이션(150)에 의해 소모되는 컨텐츠 페이로드(632)를 포함한다.
- <130> 당업자라면 잘 알고 있는 바와 같이, 도 6에 따른 엔벨로프의 내포(nesting)는 처리가 아키텍처의 임의의 처리 요소에서 행해질 수 있고 또 컨텐츠 제공자(110)가 특정의 컨텐츠가 어떻게 처리되어야 하는지를 규정할 수 있는 풍부한 동적 환경을 제공한다. 일 실시예에서, 특정의 논리 요소로 보내지는 메타데이터는 다른 처리 요소에 대해 불투명하다.
- <131> 다른 대안으로서, 서비스 제공자(120)는 또한 푸시 클라이언트(510) 또는 클라이언트 애플리케이션(150)에서의 처리를 위해 푸시 프록시(410)에서 메타데이터를 추가할 수 있다.
- <132> 도 7을 참조하면, 동 도면은 도 6의 엔벨로프 모델, 및 각각의 처리 요소가 엔벨로프를 인정하는 단계를 나타낸 것이다. 도 7에 나타낸 바와 같이, 푸시 프록시(410)는 먼저 푸시 엔벨로프(610)로부터 메타데이터를 추출한다. 이것은 단계(710)에서 행해진다.
- <133> 단계(712)에서, 푸시 프록시(410)는 푸시 클라이언트 엔벨로프(614)를 처리하기 위해 메타데이터를 사용한다. 단계(714)에서, 푸시 프록시(410)는 푸시 클라이언트 엔벨로프(614)를 푸시 클라이언트(510)로 전달한다.
- <134> 이와 유사하게, 푸시 클라이언트(510)는, 단계(720)에서, 푸시 클라이언트 엔벨로프(614)로부터 컨텐츠 프로세싱 메타데이터(622)를 추출한다. 단계(722)에서, 푸시 클라이언트(510)는 컨텐츠 엔벨로프(620)에 대해 컨텐츠 프로세싱 메타데이터(622)를 사용한다. 단계(724)에서, 푸시 클라이언트(510)는 컨텐츠 엔벨로프(620)를 클라이언트 애플리케이션(150)으로 전달한다.
- <135> 단계(730)에서, 클라이언트 애플리케이션(150)은 컨텐츠 프로세싱 메타데이터(630)를 추출하고, 단계(732)에서, 컨텐츠 페이로드(632)에 대해 컨텐츠 프로세싱 메타데이터(630)를 사용한다.
- <136> 도 8을 참조하면, 동 도면은 도 7에 나타낸 방법에 정적 또는 채널 메타데이터를 사용하는 부가적인 단계를 부가하여 나타낸 것이다. 구체적으로는, 메타데이터가 단계(710)에서 푸시 엔벨로프(610)로부터 추출된 후에, 푸시 프록시(410)는 그 다음에 단계(810)에서 푸시 클라이언트 엔벨로프를 처리하기 위해 정적 채널 메타데이터를 사용한다. 단계(712)에서, 푸시 프록시(410)는 그 다음에 컨텐츠 프로세싱 동적 메타데이터(612)를 처리한다. 푸시 프록시(410)는 그 다음에 단계(714)에서 푸시 클라이언트 엔벨로프(614)를 전달한다.
- <137> 이와 유사하게, 푸시 클라이언트(510)는 단계(720)에서 컨텐츠 프로세싱 메타데이터(622)를 추출한다. 푸시 클라이언트(510)는 이어서 단계(820)에서 컨텐츠 엔벨로프(620) 내의 컨텐츠에 대해 채널 메타데이터를 사용한다. 푸시 클라이언트(510)는 이어서, 단계(724)에서 클라이언트 애플리케이션(150)으로 컨텐츠 엔벨로프(620)를 전달하기 이전에, 단계(722)에서 컨텐츠 프로세싱 메타데이터(622) 내의 동적 컨텐츠 메타데이터를 사용한다.
- <138> 클라이언트 애플리케이션(150)은 먼저 단계(730)에서 컨텐츠 프로세싱 메타데이터(630)를 추출한다. 클라이언트 애플리케이션(150)은 이어서 단계(830)에서 컨텐츠 페이로드(632)에 대해 채널 메타데이터를 사용한다. 클라이언트 애플리케이션(150)은 이어서 단계(732)에서 컨텐츠 페이로드(632)에 대해 컨텐츠 프로세싱 메타데이터(630)를 사용한다.
- <139> 당업자라면 잘 알고 있는 바와 같이, 상기 모델은 따라서 전송되고 있는 특정의 컨텐츠와 연관되어 있는 동적 메타데이터와 함께 정적 메타데이터도 채널에 대해 적용되도록 한다.
- <140> 이제부터 도 9를 참조한다. 도 5로부터 당업자라면 잘 알고 있는 바와 같이, 푸시 클라이언트(510)는 모바일 장치 상의 다수의 타겟 애플리케이션(512)을 서비스한다. 애플리케이션이 다른 애플리케이션에 대한 서비스를 중단시키지 않고 동적 컨텐츠 전달 프레임워크에 등록을 할 수 있는 효율적인 런타임 등록 메카니즘이 요구된다.
- <141> 도 9를 참조하면, 푸시 클라이언트(510)는 푸시 클라이언트에 이미 등록되어 있는 3개의 애플리케이션, 구체적

으로는 애플리케이션(910, 912, 914)을 포함한다. 당업자라면 잘 알고 있는 바와 같이, 플러그인 모델이 중요한 이유는 새로운 장치가 애플리케이션 유형을 무제한적으로 장치 상에 설치될 수 있게 해주기 때문이다. 게다가, 애플리케이션이 동적으로 설치될 수 있어, 모바일 장치가 애플리케이션 플랫폼이 되게 한다. 이 장치가 애플리케이션 플랫폼일 수 있기 때문에, 이 장치는 새로운 애플리케이션을 동적으로 편입시킬 수 있어야만 한다.

<142> 도 9에서 알 수 있는 바와 같이, 애플리케이션(916)은 푸시 클라이언트(510)에 등록하기를 원한다. 애플리케이션(916)은 양호한 실시예에서 애플리케이션에 대한 채널 메타데이터를 제공하는 애플리케이션 매니페스트(application manifest)(918)를 포함한다. 구체적으로는, 애플리케이션 매니페스트(918)는 푸시 클라이언트(510)에 정보를 제공하며, 궁극적으로는 도 1의 푸시 프록시(410) 및 컨텐츠 제공자(110)에 애플리케이션에 대한 정적 메타데이터를 제공한다. 이것은 애플리케이션이 어느 유형의 컨텐츠를 예상하고 있는지, 컨텐츠가 어떻게 전달될 것인지, 애플리케이션이 통지를 필요로 하는지 여부, 또는 본 시스템 및 방법과 관련된 당업자에게는 명백한 다른 채널 정보를 포함할 수 있지만, 이에 한정되는 것은 아니다.

<143> 따라서, 애플리케이션(916)은 푸시 클라이언트(510)에 등록을 하고 애플리케이션 매니페스트(918)를 제공하여, 애플리케이션(916)을 서비스하기 위해 서비스 제공자로의 채널을 설정한다.

<144> 도 10을 참조하면, 대안의 모델은 도 2의 아키텍처(220)와 관련하여 기술된 모델일 수 있다. 구체적으로는, 도 10의 모델에서, 클라이언트 애플리케이션(150)은 푸시 클라이언트(140)와 짹을 이룬다. 클라이언트 애플리케이션(150)/푸시 클라이언트(140) 쌍의 각각은 푸시 컨테이너(1010)로 코디네이션된다.

<145> 애플리케이션(1020)이 푸시 컨테이너(1010)에 등록을 하고자 하는 경우, 푸시 컨테이너(1010)에 의해 클라이언트(140)가 생성되거나, 클라이언트(140)가 이미 존재한다면 푸시 컨테이너(1010)에 의해 클라이언트(140)가 사용된다. 게다가, 등록 시에, 애플리케이션(1020)은 푸시 컨테이너(1010)에 애플리케이션 매니페스트(1030)를 제공하고, 그에 의해 애플리케이션(1020)에 대한 채널 메타데이터(정적 메타데이터)를 제공한다.

<146> 도 10의 또 다른 예시가 도 11에 도시되어 있다. 구체적으로는, 푸시 컨테이너(1110)는 푸시 클라이언트의 풀을 관리/유지한다. 애플리케이션이 컨테이너에 등록을 할 때, 애플리케이션은 전용의 푸시 클라이언트(510)를 획득하고, 이 푸시 클라이언트(510)은 간단한 경우에 소켓 리스너(socket listener)(1130)와 컨테이너 핸들러(container handler)의 쌍으로 표현될 수 있다. 애플리케이션이 컨테이너(및 컨텐츠 전달 서비스)로부터 등록 취소(unregister)되거나 장치로부터 삭제될 때, 푸시 클라이언트가 풀(pool)로 반환된다.

<147> 푸시 컨테이너(1110)는 통신을 위한 소켓(1120)을 포함한다. 게다가, 푸시 컨테이너(1110)는 특정의 소켓에 할당된 소켓 리스너(1130) 및 컨텐츠 프로세서(1140)를 포함한다.

<148> 도 11에서 알 수 있는 바와 같이, 여러가지 컨테이너 프로세서 및 소켓 리스너 쌍이 이전에 등록된 애플리케이션(150)에 의해 사용된다.

<149> 새로운 애플리케이션(1150)이 푸시 컨테이너(1110)에 등록하고자 하는 경우, 새로운 컨테이너 프로세서 및 소켓 리스너(1120, 1130)가 애플리케이션(1150)을 서비스하기 위해 할당된다.

<150> 따라서, 이상의 설명은 새로운 클라이언트 애플리케이션(150)이 구현될 수 있고 푸시 클라이언트(510) 또는 푸시 컨테이너(1010 또는 1110)에 등록될 수 있으며, 그에 따라 그 장치가 새로운 애플리케이션을 동적으로 편입시킬 수 있는 애플리케이션 플랫폼이 될 수 있게 해주는 일반적인 푸시 프레임워크를 제공한다. 상기 도 9 및 도 10의 애플리케이션 매니페스트(1030 또는 918)의 전달은 채널 메타데이터의 설정을 가능하게 해주며, 그에 따라 컨텐츠가 애플리케이션의 요구사항에 따라 처리될 수 있게 해준다.

<151> 도 12를 참조하면, 컨텐츠 제공자(110)는 이와 유사하게 푸시 프록시(410)에 등록을 할 필요가 있다. 도 12에서 알 수 있는 바와 같이, 푸시 프록시(410)는 푸시 프록시(410)에 이미 등록되어 있는 3개의 컨텐츠 제공자, 즉 1210, 1212, 및 1214를 포함한다. 컨텐츠 제공자(1216)는 푸시 프록시(410)에 등록하기를 원한다.

<152> 푸시 클라이언트(510)에 등록을 할 때 애플리케이션(916)에 의해 제공되는 도 9에 나타낸 애플리케이션 매니페스트(918)와 유사하게, 컨텐츠 제공자(1216)는 컨텐츠 제공자(1216)가 등록을 할 때 푸시 프록시(410)로 전달되는 서비스 매니페스트(1218)를 포함한다. 서비스 매니페스트(1218)는 컨텐츠 제공자가 제공하는 정보의 유형, 컨텐츠 제공자가 이 정보를 얼마나 자주 제공하는지, 정보의 포맷, 및 서비스를 위해 또는 서비스의 광고를 위해 유용한 임의의 다른 정보에 관한 정보를 포함한다. 다른 정보도 가능하다.

<153> 따라서, 푸시 프록시(410)는 컨텐츠 제공자(1216)에 대한 채널(정적) 메타데이터를 설정하기 위해 서비스 매니

페스트(1218)를 사용한다.

<154> 도 13을 참조하면, 도 2의 아키텍처(230)로 표현된 또 다른 실시예는 다수의 푸시 프록시(122) 및 컨텐츠 제공자(110) 쌍을 갖는 푸시 컨테이너를 갖는 것이다. 도 12에서와 같이, 여러가지 애플리케이션이 이미 푸시 컨테이너(1310)에 등록되어 있을 수 있으며, 도 12의 예에서, 애플리케이션(1312, 1314, 1316)은 이미 푸시 프록시(1313, 1315, 1317)에 각각 등록되어 있다.

<155> 새로운 애플리케이션(1320)이 푸시 컨테이너(1310)에 등록하기를 원한다. 따라서, 푸시 컨테이너(1310)는 새로운 프록시(도시 생략)를 생성하거나 기존의 프록시(도시 생략)를 사용하며, 푸시 컨테이너(1310)는 컨텐츠 제공자(1320)를 이들 프록시와 연관시킨다. 게다가, 컨텐츠 제공자(1320)는 컨텐츠 제공자(1320)가 제공하게 될 컨텐츠를 기술하는 서비스 매니페스트(1322)를 제공하며, 그에 의해 채널 메타데이터의 설정을 가능하게 해준다.

<156> 당업자라면 잘 알고 있는 바와 같이, 도 9 및 도 10의 실시예는 공유 애플리케이션을 갖거나 애플리케이션마다 전용의 푸시 클라이언트를 갖는, 푸시 클라이언트에 대한 2가지 옵션을 나타내고 있다. 당업자라면 다른 실시예가 가능하다는 것을 잘 알 것이다. 이와 유사하게, 도 12 및 도 13과 관련하여, 다수의 컨텐츠 제공자가 등록되어 있는 푸시 프록시가 도시되어 있거나, 푸시 컨테이너에 구현된 각각의 컨텐츠 제공자에 대한 전용의 푸시 프록시가 도시되어 있다.

<157> 도 14와 관련하여, 컨텐츠 제공자(110)와 클라이언트 애플리케이션(150) 간의 메시징이 도시되어 있다. 컨텐츠 제공자(110)는 푸시 프록시(410)에 등록 메시지를 제공한다. 이 메시지는 채널 메타데이터를 푸시 프록시(410)에 제공하기 위해 사용될 수 있는 서비스 매니페스트를 포함할 수 있다. 이것은 단계(1410)에서 행해진다.

<158> 컨텐츠 제공자(110)는 그에 부가하여 또는 다른 대안으로서, 단계(1412)에 나타낸 바와 같이, 차후의 메시지에서 채널 메타데이터를 제공할 수 있다.

