



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2010년06월01일
(11) 등록번호 10-0960502
(24) 등록일자 2010년05월20일

- (51) Int. Cl.
G06F 21/24 (2006.01) G06F 17/00 (2006.01)
- (21) 출원번호 10-2004-7014430
- (22) 출원일자(국제출원일자) 2003년03월14일
심사청구일자 2008년02월21일
- (85) 번역문제출일자 2004년09월14일
- (65) 공개번호 10-2004-0098663
- (43) 공개일자 2004년11월20일
- (86) 국제출원번호 PCT/US2003/007847
- (87) 국제공개번호 WO 2003/079269
국제공개일자 2003년09월25일
- (30) 우선권주장
60/363,932 2002년03월14일 미국(US)
- (56) 선행기술조사문헌
US06236971 B1*
*는 심사관에 의하여 인용된 문헌

- (73) 특허권자
콘텐츠가드 홀딩즈 인코포레이티드
미국 델라웨어 19803 윌밍턴 스위트 200-엠 포울 크 로드 103
- (72) 발명자
칼리마이클, 씨.
미국 90242 캘리포니아 다운니 버두라 애비뉴 12834
데마티니토마스
미국 90230 캘리포니아 켈버 씨티 #130 그린 벨리 씨클 6410
(뒷면에 계속)
- (74) 대리인
박희진

전체 청구항 수 : 총 12 항

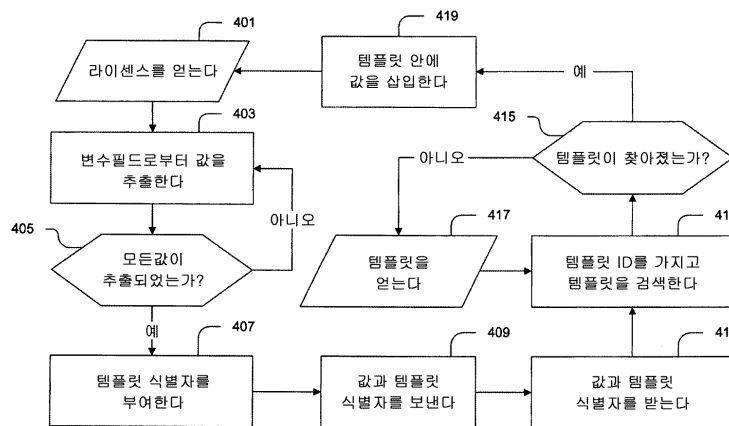
심사관 : 신창우

(54) 템플릿과 프로파일을 사용하는 권리 표현 프로파일시스템과 방법

(57) 요약

그 권리 표현에 따라 아이템의 사용을 제어하기 위한 시스템에서 사용하기 위해 아이템과의 연관성에 관한 권리 표현을 생성하기 위한 시스템과 방법으로서, 아이템의 사용 방법을 지시하는 권리표현정보를 특정하는 것과, 여기서 권리표현정보는 적어도 하나의 구성요소를 포함하며, 구성요소는 변수와 변수에 관한 대응 값을 가지며, 인코딩 프로세스를 수행하는 것을 구비하며, 인코딩 프로세스는, 권리표현정보에 대응하는 템플릿과 연관되는 식별자(identifier)를 결정하는 것과, 권리표현정보로부터 구성요소에 대응하는 변수에 관한 값을 추출하는 것과, 변수와 식별자에 의거하여 어떤 장치상에서 행사되도록 적응된(adapted) 라이선스를 인코딩하는 것을 포함하며, 라이선스는 템플릿의 식별정보(identification)와 변수에 관한 상기 값을 포함한다. 그 시스템과 방법에서, 권리표현정보의 프로파일이 생성될 수 있고, 변수에 관한 값을 구성요소로부터 제거하는 것과, 그리고 프로파일에 관한 식별정보를 생성하는 것을 포함하며, 그것에 의해 변수와 프로파일에 관한 식별정보에 의거하여 권리표현정보가 어떤 장치상에서 행사될 수 있다.

대표도



(72) 발명자

길리암찰스피.

미국 06820 코네티컷 다리엔 비치 드라이브 27

햄마누엘

미국 90241 캘리포니아 다운니 팡본 애비뉴 10259

라오겔러모

미국 90503 캘리포니아 토란스 로나 스트리트 5531

타다연비잔

미국 20876 메릴랜드 저먼타운 스코즈버리 드라이브 20920

특허청구의 범위

청구항 1

아이템의 사용을 권리표현에 근거하여 제어하는 시스템에서 사용되고, 그 아이템과 연관된 권리표현을 생성하기 위한 방법으로서,

상기 권리표현은 상기 아이템의 사용 방법을 지시한 권리표현정보를 포함하고,

상기 생성 방법은 엔코딩실행수단이 권리표현정보를 엔코딩하기 위한 엔코딩 프로세스를 실행하는 단계를 포함하고,

상기 엔코딩 프로세스는

획득수단이 상기 권리표현정보에 대응하면서 적어도 하나의 값이 삽입될 템플릿에 연관된 식별자를 획득하는 단계와,

추출수단이 상기 권리표현정보로부터 상기 값을 추출하는 단계를 포함하고,

상기 생성 방법은 또한 송신수단이 그 획득된 템플릿 식별자 및 그 추출된 값을 송신하는 단계를 포함하는 것을 특징으로 권리표현 생성 방법.

청구항 2

제1항에 있어서, 상기 권리표현정보는 상기 아이템의 사용 방법을 장치 상에서 수행하기 위해 충족시켜야 하는 조건을 포함하는 것을 특징으로 하는 권리표현 생성 방법.

청구항 3

제1항에 있어서, 상기 적어도 하나의 값은 상기 아이템을 사용할 수 있는 장치를 나타내는 값인 것을 특징으로 하는 권리표현 생성 방법.

청구항 4

제1항에 있어서, 상기 생성 방법은,

수신수단이 그 송신된 식별자 및 추출된 값을 수신하는 단계; 및

디코딩실행수단이 그 엔코딩된 권리표현정보를 디코딩하기 위한 디코딩 프로세스를 실행하는 단계를 더 포함하며,

상기 디코딩 프로세스는,

검색수단이 그 수신된 식별자에 대응하는 템플릿을 검색하는 단계; 및

삽입수단이 그 수신된 값을 그 검색된 템플릿에 삽입하는 단계를 포함하는 것을 특징으로 하는 권리표현 생성 방법.

청구항 5

제4항에 있어서, 상기 검색수단은 상기 식별자에 근거하여 상기 템플릿을 템플릿 서버에서 검색하는 것을 특징으로 하는 권리표현 생성 방법.

청구항 6

제4항에 있어서,

상기 권리표현정보는 디지털 서명을 포함하고,

상기 획득수단은 상기 권리표현정보로부터 상기 디지털 서명을 획득하고,

상기 송신수단은 그 획득된 디지털서명을 송신하고,

상기 수신수단은 그 송신된 디지털 서명을 수신하고,

상기 생성 방법은 확인수단(validating means) 이 그 수신된 디지털 서명을 이용하여 그 디코딩된 권리표현정보를 확인하는 단계를 더 구비하는 것을 특징으로 하는 권리표현 생성 방법.

청구항 7

아이템의 사용을 권리표현에 근거하여 제어하고, 그 아이템과 연관된 권리표현을 생성하기 위한 시스템으로서

상기 권리표현은 상기 아이템의 사용 방법을 지시한 권리표현정보를 포함하고,

상기 시스템은 권리표현정보를 엔코딩하기 위한 엔코딩 프로세스를 실행하는 엔코딩 프로세스 실행 수단을 포함하고,

상기 엔코딩 프로세스는,

획득 수단이 상기 권리표현정보에 대응하면서 적어도 하나의 값이 삽입될 템플릿에 연관된 식별자를 획득하는 기능과,

추출 수단이 상기 권리표현정보로부터 상기 값을 추출하는 기능을 포함하고,

상기 시스템은 또한 그 획득된 템플릿의 식별자 및 그 추출된 값을 송신하는 송신수단을 포함하는 것을 특징으로 하는 권리표현 생성 시스템.

청구항 8

제7항에 있어서, 상기 권리표현정보가 상기 아이템의 사용 방법을 장치 상에서 수행하기 위해 충족시켜야 하는 조건을 포함하는 것을 특징으로 하는 권리표현 생성 시스템.

청구항 9

제7항에 있어서, 상기 적어도 하나의 값은 상기 아이템을 사용할 수 있는 장치를 나타내는 값인 것을 특징으로 하는 권리표현 생성 시스템

청구항 10

제7항에 있어서, 상기 시스템은

그 송신된 식별자 및 추출된 값을 수신하는 수신수단과,

그 엔코딩된 권리표현정보를 디코딩 하기 위한 디코딩 프로세스를 실행하는 디코딩 프로세스 실행 수단을 더 포함하며,

상기 디코딩 프로세스는,

검색수단이 그 수신된 식별자에 대응하는 템플릿을 검색하는 기능과,

삽입수단이 그 수신된 값을 그 검색된 템플릿에 삽입하는 기능을 포함하는 것을 특징으로 하는 권리표현 생성 시스템.

청구항 11

제10항에 있어서, 상기 검색 수단은 상기 식별자에 근거하여 상기 템플릿을 템플릿 서버에서 검색하는 것을 특징으로 하는 권리표현 생성 시스템.

청구항 12

제10항에 있어서,

상기 권리표현정보는 디지털 서명을 포함하고,

상기 획득수단은 상기 권리표현정보로부터 상기 디지털 서명을 획득하고,

상기 송신수단은 그 획득된 디지털 서명을 송신하고,

상기 수신수단은 그 송신된 디지털 서명을 수신하고,

상기 시스템은 그 수신된 디지털 서명을 이용하여 그 디코딩된 권리표현정보를 확인하는 확인수단을 더 구비하는 것을 특징으로 하는 권리표현 생성 시스템.

청구항 13

삭제

청구항 14

삭제

청구항 15

삭제

청구항 16

삭제

청구항 17

삭제

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

청구항 21

삭제

청구항 22

삭제

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

청구항 27

삭제

청구항 28

삭제

청구항 29

삭제

청구항 30

삭제

청구항 31

삭제

청구항 32

삭제

청구항 33

삭제

청구항 34

삭제

청구항 35

삭제

청구항 36

삭제

청구항 37

삭제

청구항 38

삭제

청구항 39

삭제

청구항 40

삭제

청구항 41

삭제

청구항 42

삭제

청구항 43

삭제

청구항 44

삭제

청구항 45

삭제

청구항 46

삭제

청구항 47

삭제

청구항 48

삭제

청구항 49

삭제

청구항 50

삭제

청구항 51

삭제

청구항 52

삭제

청구항 53

삭제

청구항 54

삭제

청구항 55

삭제

청구항 56

삭제

청구항 57

삭제

청구항 58

삭제

청구항 59

삭제

청구항 60

삭제

청구항 61

삭제

청구항 62

삭제

청구항 63

삭제

청구항 64

삭제

명세서

기술분야

[0001] 본 발명은 일반적으로, 콘텐츠 또는 다른 아이템들과 관련되는 권리 표현들을 통하여, 콘텐츠 또는 다른 아이템들의 사용을 제어하는 것에 관한 것으로서, 특히 템플릿과 권리 표현들의 프로파일들을 생성하는 시스템과 방법에 관한 것이다.

배경기술

[0002] 문서들, 음악, 영화들, 소프트웨어, 정보 등과 같은 디지털 콘텐츠를, 전자적 수단과 특히 인터넷을 통하여, 컴퓨팅 장치에 의해 사용할 수 있는 형태로, 광범위하게 배포하는 것에 관한 가장 중요한 문제들 중의 하나는, 그러한 배포동안에 지적 소유권들을 행사하고 그 디지털 콘텐츠를 사용하는 능력의 제공이다. 이 문제를 해결하기 위한 기술들은 이 글 속에 디지털 권리 관리(Digital Rights Management: DRM) 로 부르게 된다. 그러나 DRM 시스템을 effecting 할 때 고려해야 할 문제들이 많이 있는데, 몇 가지 예만 들면, 인증(authentication), 인가(authorization), 회계(accounting), 지불과 금전적 청산(payment and financial clearing), 권리 명세(rights specification), 권리 확인(rights verification), 권리 행사(rights enforcement), 그리고 문서보호문제 등이 있다.

[0003] 예를 들면, 인쇄되는 문서들과 다른 물리적인 콘텐츠의 세계에서는 저자에 의해 만들어지는 작업물(work)은 출판사에 보통 제공되어, 그 작업물의 다수의 복사본을 형식화(format)하여 인쇄한다. 그리고나서 그 복사본들은 배급업자에 의해 서점들 또는 다른 소매점들로 보내어지고, 최종 소비자들은 거기에서 그 복사본들을 구입하게 된다. 저품질의 복사와 고비용의 인쇄물 배포는 대부분의 인쇄 문서들의 비인가된 복사에 대하여 방해물로서 작용해온 반면, 무방비의 디지털 콘텐츠를 복사하고, 수정하고 재배포하는 것은 매우 더 쉽다. 그러므로 디지털 콘텐츠를 보호하는 메커니즘의 필요성이 있다.

[0004] 오늘날의 범용 컴퓨터 및 통신 시스템들 예컨대 개인용 컴퓨터, 워크스테이션 그리고 통신망-예컨대 랜(LANs), 인터넷, 그리고 인터넷-과 연결된 다른 장치들 안에서, 사람들이 인가받지 않고 전자 콘텐츠의 복사본을 만드는 것을 막는 것 혹은 심지어 그만두게 하는 것과 관련한 어려움은 널리 인식되어 있다. 비인가된 복사를 방지하기 위한 하드웨어 기반의 솔루션들을 제공하기 위한 많은 시도들이 성공적이지 못하여왔다. 더욱이

높은 대역폭 또는 광대역 통신 기술들의 전개, 그리고 현재 국가 정보 기반구조(National Information Infrastructure: NII) 로 알려진 것의 전개는, 예컨대 무삭제(full length) 동영상과 같은 비디오 파일을 포함하여, 많은 문서를 전자적으로 배포하는 것을 더욱 편리하게 만들어주고 있다. 그리고 이것은 디지털 콘텐츠의 비인가 복사와 배포를 증식시키는 것을 더욱 쉽게 만든다. 그러므로 DRM 기술들의 더 많은 발전의 필요성은 높은 우선순위가 되고 있다.

[0005] 따라서, 공통적으로 부여된 미국특허번호 제 5,634,012호는 디지털 콘텐츠의 배포를 제어하는 DRM 시스템을 개시한다. 그 DRM 시스템의 장치들은 그와 연관되어 있는 저장소(repository)를 포함한다. 소정 세트의 사용 처리 단계들은 그 콘텐츠와 연관된 사용권을 행사하기 위한 저장소들에 의해 사용되는 프로토콜을 정의한다. 사용 권리들은 콘텐츠와 함께 존속하고(persist), 그 콘텐츠와 관련되는 사용 권리들은 디지털 작업물을 구비한다. 그 사용권들은 그 콘텐츠의 사용에 관한 다양한 방법들, 예컨대 그 콘텐츠를 보거나 인쇄하거나 디스플레이 하는 권리, 그 콘텐츠를 배포 또는 재배포하는 권리 등을 허락한다. 그러한 사용 권리들은 지불 또는 다른 조건들에 관해 부수적으로 되게 할 수 있다. 그러나 사용권들 및/또는 콘텐츠와 관련된 조건들을 유연하고 강한 방법으로 표현하고 행사하기 위한 시스템들과 방법들에 관한 더 많은 필요성이 여전히 존재한다.

발명의 상세한 설명

[0006] 위에서 언급한 것과 다른 필요성들은, 콘텐츠와 다른 아이템들에 관한 사용권들을 변조되거나 변경된 신호들 혹은 사용권의 그래픽 표현들에 의거하여 나타내는 개선된 시스템과 방법을 제공하는 본 발명의 실시예들에 의해 언급된다.

[0007] 따라서, 본 발명의 일 실시예의 일 측면에 있어서, 그 권리 표현에 따라서, 아이템의 사용을 제어하기 위한 시스템에서 사용하기 위한 아이템과의 관련성에 관한 권리표현을 생성하기 위한 시스템과 방법이 제공된다. 이 시스템과 방법은 아이템의 사용 방법을 나타내는 권리표현정보를 특정하는 것을 포함하며, 상기 권리표현정보는 적어도 하나의 구성요소를 포함하며, 상기 구성요소는 변수와 상기 변수에 관한 대응값을 가지며; 상기 시스템과 방법은 상기 권리표현정보에 대응하는 템플릿과 관련 있는 식별정보(identifier)를 결정하는 것과, 상기 권리표현정보로부터 상기 구성요소에 대응하는 변수에 관한 값을 추출하는 것과, 어떤 장치에 관해 행사되도록 변경된(adapted), 상기 템플릿의 아이디(identification)와 상기 변수에 관한 상기 값을 포함하는, 라이선스를 상기 변수와 상기 식별정보에 의거하여 엔코딩하는 것을 포함하여, 엔코딩 프로세스를 수행하는 것을 포함한다.

[0008] 권리 표현에 따라 아이템의 사용을 제어하기 위한 시스템에서 사용하기 위해 아이템과의 관련성에 관한 권리표현을 생성하기 위한 시스템과 방법으로서, 아이템의 사용 방법을 지시하는, 권리표현정보를 특정하는 것과, 여기서 상기 권리표현정보는 적어도 하나의 구성요소를 포함하며, 상기 구성요소는 변수와 상기 변수에 관한 대응 값을 가지며; 상기 변수에 관한 상기 값을 상기 구성요소로부터 제거하는 것을 포함하는, 상기 권리표현정보의 프로파일을 생성하는 것과; 그리고 상기 프로파일에 관한 식별정보(identification)를 생성하는 것을 구비하고, 그것에 의해 상기 변수와 상기 프로파일에 관한 상기 식별정보에 의거하여 상기 권리표현정보가 어떤 장치상에서 행사될 수 있는 것을 특징으로 하는 시스템과 방법이 제공된다.

[0009] 본 발명을 실시하는 데 최상의 실시예를 비롯하여, 많은 전형적인 실시예들(exemplary embodiments)과 구현들을 단순히 도시하는 것에 의해, 본 발명의 다른 측면들, 특징들 그리고 장점들이 이하의 상세한 기술로부터 명백하다. 본 발명은 또한 기타 다른 실시예가 가능하며 그것의 몇 가지 상세한 것들은 본 발명의 본질과 범위를 벗어남이 없이 다양한 관점에서 변경될 수 있다. 따라서 도면들과 설명들은 본질적으로는 예시적인 것이고 한정적인 것이 아니다.

실시예

- [0021] [0021] 권리표현들의 개요를 작성하는(profiling) 시스템과 방법이 기술된다. 이하의 기술 안에서, 설명의 목적들을 위해, 다수의 특성의 세부사항들은 본 발명의 완전한 이해를 제공하기 위해 기재하게 된다. 그러나 본 발명은 이 특성의 세부사항들 없이 또는 등가적인 장치들을 가지고서 실시될 수 있다는 것은 당업자에게 명백하다. 몇 가지 예들에 있어서, 잘 알려진 구조들과 장치들은 블록도 형태로 도시되는데, 이는 본 발명을 불필요하게 가리는 것을 피하기 위함이다.

- [0022] [0022] 위에서 언급한 바와 같이, 인증, 인가, 회계, 지불과 금전적 청산, 권리 명세, 권리 행사 및 문서 보호 문제들은 DRM 시스템에 의해 대처되어야 한다. 공통적으로 부여된 미국 특허번호 제5,530,235, 제 5,629,980, 제5,634,012호, 제5,638,443호, 제5,715,403호, 제 6,233,684호와 제6,236,971호는, 이들 전부의 모든 개시 내용이 여기에서 참조내용으로 통합되는데, 이것들과 다른 문제들에 대해 대처하고 있는 DRM 시스템들을 개시한다.

- [0023] [0023] 확장성 권리 생성 언어(eXtensible Right Markup Language: XrML) 또는 확장성 생성 언어(eXtensible Markup Language: XML) 기반 언어같이 사람이 읽을 수 있고, 기계가 읽을 수 있는 권리표현언어(rights expression language)는 권리표현을 나타내기 위해 사용될 수 있다. 유리하게(advantageously), 예컨대 라이선스와 같은 권리표현은 예컨대 디지털 콘텐츠, 서비스와 같은 어떤 아이템의 사용에 관한 허락된 방법을 특정하는 사용권을 나타낼 수 있다. 그러나 많은 디지털 콘텐츠 소비 장치들, 예컨대 MP3 플레이어, 이동전화들, 핸드헬드(handheld) 장치, 피디에이(PDAs), 셋톱박스들과 인터넷 응용품들과 같은 자원-계약적 장치들은 저장 능력과 처리 능력을 제한할 수 있었다. 따라서 그런 장치들이 XrML 해석기들 혹은 XML 해석기와 같은, 풍부한 기능을 갖춘(full-featured) 권리표현 해석기들을 권리표현들을 처리하기 위해 포함하는 것이 어려울 수 있다.

- [0024] [0024] 본 발명은 자원-계약적인 장치들이, 예를 들면, 디지털 콘텐츠를 소비하기 위해, 권리 표현을 사용하는 것에서 도움이 될 수 있다는 인식을 포함한다. 그러나 관련된 권리표현 파싱 도구들, 전달 메커니즘들, 그리고 인코딩 메커니즘들은 그러한 장치들을 혹사하는(taxing) 것을 증명할 수 있고, 이는 결과적으로 디지털 콘텐츠의 최적의 소비보다 낮은 소비를 가져오게 된다.

- [0025] [0025] 그러므로 권리표현들, 권리표현들을 해석하기 위해 채용된 자원들을 최적화하는 것의 필요성이 존재하며, 그래서 자원-계약적인 장치들이 권리표현들을 채용하는 것의 완전한 혜택을 효율적이고 강한 방식으로 누릴 수 있다. 여기에 기술된 것과 같이, 전형적인 실시예들은 전형적인 프로파일들과 템플릿들에 의거하는 전형적인 기술들을 제공하는 것에 의해 상기 문제들과 기타 다른 문제들에 대처한다.

- [0026] [0026] 도면들을 참조하면, 여러 도면에 걸쳐서 같은 참조번호는 동일하거나 대응 부분들을 지정한다. 그리고 도 1에 대하여 보다 상세하게는, DRM 시스템(100)이 예시되어 있는데, 그것은 언급된 전형적인 실시예들과 관련하여 사용되어 콘텐츠, 서비스들 또는 다른 자산에 관한 사용권들을 특정하고 행사할 수 있다. 도 1에서, DRM 시스템(100)은 사용자 활성화 요소를 활성화 서버(110)의 형태로 포함하며, 활성화 서버(110)는 공개 키(public key)와 비밀 키(private key) 쌍 세트(112)를 발행한다. 일반적으로, 사용자가 처음으로 DRM 시스템(100)을 사용할 때, 사용자는 예컨대 특별한 콘텐츠 포맷을 위한 렌더링 애플리케이션(rendering application)과 함께 작용 또는 포함하는 소프트웨어를 설치한다.

- [0027] [0027] 그 소프트웨어는 예컨대, 사용자와 관련되는 컴퓨팅 장치와 같은, 클라이언트 환경(120)에 설치된다. 그 소프트웨어는 DRM 시스템(100)의 일부일 수 있고, 보호되고 있는 콘텐츠(134)에 액세스하는 데 사용될 수 있다. 설치 후, 그 소프트웨어는 활성화 시킬 수 있다. 활성화 처리동안, 몇몇 정보가 활성화 서버(110)와 클라이언트 환경(120)의 사이에서 교환된다. 클라이언트 소프트웨어 구성 요소(122)는 클라이언트 환경(120)에서 다운로드되어 설치될 수 있다. 클라이언트 소프트웨어 구성 요소들(122)은 부정조작 불가성(tamper resistant)

을 가지며, 다른 구성요소들 뿐만 아니라 활성화 서버(110)가 발행한 그 공개 및 비밀 키 세트를 포함할 수 있다.

[0028] 권리제공(right offer)(132)은 보호되는 콘텐츠(134)와 연관될 수 있으며, 예컨대 하나 이상의 대응 조건들(132B)이 만족될 때, 최종 사용자에게 이용될 수 있는 사용권(132A)을 특정할 수 있다. 라이선스 서버(140)는 암호화키들(112)을 관리하고, 보호되는 콘텐츠(134)용 라이선스(142)를 발생한다. 그 라이선스(142)는 사용권들(132A)의 실제적인 허락을 최종 사용자에게 구체화할 수 있다. 예를 들면, 권리제공(132)은 보호되는 콘텐츠(134)를 5달러의 요금을 지불하는 조건으로 보게 하는 권리, 그리고 그 보호되는 콘텐츠를 10달러의 요금을 지불하는 조건으로 프린트할 권리를 최종 사용자에게 허락할지도 모른다. 라이선스(142)는, 5 달러의 요금이 지불되었을 때는 보는 권리에 관해 그리고/또는 10 달러의 요금이 지불되었을 때는 인쇄할 권리에 관해, 발행될 수 있다. 클라이언트 구성 요소(122)는 라이선스(142)에서 특정되어왔던 권리들을 해석하고 행사한다.

[0029] 예컨대 비보호 콘텐츠 등과 같은 순수한 콘텐츠(136)는 콘텐츠 출판사, 콘텐츠 배급업자, 콘텐츠 서비스 제공자 혹은 어떤 다른 적절한 당사자와 연관있는 컴퓨터(130)에 설치된, 예를 들면 문서 준비 애플리케이션과 같은, 어떤 애플리케이션(138)을 가지고서 준비될 수 있다. 순수한 콘텐츠(136)의 준비는 사용권(132A)과 그 순수한 콘텐츠(136)가 사용될 수 있는 조건들(132B)을 특정하는 것, 권리제공(132)을 순수한 콘텐츠(136)와 연관 짓는 것, 그리고 순수한 콘텐츠(136)를 예컨대 암호작성 알고리즘을 가지고서 보호받는 콘텐츠(134)를 생성하여 보호하는 것을 포함할 수 있다. XrML, 확장성 액세스 제어 생성 언어(eXtensible Access Control Markup Language: XACML), 개방형 디지털 권리 언어(Open Digital Rights Language: ODRL) 등과 같은 권리언어는 권리제공(132)을 특정하기 위해 사용될 수 있다. 그러나 권리제공(132)은 어떤 적절한 방법으로도 특정될 수 있다. 여기서 사용되는 것처럼, "권리 정보를 특정하는 것"이라는 어구는 권리표현들과 관련 있는 정보를 생성하는 것, 유추해내는 것, 또는, 그렇지 않다면, 활용하는 것을 언급한다. 또한, 권리제공(132)은 보호받는 콘텐츠(134)와 연관될 수 있는 미리 정의된 명세, 프로파일, 템플릿 등의 형태로 있을 수 있다. 따라서 권리제공(132)을 특정하는 과정은 권리들, 조건들 등을 콘텐츠와 연관짓는 어떤 적절한 프로세스를 포함할 수 있다. 보호받는 콘텐츠(134)와 순수한 콘텐츠(136)를 암호화하는 데 사용되는 암호화 키(112)와 연관된 권리제공(132)은 라이선스 서버(140)에 전달될 수 있다.