<159> 푸시 프록시(410)는 이어서 단계(1414)에서 이용가능한 서비스의 목록(서비스 카탈로그)에 서비스를 추가한다.

<160> 도 14의 예에서의 선택적인 단계는 단계(1416)에서 푸시 프록시(410)가 푸시 클라이언트(510)에 이용가능한 새로운 서비스를 통지하는 것이며, 단계(1418)에서 이 통지는 클라이언트 애플리케이션(110)으로 전달될 수 있다.

<161> 당업자라면 잘 알고 있는 바와 같이, 단계(1416, 1418)는 선택적이며, 다른 대안은 클라이언트 애플리케이션(150)이 새로운 서비스를 보기 위해 서비스 카탈로그를 푸시 프록시(410)로부터 주기적으로 풀링하는 것을 포함한다.

<162> 클라이언트 애플리케이션(150)에 대한 사용자 또는 서비스 제공자가 클라이언트 애플리케이션(150)이 서비스에 가입해야만 하는 것으로 결정할 때, 클라이언트 애플리케이션(150)은 단계(1420)에서 가입 메시지를 전송한다. 이 가입 메시지는 또한 단계(1422)에서 푸시 프록시(410)로 전달된다.

<163> 푸시 프록시(410)가 단계(1422)에서 가입 메시지를 수신하면, 2가지 옵션이 이용가능하다. 첫번째 옵션은 가입을 위해 컨텐츠 제공자(110)로 메시지(1424)를 전송하고 이어서 단계(1426)에서 다시 메타데이터를 포함하는 메시지 엔VELOP를 수신하는 것이다. 이 메타데이터는 장치 또는 장치 유형 특성일 수 있다.

<164> 다른 대안으로서, 푸시 프록시(410)는 단계(1422)에서 가입 메시지를 수신할 수 있고, 곧바로 컨텐츠 제공자(110)에 의해 이미 제공되어 푸시 프록시(410)에 저장되어 있는 정보에 기초하여, 단계(1430)에서 푸시 클라이언트(510)에 응답한다. 이 응답은 단계(1432)에서 클라이언트 애플리케이션(150)으로 전달된다. 당업자라면 잘 알고 있는 바와 같이, 이 응답은 컨텐츠 제공자(110)에 고유한 채널 메타데이터를 포함할 수 있다.

<165> 모델들에서의 차이점은 누가 애플리케이션에 대한 데이터를 커스터마이즈하느냐에 의존할 수 있다. 당업자라면 잘 알고 있는 바와 같이, 컨텐츠 제공자(110)는 다른 처리 요소와 비교하여 최상의 컨텐츠 커스터마이제이션(customization of content)을 제공한다. 그렇지만, 서비스 제공자(120)는 또한, 푸시 프록시(410)를 통해, 컨텐츠 커스터마이제이션을 제공할 수 있다.

<166> 게다가, 당업자라면 잘 알고 있는 바와 같이, 컨텐츠의 구조는 애플리케이션이 요구하는 데이터에 의존할 수 있다. 예를 들어, 금융 애플리케이션에서, 이 애플리케이션은 주식 시세 및 환율 둘다를 원할 수 있다. 이하의 XML이 사용될 수 있다.

<167> <FIN>

<168> <quotes>

<169> <quote ticker = ABC>

<170> 18.54

<171> </quote>

<172> <quote ticker = XYZ>

<173> 123.45

<174> </quote>

<175> </quotes>

<176> <rates>

<177> <rate id = "US-CAN">

<178> 1.15

<179> </rate>

<180> <rate id = "US-EURO">

<181> 0.85

<182> </rate>

<183> </rates>

<184> </FIN>

<185> 사용자가 주식 시세만을 원하고 환율은 원하지 않은 경우, 이 구조는 이하의 것으로 변할 수 있다.

<186> <FIN>

<187> <quote ticker = ABC>

<188> 18.54

<189> </quote>

<190> <quote ticker = XYZ>

<191> 123.45

<192> </quote>

<193> </FIN>

<194> 이 메타데이터는 전달되고 있는 데이터의 구조에 관한 정보를 애플리케이션에 제공할 수 있다.

<195> 따라서, 2가지 모델이 존재한다. 정적 메타데이터는 등록 동안이나 그 이후에 푸시 프록시(410) 및 푸시 클라이언트(510)에 제공될 수 있다. 다른 대안으로서, 푸시 프록시(410) 및 푸시 클라이언트(510)에 대한 메타데이터는 사전 제공되어 있을 수 있다, 즉 애플리케이션이 클라이언트에 등록할 때까지 정보가 푸시 클라이언트 또는 푸시 프록시에 저장되어 있다.

<196> 이제부터, 도 15를 참조한다. 도 15는 애플리케이션이 푸시 클라이언트(510)에 등록을 할 때에 일어나는 논리적 단계를 나타낸 것이다.

<197> 애플리케이션이 푸시 클라이언트(510)에 등록을 하면, 첫번째 단계(1510)는 등록된 애플리케이션을 애플리케이션에 의해 요구되는 컨텐츠 유형과 일치시키는 것이다. 이것은 도 9에 나타낸 애플리케이션 매니페스트(918)로부터 알 수 있다.

<198> 두번째 단계(1520)는 애플리케이션에 대한 환경을 설정하는 것이다. 이들은 애플리케이션에 대한 저장 및 전달 옵션을 포함하지만 이에 한정되는 것은 아니다. 예를 들어, 애플리케이션은 전송을 미리 정해진 양의 데이터로 제한할 수 있다. 푸시 클라이언트(510)는, 흐름 제어 이벤트에 있거나 애플리케이션 또는 클라이언트가 연결되

어 있지 않은 경우, 애플리케이션을 위해 데이터의 캐싱을 필요로 할 수 있고 선택적으로 데이터가 기다리고 있음을 애플리케이션에 통지하는 것을 필요로 할 수 있다.

<199> 세번째 단계(1530)는 푸시 프록시(410)에 애플리케이션 설정을 통지하는 것이다. 이것은 예를 들어 애플리케이션 또는 푸시 클라이언트(510)에 대한 이용가능한 저장을 포함한다. 당업자라면 잘 알고 있는 바와 같이, 푸시 프록시(410)는 푸시 클라이언트(510)가 저장할 수 있는 것보다 많은 데이터를 푸시해서는 안된다. 따라서, 애플리케이션 설정은 전달되는 데이터의 상한을 포함할 수 있다. 도 4 및 도 5를 참조하면, 이것은, 컨텐츠가 애플리케이션이 처리할 수 있는 것보다 클 경우, 컨텐츠 프래그먼트화 블록(464)이 컨텐츠를 프래그먼트화하도록 요청할 수 있다. 또한, 데이터가 비선형적인 경우, 컨텐츠 의존성 블록(462)은, 컨텐츠 의존성 블록(564)이 데이터를 재구성할 수 있도록 해주기 위해, 도 5의 컨텐츠 의존성 블록(564)을 위한 메타데이터를 생성해야만 할 수 있다.

<200> 다시 도 15를 참조하면, 단계(1530)는 또한 데이터 전달에 관한 선호도를 나타낼 수 있다. 예를 들어, 애플리케이션은 어떤 유형의 데이터를 다른 것들보다 선호할 수 있고, 이들 유형의 데이터가 우선순위를 부여받을 수 있다. 따라서, 단계(1530)는 유형 "A"의 데이터가 즉시 전달되는 한편 유형 "B"의 데이터가 지연된 시간에 전달될 수 있는 전달 스케줄을 확립하는데 사용될 수 있다.

<201> 이제, 도 16을 참조한다. 컨텐츠 제공자(110)가 푸시 프록시(410)에 등록을 할 때, 여러가지 단계가 수행된다. 첫번째 단계(1610)는 컨텐츠 저장 및 전달을 위해 요구되는 클라이언트 설정을 분석하는 것을 포함한다. 이것은, 예를 들어, 컨텐츠 제공자(110)로부터의 컨텐츠를 소비할 수 있는 장치 상의 푸시 클라이언트(510)를 식별하기 위해 서비스 광고에 대해 사용될 수 있다.

<202> 두번째 단계(1620)는 푸시 프록시(410)가, 무엇보다도 프록시 저장, 전달 옵션, 변환 옵션을 포함한 환경을 설정할 수 있게 해준다.

<203> 세번째 단계(1630)에서, 푸시 프록시(410)는 애플리케이션이 컨텐츠 제공자(110)로부터의 컨텐츠를 획득하기 위해 이미 등록되어 있는지 여부를 검사할 수 있다. 이러한 경우, 애플리케이션은, 전달 채널이 설정되어 있고 또 애플리케이션이 컨텐츠에 대해 준비되어 있다는 푸시 프록시(410)로부터 컨텐츠 제공자(110)로의 통지와 컨텐츠를 수신할 준비를 한다.

<204> 예를 들어, 컨텐츠 제공자(110)가 온라인으로 되기 이전에 애플리케이션이 장치 상에 사전 설치되어 있는 경우에, 단계(1630)가 행해질 수 있다. 따라서, 애플리케이션은 컨텐츠 제공자(110)가 이용가능하게 되기를 기다리거나, 애플리케이션은 일반적인 유형(예를 들어, 브라우저 또는 RSS 뷰어)이고 다수의 컨텐츠 제공자로부터의 정보를 소비할 수 있다. 대안의 설정에서, 애플리케이션이 설치되기 이전에 컨텐츠 제공자(110)가 이미 이용가능한 경우, 컨텐츠 제공자(110)로부터 클라이언트 애플리케이션(150)으로 이동하기 시작하는 컨텐츠를 개시하기 위해 도 15에서의 통지 단계(1530)가 사용될 수 있다.

<205> 도 16을 참조하면, 당업자라면 잘 알고 있는 바와 같이, 클라이언트 설정은, 무엇보다도, 컨텐츠 분할을 위해 사용되는 이용가능한 저장 장치 크기, 흐름 제어를 위해 사용되는 큐 크기, 푸시 간격을 포함하는 전달 스케줄링, 클라이언트가 프록시로부터 정보를 불러내는지, 의사-푸시 모드(pseudo-push mode)를 생성하고 있는지, 모바일 장치의 화면 크기 등의 커스터마이제이션 옵션 등의 어떤 정보를 포함할 수 있다.

<206> 또한, 당업자라면 잘 알고 있는 바와 같이, 서로 다른 클라이언트에 대해 서비스 카탈로그가 다를 수 있다. 예를 들어, 어떤 클라이언트는 더 많은 데이터를 이용하여, 이러한 양의 정보를 처리할 수 없는 장치가 보다 작은 스크린 크기 등을 갖게 하는 것보다 클라이언트가 컨텐츠 제공자(110)에 더 적합하도록 하는 상이한 스크린 크기나 그 외 조건 등을 가질 수 있다. 따라서, 푸시 프록시(410)는 그 클라이언트 애플리케이션에 대해 아는 것에 기초하여 특정의 클라이언트 애플리케이션에 대한 서비스 카탈로그를 생성할 수 있으며, 그 클라이언트 애플리케이션(150)이 설치되어 있는 그 장치만이 컨텐츠 제공자에 관한 정보를 수신할 수 있다.

<207> 또한, 당업자라면 잘 알고 있는 바와 같이, 어떤 경우에, 애플리케이션은 사용자 개입없이 서비스 제공자 및 컨텐츠 제공자에 기초하여 설치될 수 있다. 예를 들어, 컨텐츠 제공자(110)가 푸시 프록시(410)에 등록하고 있는 경우, 모바일 장치의 사용자는 어떤 애플리케이션을 수락하도록 계약 의무를 가질 수 있다. 따라서, 푸시 프록시(410)는 애플리케이션을 설치할 준비가 되어 있고 또 애플리케이션을 푸시 클라이언트(510)로 푸시할 준비가 되어 있음을 푸시 클라이언트(510)에 통지할 수 있다. 이것은, 예를 들어, 그의 모바일 계획(mobile plan)에 관한 선호 요금(preferred rate)을 받기 위해 매월 어떤 수의 광고를 수신하기로 합의한 사용자를 포함할 수 있다. 컨텐츠 제공자(110)는 광고 제공자일 수 있고, 푸시 프록시(410)는 따라서 애플리케이션을 디스플레이하

는 광고를 푸시 클라이언트(510)(이 푸시 클라이언트(510)는 푸시 클라이언트(410)에 등록되어 있는 애플리케이션 인스톨러(application installer)에 의해 서비스될 수 있음)로 푸시할 수 있으며, 따라서 컨텐츠 제공자(110) 및 서비스 제공자(120)가 프로세스를 완전히 구동하게 할 수 있다.

<208> 따라서, 이상의 설명은 각각의 애플리케이션 또는 컨텐츠 제공자가 등록을 하고 애플리케이션 매니페스트 또는 서비스 매니페스트를 각각 제공하는 푸시 프레임워크에서의 플러그인 등록 모델을 제공한다. 애플리케이션 매니페스트 또는 서비스 매니페스트는 등록 동안이나 그 이후에 푸시 프록시(410) 및 푸시 클라이언트(510)에서 채널 메타데이터를 설정하는데 사용된다. 그 후에 애플리케이션(150)이 등록을 하고 또 컨텐츠 제공자(110)가 등록을 할 때, 컨텐츠가 애플리케이션(150)과 컨텐츠 제공자(110) 간에 이동하기 시작할 수 있다.

<209> 도 4 및 도 5를 참조하면, 채널 메타데이터는 채널 메타데이터 저장소(470, 570)에 저장되어 있다. 그렇지만, 또한 동적 메타데이터가 반복되는 경우 아키텍처(100) 내의 여러가지 처리 요소에 동적 메타데이터를 저장하는 것도 바람직하다. 당업자라면 잘 알고 있는 바와 같이, 이것은, 현재의 메타데이터 추출기(450)가 동일한 메타데이터를 여러번 추출할 필요가 없기 때문에, 푸시 프록시(410)에서의 처리를 절감시켜 준다. 게다가, 컨텐츠 만료 및 교체 모듈(466 또는 566) 등의 여러가지 모듈에 의한 처리는 전달되는 각각의 컨텐츠마다 갱신될 필요가 없다. 푸시 프록시(410)가 많은 수의 푸시 클라이언트(510)와 함께 동작할 수 있기 때문에, 각각의 컨텐츠 메시지에 대한 이러한 처리 절감이 상당할 수 있다. 게다가, 컨텐츠 제공자(110)와 푸시 프록시(410) 사이에서 고정된 회선을 통해 또는 푸시 프록시(410)와 푸시 클라이언트(510) 사이에서 고정된 회선(fixed line)을 통해 메타데이터를 전달할 필요가 없어 대역폭이 절감될 수 있다.