[0030] DRM 시스템(100)을 위한 전형적인 일의 흐름은 보호받는 콘텐츠(134)를 활성화 서버(110)에 의해 수신하기 위해 활성화되어 있는 클라이언트 환경(120) 안에서 작용하는 사용자를 포함할 수 있다. 활성화 프로세스는 결과적으로 공개 및 비밀 키 쌍(112), 그리고 예를 들면, 몇몇 사용자 그리고/또는 클라이언트 환경(120)에 클라이언트 소프트웨어 구성요소(122)의 형태로 다운로드 되고 있는 기계-특정(machine specific) 정보 등을 낳는다. 활성화 프로세스는, 예를 들면, 라이선스(142)의 발행에 앞서서 어떤 적절한 시점에서 완성될 수 있다.

[0031] 사용자가 보호되고 있는 콘텐츠(134)를 사용하고 싶을 때, 그 사용자는 보호되고 있는 콘텐츠(134)에 관한 요청을 만든다. 예를 들면, 사용자는 클라이언트 환경(120)에 설치되어 있는 브라우저를 사용하여, 웹 서버(150) 상에서 운영되고 있는 웹 사이트를 열람하고, 그 보호되는 콘텐츠(134)를 다운로드하려 할지도 모른다. 이러한 프로세스 동안에, 사용자는 콘텐츠의 판매에 있어서 요금 처리, 정보의 수집 등과 같은 다른 처리들을 포함하는 일련의 단계들을 거칠 지도 모른다. 요금의 징수와 그 사용자가 활성화되어 있다는 확인처럼 적절한 조건들과 다른 전제 조건들이 만족될 때, 웹 서버(150)는, 예를 들면, 안전한 통신 채널, 예컨대 안전 소켓층(Secure Sockets Layer: SSL)과 같은 것을 통해 라이선스 서버(140)와 접촉할 수 있다. 라이선스 서버(140)는 그런 다음, 보호받는 콘텐츠(134)를 위한 라이선스(142)를 생성하고, 웹서버(150)는 그 보호받는 콘텐츠(134)와 그 라이선스(142) 양자를 다운로드한다. 라이선스(142)는 사용권들(132A)의 적절한 사용권들을 포함할 수 있고 라이선스 서버(140) 혹은 연관된 장치로부터 다운로드 될 수 있다. 보호받는 콘텐츠(134)는 발행자, 배포자, 혹은 다른 당사자들과 연관되는 컴퓨터(130)로부터 다운로드 될 수 있다. 권리제공(132)은 지속적인 수 있고 그리고 보호되는 콘텐츠(134)와 연관되어 계속 유지될 수 있다.

[0032] 클라이언트 환경(120)의 클라이언트 소프트웨어 구성 요소(122)는 그 때 라이선스(142)를 해석하는 것을 계속할 수 있고, 라이선스(142)에 특정되어 있는 권리(132A)와 조건들(132B)에 의거하여 보호되는 콘텐츠(13

4)의 사용을 허락할 수 있다. 예를 들면, 사용 권리들의 해석과 시행은 공통적으로 부여되어 있는 미국 특허번호 제5,530,235호, 제5,629,980호, 제5,634,012호, 제5,638,443호, 제5,715,403호, 제6,233,684호와 제6,236,971호에 더 기술되어 있다. 상기 단계들은 순차적으로, 대략 동시적으로, 다양한 순서로 일어날 수 있다.

[0033] DRM 시스템(100)은 보호되는 콘텐츠(134)를 보호하는 것의 안전에 관한 측면에도 대처한다. 특히, DRM 시스템(100)은 라이선스 서버(140)에 의해 발생되어 온 라이선스(142)를 인증할 수 있다. 그러한 인증을 달성하는 한 가지 방식은 클라이언트 소프트웨어 구성요소(122)가 그 라이선스(142)가 신뢰할 수 있는 것인지를 결정하는 것이다. 달리 말하면, 클라이언트 소프트웨어 (122)는 암호의 서명을 검증 및/또는 확인할 능력 혹은 라이선스(142)의 특징을 식별하는 다른 능력을 포함할 수 있다. 앞서 기술된 활성화 단계 동안에, 클라이언트 환경(120)과 라이선스 서버(140)는 부정조작 불가성 소프트웨어 패키지 내의 키들의 세트(112)를 수신할 수 있으며, 그 소프트웨어 패키지는, 예를 들면, 활성화된 클라이언트 환경(120)을 위한 클라이언트 소프트웨어 구성요소(122)같은 다른 구성요소들을 라이선스(142)의 서명을 검증하기 위하여 포함할 수 있다.

[0034] DRM 시스템(100)은 전형적인 특징이고, 다수의 다른 등가적인 구성으로 구현될 수 있다. 예컨대, 라이선스(142)와 보호되는 콘텐츠(134)는 다른 실체들로부터 배포될 수 있다. 다른 하나의 예로서, 권리제공(132)은 보호되는 콘텐츠(134)를 준비하고 있는 당사자 이외의 당사자에 의해 그 보호되는 콘텐츠(134)와 관련될 수 있다. 또 다른 예로서, 정보 센터(160)는 결제 처리작업들을 처리하고 그리고 라이선스(142)를 발행하기 전에 결제를 검증하는 데 사용될 수 있다. 더욱이 여러 가지 프로세스들과 트랜잭션들은, 예를 들면 온라인 및/또는 오프라인 환경들 및/또는 그것들의 조합을 통하여 수행될 수 있다. 예를 들면, 최종 사용자는 콘텐츠를 컴퓨터로 다운로드 할 수 있고, 그런 다음 그 콘텐츠를 컴퓨터로부터 개인 휴대 단말기(PDA)로 옮길 수 있었다. 최종 사용자는 그런 다음, 슈퍼마켓 키오스크(kiosk), 금전 등록기, 선불 라이선스 카드 등을 통하여, 그 콘텐츠용으로 라이선스를 구입할 수 있다. 그 후, 최종 사용자는 PDA 및/또는 컴퓨터 상에서 그 콘텐츠를 사용하기 위해 활성화 할 수 있다. 이와 같은 오프라인 시나리오에 있어서, 다양한 장치들은, 할 필요는 없지만, 다른 장치들과 직접 통신할 수 있고, 정보는 장치들 간에 매체를 물리적으로 이동시키는 것과 같은 적절한 어떤 방법으로 교환될 수 있다.

[0035] [0001] 여기서 기술된 것처럼, 전형적인 프로파일과 템플릿은 XrML 권리 표현 등과 같은 권리표현의 사이즈를 압축하기 위해 채용될 수 있다. 유리하게, 전형적인 프로파일들과 템플릿들은 자원-제한적인 애플리케이션, 장치, 시스템, 플랫폼 등에 의해 권리 표현을 해석하기 위한 것으로 채용된 프로세싱 자원들의 양을 최소화하기 위한 것으로 채용될 수 있다.

[0036] [0002] 전형적인 실시예에 따르면, 만일 X가 권리 표현 언어의 모든 동형의(conformant) 표현들의 집합으로 간주될 수 있으면, 그 권리 표현 언어 프로파일은 X의 몇몇 비어있지 않은 부집합(non-empty subset) Y로서 정의될 수 있다. 만약 주어진 표현 E가 Y의 요소가 될 수 있다면, E는 그 프로파일 안에 있다고 할 수 있다. 그러나 만일 E가 Y의 요소가 아니라면, E는 그 프로파일에 대하여는 무효라고 말할 수 있다. 따라서 전형적인 실시예에 따르면, 권리표현 언어의 프로파일링(profiling)은 특수한 해석기가 지원하고 해석할 수 있는 권리표현 언어의 부집합들을 식별하는 것을 포함할 수 있다. 유리하게, 프로파일링은 자원-제한적인 장치가 권리-해석된 언어 시스템에 참여하도록 허용하기 위해 채용될 수 있다.

[0037] [0003] 전형적인 실시예들에 따르면, 예컨대 템플릿에 의거한 메커니즘을 채용하는 것과 같이, 권리표현 언어 프로파일을 생성하는 다수의 형태와 방식들이 있다. 템플릿들과 연관된 기술들, 예컨대 템플릿들의 예를 들면 바이너리 토큰과 같은 목적물에 대한 매핑은 그러한 기술들의 수집을 기술하기 위해 템플릿 매핑으로 불려질 수 있다.

[0038] [0004] 전형적인 템플릿 매핑과 프로파일링 기술들은 권리 표현의 크기를 압축한다. 예를 들면, 권리 표현들 언어를 위해 이뤄질 수 있는 최고의 압축은 주어진 영역 안에 존재할 수 있고, 각각의 그리고 모든 권리표현에 참조 값을 부여할 수 있는 모든 별개의 권리 표현들을 확인하는 것 일 수 있다. 이런 식으로, 권리표현 언어

해석기는 대응 권리표현을 해석하기 위해, 상기 참조 값으로 주어진 권리표현을 검색하는 것이 단지 필요할 뿐이다. 전형적인 템플릿 매핑 및 프로파일링 기술들은, 전형적인 템플릿, 그리고 프로파일 인코딩, 그리고 디코딩 프로세스들을 통하여, 그런 수준의 압축에 이르려고 시도한다.

[0039] [0005] 하나의 전형적인 인코딩 프로세스는 권리 표현용 템플릿 또는 프로파일을 생성하기 위해 권리표현으로부터 변수 필드들에 관한 하나 이상의 값들을 제거하는 것과, 그 템플릿 혹은 프로파일에 식별자(ID)를 부여하는 것을 포함할 수 있다. 유리하게도, 식별자는, 전체 혹은 가공되지 않은 권리표현을 보내는 대신에, 상기 권리표현의 변수 필드들에 관한 하나 이상의 값들과 함께, 예를 들면, 자원-제한적인 장치로 보내질 수 있다.

[0040] [0006] 전형적인 디코딩 프로세스는 상기 식별자 그리고 상기 변수 필드들에 관한 하나 이상의 값들을 수신하는 것과, 보내진 값들에 의거하여, 예컨대 자원-제한적인 장치에 관한 권리표현을 해석하는 것을 포함할 수 있다. 유리하게도, 그런 해석자는, 전체 혹은 가공되지 않은 권리표현을 해석하기 위해 사용되곤 했던 완전히 갖춘 권리표현 언어 파서(parser) 대신에, 최소한의 컴퓨팅 자원들을 포함할 수 있다.

[0041] [0007] 나아가, 전형적인 인코딩, 그리고 디코딩 기술들은, 유리하게도, 손실 없는 압축과 압축해제를 제공할 수 있다. 예를 들면, 변수 필드들에 관한 하나 이상의 값들을 그 원래의 권리표현으로부터, 대응 식별자에 의해 확인된 바대로, 상기 원래의 권리표현에 대응되는 템플릿 혹은 프로파일 안으로 자리를 차지하도록 하는 것에 의해, 원래의 권리표현이 생성되고, 소생시키고(recreated), 개조되거나(reconstructed) 복사될 수 있다. 최고급(high-end) 권리 관리 애플리케이션들, 장치들, 그리고 시스템들은 자원-제한적인 장치들과 정보처리 상호운용가능성(interoperability)을 제공하는 개조된 권리표현을, 유리하게도, 채용할 수 있다.

[0042] [0008] 표 1에서 나타낸 것처럼, 다음의 전형적인 명칭공간(namespace) 테이블은, 음악과 오디오에 관련된 전형적인 XrML 라이선스의 형태로 전형적인 권리 표현들에 적용된 것처럼, 전형적인 템플릿 매핑 및 프로파일링 기술들을 도시한다. XrML 명세(specification)는 월드와이드 웹(www)의 xrml.org에서 얻을 수 있다. 확장 스키마(schema/2001/11/compactMusic)는 첨부물로서 제공된다.

[0043] 표 1: 명칭공간 테이블

[0044]

접두어	명칭 공간	스키마 설명
(r)	http://www.xrml.org/schema/2001/11/xrml2core	XrML 2. 0 Core
Sx	http://www.xrml.org/schema/2001/11/xrml2sx	Standard Extension
Cx	http://www.xrml.org/schema/2001/11/xrml2cx	Content Extension
Cm	http://www.xrml.org/schema/2001/11/xrml2compactMusic	Sample Extension
Dsig	http://www.w3.org/2000/09/xmlsig#	Digital Signature
Xsi	http://www.w3.org/2001/XMLSchema-instance	Schema Instance

[0045] [0009] 정체가 확인된 장치에 의해 발행된 XrML 라이선스의 형태인 권리표현은, 그 라이선스에서 확인된 장치는 확인된 노래 파일을, 다음의 제1 전형적 라이선스(L1)에 표현된 것처럼, 재생할 수 있다는 것을 특정하는 데 사용될 수 있다.

[0046] [0010] 제1의 전형적인 라이선스 (L1) :

```
<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>
```

[0047]

[0048] [0011] 전형적인 라이선스 L1에 관하여, 다음을 가정할 수 있다.

[0049] (i)라이선스 재분배는 채용된 비즈니스 모델이 아니며, (ii) 채용된 장치는 어떤 적당한 노래 파일 입력 매커니즘을 사용할 수 있고, (iii) 채용된 장치는 노래 파일을 재생할 수 있으며, (iv) 보안은, 서명을 채용하지 않고서도, 채용된 장치가 ID 번호에 의해 식별할 수 있고, 전형적인 라이선스 L1는 발행 장치를 식별할 수 있으므로, 관심사항이 아니고, (v) 전형적인 라이선스 L1는 노래 파일과 함께 진행할 수 있어 그 전형적인 라이선스 L1에 관한 검색은 전혀 필요하지 않으며, 그리고 (vi) 채용된 장치에 대해서는 인터넷 액세스가 가능하지 않다. 전형적인 라이선스 L1에 따르면, ID 번호 123을 가진 장치는, ID 번호 024를 가진 발행장치에 따르면, ID 번호 456을 가진 노래 파일을 무제한의 횟수만큼 재생할 사용권이 허여될 수 있다. 장치 123이, 다음의 제2 전형적인 라이선스(L2)에 표현된 것처럼, 노래파일 456을 재생할 수 있는 횟수에 관한 한계를 라이선스에 특정하는 것에 의해 부가적인 유연성이 채용될 수 있다.

[0050] [0012] 제2의 전형적인 라이선스 (L2) :

```
<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
    <sx:exerciseLimit>
      <sx:stateReference>
        <cm:countKeeper key="789">
          <cm:device cmid="123"/>
        </cm:countKeeper>
      </sx:stateReference>
    </sx:exerciseLimit>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>
```

[0051]

[0052] [0013] 따라서, 전형적인 라이선스 L2는, 볼드체로 표시된 부분을 제외하고는, 전형적인 라이선스 L1과 비슷할 수 있는데, 그 볼드체 부분은 노래파일 456이 장치 123에 의해 재생될 수 있는 횟수에 대한 제한이 있을 수 있다는 것과, 재생될 수 있는 노래파일 456의 남은 횟수가 카운트되어야 하고 장치 123에 검색 키 789로 저장되어야 한다는 것을 특징할 수 있다. 그렇지만, 다음의 제3 전형적인 라이선스 L3에 표현되어 있는 것처럼, 무제한적인 라이선스 재배포를 정체가 확인된 장치들에 대하여 허용하기 위해 라이선스가 채용될 수 있다.

[0053] [0014] 제3의 전형적인 라이선스 (L3) :

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <grant>
    <cm:device cmid="123"/>
    <issue/>
    <grant>
      <cm:device cmid="111"/>
      <cx:play/>
      <cm:song cmid="456"/>
    </grant>
  </grant>
  <grant>
    <cm:device cmid="123"/>
    <issue/>
    <grant>
      <cm:device cmid="333"/>
      <cx:play/>
      <cm:song cmid="456"/>
    </grant>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0054]

[0055] [0015] 전형적인 라이선스 L3은 볼드체 부분을 제외하고는 전형적인 라이선스 L1과 유사할 수 있으며, 볼드체 부분은 상기 장치들을 노래 파일 456이 재배포될 수 있는 장치인 111과 333으로서 확인할 수 있다. 따라서 전형적인 라이선스 L3을 가지고서, 장치 123은 장치 111과 333이 노래파일 456을 재생하는 것에 관한 허락을 발행할 권리를 갖는다. 유리하게도, 장치 111과 333은, 발행된 라이선스들이 전형적인 라이선스 L1과 거의 동일하게 보이는 것과 같이, 단지 장치 확인(identifications)에 있어서만 차이가 있는 장치 123 만큼 세련될 필요는 없다. 그러나 라이선스는, 다음의 제4 전형적인 라이선스 (L4)에 표현된 것처럼, 확인되지 않은(unidentified) 장치들에 대한 제한된 라이선스 재배포를 허락할 수 있다.

[0056] [0016] 제4의 전형적인 라이선스 (L4) :

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <grant>
    <forAll varName="otherDevice"/>
    <cm:device cmid="123"/>
    <issue/>
    <grant>
      <cm:device varRef="otherDevice"/>
      <cx:play/>
      <cm:song cmid="456"/>
    </grant>
    <sx:exerciseLimit>
      <sx:stateReference>
        <cm:countKeeper key="555">
          <cm:device cmid="123"/>
        </cm:countKeeper>
      </sx:stateReference>
    </sx:exerciseLimit>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0057]

[0058] [0017] 전형적인 라이선스 L4는 볼드체 부분을 제외하고는 전형적인 라이선스 L3과 유사할 수 있으며, 볼드체 부분은, 노래파일 456을 재생할 수 있는 라이선스를 확인되지 않은 장치들에게, 제한된 횟수만큼, 검색 키 555 하의 장치 123에 저장되어 유지되고 있는 횟수를 가지고서, 장치 123이 선택적으로 배포하도록 허락하거나 또는 요구하는 데 이용될 수 있다. 유리하게도, 장치 123이 라이선스들을 발행한 장치들은, 발행된 라이선스들이 전형적인 라이선스 L1과 거의 동일하게 보이는 것과 같이, 단지 장치 확인(identifications)에 있어서만 차이가 있는 장치 123 만큼 세련될 필요는 없다.

[0059] [0018] 라이선스는 또한, 확인된 장치에서, 고정된 시간 간격으로 제한되도록, 다음의 제5의 전형적 라이선스 L5에 표현된 것처럼, 확인된 노래 파일의 재생을 특정할 수 있다.

[0060] [0019] 제5의 전형적인 라이선스 (L5):

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
    <validityInterval>
      <notBefore>2002-03-04T05:06:07</notBefore>
      <notAfter>2002-03-11T05:06:07</notAfter>
    </validityInterval>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0061]

[0062] [0020] 전형적인 라이선스 L5는, 볼드체 부분을 제외하고는, 전형적인 라이선스 L1과 비슷할 수 있으며, 그 볼드체 부분은 장치 123이 특정된 시간 간격 동안 노래파일 456을 재생하는 것에 한정될 수 있도록 특정하는 것에 사용될 수 있다. 확인된 장치상에서, 확인된 노래 파일의 재생은 또한, 다음의 제6의 전형적인 라이선스 L6에

표현된 것과 같이, 고정된 시간 간격으로 제한될 수 있으며, 그 시간간격은 노래 파일이 재생될 수 있는 최초 시간에 시작될 수 있다.

[0063] [0021] 제6의 전형적인 라이선스 (L6) :

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
    <sx:validityIntervalFloating>
      <sx:stateReference>
        <cm:intervalKeeper key="777">
          <cm:device cmid="123"/>
        </cm:intervalKeeper>
      </sx:stateReference>
    </sx:validityIntervalFloating>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0064]

[0022] 전형적인 라이선스 L6은, 볼드체 부분을 제외하고는, 전형적인 라이선스 L1에 유사하며, 그 볼드체 부분은 장치 123이 노래파일 456이 재생되는 맨 처음 시간부터 그 노래파일 456을 재생하는 것에 한정될 수 있다는 것을 특정하는 데 사용되며, 남은 시간과 시작시간은 장치 123에 의해 검색 키 777 하에 저장된다. 또 다른 예에서, 다음의 제7의 전형적인 라이선스 L7에 표현된 것처럼, 확인된 장치에 의해 확인된 노래파일을 소정 개수까지의 다른 장치로 대여하는 것을 허락하는 데 라이선스가 사용될 수 있다.

[0066] [0023] 제7의 전형적인 라이선스 (L7) :

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <grant>
    <forAll varName="otherDevice"/>
    <forAll varName="interval"/>
    <cm:device cmid="123"/>
    <issue/>
    <grant>
      <cm:device varRef="otherDevice"/>
      <cx:play/>
      <cm:song cmid="456"/>
      <validityInterval varRef="interval"/>
    </grant>
    <cm:copyManagement>
      <sx:stateReference>
        <cm:copyKeeper key="999">
          <cm:device cmid="123"/>
        </cm:copyKeeper>
        <serviceParameters>
          <datum>
            <validityInterval
              varRef="interval"/>
          </datum>
        </serviceParameters>
      </sx:stateReference>
    </cm:copyManagement>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0067]

[0068] [0024] 전형적인 라이선스 L7은 볼드체 부분을 제외하고 전형적인 라이선스 L4와 비슷할 수 있으며, 그 볼드체 부분은 가변 시간 제약, 복사 관리 제약을 특정할 수 있다. 예를 들면, 볼드체 부분의 문자 그대로의 해석은 "장치 123이 하나의 장치와 간격을 선택할 수 있고, 그런 다음, 만약 장치 123이 선택된 간격을 장치 123 상에 있는 검색 키 999를 사용하는 복사 파수꾼(copy keeper)에 전달하고 그 복사 파수꾼이 선택된 간격 동안에 복사본이 여전히 남아있다는 것을 검증한다면, 선택된 장치가 선택된 간격동안 노래파일 456을 재생하는 것을 허락하는 라이선스를 발행할 수 있다.

[0069] [0025] 유리하게도, 장치 123이 노래 파일 456을 대여해준 장치들은 장치 123 만큼 세련될 필요는 없다. 그 발행된 라이선스들은 전형적인 라이선스 L5와 거의 동일하게 보일 것이고 장치 확인과 유효화 간격에 있어서만 다를 뿐이기 때문이다. 이에 비해, 다음의 제8의 전형적인 라이선스 L8에 표현된 것처럼, 노래는 무제한의 횟수 만큼 재생되도록 세련될 수 있다.

[0070] [0026] 제8의 전형적인 라이선스 (L8) :

```

<license>
  <grant>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0071]

[0072] [0027] 전형적인 라이선스 L8은 확인된 장치 123의 생략부분을 제외하고는 전형적인 라이선스 L1과 유사할 수 있다. 유리하게, 어떤 적당한 장치는 전형적인 라이선스 L8을 가지고 노래 파일 456을 재생할 수 있고, 반면에 다음 예를 가지고서 동일한 도메인 안의 다수의 장치들은, 다음의 제9의 전형적인 라이선스 (L9)에 표현된 것처럼, 확인된 노래 파일을 재생할 수 있다.

[0073] [0028] 제9의 전형적인 라이선스 (L9) :

```

<license>
  <grant>
    <forall varName="device">
      <everyone>
        <cm:myDomain/>
        <trustedIssuer>
          <cm:device cmid="000"/>
        </trustedIssuer>
      </everyone>
    </forall>
    <cm:device varRef="device"/>
    <cx:play/>
    <cm:song cmid="456"/>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="024"/>
  </cm:issuerDevice>
</license>

```

[0074]

[0075] [0029] 전형적인 라이선스 L9는 볼드체 부분을 제외하고는 전형적인 라이선스 L1과 유사할 수 있으며, 그 볼드체 부분은 "장치 000가 그의 영역 안에 있을 수 있다고 말하는 모든 장치가 노래 파일 456과 기타를 재생할 수 있는 것으로 해석될 수 있다."고 이해될 수 있다. 유리하게, 이하의 제10의 전형적인 라이선스 (L10) 에서 표현되는 것처럼, 장치 000가 다른 약간의 장치가 그 도메인 안에 있을 수 있는 것을 특정하는 방법을 제공하기 위해 라이선스가 채용될 수 있다.

[0076] [0030] 제10의 전형적인 라이선스 (L10) :

```

<license>
  <grant>
    <cm:device cmid="123"/>
    <possessProperty/>
    <cm:myDomain/>
    <validityInterval>
      <notBefore>2002-03-01T00:00:00</notBefore>
      <notAfter>2002-03-15T00:00:00</notAfter>
    </validityInterval>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid="000"/>
  </cm:issuerDevice>
</license>

```

[0077]

[0078] [0031] 전형적인 라이선스 L10은 볼드체 부분을 제외하고는 전형적인 라이선스 L1과 유사할 수 있으며, 여기서 발행자는 장치 000일 수 있고, 그것을 장치 123이 특정된 시간 간격 동안에 장치 000의 영역에 있다는 것을 증명할 수 있다. 예를 들면, MP3 플레이어와 같은 장치 123은 특정된 시간 간격동안에 노래 파일 456을 재생할 수 있다. 만약 개인용 컴퓨터와 같은 장치 000이, 특정된 시간 간격동안에, 장치 123이 그 영역 안에 있다는 것 예컨대, 장치 000의 동일한 네트워크 상에 있다는 것을 증명할 수 있다. 유리하게, 만일 장치 123이 지정받았던 시간 간격동안 장치 000로부터 전형적인 라이선스 L10을 받으면, 장치 123은 장치 123이 노래 파일 456을 재

생할 수 있었다라는 것을 알았을 것이다. 전형적인 라이선스 L9는 아마 노래 파일 456과 함께 장치 123에 도착할 것이다. 전형적인 라이선스 L10은 그러나 아마 다른 수단을 통하여 도착할 것이다. 예를 들면, 장치 123이 장치 000 도메인에서 가정의 네트워크에 연결될 때마다, 장치 000은 장치 123에게 새롭거나 갱신하게 되었던 전형적인 라이선스 L10를 보낼지도 모른다.

[0079] [0032] 이하의 제11의 전형적인 라이선스 (L11) 에서 나타내게 되는 것처럼 최종적인 예는 예를 들면, 안전 모델 또는 신뢰 정책이 장소에 넣어질 수 있게, 실체들의 안전한 신원확인(identification)을 가능하게 하는 라이선스에 방향이 맞춰진다.

[0080] [0033] 제11의 전형적인 라이선스 (L11) :