<210> 이제 도 17을 참조한다. 도 17은 마지막 메타데이터 버전이 처리 요소에 의해 저장되는 런타임 흐름의 일례를 나타낸 것이다.

<211> 도 17에 나타낸 바와 같이, 컨텐츠 제공자(110)는 컨텐츠  $[C_1+M(p,c,a)_1]$ 를 포함하는 컨텐츠 엔벨로프를 제공한다. 이것은 제1 컨텐츠 페이로드가 프록시 메타데이터, 클라이언트 메타데이터 및 애플리케이션 메타데이터를 포함하는 메타데이터와 함께 전송되고 있음을 의미한다. 이것은 단계(1710)에서 전송된다.

<212> 단계(1712)에서, 푸시 프록시(410)는 문구 " $M(p)_1$  사용"으로 나타낸 바와 같이 프록시 메타데이터를 사용한다. 게다가, 단계(1714)에서, 컨텐츠와, 클라이언트 메타데이터 및 애플리케이션 메타데이터를 포함하는 메타데이터가 푸시 클라이언트(510)로 전달된다.

<213> 단계(1716)에서, 푸시 클라이언트(510)는 클라이언트 메타데이터를 사용하며, 또한 단계(1718)에서, 컨텐츠 페이로드를 클라이언트 애플리케이션(150)으로 전달한다. 클라이언트 애플리케이션(150)은, 단계(1720)에서, 애플리케이션 메타데이터를 사용하며 또한 컨텐츠 페이로드를 소비한다.

<214> 단계(1722)에서 알 수 있는 바와 같이,  $C_2$ 로 나타낸 제2 컨텐츠 페이로드는 제1 컨텐츠 페이로드와 동일한 메타데이터를 갖는다. 각각의 처리 요소, 즉 푸시 프록시(410), 푸시 클라이언트(510) 및 클라이언트 애플리케이션(150)이 컨텐츠 제공자(110)를 위한 메타데이터를 캐싱하였기 때문에, 메타데이터가 다시 전달될 필요가 없고 그 대신에 처리 요소 상에 이미 존재한다.

<215> 그 후에, 단계(1724)에서, 푸시 프록시(410)는 푸시 프록시(410)에 대해 이전에 캐싱되었던 메타데이터를 사용한다. 이와 유사하게, 단계(1726, 1728)에서, 각각, 푸시 클라이언트(510)는 클라이언트 메타데이터를 사용하고, 클라이언트 애플리케이션(150)은 애플리케이션 메타데이터를 사용한다. 단계(1725, 1727)에서, 컨텐츠가 메타데이터 없이 전달된다.

<216> 단계(1740)에 나타낸 바와 같이, 컨텐츠는 푸시 클라이언트(510) 및 클라이언트 애플리케이션(150)에 대한 새로운 메타데이터를 가질 수 있지만, 푸시 프록시(410)에 대한 이전의 메타데이터를 보유하고 있을 수도 있다. 이러한 경우에, 단계(1740)에서 전달되는 메타데이터는 클라이언트 메타데이터 및 애플리케이션 메타데이터만을 포함한다. 단계(1742)에서, 푸시 프록시(410)는 캐싱된 프록시 메타데이터를 사용하고, 단계(1744)에서, 새로운 클라이언트 메타데이터 및 애플리케이션 메타데이터와 함께 컨텐츠 페이로드를 전달한다.

<217> 단계(1746)에서, 푸시 클라이언트(510)는 그에게로 전달된 새로운 클라이언트 메타데이터를 사용하고, 또한 단계(1748)에서, 컨텐츠 페이로드 및 애플리케이션 메타데이터를 전달한다.

<218> 단계(1750)에서, 클라이언트 애플리케이션은 새로운 애플리케이션 메타데이터를 사용하고, 또한 컨텐츠 페이로드를 소비한다.

- <219> 당업자라면 잘 알고 있는 바와 같이, 어느 메타데이터가 변경되었는지 및 어느 메타데이터가 그대로 있는지에 관한 여러가지 구성이 존재할 수 있으며, 변경된 메타데이터만이 그를 필요로 하는 처리 요소로 전달된다. 당업자라면 잘 알고 있는 바와 같이, 처리 요소는, 새로운 메타데이터를 수신하지 않는 경우, 그가 저장한 캐싱된 메타데이터로 되돌아가서 그 캐싱된 메타데이터를 컨텐츠 페이로드에 대해 사용한다.
- <220> 다른 대안의 실시예에서, 메타데이터에 대해 충분적 변경도 행해질 수 있다. 예를 들어, 단계(1760)에서, 엘타 메타데이터 버전과 함께 새로운 컨텐츠 페이로드가 서비스 프록시(410)로 전달될 수 있다. 프록시 메타데이터의 엘타는 이전에 전달된 프록시 메타데이터와 현재의 메타데이터(이 메타데이터로 컨텐츠가 처리되어야만 함) 간의 차이를 포함할 수 있다. 푸시 프록시(410)는 이전의 메타데이터에 엘타를 부가하고 이어서 단계(1762)에서 이것을 컨텐츠 페이로드를 처리하는데 사용함으로써 메타데이터를 작성한다. 그 후에, 변화가 없기 때문에, 단계(1764)에서, 컨텐츠 페이로드가 단독으로 전송되고, 단계(1766)에서, 푸시 클라이언트(510)는 이전에 캐싱된 클라이언트 메타데이터를 사용한다.
- <221> 이어서, 푸시 클라이언트는 단계(1768)에서 클라이언트 애플리케이션(150)으로 컨텐츠 페이로드를 전달하고, 이 클라이언트 애플리케이션(150)은 단계(1770)에서 컨텐츠 페이로드에 대해 이전에 캐싱된 위치 메타데이터를 사용하고, 이어서 컨텐츠 페이로드를 소비한다.
- <222> 충분적 데이터가 사용될 수 있는 경우의 예는 컨텐츠 제공자가 컨텐츠 페이로드 내의 기존의 필드(30)가 클라이언트 애플리케이션(150)으로 전송하기 위해 추출되어야만 함을 프록시에게 말하는 상황이다. 후속하는 트랜잭션에서, 그 컨텐츠 페이로드에 중요한 2개의 부가적인 필드가 컨텐츠 제공자(110)에 의해 클라이언트 애플리케이션(150)으로 전달될 필요가 있는 것으로 생각될 수 있다. 따라서, 컨텐츠 제공자는, 충분적 변경을 사용하여, 2개의 부가적인 필드를 추출하고 이전에 추출된 30 필드에 이들을 추가하도록 푸시 프록시에 말할 수 있다. 엘타, 즉 2개의 부가적인 필드만을 전달하면 되므로, 푸시 프록시(410)에서 메타데이터를 추출하기 위한 처리 시간이 감소되고, 그에 따라 프로세스를 최적화한다.
- <223> 또한, 잘 알고 있는 바와 같이, 메타데이터가 여러가지 형태로 올 수 있다. 메타데이터는 자바 또는 C# 등의 네이티브 코드(native code) 또는 인터프리트된 코드(interpreted code)로서 컴파일될 수 있다. 메타데이터는 또한 어떤 프로퍼티를 사용할지를 나타내는 데이터/프로퍼티 파일일 수 있다. 다른 대안의 실시예에서, 메타데이터는 바이너리 컨텐츠, 예를 들어, XML 문서에 대한 XSLT 변환 등의 변환일 수 있다.
- <224> 상기한 것은 특정의 클라이언트 애플리케이션으로 전송되는 컨텐츠에 지능을 제공하기 위해 여러가지 응용에 사용될 수 있다. 상기한 것은 또한, 컨텐츠 제공자가 그의 데이터와 함께 제공하는 메타데이터에만 기초하여, 여러가지 애플리케이션에 컨텐츠를 제공할 수 있는 풍부한 컨텐츠 제공자를 제공할 수 있다. 이것은 도 18에 예로서 나타내어져 있다.
- <225> 컨텐츠 제공자(110)는, 예를 들어, 온라인 서점일 수 있다. 애플리케이션은 특정 장르의 새로운 발매에 대해 통보해주기를 원한다는 것을 온라인 서점에 알려주기 위해 온라인 서점에 등록을 할 수 있다. 이것은 매일 또는 매주 또는 매월 행해질 수 있다.
- <226> 컨텐츠 제공자(110)는, 예를 들어, 도서 목록(1812)을 갖는 컨텐츠 엔벨로프(1810)를 푸시 프록시(410)로 매주 전송하게 된다. 컨텐츠 제공자(110)는 또한, 예를 들어, 애플리케이션이 특정의 컨텐츠를 수신한 것에 기초하여 그 컨텐츠를 변환하기 위한 URL 링크일 수 있는 변환 메타데이터(1814)도 전송할 수 있다.
- <227> 일 실시예에서, 도서 목록(1812)은 수많은 책을 포함할 수 있으며, 각각의 도서의 설명은 작가 및 도서의 줄거리로 포함한다. 이 파일은, 예를 들어, 크기가 100 KB일 수 있다.
- <228> 푸시 프록시(410)는 이 큰 파일을 수신할 수 있고 또, 서비스되고 있는 클라이언트 애플리케이션에 기초하여, 예를 들어, 10킬로바이트의 정보만을 수신할 수 있는 클라이언트에 더 잘 대처하기 위해 큰 컨텐츠 파일로의 변환이 행해질 필요가 있다는 것을 인식할 수 있다. 프록시 메타데이터로서 전달되는 이 변환은 따라서, 도서 목록을 10 KB의 수정된 문서(1820)로 감소시키기 위해, 도서 목록에 적용될 수 있다. 이것은, 예를 들어, 줄거리를 제거하고 도서에 순위를 매기며 상위 50 개만을 포함하는 것 또는 당해 기술 분야에 잘 알려진 다른 변환에 의해 행해질 수 있다.
- <229> 변환이 완료되면, 수정된 문서(1820)가 푸시 클라이언트(510)로 전송된다.
- <230> 게다가, 도 4에서 알 수 있는 바와 같이, 지역 호출 메시지 저장 장치(452)는 변환 프로세스에서 제거된 여분의 컨텐츠를 저장하는데 사용될 수 있다.

- <231> 상기한 것의 이점은 서점이 하나의 사이트를 가질 수 있으며 그의 클라이언트를 전부에게 하나의 리스트를 전송 할 수 있다는 것이다. 여러 클라이언트가 모바일 무선 클라이언트가 아니기 때문에, 100 KB 파일이 이를 클라이언트에 적합할 수 있다. 또한, 변환 메타데이터도 제공함으로써, 서점은 모든 클라이언트에게 전송하는 하나의 리스트를 가질 수 있다. 당업자라면 잘 알고 있는 바와 같이, 대부분의 현재의 웹 기술은 모바일 클라이언트를 위한 별개의 웹 사이트를 필요로 하며, 이것은 상기한 해결책에 의해 극복된다.
- <232> 상기한 것은 또한 신디케이션 모델에 적합하며, 이제부터 도 19를 참조한다.
- <233> 당업자라면 잘 알고 있는 바와 같이, 네트워크 상태가 많은 양의 데이터를 수신하기에 최적이 아닌 경우에, 모바일 장치는 많은 양의 데이터를 수신하기를 원하지 않을 수 있다. 게다가, 네트워크 운영자는 네트워크 트래픽을 시간에 따라 보다 균일하게 분산시키기 위해 대역폭 사용의 피크 기간 동안 많은 양의 데이터를 전송하는 것을 회피하고자 할 수 있다. 이것은 도 19에 나타낸 바와 같은 푸시/풀 모델을 사용하여 달성될 수 있다.
- <234> 상기 도 4를 참조하여 기술한 바와 같이, 사용자가 현재 필요로 할 수 있는 것보다 더 많은 정보를 포함하는 컨텐츠가 제공될 수 있다. 예를 들어, 사용자가 그의 지역 내의 레스토랑에 대한 위치 정보를 요청하는 경우, 서비스 제공자는 그 지역에서 이용 가능한 다른 서비스 등의 광고를 추가하고자 할 수 있다. 그렇지만, 서비스 제공자는 이 부가적인 컨텐츠를 사용자에게 즉각 푸시하지 않고 부가적인 컨텐츠를 보여주는 컨텐츠의 제목(headline) 또는 테이블 등의 안내서(primer)를 제공하고자 할 수 있다.
- <235> 다른 상황에서, 컨텐츠가 너무 커서 사용자에게 전송할 수 없고, 사용자는 컨텐츠의 제1 부분만을 수신할 수 있고, 컨텐츠의 나머지는 지역 호출 메시지 저장 장치(452)에 저장된다.
- <236> 그 후에, 저장된 컨텐츠는 푸시 프록시(410)에 의해 또는 내 푸시 클라이언트(510)에 대해 요청될 때 푸시 클라이언트(510)로 전달될 수 있다.
- <237> 푸시 클라이언트(510)는 네트워크의 상태를 모니터링 할 수 있는 네트워크 상태 모니터(1910)를 포함한다. 푸시 클라이언트(510)는 어떤 상황에서 여분의 데이터만을 수신하고자 할 수 있다. 예를 들어, WiFi 및 셀룰러 옵션을 갖는 복합 모바일 장치에서, WiFi 연결을 통해 데이터를 제공하는 것이 더 저렴하며, 따라서 네트워크 상태 모니터(1910)는, 지연된 컨텐츠를 받기 이전에, 푸시 클라이언트(510)가 WiFi 네트워크에 연결될 때까지 기다릴 수 있다. 다른 대안으로서, 네트워크 상태 모니터는 로밍 요금(roaming charge)을 최소화하기 위해 클라이언트가 외부 네트워크(foreign network)에서 로밍 중인지 홈 네트워크(home network)에 연결되어 있는지를 검사할 수 있다. 네트워크 상태 모니터는 또한 장치에 대해 전용 데이터 채널이 설정되어 있는지 여부를 알아보기 위해 검사를 할 수 있다. 당업자라면, 네트워크 상태 모니터(1910)가 또한, 지연된 데이터를 푸시 클라이언트(510)로 전달하라고 요청하기 이전에, 여러가지 다른 전제 조건에 대해 검사를 할 수 있다는 것을 잘 알고 있을 것이다.
- <238> 무선 네트워크(130)는 또한 푸시 클라이언트(510) 및 푸시 프록시(410) 중 어느 하나 또는 그 둘다에 데이터 전달의 비용에 관한 정보를 제공할 수 있다. 당업자라면 잘 알고 있는 바와 같이, 컨텐츠의 전달에 대해 여러가지 피크 기간이 있다. 교통 정보의 경우에, 피크 기간은 사람들이 직장에 가고 직장으로부터 오는 평일의 시작 및 끝에 있을 수 있다. 주식 시세의 경우, 피크 기간은 시장이 열려 있는 시간 동안일 수 있다. 다른 피크 기간들도 존재한다. 데이터 트래픽을 평균화하기 위해, 네트워크에서의 현재의 데이터 사용에 기초하여 네트워크가 서로 다른 요금을 부과하는 것이 바람직할 수 있다. 따라서, 피크 기간 동안에, 한밤중 등의 비피크 기간 동안보다 더 높은 요금이 부과될 수 있다. 따라서, 무선 네트워크(130)는 푸시 클라이언트(510) 상의 지역 호출 관리자(552)에 또 푸시 프록시(410) 상의 푸시 스케줄러(454)에 전달 비용 통지를 제공한다.
- <239> 일 실시예에서, 컨텐츠 제공자(110)로부터 푸시 프록시(410)로 전달되는 데이터는 클라이언트에 대한 그의 중요도에 기초하여 순위가 매겨질 수 있다. 어떤 정보는 즉시 전달되도록 메타데이터를 통해 지정될 수 있다. 다른 정보는 네트워크 비용이 제1 값(예를 들어, 메가바이트당 10¢)보다 작을 때 전달되도록 지정될 수 있고, 다른 데이터는 네트워크 비용이 제2 값(예를 들어, 메가바이트당 5¢)보다 아래로 떨어질 때 전달되도록 지정될 수 있다. 따라서, 푸시 스케줄러(454)는 지역 호출 메시지 저장 장치(452)에 저장되어 있는 데이터를 고려하고 푸시 에이전트(push agent)(444)에게 푸시 클라이언트(510) 상의 푸시 에이전트(544)에게 지역 데이터를 전달하도록 지시한다.
- <240> 다른 대안으로서, 지역 호출 관리자(552)는 또한 무선 네트워크(130)로부터 전송되는 바와 같이 네트워크 상황을 모니터링 할 수 있고, 데이터 레이트가 어떤 레이트보다 아래에 있는 경우, 컨텐츠 풀 브로커(content pull broker)(554)에게 지역 호출 메시지 저장 장치(452)로부터 컨텐츠를 풀링하도록 요청할 수 있다.