```

<license>
  <grant>
    <keyHolder>
      <info>
        <dsig:KeyValue>
          <dsig:RSAKeyValue>
            <dsig:Modulus>oRUTUiTQkM.....</dsig:Modulus>
            <dsig:Exponent>AQABAA==</dsig:Exponent>
          </dsig:RSAKeyValue>
        </dsig:KeyValue>
      </info>
    </keyHolder>
    <cx:play/>
    <digitalResource>
      <secureIndirect URI="http://www.xrml.org/.../songs/456">
        <dsig:DigestMethod Algorithm="http://.../xmldsig#sha1"/>
        <dsig:DigestValue>PB4QbKQCo941tTExbjl/Q==</dsig:DigestValue>
      </secureIndirect>
    </digitalResource>
    <sx:exerciseLimit>
      <sx:stateReference>
        <uddi>
          <serviceKey>
            <uuid>D04951E4-332C-4693-B7DB-D3D1D1C20844</uuid>
          </serviceKey>
        </uddi>
      </sx:stateReference>
    </sx:exerciseLimit>
  </grant>
  <issuer>
    <dsig:Signature>
      <dsig:SignedInfo>
        <dsig:C...14...NMethod Algorithm="http://.../REC-xml-c14n-20010315"/>
        <dsig:SignatureMethod Algorithm="http://.../xmldsig#rsa-sha1"/>
        <dsig:Reference>
          <dsig:Transforms>
            <dsig:Transform Algorithm="http://.../xrml2core#license"/>
          </dsig:Transforms>
          <dsig:DigestMethod Algorithm="http://.../xmldsig#sha1"/>
          <dsig:DigestValue>PB4QbKQCo941tTEx...</dsig:DigestValue>
        </dsig:Reference>
      </dsig:SignedInfo>
      <dsig:SignatureValue>AYmqOhSHbiP9JadD2...</dsig:SignatureValue>
      <dsig:KeyInfo>
        <dsig:KeyValue>
          <dsig:RSAKeyValue>
            <dsig:Modulus>X0j9q9OAxvhre4NP6q...</dsig:Modulus>
            <dsig:Exponent>AQABAA==</dsig:Exponent>
          </dsig:RSAKeyValue>
        </dsig:KeyValue>
      </dsig:KeyInfo>
    </dsig:Signature>
  </issuer>
</license>

```

[0081]

[0082] [0034] 전형적인 라이선스 L11는 첫 번째에서 네 번째 볼드체 부분까지를 제외하고는 전형적인 라이선스 L2에 유사할 수 있다. 여기서 장치 식별자 123은 장치 123의 공개키 식별정보(identification)로 대체할 수 있으며,

노래 파일 456은 해쉬 값을 통하여 노래 파일 456에 대한 암호표기법상 안전한 참조로 대체할 수 있으며, 장치 123상에 있는 검색 키 789는 온라인 카운터 서비스의 웹서비스 등록저장소(Universal Description, Discovery and Integration: UDDI) 서비스 키로 대체될 수 있으며, 발행자 장치 024는 발행자의 공개 키 식별정보와 전형적 라이선스 L1의 발행자 서명으로 대체될 수 있다.

[0083] [0035] 유리하게도, 효율적인 전형적인 프로파일들은 전형적인 라이선스 L1-L11에 의거하여 구축될 수 있다. 전형적인 라이선스 L1에 의거하여, 프로파일 B라 불리는 전형적인 프로파일이 만들어지고 생성될 수 있다. 만약 다른 cmid 값을 갖는 가능한 예외를 가지면서, 라이선스가 전형적인 라이선스 L1과 유사할 수 있다면, 라이선스는 전형적인 프로파일 B에 있을 수 있다.

[0084] [0036] 예를 들면, 전형적인 라이선스 L3가 전형적인 프로파일 B에 있을 수 없는데, 이는 전형적인 프로파일 B에 포함될 수 없었던 전형적인 라이선스 L3에 포함된 추가적인 cmid 값 111과 333 때문이다. 그러나 장치 123이 장치 111 혹은 333에게 발행할 수 있는 라이선스는, 예를 들면, 전형적인 무제한 재배포 라이선스 L3 등에 준하여, 전형적인 프로파일 B에 있을 수 있다. 이 경우에, 장치 123에 의해 발행된 라이선스는, 발행자 장치가 024에서부터 123까지 변하는 것과 재생장치 123이 111 혹은 333 등으로 변하는 것을 제외하고는, 전형적인 라이선스 L1과 유사할 수도 있다. 또한, 재미없게도 전형적인 라이선스 L1이 전형적인 프로파일 B 안에 있을 수 있다.

[0085] [0037] 유사한 방법으로, 프로파일 C로 불리는, 전형적인 프로파일은, 전형적인 라이선스 L2에 의거하여, 구축되고 생성된다. 예를 들면, 만약에 (i) 그 라이선스가 예를 들면, 전형적인 라이선스 L2의 cmid 값들인 123, 456, 024에 대응되는 다른 cmid 값들을 갖는 것과 같은 가능한 예외를 가지면서, 전형적인 라이선스 L2와 유사할 수 있고, (ii) 그것에 관해 카운트가 유지되는 그런 장치의 cmid 값은 정체가 확인된 노래 파일 등을 재생할 권리를 행사할 수 있는 그 장치의 cmid 값과 같을 수 있다면, 라이선스는 전형적인 프로파일 C 안에 있을 수 있다.

[0086] [0038] 도 2는 권리표현 프로파일링 처리를 나타내는 흐름도이다. 도 2에서, 단계201에서는 예를 들어 프로파일링 처리는 라이선스를 획득하는 과정을 포함할 수 있다. 단계203에서, 예를 들어 라이선스는 프로파일에 대하여 비교될 수 있다. 단계205에서, 예를 들어 라이선스와 프로파일을 비교함으로써 라이선스가 프로파일내에 존재하는지 아닌지가 결정하여, 라이선스와 프로파일이 하나이상의 대응하는 변수값들을 가지는 것을 제외하고는 서로 비슷한지를 결정한다. 단계205에서, 라이선스가 프로파일에 있는 것으로 결정된다면, 라이선스는 단계 207에서 프로파일내에 존재하는 것으로서 분류될 수 있다. 그렇지않다면, 단계209에서 라이선스가 프로파일에 대하여 무효하다라고 분류될 수 있다.

[0087] [0039] 바람직하게는, 프로파일링은 장치에게 예를 들어 그들의 성능 등을 쉽게 분류하도록 허락할 수 있다. 예를 들어, 단지 프로파일 B를 지원하는 장치는 식별된 노래파일 등을 연주하는 권리의 실시와 관련된 제한들을 지원하지 않는다. 이에 비해, 그 상태가 동일한 장치에서 유지될 수 있는 한, 프로파일 C를 지원하는 장치는 실시 제한들을 지원할 수 있다.

[0088] [0040] 프로파일링 및 템플릿 매핑은, 바람직하게는 표 2에 도시된 전형적인 바이트 테이블내에 도시된 바와 같이, 권리표현의 압축을 위하여 채용될 수 있다.

[0089] 표 2: 전형적인 바이트 테이블.

[0090]

바이트	의미
-----	----

0	값	압축 알고리즘
	0x00	예약됨
	0x01	프로파일 B용 알고리즘 B
	0x02	프로파일 C용 알고리즘 C
	0x03-0xFE	미래의 알고리즘들
	0xFF	비압축됨
1-3	(비)압축된 데이터의 사이즈	
4에서x+3까지	(비)압축된 데이터	

[0091] 표 2에서 보는 바와 같이, 예를 들어 바이트 0은 채용된 알고리즘의 식별번호, 프로파일, 템플릿과 같은 정보를 저장하기 위하여 사용될 수 있고, 바이트 1-3은 압축되거나 비압축된 데이터를 저장하기 위하여 채용되는 페이로드(payload) 바이트들의 수 x 를 지정하기 위하여 사용될 수 있고, 바이트 4부터 $x+3$ 까지는 압축되거나 비압축된 데이터 등을 저장하기 위하여 사용될 수 있다.

[0092] [0042] 전형적인 실시예에 따르면, 프로파일 B를 위한 알고리즘 B, 또는 그와 통합된 식별번호(예를 들어, 1)를 갖는 프로파일 B에 의거한 템플릿 T1은 멤버 표현 E를 12 페이로드 바이트(예를 들어, $X = 12 = 4$ 바이트 곱하기 대응하는 3개의 표현들의 3개의 *cmid* 값들) 아래로 압축하기 위하여 채용될 수 있다. 그런 예시적인 비트 구조로 압축된 전형적인 라이선스 L1은 표 3에서 보는 바와 같이, 전형적인 압축된 바이트 테이블 내에 도시될 수 있다.

[0093] 표 3 : 전형적인 압축된 바이트 테이블.

[0094]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
16 진수	01	00	00	0C	00	00	00	0B	00	00	01	C8	00	00	00	18
10 진수	1	12			123				456				24			
	<i>id</i>	<i>x</i>	<i>cmid 1</i>			<i>cmid 2</i>				<i>cmid 3</i>						

[0095] [0043] 표 3에서 보는 바와 같이, 예를 들어 바이트 0은 프로파일 B 또는 템플릿 T1을 위한 알고리즘 B를 위한 식별번호 1을 저장하는 데 사용될 수 있고, 바이트 1-3은 채용된 페이로드 바이트의 수 $x(x=12)$ 를 저장하는 데 사용될 수 있고, 바이트 4에서 15($x+3$)까지는 전형적인 라이선스 L1으로부터의 *cmid*값들 123,456 및 024를 저장하는 데 사용될 수 있다.

[0096] [0044] 전형적인 실시예에 따르면, 프로파일 C 또는 그와 통합된 식별번호(예를 들어, 2)를 갖는 프로파일 C에 의거한 템플릿 T2를 위한 알고리즘은 어떤 그것의 멤버표현 E를 16 페이로드 바이트 이하로 압축하기 위하여 채용될 수 있다(예를 들어, $x = 14 = 4$ 바이트 곱하기 대응하는 4개의 표현들의 4개의 *cmid* 값들). 전형적인 비트 구조로 압축된 전형적인 라이선스 L2는 표 4에서 보는 바와 같이, 전형적인 압축된 바이트 테이블내에 도시될 수 있다.

[0097] 표 4. 전형적인 압축된 비트 테이블.

[0098]

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
16 진수	02	00	00	10	00	00	00	7B	00	00	01	C8	00	00	03	15	00	00	00	18
10 진수	1	16			123				456				789				24			
	<i>id</i>	<i>x</i>	<i>cmid 1</i>			<i>cmid 2</i>				<i>cmid 3</i>				<i>cmid 4</i>						

[0099] [0045] 표 4에서 보는 바와 같이, 예를 들어 바이트 0은 프로파일 C 또는 템플릿 T2를 위한 알고리즘 C를 위한 식별번호 2를 저장하는 데 사용될 수 있고, 바이트 1-3은 채용된 패이로드 바이트의 번호 x (x=16)를 저장하는 데 사용될 수 있고, 바이트 4에서 19(x+3)까지는 전형적인 라이선스 L2로부터의 cmid값들 123, 456, 789 및 024를 저장하는 데 사용될 수 있다.

[0100] [0046] 바람직하게는, 전형적인 비트구조는 추가적인 압축 알고리즘, 프로파일 또는 템플릿을 전형적인 비트구조 이득성능을 채용하는 장치들로서, 새로운 알고리즘, 프로파일로서 추가하도록 허락하고, 템플릿은 창작되고, 생성되고, 파생되고, 결정된다. 또한, 전형적인 비트구조는 바람직하게는 비압축되거나 비가공된(raw) 데이터, 예를 들어 비압축되거나 비가공된 권리포현의 송수신을 허락한다.

[0101] [0047] 전형적인 프로파일 B 또는 템플릿 T1은 도 5에서 보는 바와 같이, 전형적인 프로파일 또는 템플릿 테이블내에 도시된 바와 같이, 전형적인 라이선스 LI를 4부분 B1-B4로 분할하는데 사용될 수 있다.

[0102] 표 5 : 전형적인 프로파일 또는 템플릿.

자원 스트링 명칭	값
B1	<pre><license> <grant> <cm:device cmid="</pre>
B2	<pre>"/> <cx:play/> <cm:song cmid="</pre>
B3	<pre>"/> </grant> <cm:issuerDevice> <cm:device cmid="</pre>
B4	<pre>"/> </cm:issuerDevice> </license></pre>

[0104] [0048] 바람직하게는, 부분 B1-B4 는 비압축되거나 비가공된 권리포현과 같은 비압축되거나 비가공된 데이터를 보내거나 받는 자원-제한적인 장치(resource-constrained device)와 같은 장치 내에 자원 스트링(string)으로 저장될 수 있다. 자원-제한적인 장치와 같은 프로파일 B장치는 전형적인 프로파일 B 또는 템플릿 T1에 대응하는, 비가공되거나 비압축된 권리포현 스트링과 같은 비가공되거나 비압축된 데이터 스트링 W를 수신할 수 있다.

[0105] [0049] 프로파일 B장치는 표 6에 그리고 아래 위조 코드에 도시된 바와 같이, 호출하고, 주문하고(involve), 실행하도록 구성될 수 있다.

[0106] 바람직하게는, 제1전형적 기능은 프로파일 B장치가 노래를 연주할 수 있는지를 결정하는데 채용될 수 있다.

[0107] 표 6 : 제1의 전형적 함수.

[0108]

아이템	명칭	설명
함수	<i>interpretXrML_B</i>	프로파일 B 장치가 노래를 재생할 수 있는지를 결정한다.
입력	<i>W</i>	가공되지 않은 XrML
입력	<i>deviceId</i>	장치의 id
입력	<i>songId</i>	노래의 id
출력	<i>result</i>	만약 라이선스가 프로파일 B에 없으면, 에러 1 만약 장치가 부정확하면, 에러 2 만약 노래가 부합하지 않으면, 에러 3 그렇지 않으면, 결과는 발행자 장치의 id를 나타내는 에러-없음이다. 발행자 장치가 신뢰할 수 있는지를 검증하는 것은 호출자의 책임이다.

[0109]

[0050] 비압축 날짜 *W*를 갖는 의사코드(pseudo-Code)내의 제1의 전형적인 함수:

```

Function interpretXrML_B(W, deviceId, songId) {
  Read through W comparing to B1
  If mismatch before end of B1 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  If temp does not equal deviceId then ERROR 2
  Read through W comparing to B2
  If mismatch before end of B2 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  If temp does not equal songId then ERROR 3
  Read through W comparing to B3
  If mismatch before end of B3 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  Set result = temp
  Read through W comparing to B4
  If mismatch before end of B4 then ERROR 1
  Success. Return result.
}
    
```

[0110]

[0111]

[0051] 바람직하게는, 제1전형적 함수의 성능은 아래 나타낸 바와 같이, 압축된 데이터 스트링 *W*상에서 동작할 때, 최적화될 수 있다.

[0112]

[0052] 압축된 데이터 *W*를 가진 의사-코드내에서의 첫번째 전형적인 함수:

```

Function interpretXrML_B(w, deviceId, songId) {
  algorithm = readBytes(w, 1);
  size = readBytes(w, 3);
  if(algorithm != 1 OR size != 12) {
    skipBytes(w, size);
    ERROR 1;
  }
  if(deviceId != readBytes(w, 4)) {
    skipBytes(w, 8);
    ERROR 2;
  }
  if(songId != readBytes(w, 4)) {
    skipBytes(w, 4);
    ERROR 3;
  }
  result = readBytes(w, 4);
  return result;
}
    
```

[0113]

[0114] [0053] 라이선스와 같은 권리표현의 엔코딩은 XrML과 같은 원래 그대로의 형태로부터 권리표현을 자원-계약적인 장치상에 채용될 수 있는 라이선스의 템플릿 또는 프로파일 대응부(counterpart)로 변환하는 과정을 포함할 수 있다. 그런 엔코딩을 수행하려는 접근은 있는 그대로의 엔코딩(explicit encoding, 이하 '익스플리시트 엔코딩')으로서 지칭되고, 그것은 일반적으로 원 라이선스가 라이선스의 템플릿 또는 프로파일 대응부에 거의 일치될 수 있을 때, 가장 잘 동작한다.

[0115] [0054] 도 3은 도 1의 디지털 권리 관리 시스템내의 자원-계약적인 장치들을 위한 전형적인 권리표현 엔코딩 및 디코딩 서브-시스템을 도시한다. 도 3의 서브시스템에서, 예를 들어 XrML 파서(parser), XrML 해석기와 같은 최상의 자원은 자원-계약적인 장치상에서의 명백한 엔코딩을 위하여 채용될 필요는 없다. 예를 들어, 장치(301)는 비엔코드 된 라이선스(303)를 수신하고, 예를 들어 자원 제약 때문에 라이선스(303)를 파싱(parsing)하는 대신에, 장치(301)가 나중에 처리할 수 있는 형태로 라이선스(303)을 엔코드하기 위해 익스플리시트 엔코더(explicit encoder)(305)를 채용할 수 있다. 이 경우, 유리하게도, 범용 해석기를 채용할 필요 없이, 오히려 자원계약적이거나 혹은 풍부한 기능을 갖춘 해석기보다 저급의 해석기(307)를 가지고 채용한다.

[0116] [0055] 아래에서 보는 바와 같이, 제1의 전형적 라이선스 L1에 대응하는 템플릿 T1 또는 프로파일 B는, 예컨대 cmid 변수들에 관한 값들처럼, 전형적 라이선스 L1의 하나의 인스턴트 생성(instantiation)으로부터 다른 것으로 가장 잘 변하기 쉬운 것 같은 파라미터들, 값 또는 데이터를 제거함으로써 생성될 수 있다.

[0117] [0056] 전형적 템플릿 T1 또는 프로파일 B :

```

<license>
  <grant>
    <cm:device cmid=""/>
    <cx:play/>
    <cm:song cmid=""/>
  </grant>
  <cm:issuerDevice>
    <cm:device cmid=""/>
  </cm:issuerDevice>
</license>

```

[0118]

[0119] 057] 전형적인 템플릿 T1 또는 프로파일 B는 XrML 기반의 템플릿, 프로파일 등의 형태로 될 수 있다. 전형적인 템플릿 TL 또는 프로파일 B는 예를 들어 특별한 장치에 특별한 노래를 연주하는 권리를 허락할 수 있는 라이선스를 표현하기 위하여 사용될 수 있고, 전형적인 라이선스 L1등을 발행할 수 있는 장치를 참조할 수 있다.

[0120] [0058] 전형적인 템플릿 T1 또는 프로파일 B는 표 5에서 나타내게 되는 것처럼 전형적인 라이선스 L1를 4개의 부분 B1-B4로 나누는 3개의 가변 요소들을 포함할 수 있다. 유리하게, 부분 B1-B4는 자원 스트링으로서 장치(301) 안에 저장될 수 있어, 장치(301)이 엔코드 되지 않은 라이선스(303) 예를 들면, 가공되지 않거나 압축되지 않은 권리표현들(W)을 보낼 수 있고 그리고 받을 수 있도록 할 수 있다.

[0121] [0059] 만일 장치(301)가 가공되지 않은 XrML 스트링 W를 받을 수 있으면, 장치(301)은, 장치(301)가 사용방법, 예컨대 노래(309)를 연주하는 것 같은 것을 특정하는 사용권을 행사할 수 있는지를 결정하는 제1 전형적인 함수를 부를 수 있다. 비슷한 방법으로, 장치(301)은, 표 7과 아래의 의사-코드에 나타난 것처럼, 제2의 전형적인 함수를 불러서 예컨대 전형적인 라이선스 L1과 같은 엔코드된 라이선스를 발생시킬 수 있다. 제1 전형적인 함수는 해석기(307) 기능성의 일부로서 포함될 수 있으며, 제2 전형적인 함수는 엔코더(305) 기능성의 일부로서 포함될 수 있다.

[0122] 표 7 : 제2의 전형적인 함수

아이템	명칭	설명
함수	<i>iEncodeXrML_T1</i>	프로파일 B 장치가 라이선스를 엔코드할 수 있는지를 결정한다.
입력	<i>W</i>	가공되지 않은 XrML.
입력	<i>deviceId</i>	장치의 id.
입력	<i>songId</i>	노래의 id.
출력	<i>result</i>	만약 라이선스가 T1이 아니라면, 에러 1 엔코드된 T1.

[0124] [0060] 의사코드내의 제2의 전형적인 함수 :

```

Function EncodeXrML_T1(W,deviceId, songId) {
  Read through W comparing to B1
  If mismatch before end of B1 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  Read through W comparing to B2
  If mismatch before end of B2 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  Read through W comparing to B3
  If mismatch before end of B3 then ERROR 1
  Read integer from W into temp until reaching end of numbers
  Read through W comparing to B4
  If mismatch before end of B4 then ERROR 1
  Success. Return the variables from L1.
}
    
```

[0125]

[0126] [0061] T1에서 기술된 것처럼 만약 전형적인 라이선스 L1이 장치 (301)의 프로파일 B에 있을 수 있다면, 전형적인 라이선스 L1은 장치 (301)에 의해 표 8 등에 도시된 것처럼 엔코드된 라이선스 표에서 엔코드될 수 있다.

[0127] 표 8 : 전형적인 엔코드된 라이선스

	0	1	2	3	4	5	6	7	8	9	10	11	12
16 진수	01	00	00	00	7B	00	00	01	C8	00	00	00	18
10 진수	1	123				456				24			
	<i>id</i>	<i>cmid 1</i>				<i>cmid 2</i>				<i>cmid 3</i>			

[0129] [0062] 표 8에서, 상응하는 12 바이트의 표현은 장치 (301)에 의해, 예를 들면, "식별번호 1을 갖는 템플릿을 사용하고 최초 3개의 변수에 값 123, 456 그리고 024를 채워넣어라"로 의미하도록 해석될 수 있다.

[0130] 유리하게, 예를 들면, 전형적인 라이선스 L1에 상응하는, 엔코드된 라이선스 (311)은, 템플릿, 혹은 프로파일, 그리고 그것들의 관련 변수들을 참조하기 위해 단지 갖는 것에 의해 극히 간결할 수 있으며, 그에 의해 전형적인 라이선스 L1의 전달 비용과 저장 비용을 줄어든다.

[0131] [0063] 도 4는 도 3의 전형적인 권리표현 엔코딩 및 디코딩 처리를 나타내는 흐름도이다. 도 4에서, 단계 401에서, 예를 들면, 엔코딩 프로세스는 라이선스를 얻는 것을 포함할 수 있다. 단계 403에서, 채용된 값들이 라이선스로부터 제거될 때까지, 단계 405에서 결정된 대로, 예를 들면 라이선스의 변수필드들로부터 하나 이상의 값이 추출될 수 있어 그 라이선스에 상응하는 템플릿을 발생시킨다. 단계 407에서는, 예를 들면, 템플릿 식별자가 부

여될 수 있으며, 단계 409에서는 추출된 값들과 템플릿 식별자는 인코딩된 라이선스로서 보내질 수 있다.

- [0132] [0064] 단계 411에서, 예를 들면, 디코딩 프로세스는 추출된 값들과 함께 템플릿 식별자를 수신하는 것과, 단계 415에서 정해진 바대로 템플릿 식별자에 상응하는 템플릿을 검색하는 단계 413을 포함할 수 있다. 만약 상응하는 템플릿이 발견될 수 없다면, 단계 417은 그 템플릿 식별자에 상응하는 템플릿을, 예를 들면 전형적인 템플릿 공간 서비스로부터, 후술하는 것처럼, 얻기 위해 채용될 수 있다. 일단 그 템플릿 식별자에 상응하는 템플릿이 얻어지면, 단계 419에서 그 받은 값들은 템플릿 식별자에 상응하는 템플릿 안으로 삽입될 수 있어 원래의 라이선스를 유리하게도 손실없는 방식으로 재구축할 수 있다.
- [0133] [0065] 본 발명은 예를 들면, 모든 라이선스들이 템플릿에 명백하게 매핑될 수 있는 것은 아니라는 것의 인가 혹은 특별한 장치 예컨대 슬레이브 장치가 지원할 수 있는 프로파일을 포함할 수 있다. 유리하게, 신뢰할 수 있는 마스터 장치는 라이선스를 해석할 수 있고 해석된 권리표현들과 슬레이브 장치가 지원하는 템플릿 세트 간의 매핑을 찾을 수 있고, 그 결과 얻어지는 템플릿을 슬레이브 장치에 발행할 수 있다.
- [0134] [0066] 예를 들면, 도 5는 도 1의 DRM 시스템내의 서브-시스템(500)을 매핑하는 전형적인 권리표현 템플릿을 도시한다.
- [0135] 도 5에서, 예를 들면, 라이선스 A(507)는 콘텐츠의 일부를 막연히 장치 A(505)의 도메인과 같은 특정의 홈 도메인 안에서 재생할 수 있는 권리를 허용할 수 있다. 그러나 장치 B(501)는 라이선스 템플릿 B(509)을 단지 지원할 수 있는데, 장치 B(501)는 템플릿 B(509)에 의해 주어진 것처럼, 날짜 D1부터 D2까지 사이에 그 콘텐츠를 재생할 수 있다.
- [0136] [0067] 일반적으로, 장치 B(501)가, 이 경우는 슬레이브 장치가 라이선스 A(507)를 해석하는 것은 불가능할 지도 모르는데, 왜냐하면 라이선스 A(507)는 예를 들면 구문, " 날짜 D1과 D2 사이에서 재생하는 것"의 의미를 포함하지 않기 때문이다. 사용자는 다른 신뢰할 수 있는 장치, 예컨대 장치 A(505), 이 경우에는 마스터 장치를, 아주 다양한 라이선스 예컨대 라이선스 A(507)를 해석하는 홈 도메인에서, 채용할 수 있다. 그러나, 보다 더 전형적인 실시예에서, 장치 A(505)는 장치 B(501)와 상호작용을 할 수 있어, 예를 들면, 장치 B(501)가 지원하는 템플릿의 리스트, 예를 들면 템플릿 B(509) 등을 얻을 수 있다.
- [0137] [0068] 또 다른 전형적인 실시예에서, 장치 A(505)는, 장치 B(501)가 예컨대 구성, 그리고 등록 메커니즘들과 같은 다른 방법을 통하여 지원할 수 있는 템플릿 리스트를 포함하는 지식 데이터베이스를 수립할 수 있다. 그러나 장치 A(505)가 예를 들면 템플릿 B(509)와 라이선스 A(507)를 해석할 수 있다면, 장치 A(505)는 라이선스 B(503)를 장치 B(501)에 발행할 수 있는데, 왜냐하면 템플릿 B(509)에 의거한 라이선스는 라이선스 A(507) 안에서 발행될 수 있는 권리들의 부집합을 포함한다고 간주될 수 있기 때문이다. 유리하게, 라이선스 B(503)는 템플릿 B(509)에 의거할 수 있고, 명백하게 인코딩될 수 있어서 장치 B(501)가 해석할 수 있는 인코딩된 대응 라이선스 B(511)을 발생시킬 수 있다. 따라서 라이선스 A(507)와 같은 일반적인 라이선스는 예컨대 장치 B(501)와 같은 장치들로 매핑되는 템플릿일 수 있는데, 그 장치 B(501)는 예를 들면 제한된 양의 템플릿들 혹은 프로파일들을 해석할 수 있는 것이어서 자원-계약적이다.
- [0138] [0069] 따라서 도 6은 도 5의 서브시스템의 처리들을 매핑하는 전형적인 템플릿을 도시하는 흐름도이다. 도 6의 단계 601에서, 예를 들면, 도메인 라이선스는 홈 컴퓨터 네트워크와 같은 홈 도메인에게, 확인된 콘텐츠의 일부 예를 들면 음악 파일, 영화파일, 이미지 파일 혹은 소프트웨어 파일을 재생하고, 보고, 표현하고, 실행하는 무제한의 권리를 허용하는 것을 얻을 수 있다. 사용자는 그런 다음 예를 들면, 홈 네트워크 상의 개인용 컴퓨터와 같은 마스터 장치로부터, 그 도메인 라이선스와 연관된 콘텐츠는 MP3 플레이어와 같은 자원-계약적인 장치에서 재생되어야 하는 것을 요청할 수 있다.
- [0139] [0070] 따라서 단계 603에서, 예를 들면, 마스터 장치는, 슬레이브 장치가 지원할 수 있는 라이선스들, 프로파

일들, 혹은 템플릿의 타입을 결정하기 위하여, 슬레이브 장치에 조회할 수 있다. 단계 605에서, 예를 들면, 마스터 장치는 슬레이브 장치는 도메인 라이선스를 지원할 수 없다는 것을 결정할 수 있고, 단계 607에서 예를 들면, 슬레이브 장치가 예를 들면 템플릿 서비스로부터 혹은 슬레이브 장치 그 자신으로부터 지원할 수 있는 라이선스에 상응하는 템플릿을 검색할 수 있다.

[0140] [0071] 단계 609에서, 예를 들면, 마스터 장치는 슬레이브 장치가 지원할 수 있는 라이선스를 검색된 템플릿에 의거하여 발행할 수 있다. 유리하게 슬레이브 장치는 도메인 라이선스를 해석할 수 있을 필요는 없으나, 단계 609에서 발행된 라이선스를 해석할 수는 있다. 유리하게, 발행된 라이선스는 도메인 라이선스에 특정된 권리들의 부집합을 포함할 수 있다.

[0141] [0072] 단계 611에서, 예를 들면, 마스터 장치, 슬레이브 장치 혹은 다른 장치는, 예를 들면 전형적인 실시예들에 관해서 기술된 것과 같은 적당한 방법으로 라이선스를 엔코드 하는 데 사용될 수 있다. 단계 613에서, 슬레이브는 엔코드된 라이선스를 해석하기 위한 간단한 것을 받을 수 있으며, 슬레이브는 단계 615에서 전형적인 실시예들에 관해 기술된 것처럼 어떤 적절한 방법으로 해석할 수 있다.

[0142] [0073] 유리하게, 기술된 전형적인 실시예들의 압축과 압축해제의 전형적인 형태는 손실이 없는 것으로 간주될 수 있으며, 슬레이브 장치에 대하여 허락을 하여, 변수들에 관한 하나 이상의 값들을 템플릿 혹은 프로파일의 저장된 복사본 안으로 되돌려놓음으로써, 발행된 라이선스의 엔코드된 형태로부터 발행된 라이선스를 복사, 재구축, 재발생 혹은 복제하도록 한다. 그리하여, 엔코드된 라이선스, 관련 템플릿 혹은 프로파일을 채용함으로써, 엔코딩 프로세스는 역전될 수 있어 원래의 라이선스가 유리하게도 그것의 엔코드된 형태로부터 재구축될 수 있다.

[0143] [0074] 전형적인 엔코딩과 디코딩 프로세스들은, 유리하게, 디지털 서명들에 채용될 수 있어서 발행된 라이선스가 교묘히 바꾸거나 절충하여 조작되어 버린 것은 아니라는 것을 확인하거나(validate), 인증하거나, 또는 보장한다. 예를 들면, 디지털 서명에 관한 값은 라이선스의 엔코드된 형태에 채용되는 변수 일 수 있다.

[0144] [0075] 도 7는, 그런 라이선스 서명 값 (705)에 의거하여, 도 3의 전형적인 권리표현 엔코딩 및 디코딩 서버-시스템을 더 도시한다. 도 7에서, 예를 들면, 서명값 (705)는, 하나 이상의 값들 (701)-(703)과 함께, 전형적인 라이선스 LUI 1와 같은 라이선스 (707)로부터 추출될 수 있는 템플릿 혹은 프로파일에 대응하는 변수들에 관한 값으로서, 채용될 수 있다.

[0145] [0076] 유리하게, 라이선스 (707)는, 값 (701)-(703)을 배달될 수 있는 엔코드된 라이선스 (709)의 형태로 보내는 것에 의해 엔코드될 수 있다. 상기 값들 (701)-(703)은 상기 대응하는 변수들을 템플릿 혹은 프로파일의 저장된 복사본 안에 자리잡아 디코드된 라이선스 (711)을 발생시키도록 하는 데 사용될 수 있다. 유리하게, 서명값 (705)는 대응하는 템플릿, 프로파일 등의 대응하는 변수 필드들 안으로 회복될 수 있고, 서명 확인, 인증 등의 기술들은 앞서 설명된 전형적인 실시예들에 따르는 것처럼, 디코드된 라이선스 (711)에 채용될 수 있다.

[0146] [0077] 전형적인 서명 기술은, 유리하게도, 라이선스 (709)를 엔코드할 수 있는 자원-제한적인 장치에 채용될 수 있지만, 편리함이 부족하거나 또는 엔코드된 라이선스 (709)를 서명 또는 재서명할 수도 있는데, 왜냐하면 그러한 장치는 신뢰 도메인의 일부가 아닐 수 있거나 세련된 프로세싱 능력들을 갖지 못 할 수 있기 때문이다. 더 전형적인 실시예에서는, 그러나, 라이선스의 엔코드된 형태가, 예를 들면 앞서 설명된 전형적인 실시예에 따르는 것과 같은, 어떤 적절한 디지털 서명 기술을 사용하여, 서명될 수 있다.

[0147] [0078] 예를 들면, 도 8은 도 3의 전형적인 권리표현 엔코딩 및 디코딩 서버-시스템을 라이선스 변수들 (801)-(803)에 관한 서명 값 (805)에 의거하여 더 도시한다. 도 8에서, 예를 들면, 만약 전형적인 엔코딩 프로세스들을 수행할 수 있는 장치가 신뢰 영역의 일부일 수 있다면, 그러한 장치는 원래의 라이선스 (807)을 엔코드된 형태 (809)로 변환하고, 라이선스 (809)의 엔코드된 형태를 서명하고, 그리고 서명값 (805)를 변수들 (801)-

(803)과 함께 보낸다.

[0148] [0079] 유리하게, 서명값 (805)는 변수들 (801)-(803)이 교묘히 바꾸거나 절충하여 조작된 것이 아니라는 것을 확인하거나, 인증하거나 혹은 보장하기 위해 채용될 수 있다. 그리하여 템플릿 식별자들, 프로파일 식별자들 그리고 템플릿 혹은 프로파일에 대응할 수 있는 변수들에 관한 값들의 무결성과 유효성이 보장될 수 있고 신뢰될 수 있다.

[0149] [0080] 전형적인 서명 방법들 중 어느 쪽이 사용될 수 있나에 의거하여, 서명값은 라이선스의 유효한 엔코드 또는 디코드된 형태들을 사용할 수 있다. 부가하여, 전형적인 템플릿 매핑이, 번역하는 동안에, 템플릿들을 프로파일링 기술로서 채용하는 것에 의해 일어날 수 있고, 채용된 권리언어 해석기는 기능이 풍부한(feature-rich) 언어 파서나 해석기일 필요는 없다. 예를 들면, 라이선스의 의미 해석은, 소정의, 미리 만들어진, 혹은 미리 설계된 대응 장치 예컨대 자원-제한적인 그런 장치안에서 일어날 수 있다.

[0150] [0081] 따라서, 유리하게, 장치가 라이선스를 받을 때, 그 장치는 그 라이선스의 의미를, 그 라이선스에 포함되어 있는 변수들과 함께, 통합할 수 있다. 예를 들면, 엔코드된 템플릿 혹은 프로파일은, 유리하게도, 다음의 의사-코드 조각에 의거하여, 해석될 수 있다.