- <241> 다른 대안으로서, 지역 호출 관리자(552)는 또한, 모바일 장치가 WiFi 네트워크와 연결되어 있는 경우와 같이, 네트워크 상태가 많은 양의 데이터를 풀링하는데 적합한 것임을 파악할 수 있고 컨텐츠 풀 브로커(554)에게 지역 호출 메시지 저장 장치(452)로부터 데이터를 풀링하도록 요청할 수 있다.
- <242> 또한, 당업자라면 잘 알고 있는 바와 같이, 사용자는 컨텐츠가 풀링되도록 항상 요청할 수 있다. 따라서, 사용자 요청(1940)은 또한 지역 호출 메시지 저장 장치(452)로부터 데이터를 풀링하도록 컨텐츠 풀 브로커(554)를 트리거하는데 사용될 수 있다.
- <243> 푸시 스케줄러(454) 및 지역 호출 관리자(552)에 저장되어 있는 규칙은 컨텐츠의 분류에 기초한 정적 메타데이터일 수 있다. 이 규칙은 또한 전달된 특정의 데이터에 대한 동적 메타데이터에 기초할 수 있다. 이 경우에, 컨텐츠 제공자(110)는 데이터를 분류하였다.
- <244> 이제부터 도 20을 참조한다. 당업자라면 잘 알고 있는 바와 같이, 데이터는 2가지 형태, 즉 선형 또는 비선형 중 어느 하나일 수 있다. 선형 데이터는, 예를 들어, 선형적 방식으로 이동하는 어레이 또는 문자열 또는 컨텐츠일 수 있다. 비선형 데이터는, 그와 반대로, 서로와 선형적으로 관련되어 있지 않은 데이터이며 컨텐츠 맵 또는 링크와의 복잡한 의존관계를 포함할 수 있다.
- <245> 선형 컨텐츠의 경우, 프래그먼트화는 단지 선형적 진행에 기초하여 데이터를 여러가지 성분으로 분할하는 것을 포함한다. 데이터는 세그먼트로 분할되고, 이 세그먼트가 푸시 클라이언트(410)로 전달된다. 도 20에 나타낸 바와 같이, 프래그먼트화 프로세서(2010)는 컨텐츠(2012)와 상호작용하여, 컨텐츠가 선형적 진행으로 파싱될 수 있는지를 결정한다. 프래그먼트화 프로세서(2010)는 그 다음에 데이터를, 도 20의 예에서, 세그먼트(2014, 2016, 2018)로 분할하고, 도 20에 나타낸 바와 같이, 제1 세그먼트(2014)를 전달하는 한편 제2 및 제3 세그먼트(2016, 2018)의 전달을 각각 지연시킨다.
- <246> 커서 관리 모듈(2030)은 어느 세그먼트가 전달되었는지를 추적하고 그 다음 세그먼트를 차례대로 전달한다.
- <247> 도 21을 참조하면, 비선형 데이터는 더 지능적인 방식으로 분할될 필요가 있다. 게다가, 다른쪽 종단에서, 세그먼트를 재구성하기 위해, 메타데이터가 필요하다.
- <248> 프래그먼트화 프로세서(2110)는 메타데이터 기반 분석에 기초하여 컨텐츠를 분석한다. 이들은, 논리적으로 요구되는 경우, 어떤 세그먼트 또는 데이터 요소를 함께 유지하는 것을 포함할 수 있다. 프래그먼트화 프로세서(2110)는 컨텐츠(2112)를 분석하고 이 컨텐츠를 논리적 규칙에 기초하여 세그먼트로 분할한다. 각각의 세그먼트는 컨텐츠 및, 예를 들어, 각각의 세그먼트에 대한 의존관계, 맵 및 내비게이션 규칙을 포함하는 메타데이터를 포함한다.
- <249> 일단 분할되면, 제1 세그먼트(2114)는 푸시 클라이언트(510)로 전송되고, 나머지 세그먼트(2116, 2118)의 전달은 도 21에 나타낸 바와 같이 지연된다. 세그먼트 내비게이션 블록(2130)은 그 다음에 어느 세그먼트를 전송할지를 처리한다. 당업자라면 잘 알고 있는 바와 같이, 제1 세그먼트(2114)는 데이터 부분 및 메타데이터 부분을 포함한다. 세그먼트(2114)의 메타데이터 부분은 컨텐츠를 어떻게 재구성할지를 컨텐츠 의존성 모듈(564)에 알려주기 위해 프래그먼트화 프로세서(2110)에 의해 부가되는 메타데이터의 계층이다. 제1 세그먼트(2114)의 데이터 부분은 컨텐츠 및 채널과 연관되거나 컨텐츠와 연관된 메타데이터 둘다를 포함할 수 있다.
- <250> 세그먼트 내비게이션 블록(2130)은 사용자가 데이터를 어떻게 순회할지를 처리하도록 구성되어 있다. 예를 들어, 데이터가 트리 포맷으로 되어 있고 또 사용자가 트리의 제1 브랜치를 따라 아래쪽으로 가는 경우, 세그먼트 내비게이션 블록(2130)은 사용자가 내비게이션한 요소로부터 도달될 수 있는 트리 내의 다른 브랜치를 푸시 클라이언트(410)로 전달할 수 있다.
- <251> 예를 들어, 트리는 회사에 대한 구조와 함께 직원 이름을 갖는 직원 데이터베이스를 포함할 수 있다. 도 21에 기초하여, 사용자가 조직의 특정 부서로 내비게이션하는 경우, 세그먼트 내비게이션 블록(2130)은 그 부서 내의 그룹들에 대한 그룹 프래그먼트를 전달할 수 있다. 이어서, 사용자가 그 부서 내의 특정 그룹으로 내비게이션하는 경우, 세그먼트 내비게이션 블록(2130)은 그 그룹 내의 직원에 관한 정보 프래그먼트를 전달할 수 있다.
- <252> 따라서, 상기한 구성은 데이터가 논리적 성분으로 분할되는 것을 필요로 한다. 모든 유형 및 컨텐츠에 식별자가 할당되고, 정보를 안내서(primer)와 함께 전달하는 구조적 정보가 생성된다.
- <253> 따라서, 상기한 구성은, 시스템의 구조를 변경하지 않고 애플리케이션 및 컨텐츠가 추가될 수 있는 일반적인 시스템에서 사용될 수 있는 동적 컨텐츠 전달을 위한 아키텍처를 제공한다. 컨텐츠가 그를 수신하는 애플리케이션

에 적합하도록 조정되고 상기한 바에 따라 프래그먼트화될 수 있다.

<254> 당업자라면 잘 알고 있는 바와 같이, 푸시 클라이언트 및 클라이언트 애플리케이션은 임의의 모바일 장치 상에 존재할 수 있다. 한 예시적인 모바일 장치에 대해 도 22를 참조하여 이하에서 기술한다. 이것은 제한하려는 것이 아니며 예시의 목적을 위한 것이다.

<255> 도 22는 본 발명의 장치 및 방법의 양호한 실시예에서 사용하기에 적합한 이동국을 나타낸 블록도이다. 이동국(2200)은 양호하게는 적어도 음성 및 데이터 통신 기능을 갖는 양방향 무선 통신 장치이다. 이동국(2200)은 양호하게는 인터넷을 통해 다른 컴퓨터 시스템과 통신할 수 있는 기능을 갖는다. 제공된 정확한 기능에 따라, 무선 장치는, 예로서, 데이터 메시징 장치, 양방향 페이저, 무선 이메일 장치, 데이터 메시징 기능을 갖는 셀룰러 전화, 무선 인터넷 기기, 또는 데이터 통신 장치라고 할 수 있다.

<256> 이동국(2200)이 양방향 통신이 가능한 경우, 이동국(2200)은 수신기(2212) 및 송신기(2214) 둘다를 포함하는 통신 서브시스템(2211)은 물론, 하나 이상의, 양호하게는 내장된 또는 내장형의, 안테나 요소(2216, 2218), 국부 발진기(LO)(2213), 및 디지털 신호 처리기(DSP)(2220) 등의 처리 모듈을 포함하게 된다. 통신 분야의 당업자에게는 명백한 바와 같이, 통신 서브시스템(2211)의 특정의 설계는 장치가 사용되도록 되어 있는 통신 네트워크에 의존한다.

<257> 네트워크 액세스 요건도 네트워크(2219)의 유형에 따라 변하게 된다. 어떤 CDMA 네트워크에서, 네트워크 액세스는 가입자 또는 이동국(2200)의 사용자와 연관되어 있다. CDMA 이동국은 CDMA 네트워크 상에서 동작하기 위해서 분리가능 사용자 식별 모듈(removable user identity module, RUIM) 또는 가입자 식별 모듈(subscriber identity module, SIM)을 필요로 할 수 있다. SIM/RUIM 인터페이스(2244)는 통상적으로 디스크 또는 PCMCIA 카드처럼 SIM/RUIM 카드가 삽입되고 배출될 수 있는 카드-슬롯과 유사하다. SIM/RUIM 카드는 대략 64 K의 메모리를 가질 수 있고 또 많은 키 구성(2251)과, 식별자 및 가입자 관련 정보 등의 다른 정보(2253)를 가질 수 있다.

<258> 요구되는 네트워크 등록 또는 활성화 절차가 완료되었을 때, 이동국(2200)은 네트워크(2219)를 거쳐 통신 신호를 전송 및 수신할 수 있다. 도 22에 나타낸 바와 같이, 네트워크(2219)는 이동국과 통신하는 다수의 기지국으로 이루어질 수 있다. 예를 들어, 하이브리드 CDMA 1x EVDO 시스템에서, CDMA 기지국 및 EVDO 기지국은 이동국과 통신하고, 이동국은 둘다에 동시에 연결되어 있다. EVDO 및 CDMA 1x 기지국은 이동국과 통신하기 위해 서로 다른 페이징 슬롯을 사용한다.

<259> 통신 네트워크(2219)를 통해 안테나(2216)에 의해 수신되는 신호는 수신기(2212)에 입력되고, 이 수신기(2212)는 신호 증폭, 주파수 다운 컨버전, 필터링, 채널 선택, 및 도 22에 나타낸 예시적인 시스템에서의 아날로그-디지털(A/D) 변환 등과 같은 통상의 수신기 기능을 수행할 수 있다. 수신된 신호의 A/D 변환은 복조 및 디코딩 등의 보다 복잡한 통신 기능이 DSP(2220)에서 수행될 수 있게 해준다. 유사한 방식으로, 전송될 신호는, 예를 들어, 변조 및 인코딩을 비롯하여, DSP(2220)에 의해 처리되고, 디지털-아날로그 변환, 주파수 업 컨버전, 필터링, 증폭, 및 안테나(2218)를 거쳐 통신 네트워크(2219)를 통한 전송을 위해 송신기(2214)에 입력된다. DSP(2220)는 통신 신호를 처리할 뿐만 아니라, 수신기 및 송신기 제어도 제공한다. 예를 들어, 수신기(2212) 및 송신기(2214)에서 통신 신호에 적용되는 이득은 DSP(2220)에 구현되는 자동 이득 제어 알고리즘을 통해 적용적으로 제어될 수 있다.