```

Function interpretXrML_B(w, songID) {
  TemplateNumber=Readbytes (w, 1)
  LicDeviceID=Readbytes (w, 4)
  LicSongID=Readbytes (w, 4)
  if(TemplateNumber!= 1) {
    return(NO_PERMISSION_PLAY)
  }
  if(LicDeviceID!= THISDEVICEID) {
    return(NO_PERMISSION_PLAY)
  }
  if(LLicSongID!= songID) {
    return(NO_PERMISSION_PLAY)
  }
  return (YES_PERMISSION_PLAY)
}
    
```

[0151]

[0152] [0082] 유리하게, 해석의 그러한 전형적인 형태는, 템플릿이나 프로파일에서 설명될 수 있는 것의 이해를 장치가 실행할 수 있는 어떠한 함수들에 조화시키기 위해 개발자에게 의존할 수 있다. 따라서 권리언어는, 자신의 템플릿들 혹은 프로파일들, 그러한 장치의 개발자, 라이선스 발행자와 소정의 템플릿들이나 혹은 프로파일들에게 매핑 라이선스들이 될 수 있는 엔코딩 장치간의 이해의 공통적인 점으로서 채용될 수 있다.

[0153] [0083] 다른 전형적인 기술들이 제공되어, 유리하게, 전형적인 템플릿 매핑이나 프로파일링을 보충 또는 강화시킬 수 있고, 그리고 자원-제한적인 장치들에 효과적인 관리 콘텐츠를 배포를 용이하게 할 수 있다. 전형적인 템플릿공간이 사용되어 전형적인 템플릿 매핑을 통해 엔코드된 라이선스로 하여금 어느 템플릿이 그것과 함께 연관될 수 있느냐에 관해 지정할 수 있다.

[0154] [0084] 만일 코드화된 라이선스가 단지 "템플릿 1을 사용하라"고 특정한다면, "템플릿 1"의 의미는 모호하다. 유리하게, 전형적인 템플릿공간은 정수 "1"에 관한 문맥을 참조된 표현안에 제공하는 명칭공간일 수 있다. 이러한 방식에 있어서, 라이선스를 표 8에 나타난 것처럼, "1", "123", "456", "24"와 같이 연속해서 단순히 엔코딩하는 대신에, --여기서 "1"은 템플릿 식별자일 수 있고, "123", "456" 그리고 "024"는 변수들의 값일 수 있다-- , 라이선스는 "Acme.L", "123", "456", "024"로 엔코드될 수 있다.

[0155] [0085] 유리하게도, Acme. 1은 "1"이 간결한 의미를 유추할 수 있었던 명칭공간을 참조할 수 있다. 예를 들면, Acme 코퍼레이션은 자기의 제품 라인이 지원하는 일련의 잘-정의된 제품들을 생성할 수 있다. 템플릿공간-엔코

드된 라이선스를 받는 장치는, 예를 들면, "Acme"와 연관되는 데이터베이스를 액세스하는 것에 의해 적당한 템플릿을 식별할 수 있고, 그런 다음 그 장치는 정수 1을 통해 그 데이터베이스 안으로 인덱스할 수 있다.

[0156] [0086] 따라서 전형적인 템플릿공간들은, 유리하게, 템플릿들을 모호하지 않고 가능할 수 있는 방식으로, 검색되고, 저장되고, 관리되고, 액세스되도록 허락한다. 전형적인 템플릿공간들은 따라서: Acme Inc., Acme Cameras, Acme PVR Model 123, 바이오스 버전 1.3, 유럽의 개인용 비디오 리코더 메이커들, acmetemplates.com, 등을 포함할 수 있었다. 유리하게도, 자원-계약적인 장치들의 메이커들이 권리언어와 같이 배포될 수 있는 콘텐츠를 채용하므로, 각각의 제품 공간들에 관한 변수인 표준화된 템플릿들의 집합이 전개될 수 있다.

[0157] [0087] 템플릿공간들은 그 자체로 템플릿 라이프사이클들의 문제에 대하여 충분히 대처할 수 없을 지도 모른다. 그러나 유리하게도 템플릿공간 서비스들은 전개될 수 있다. 예를 들면, 템플릿공간 서비스는 템플릿 그 자체를 관리하는 서비스를 포함할 수 있고, 그래서 템플릿들은 전형적인 템플릿 매핑을 수행하고 있는 장치들과, 라이선스를 그것의 원래의 형태로 다시 디코드 하기를 원하는 시스템들에 의해 나중에 검색될 수 있다. 템플릿공간 서비스는 또한 다른 기능들을, 예컨대 주어진 회사에 의해 신청된 템플릿들에 관한 검색, 변경사항들과 갱신사항들에 관한 지원을, 템플릿들 및/또는 라이선스들에 대한 그것의 매핑에게 제공할 수 있다.

[0158] [0088] 유리하게, 템플릿공간 서비스는, 예를 들면, 웹 사이트, 자원-계약적인 장치 안으로 구축된 서비스와 같은 컴퓨터 네트워크 기반의 서비스를 포함하는, 다수의 방법론을 통해 전개될 수 있다. 예를 들면, 도 9는, 예를 들면 웹사이트 또는 자원계약적인 장치 안으로 구축된 서비스와 같은 컴퓨터 네트워크 기반의 서비스를 포함하는 다수의 방법론을 통해 전개될 수 있는 전형적인 템플릿공간 서비스 (900)을 도시한다.

[0159] [0089] 도 9에서, 사용자는, 개인용 컴퓨터와 같은 마스터 장치 (901) 위에 있는 풍부한 권리언어에 의해 한정되는 콘텐츠를, 라이선스 소스 (903)를 통하여, 얻을 수 있다. 사용자는 장치 제조사 (905)로부터, 갱신으로서, 새로운 바이오스를, 핸드헬드 이동 애플리케이터(mp3) 플레이어와 같은 슬레이브 장치 (907)용으로 수신할 수 있고, 그것은 사용자가 그 콘텐츠를, 전형적인 실시예들의 템플릿의 타입을 "날짜 a-b 동안에만" 재생하는 것과 같이 새로운 라이선스 템플릿을 가지고 적당하게 취급하도록 허락한다. 예를 들면, 새로운 바이오스는 슬레이브 장치 (907) 상의 다수의 새로운 템플릿들을 지원할 수 있다.

[0160] [0090] 공개적으로 접근하기 쉬운 네트워크 서비스 등과 같은, 템플릿 서비스 (909)를 통하여, 장치 제조업자는 템플릿들을 이용할 수 있게 할 수 있다. 사용자가, 슬레이브 장치 (907)로부터의 정보에 의거하여, 콘텐츠를 슬레이브 장치 (907)로 이동시킬 때, 마스터 장치 (901)는 슬레이브장치 (907)에 의해 지원되는 템플릿의 리스트를 템플릿 서비스 (909)로부터 얻을 수 있다. 마스터 장치 (901)은 템플릿공간 네트워크 서비스 (909)로부터의 권리언어를 해석할 수 있고, 지원되는 새로운 템플릿이 콘텐츠를 마스터 장치 (901)로부터 슬레이브 장치 (907)로 전달하기에 적당할 수 있다는 것을 결정할 수 있다. 마스터 장치 (901)는 그런 다음 전형적인 인코딩 기술들에 의거하여 새로운 라이선스를 동봉하여 슬레이브 장치 (907)에 발행할 수 있다.

[0161] [0091] 또 다른 전형적인 실시예에서, 유사한 흐름과 권리 라이프사이클이 완성될 수 있다. 그러나 그 대신에, 제조사 (905)는 슬레이브 장치 (907) 그 자신 위에서 그 템플릿들을 이용할 수 있게 해준다. 나아가, 오래된 장치들(legacy devices) 혹은 DRM에 대한 지원이 거의 없는 장치들은, 그러한 장치의 행동을 기술하는 산업-창설된 포괄적인 템플릿을 채용할 수 있다. 유리하게도, 오래된 장치는 어떤 형태의 권리언어들 검색하지 않을 것이나, 오히려 콘텐츠를 출처에 관계없이 같은 것으로 취급할 것이다.

[0162] [0092] 예를 들면, 디지털 텔레비전은 "디지털 텔레비전들"을 표현하기 위해 생성된 표준 권리 템플릿을 채용할 수 있다. 그러면 디지털 텔레비전의 제조업자들은 그런 템플릿에 표현되어 있는 조항들에 동의할 수 있고, 그리고 그것들의 콘텐츠 사용 규칙들을 그러한 템플릿으로 한정하는 제품들을 개발할 수 있다. 유리하게도, 마스터

장치는 라이선스를 해석할 수 있고, 그리고 사용권들이 허락될 수 있을 때에만 그러한 장치에 대한 콘텐츠 배달을 제어할 수 있다.

[0163] [0093] 전형적인 실시예에서, 예컨대 표 3, 5 및 8에 나타난 것처럼 템플릿 혹은 프로파일의 형태로 권리표현을 엔코드 등을 하기 위해 상징적인 표시가 채용될 수 있다. 상징적 표시는 글리프(glyph, 상형문자), 변조 혹은 변경된 빛 혹은 소리 신호들에 기반하는 것과 같은 변조 혹은 변경된 신호-기반 표시에 의거하는 그래픽 표시를 포함할 수 있다. 이와 같이, 템플릿 혹은 프로파일의 상징적 표시는, 별모양의 구조들, 정사각형 모양의 구조들, 직사각형 모양의 구조들, 원형 모양의 구조들, 삼각형 모양의 구조들 또는 그것들의 부분들과 같은, 글리프 구조들에 의거하여, 엔코드되고 유추될 수 있다. 비슷한 방식으로, 변조된 혹은 변경된 신호들, 빛들, 소리들, 일련의 소리들, 광 펄스들에 의거하여, 자외선파들, X-선들, 마이크로파들, 무선파들을 사용하여, 레이저들, 변조 혹은 뒤엎겨있는(convoluted) 정보, 다른 색상들 혹은 파장들, 그리고 전계 혹은 자계 변동들을 채용하여, 템플릿 혹은 프로파일이 엔코드될 수 있고, 유추될 수 있다.