<260> 이동국(2200)은 양호하게는 장치의 전체적인 동작을 제어하는 마이크로프로세서(2238)를 포함한다. 적어도 데이터 및 음성 통신을 포함하는 통신 기능은 통신 서브시스템(2211)을 통해 수행된다. 마이크로프로세서(2238)는 또한 디스플레이(2222), 플래쉬 메모리(2224), 랜덤 액세스 메모리(RAM)(2226), 보조 입/출력(I/O) 서브시스템(2228), 직렬 포트(2230), 2개 이상의 키보드 또는 키패드(2232), 스피커(2234), 마이크로폰(2236), 단거리 통신 서브시스템 등의 다른 통신 서브시스템(2240), 및 일반적으로 2242로 표시된 임의의 다른 장치 서브시스템 등의 다른 장치 서브시스템들과 상호작용한다. 직렬 포트(2230)는 USB 포트 또는 당업자에게 알려져 있는 다른 포트를 포함할 수 있다.

<261> 도 22에 나타낸 서브시스템들 중 어떤 것들은 통신 관련 기능을 수행하는 반면, 다른 서브시스템들은 "상주형(resident)" 또는 온-디바이스(on-device) 기능을 제공할 수 있다. 주목할 만한 것은, 예를 들어, 키보드(2232) 및 디스플레이(2222) 등의 어떤 서브시스템들은 통신 네트워크를 통해 전송하기 위한 텍스트 메시지를 입력하는 것 등의 통신 관련 기능, 및 계산기 또는 작업 목록 등의 장치-상주형 기능(device-resident function) 둘다에 대해 사용될 수 있다.

<262> 마이크로프로세서(2238)에 의해 사용되는 운영 체제 소프트웨어는 양호하게는 플래쉬 메모리(2224) 등의 영속적

저장 장치에 저장되어 있으며, 이 저장 장치는 그 대신에 판독 전용 메모리(ROM) 또는 유사한 저장 요소(도시 생략)일 수 있다. 당업자라면 운영 체제, 특정의 장치 애플리케이션, 또는 그의 일부분이 일시적으로 RAM(2262) 등의 휘발성 메모리에 로드될 수 있다는 것을 잘 알 것이다. 수신된 통신 신호도 RAM(2262)에 저장될 수 있다.

<263> 도시된 바와 같이, 플래쉬 메모리(2224)는 컴퓨터 프로그램(2258) 및 프로그램 데이터(2250, 2252, 2254, 2256) 저장 둘다를 위해 서로 다른 영역으로 분할될 수 있다. 이들 서로 다른 저장 유형은 각각의 프로그램이 플래쉬 메모리(2224)의 일부분을 그 자신의 데이터 저장 요건을 위해 할당할 수 있음을 나타낸다. 마이크로프로세서(2238)는, 그의 운영 체제 기능 이외에, 양호하게는 이동국 상에서의 소프트웨어 애플리케이션의 실행을 가능하게 해준다. 예를 들어, 적어도 데이터 및 음성 통신 애플리케이션을 포함하는 기본적인 동작을 제어하는 미리 정해진 일련의 애플리케이션이 통상적으로 제조 동안에 이동국(2200) 상에 설치된다. 다른 애플리케이션은 차후에 또는 동적으로 설치될 수 있다.

<264> 양호한 소프트웨어 애플리케이션은 이메일, 일정 관리(calendar event), 음성 메일, 약속, 및 작업 항목(이에 한정되지 않음) 등의 이동국의 사용자에 관련한 데이터 항목을 구성 및 관리할 수 있는 기능을 갖는 개인 정보 관리자(PIM) 애플리케이션일 수 있다. 자연히, PIM 데이터 항목의 저장을 용이하게 해주기 위해 하나 이상의 메모리 저장 장치가 이동국 상에서 이용가능하게 된다. 이러한 PIM 애플리케이션은 양호하게는 무선 네트워크(2219)를 통해 데이터 항목을 전송 및 수신할 수 있는 기능을 가지게 된다. 양호한 실시예에서, PIM 데이터 항목은, 무선 네트워크(2219)를 통해, 호스트 컴퓨터 시스템에 저장되거나 그와 연관된 이동국 사용자의 대응하는 데이터 항목과 매끄럽게 통합, 동기화 및 갱신될 수 있다. 추가의 애플리케이션도 네트워크(2219), 보조 I/O 서브시스템(2228), 직렬 포트(2230), 단거리 통신 서브시스템(2240) 또는 임의의 다른 적당한 서브시스템(2242)을 통해 이동국(2200)에 로드되고, 마이크로프로세서(2238)에 의해 실행하기 위해 사용자에 의해 RAM(2262) 또는 양호하게는 비휘발성 저장 장치(도시 생략)에 설치될 수 있다. 애플리케이션 설치에서의 이러한 유연성은 장치의 기능을 향상시키고 또 향상된 온-디바이스 기능, 통신-관련 기능, 또는 둘다를 제공할 수 있다. 예를 들어, 보안 통신 애플리케이션은 전자 상거래 기능 및 다른 이러한 금융 거래가 이동국(2200)을 사용하여 수행될 수 있게 해줄 수 있다.

<265> 데이터 통신 모드에서, 텍스트 메시지 또는 웹 페이지 다운로드 등의 수신된 신호는 통신 서브시스템(2211)에 의해 처리되고 마이크로프로세서(2238)에 입력되며, 이 마이크로프로세서(2238)는 양호하게는 수신된 신호를 디스플레이(2222)로 또는 다른 대안으로서 보조 I/O 장치(2228)로 출력하기 위해 추가적으로 처리한다. 푸시 클라이언트(140, 510)와 동등할 수 있는 푸시 클라이언트(2260)도 이 입력을 처리할 수 있다.

<266> 이동국(2200)의 사용자는 또한 디스플레이(2222) 및 아마도 보조 I/O 장치(2228)와 함께, 양호하게는 완전한 영 숫자 키보드 또는 전화형 키패드인 키보드(2232)를 사용하여, 예를 들어, 이메일 메시지 등의 데이터 항목을 작성할 수 있다. 이러한 작성된 항목은 이어서 통신 서브시스템(2211)을 통해 통신 네트워크를 거쳐 전송될 수 있다.

<267> 음성 통신의 경우, 이동국(2200)의 전체적인 동작은, 수신된 신호가 양호하게는 스피커(2234)로 출력되고 전송을 위한 신호가 마이크로폰(2236)에 의해 생성되는 것을 제외하고는 유사하다. 음성 메시지 녹음 서브시스템 등의 대안의 음성 또는 오디오 I/O 서브시스템도 이동국(2200) 상에 구현될 수 있다. 음성 또는 오디오 신호 출력이 양호하게는 주로 스피커(2234)를 통해 달성되지만, 예를 들어, 호출측 당사자의 신원의 표시, 음성 통화의 기간, 또는 다른 음성 통화 관련 정보를 제공하기 위해 디스플레이(2222)도 사용될 수 있다.

<268> 도 22의 직렬 포트(2230)는 통상적으로 사용자의 데스크톱 컴퓨터(도시 생략)과의 동기화가 바람직할 수 있는 개인 휴대 단말기(PDA)형 이동국에서 구현되지만, 이는 선택적인 장치 요소이다. 이러한 포트(2230)는 사용자가 외부 장치 또는 소프트웨어 애플리케이션을 통해 사용자 설정(preference)을 설정할 수 있게 해주고 또 무선 통신 네트워크 이외의 것을 통해 이동국(2200)으로의 정보 또는 소프트웨어 다운로드를 제공함으로써 이동국(2200)의 기능을 확장시킨다. 대안의 다운로드 경로는 예를 들어 직접 연결(direct connection), 따라서 믿을 만한 신뢰된 연결을 통해 장치 상으로 암호화 키를 로드하고 그에 의해 안전한 장치 통신을 가능하게 해주는데 사용될 수 있다. 당업자라면 잘 알고 있는 바와 같이, 직렬 포트(2230)는 또한 모뎀으로서 동작하도록 모바일 장치를 컴퓨터에 연결하는데 사용될 수 있다.

<269> 단거리 통신 서브시스템 등의 다른 통신 서브시스템(2240)은 이동국(2200)과 다른 시스템 또는 장치(꼭 유사한 장치일 필요는 없음) 간의 통신을 제공할 수 있는 또 다른 선택적인 구성요소이다. 예를 들어, 서브시스템(2240)은 유사한 기능의 시스템 및 장치와의 통신을 제공하기 위해 적외선 장치와 관련 회로 및 구성요소 또는 블루투스™ 통신 모듈을 포함할 수 있다.

<270> 본 명세서에 기술된 실시예는 본 발명의 기술의 구성요소에 대응하는 구성요소를 갖는 구조, 시스템 또는 방법의 예이다. 이 기술된 설명은 당업자로 하여금 마찬가지로 본 발명의 기술의 구성요소에 대응하는 대안의 구성요소를 갖는 실시예를 제조 및 사용할 수 있게 해줄 수 있다. 본 발명의 기술의 의도된 범위는 따라서 본 명세서에 기술된 본 발명의 기술과 다르지 않은 기타의 구조, 시스템 또는 방법을 포함하며, 또한 본 명세서에 기술된 본 발명의 기술과 실질적으로 다르지 않은 기타의 구조, 시스템 또는 방법도 포함한다.

### 발명의 효과

<271> 본 발명의 구성에 따르면, 아키텍처의 변경없이 애플리케이션 및 컨텐츠 제공자가 추가될 수 있는 일반적인 동적 컨텐츠 전달 아키텍처를 제공할 수 있다.

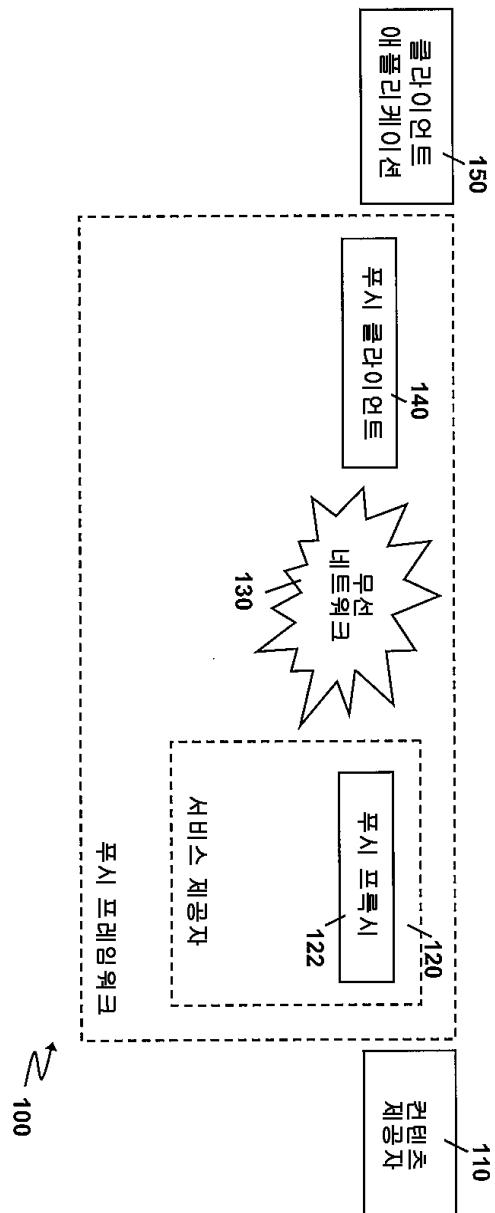
### 도면의 간단한 설명

- <1> 도 1은 동적 컨텐츠 전달 시스템에 대한 기본적인 아키텍처의 블록도.
- <2> 도 2는 도 1의 동적 컨텐츠 전달 시스템의 또 다른 아키텍처를 나타낸 블록도.
- <3> 도 3은 컨텐츠 및 메타데이터 흐름을 나타낸 도 1의 블록도.
- <4> 도 4는 본 발명의 시스템 및 방법과 관련하여 사용될 수 있는 푸시 프록시(push proxy)를 나타낸 블록도.
- <5> 도 5는 본 발명의 시스템 및 방법과 관련하여 사용될 수 있는 푸시 클라이언트(push client)를 나타낸 블록도.
- <6> 도 6은 컨텐츠 및 메타데이터의 다층 엔베로프 모델(multilayer envelope model)을 나타낸 블록도.
- <7> 도 7은 각각의 엔베로프에 대한 처리 단계 동적 메타데이터를 나타낸 도 6의 블록도.
- <8> 도 8은 정적 및 동적 메타데이터를 사용하는 프로세싱을 추가하여 나타낸 도 6의 블록도.
- <9> 도 9는 단일의 공유 푸시 클라이언트(shared push client)에 대한 애플리케이션의 등록 프로세스를 나타낸 블록도.
- <10> 도 10은 푸시 클라이언트의 폴(pool)을 관리하는 푸시 컨테이너(push container)에 대한 애플리케이션의 등록 프로세스를 나타낸 블록도.
- <11> 도 11은 애플리케이션이 컨텐츠 프로세서 및 소켓 리스너(socket listener)에 등록하는 것을 나타낸 블록도.
- <12> 도 12는 컨텐츠 제공자가 단일의 공유 푸시 프록시에 등록하는 것을 나타낸 블록도.
- <13> 도 13은 컨텐츠 제공자가 푸시 프록시의 폴을 관리하는 푸시 컨테이너에 등록하는 것을 나타낸 블록도.
- <14> 도 14는 컨텐츠 제공자와 클라이언트 애플리케이션 간의 등록 메시지를 나타낸 흐름도.
- <15> 도 15는 푸시 클라이언트와 푸시 프록시 간의 등록 동안의 상호작용을 나타낸 블록도.
- <16> 도 16은 푸시 프록시와 컨텐츠 제공자 간의 등록 동안의 상호작용을 나타낸 블록도.
- <17> 도 17은 컨텐츠 제공자와 처리 요소 간의 컨텐츠 및 메타데이터의 흐름을 나타낸 흐름도.
- <18> 도 18은 컨텐츠에 대한 예시적인 변환 애플리케이션을 나타낸 블록도.
- <19> 도 19는 컨텐츠 신디케이션 모델(content syndication model)의 블록도.
- <20> 도 20은 선형 프레그먼트화 프로세스(linear fragmentation process)의 블록도.
- <21> 도 21은 비선형 프레그먼트화 프로세스(non-linear fragmentation process)의 블록도.
- <22> 도 22는 본 발명의 방법 및 시스템과 관련하여 사용될 수 있는 예시적인 모바일 장치의 블록도.
- <23> <도면의 주요 부분에 대한 부호의 설명>
- <24> 100: 푸시 프레임워크
- <25> 110: 컨텐츠 제공자
- <26> 120: 서비스 제공자

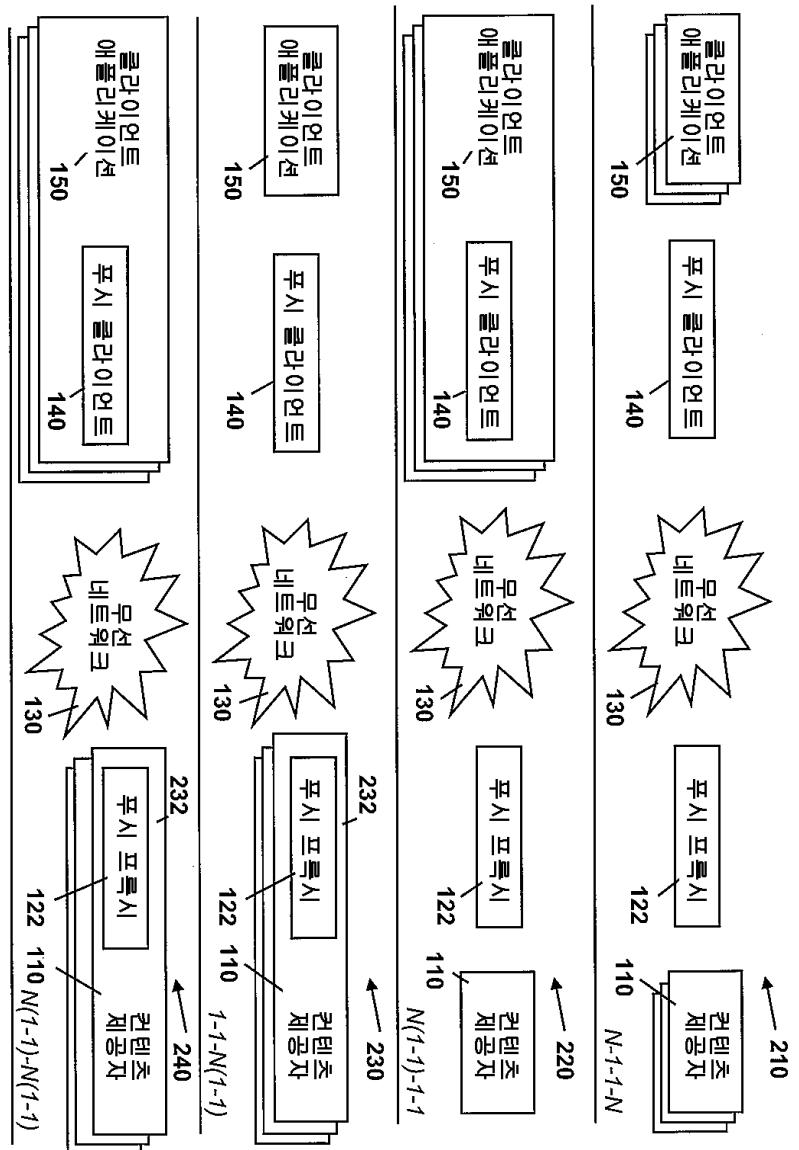
- <27> 122: 푸시 프록시
- <28> 130: 무선 네트워크
- <29> 140: 푸시 클라이언트
- <30> 150: 클라이언트 애플리케이션

## 도면

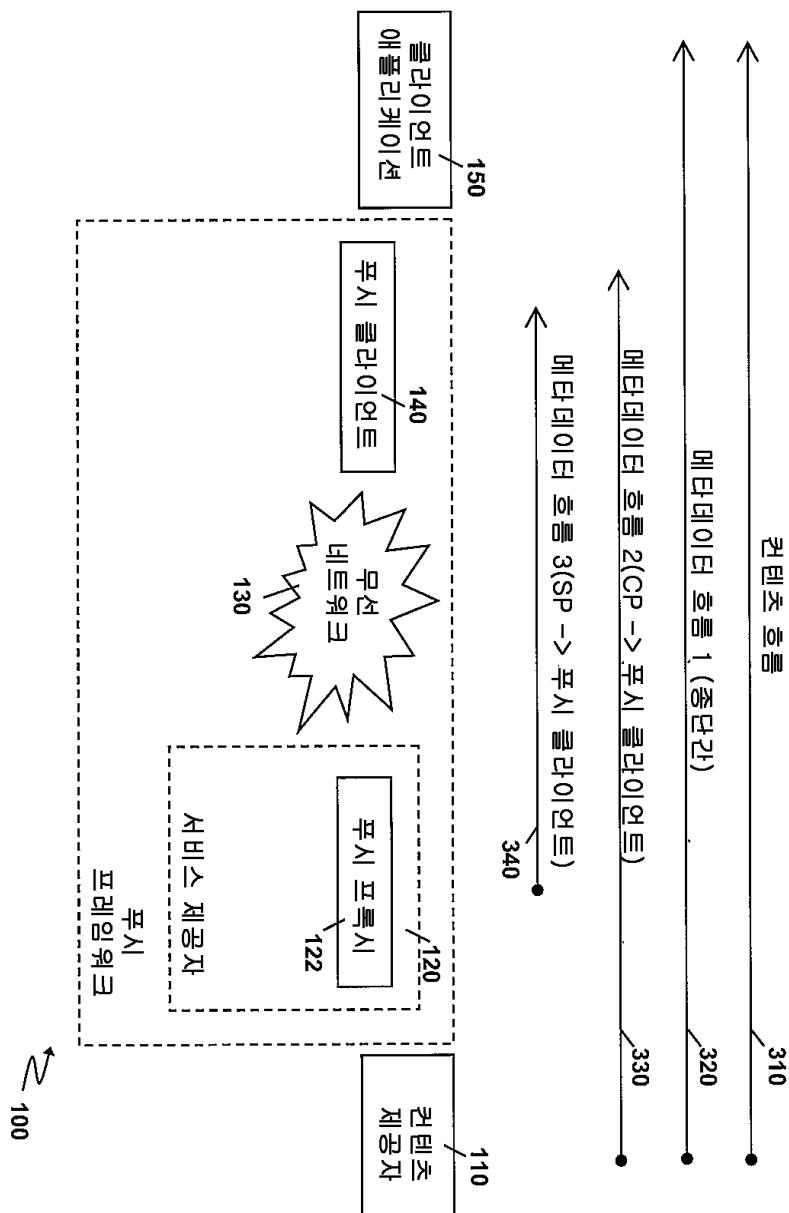
도면1



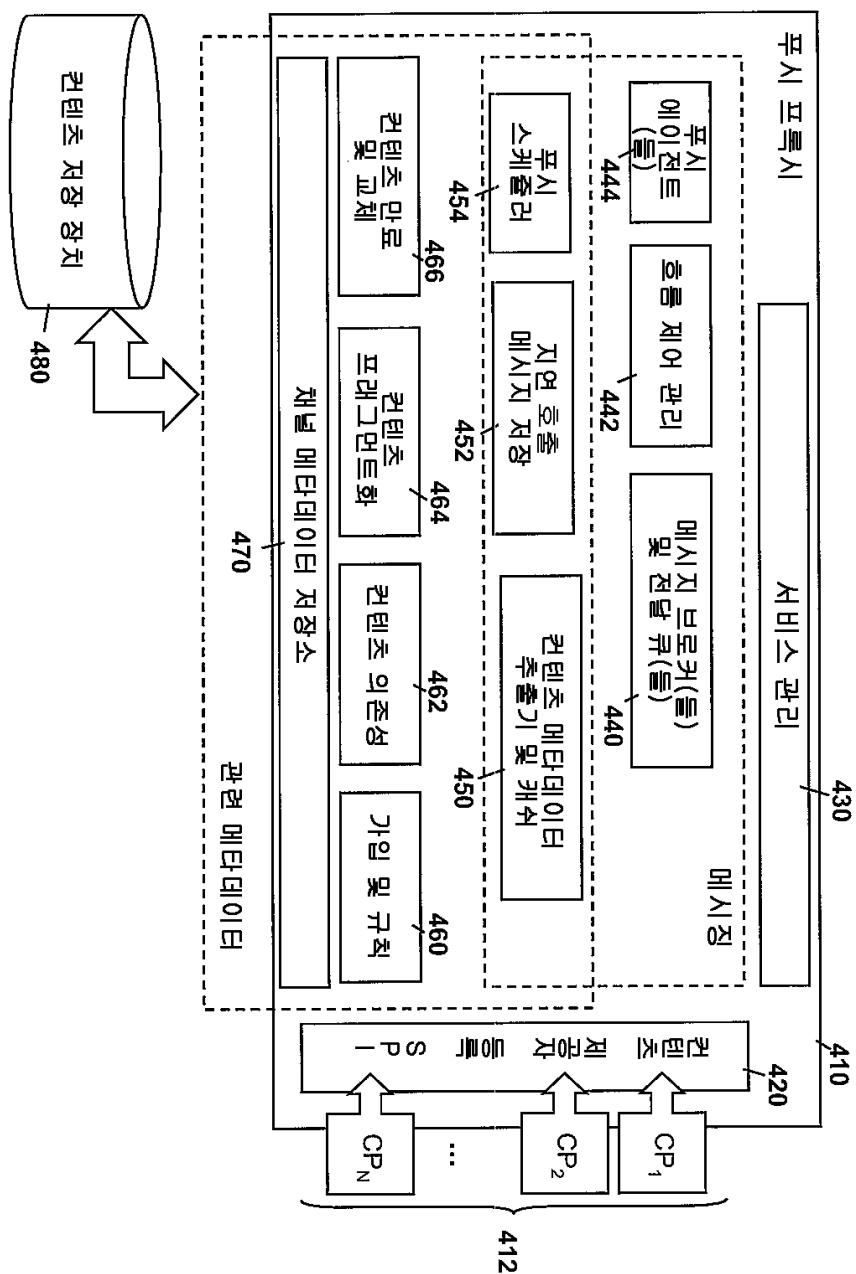
도면2



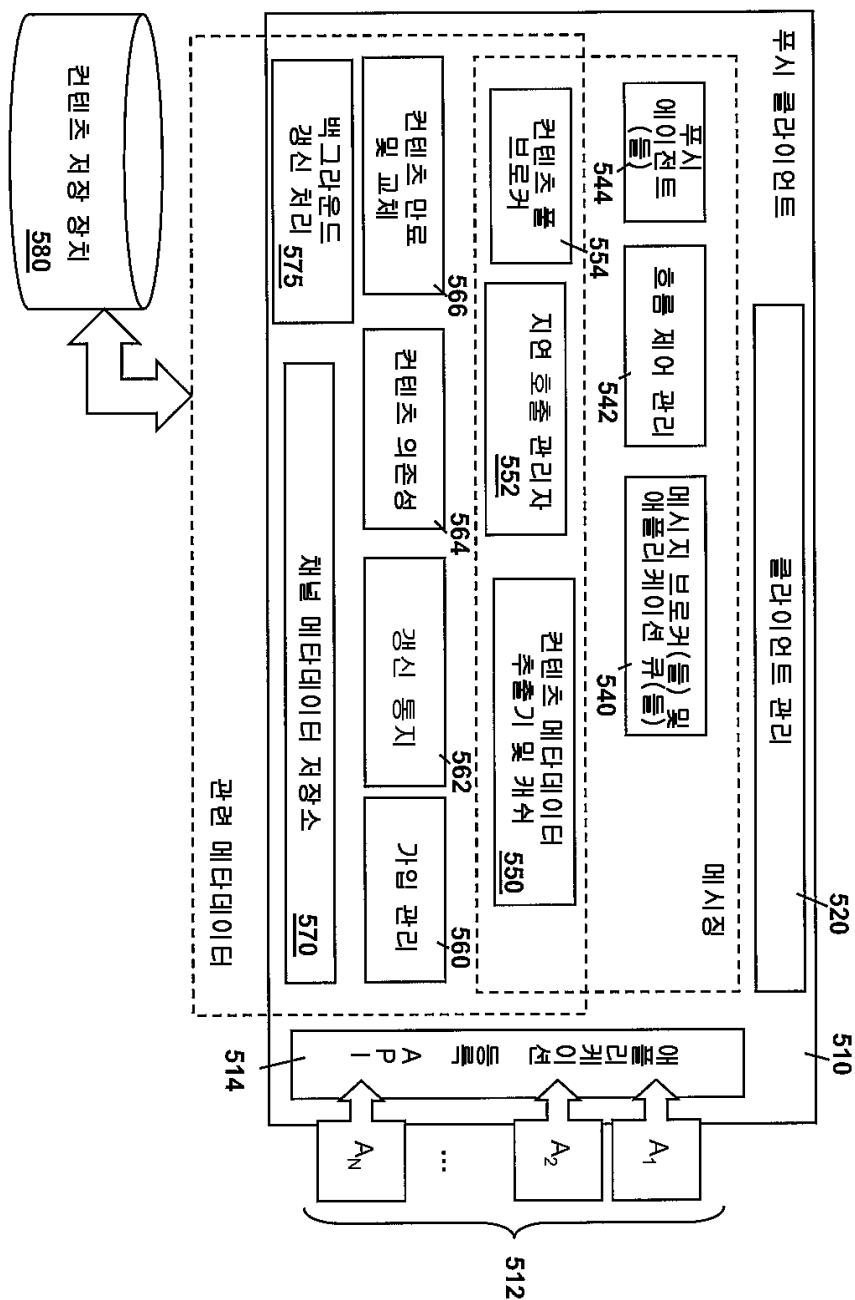
도면3



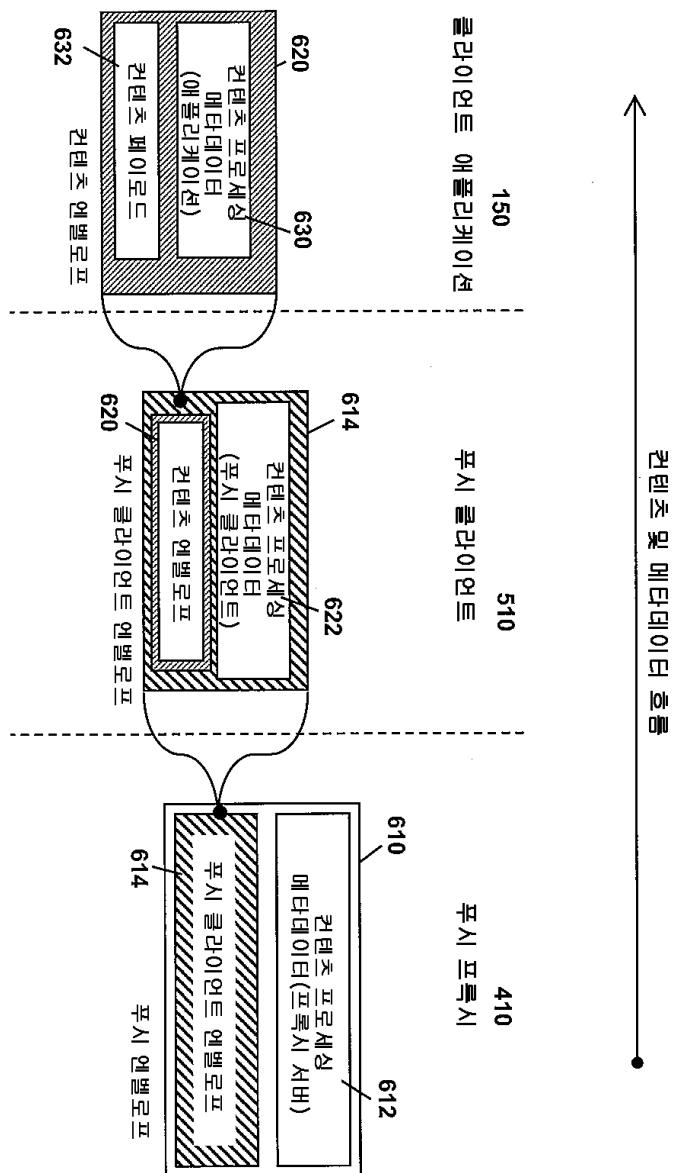
도면4



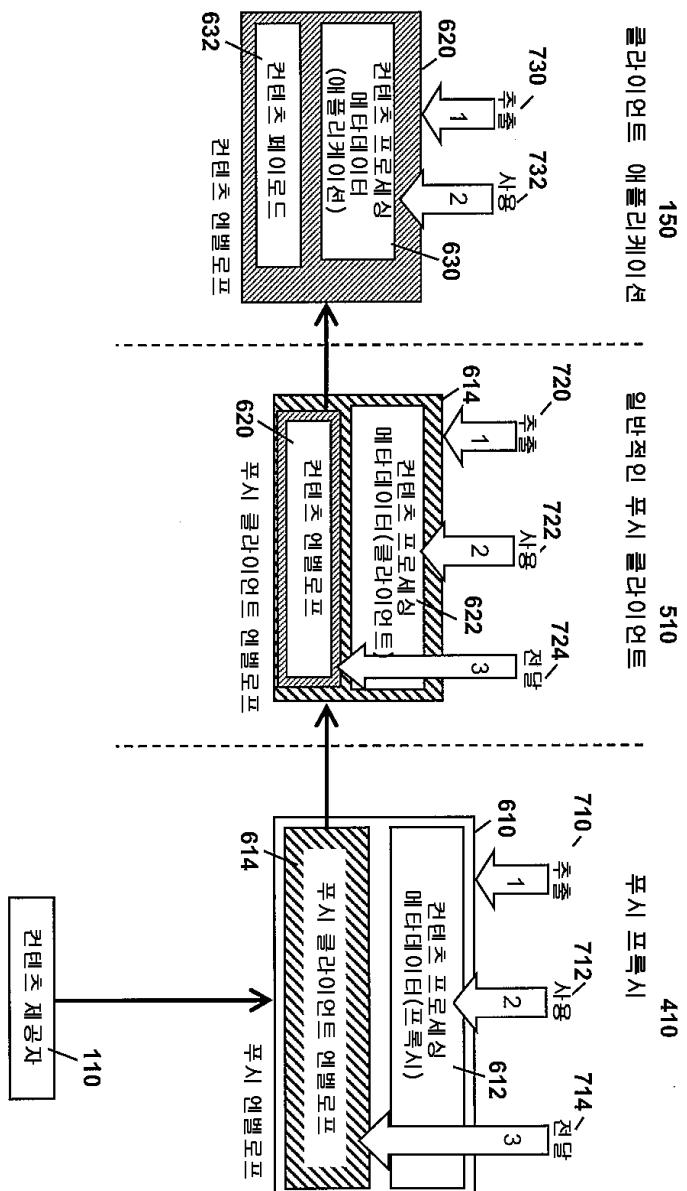
도면5



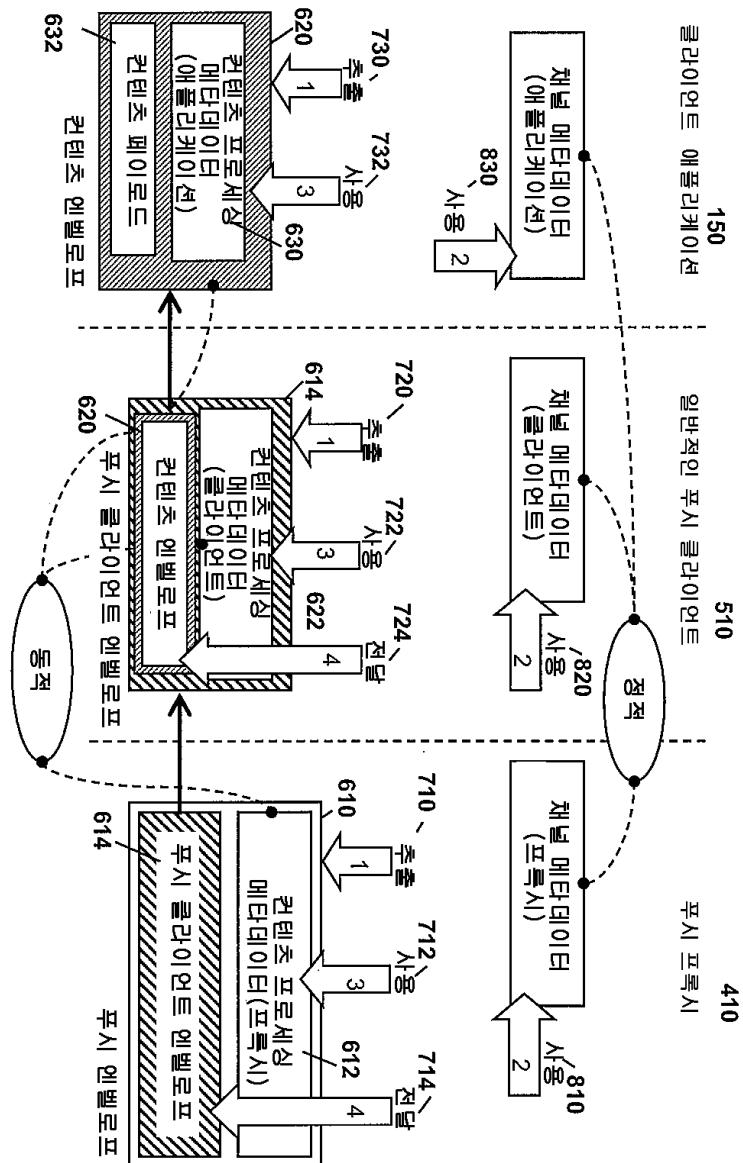
도면6



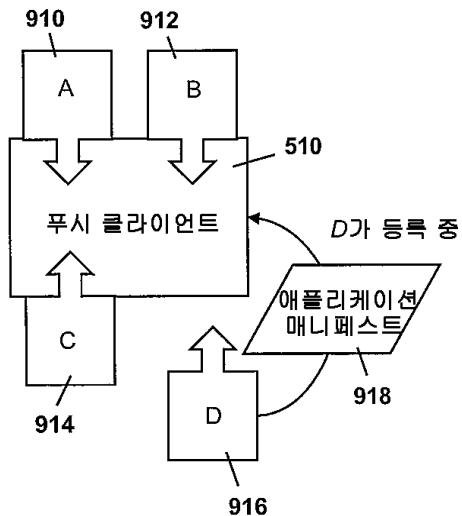
도면7



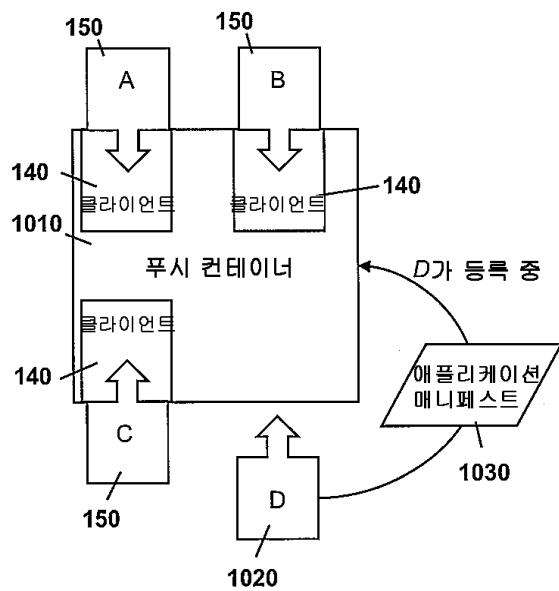
도면8



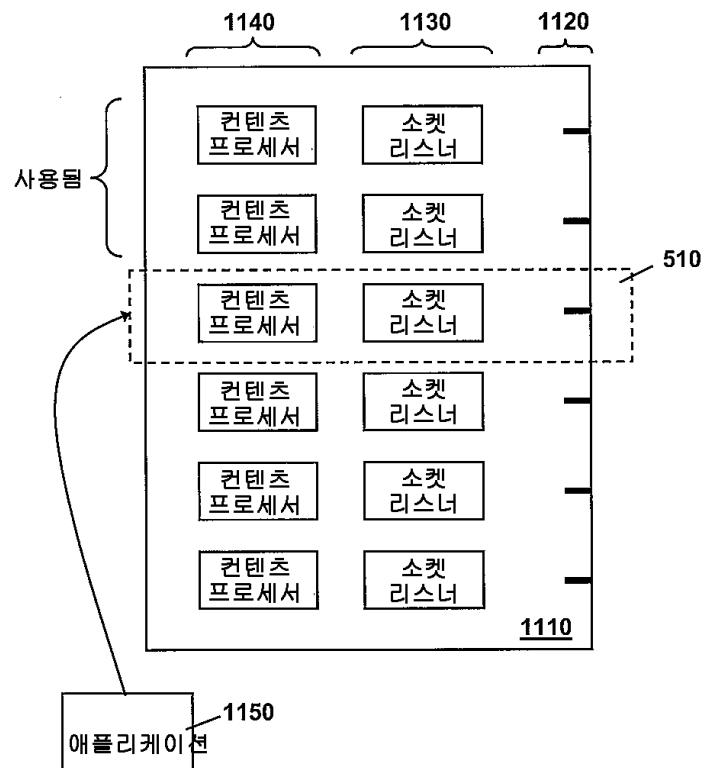
도면9



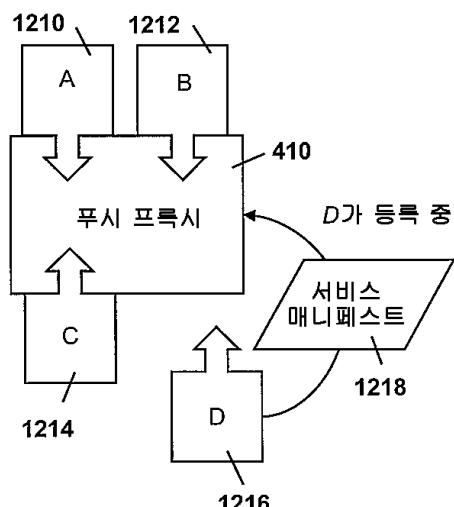
도면10



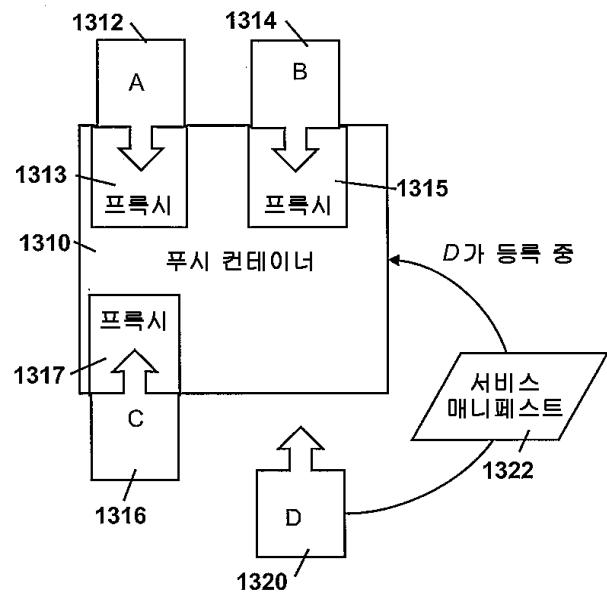
도면11



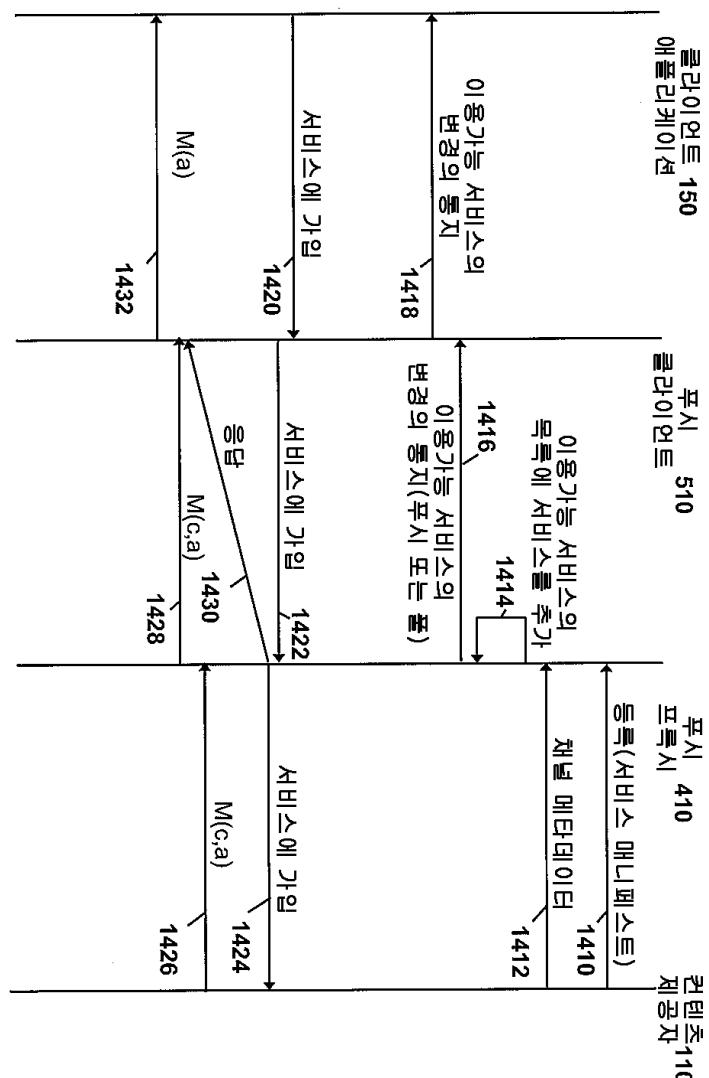
도면12



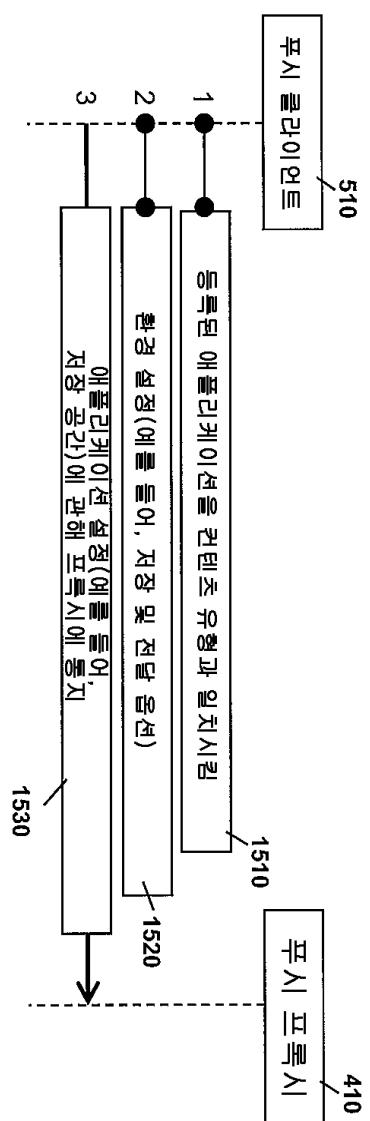
도면13



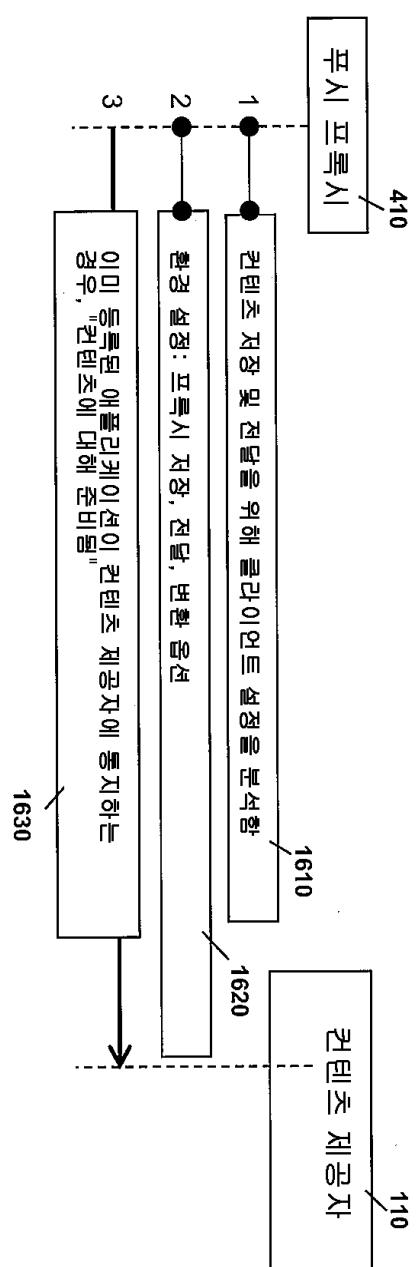
도면14



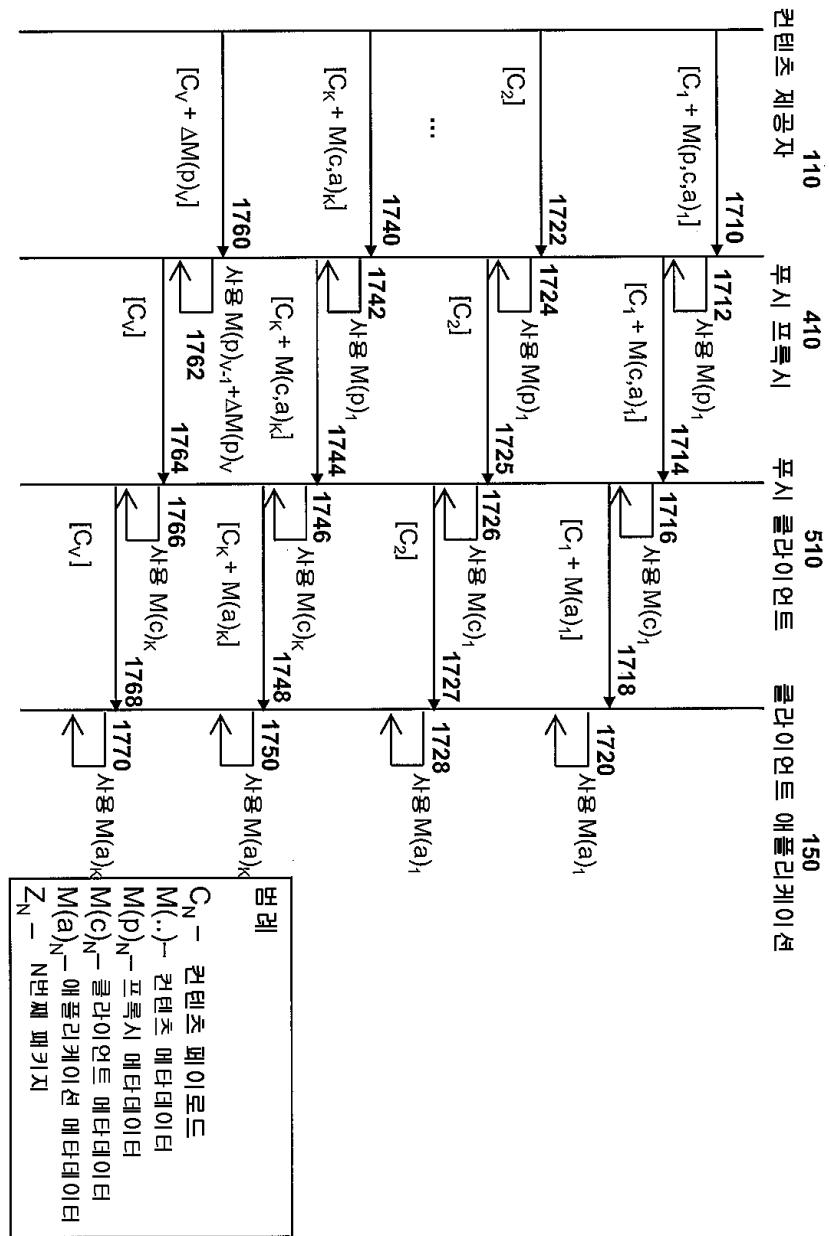
도면15



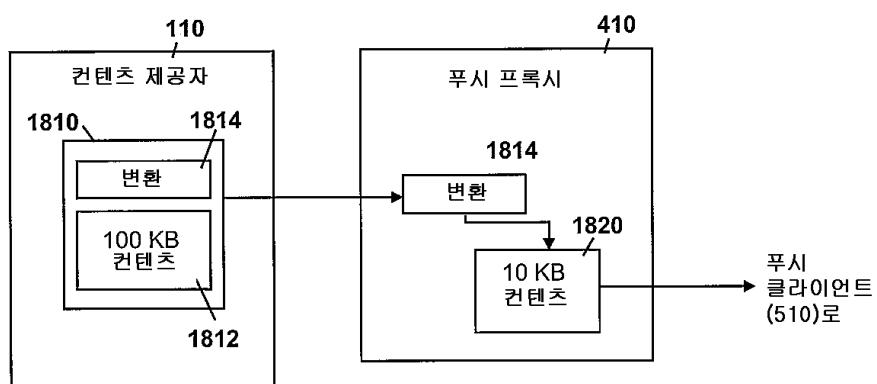
도면16



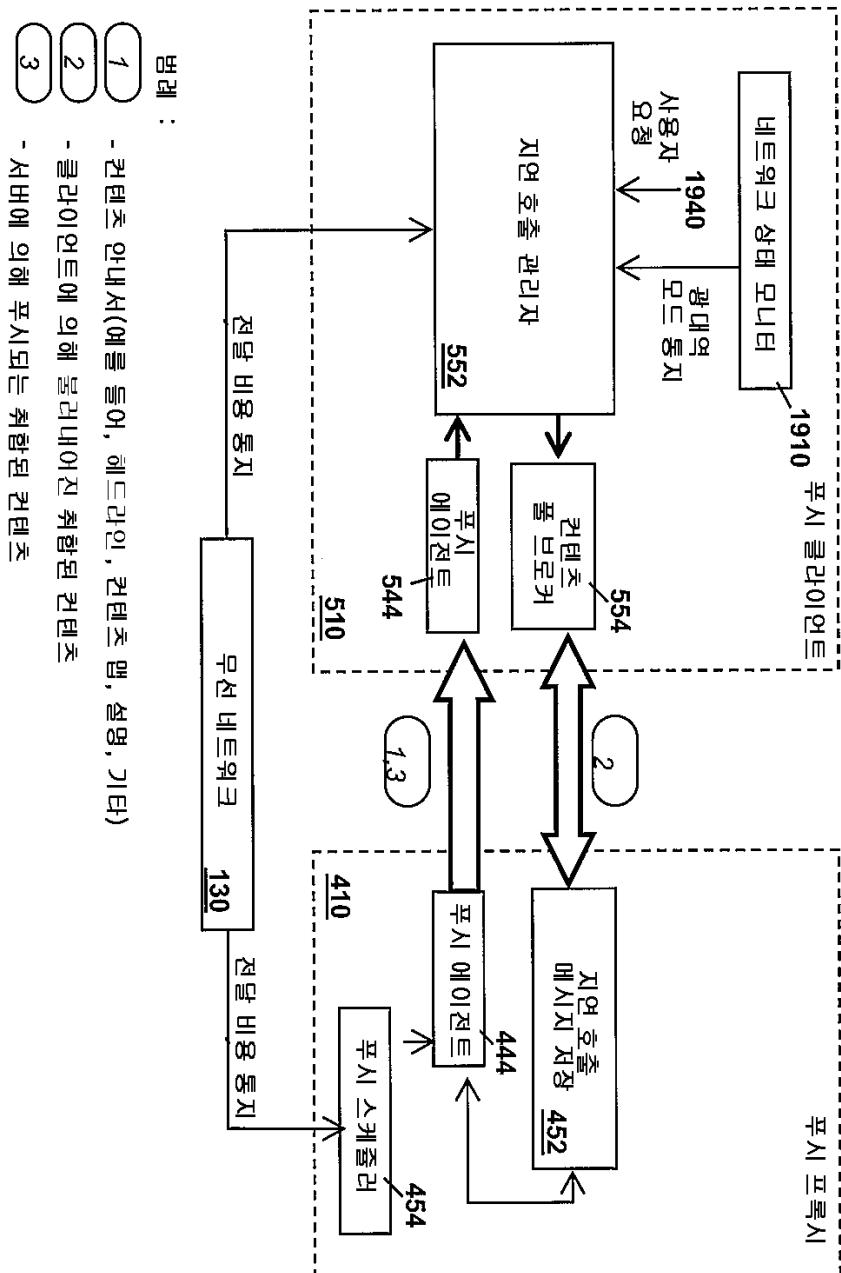
도면17



도면18



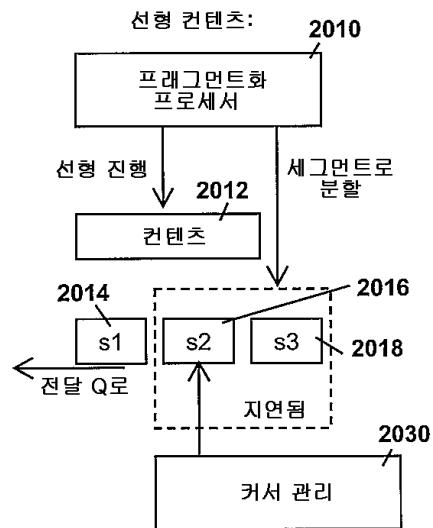
도면19



四

- 컨텐츠 안내서(예를 들어, 해드라인, 컨텐츠 맵, 설명, 기타)
  - 클라이언트에 의해 물려내어진 취합된 컨텐츠
  - 서버에 의해 푸시되는 취합된 컨텐츠

## 도면20

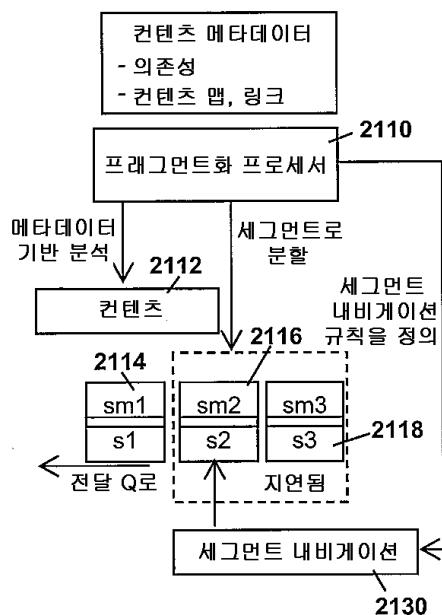


범례 :

**sX** - 세그먼트 X**smX** - 세그먼트 X에 대한 메타데이터(의존성, 맵, 내비게이션, 규칙)

## 도면21

비선형 컨텐츠 :



도면22