[0164] [0094] 예를 들면, 도 10은 표 3, 5 혹은 8에 대응하는 템플릿들의 혹은 프로파일들의 전형적인 상징적 표시를 도시한다. 도 10에서, 예를 들면, 전형적인 상징적 표시는 예를 들면 별 모양의 구조 등에 의거하여 글리프 구조로부터 유추될 수 있고, 표 3, 5 혹은 8에 대응하는 템플릿들 혹은 프로파일들을 표현하기 위해 사용될 수 있다. 전형적인 실시예에 있어서, 별 모양의 구조로부터 유추된 심볼 (1001)-(1004)는 적, 녹, 그리고 청 색공간으로부터 취해질 수 있다. 이와 같이 별 모양 구조의 심볼 (1001)-(1004)는 적(R), 녹(G) 및/또는 청(B)의 색깔을 포함할 수 있다. 따라서 기본적인 경우에 있어서, 어떤 색깔들도 조합될 수 없는 그런 점에서, 12개의 심볼들이 생성될 수 있다 (예를 들면, 4개의 심볼 곱하기 3가지 색깔). 그렇지만, 구성원들 혹은 심볼들의 숫자를 늘리기 위해, 도 10에서 나타난 것처럼 중첩되는 색깔을 갖는 복합심볼들 (1011)-(1019)이 생성될 수 있다.

[0165] [0095] 이와같이, 예를 들면, 물리적인 장소에 겹쳐지기 위해, 그 별모양의 각 부분에 관해 다른 색깔들 혹은 색깔들의 조합인 별 모양 구조의 부분 혹은 모두를 잠재적으로 생산하기 위해, 모든 12개의 심볼들까지의 다른 색깔들의 적어도 2개 이상의 심볼들은 조합될 수 있다. 예를 들면, 복합심볼 (1011)의 결과로서 생기는 수평의 심볼들 (1001)의 색은 녹색과 적색 심볼들 (1001)의 조합에 의거할 수 있다. 이런 식으로, 여러 가지 색깔들의 조합들 안에서, 심볼 (1001)-(1004)는 도 10에서 나타내게 되는 것처럼 복합 심볼 (1011)-(1019)를 생성하기 위해 사용될 수 있다.

[0166] [0096] 따라서 도 10에서 예시된 전형적인 심볼 세트의 다수의 복합 심볼들은, 예를 들면, 빈 공간이 심볼로 간주되지 않는다는 가정하에, (212-1) 또는 4095 심볼들이 될 수 있다. 유리하게도, 전형적인 실시예에 의해 제공되는 복합 심볼들에 관한 많은 선택권을 가지고서, 문법, 언어 등에 관한 정보와 같은, 권리표현들에 관한 그에 상응하여 복잡적이고 포괄적인 상징적인 정보의 표시들이 생성될 수 있다.

[0167] [0097] 예를 들면, 도 10에서 나타내게 되는 것처럼, 전형적인 권리 표현 (1021)은 표 3, 5 또는 8에 대응하고 있는 템플릿들 또는 프로파일들을 가리킬 수 있다. 도 10에서, 예를 들면, 심볼 (1011)-(1019)는 템플릿 또는 프로파일 ID "1"을 나타내기 위해 사용될 수 있고; 그 "1" 다음에는 그 템플릿의 첫 번째 cmid 값 "123" 혹은 그 프로파일의 페이로드 바이트 "x"의 개수가 뒤따르고; 그 템플릿의 두 번째 cmid 값 "456" 혹은 그 프로파일의 첫 번째 cmid 값 "123"이 뒤따르며; 그 템플릿의 세 번째 이면서 마지막 cmid 값 "024" 혹은 그 프로파일의 두 번째 cmid 값 "456"이 뒤따르며; 그 프로파일의 세 번째 이면서 마지막 cmid 값 "024"가 뒤따른다. 이러한 방식으로 심볼들은, 표 3, 5, 혹은 8에 상응하는 템플릿들 혹은 프로파일들을 표현하기 위해 생성되고 보내질 수 있다.

[0168] [0098] 전형적인 실시예들에 있어서, 권리표현에 관한 정보를 나르는 데 사용되는 전형적인 실시예들의 글리프들, 심볼들, 단어들, 메시지들 등은, 빛, 소리, 일련의 소리들, 광 펄스들 등을 이용하여, 어떤 적절한 주파수, 진폭 등을 이용하여, 엔코드될 수 있고, 자외선 파들, x-선들, 마이크로파들, 무선파들 등을 포함할 수 있고,

레이저들, 변조되거나 혹은 뒤엎겨진 정보, 다른 색깔 혹은 파장들, 전계 또는 자계 변이 등을 채용할 수 있다.

- [0169] [0099] 하나의 전형적인 실시예에 있어서, 상기 전형적인 실시예들의 글리프들, 심볼들, 단어들, 메시지들 등의 어떤 적당한 시리즈라도, 전송되는 메시지에 어떤 의미를 제공하는 자기 자신의 문법, 규칙 등을 가질 수 있다.
- [0170] 예를 들면, 사람의 귀나 눈이 검출할 수 없는 주파수 범위에 있는 소리 또는 빛이, 상기 전형적인 실시예들의 글리프들, 심볼들, 단어들, 메시지들 등을, 보안 적용과 프라이버시 적용 동안에, 보내기 위해 채용될 수 있다. 이런 방식에 있어서, 보내진 메시지는 사용자나 혹은 사용자들에게 투명하게 만들어질 수 있으나, 동시에 적합한 센서, 검출기 등에 의해 검색되고 이해될 수도 있다. 전형적인 실시예에 있어서, 센싱된 정보, 메시지 등은 숫자들, 비트들, 바이트들 등에 의거하여, 양자화되고, 디지털화되고, 표현될 수 있지만 한다면, 상기 실시예들의 글리프들, 단어들, 메시지들 등은 어떤 적합한 사람 혹은 기계 센싱 기관 혹은 장치를 사용하여 표현될 수 있다.
- [0171] [00100] DRM 시스템(100)의 장치들과 서브시스템들은 예를 들면, 도 1-10에 관해서 기술된 것처럼, 예를 들면 통신 네트워크(170)의 위에서 통신할 수 있고, 그리고 기술된 전형적인 실시예들의 프로세스들을 수행할 수 있는, 예를 들면 어떤 적합한 서버들, 워크스테이션들, 개인용 컴퓨터들(PCs), 랩탑 컴퓨터들, PDA들, 인터넷 기구들, 셋탑 박스들, 모뎀들, 핸드헬드 장치들, 기타 장치들을 포함할 수 있다. DRM 시스템(100)의 장치들과 서브시스템들은, 예를 들면 어떤 적합한 프로토콜을 사용하여 서로간에 통신을 할 수 있고, 그리고 범용 컴퓨터 시스템을 사용하여 구현될 수 있다. 인터넷 액세스, 예컨대 음성, 모뎀 등과 같은 어떤 적합한 형태의 원거리 통신들, 무선통신 매체 등을 포함하여, 하나 이상의 인터페이스 메커니즘이 DRM 시스템(100)에서 사용될 수 있다. 따라서, 통신망(170)은 예를 들면, 무선통신망, 셀룰라 통신망, 공중전화망(PSTNs), 패킷데이터망(PDNs), 인터넷, 인트라넷, 하이브리드 통신망, 이들의 조합 등을 포함할 수 있다. 나아가, 통신망(170)은 같거나 다른 네트워크일 수 있다.
- [0172] [00101] 위에서 유의한 것처럼, 개시된 전형적인 실시예들을 구현하는 데 사용되는 특수한 하드웨어의 많은 변경들이 가능하므로, DRM 시스템(100)은, 도 1-10에 관해 설명된 것처럼, 예시적인 목적을 위한 것이 이해되어야 하겠다. 예를 들면, DRM 시스템(100)의 장치들과 서브시스템들의 기능성은 하나 이상의 프로그램된 컴퓨터 시스템 혹은 장치들을 통하여 구현될 수 있다. 다른 변경들 뿐 아니라 그러한 변경들을 구현하기 위하여 단일의 컴퓨터 시스템은 DRM 시스템(100)의 하나 이상의 장치 및 서브시스템의 특수 목적 함수들을 수행하도록 프로그램될 수 있다. 한편, 두 개 이상의 프로그램된 컴퓨터 시스템 혹은 장치들은 DRM 시스템(100)의 장치와 서브시스템들 중 어느 하나와 치환될 수 있다. 따라서 잉여성(redundancy), 복제성(replication) 등과 같은 분산 처리의 이론들과 장점들 역시 원하는 바대로, DRM 시스템(100)의 견고성과 성능을 향상시키기 위해 구현될 수 있다.
- [0173] [00102] DRM 시스템(100)은 예를 들면, 도 1-10에 관해서 기술한 것처럼, 여기서 설명된 다양한 프로세스들 o에 관련된 정보를 저장하는 데 사용될 수 있다. 이 정보는 DRM 시스템(100)의 장치들과 서브시스템들의 하드디스크, 광디스크, 광자기 디스크, 램(RAM) 등과 같은 하나 이상의 메모리들안에 저장될 수 있다. DRM 시스템(100)의 장치들과 서브시스템들의 하나 이상의 데이터베이스들은 전형적인 실시예들을 구현하기 위해 사용되는 정보를 저장할 수 있다. 그 데이터베이스들은 앞서 열거된 메모리들과 같은 하나 이상의 메모리들에 포함되어 있는, 레코드들, 테이블들, 어레이들, 필드들, 그래프들, 트리들, 리스트들 등과 같은 데이터 구조들을 이용하여 조직될 수 있다.
- [0174] [00103] DRM 시스템(100)의 전부 또는 일부는 예를 들면, 도 1-10에 관해서 기술된 것처럼, 개시된 전형적인 실시예들의 기술에 따라서 프로그램된, 하나 이상의 범용 컴퓨터 시스템들, 마이크로 프로세서들, 디지털 신호의 프로세서들, 마이크로 컨트롤러들 등을 사용하여 편리하게 구현될 수 있다. 개시된 전형적인 실시예들의 기술에 의거하여 평균적인 기술을 가진 프로그래머들에 의해 적절한 소프트웨어가 별문제 없이 준비될 수 있다. 나아가, DRM시스템(100)은 주문형 집적회로(application-specific integrated circuit)의 준비 혹은 부품회로

들의 적절한 네트워크의 상호연결에 의해 구현될 수 있다.

산업상 이용 가능성

- [0175] [00104] 유리하게, 여기서 설명된 전형적인 실시예들은, 오프라인 시스템들과 온라인 시스템들, 그리고 TV 애플리케이션들, 컴퓨터 애플리케이션들, DVD 애플리케이션들, VCR 애플리케이션들, 응용품 애플리케이션들, CD 플레이어 애플리케이션들 등과 같은 애플리케이션들 내에서 채용될 수 있다. 나아가, 전형적인 실시예들의 글리프, 심볼, 단어, 메시지, 프로파일, 템플릿 등을 보내기 위해 채용된 신호들은 사람의 가청 스펙트럼 안에서, 사람의 가청 스펙트럼 안에서, 사람의 가청 스펙트럼을 벗어나서, 사람의 가청 스펙트럼을 벗어나서, 이들의 조합에서, 전송되도록 구성될 수 있다.

- [0176] [00105] 전형적인 실시예들이 비록 음악 파일들, 오디오 파일들 등과 같은 아이템들에 관하여 설명되었지만, 그 전형적인 실시예들은 디지털 콘텐츠, 장치들 소프트웨어, 서비스들, 상품들, 자원들 등과 같은 어떤 적합한 아이템에도 적용될 수 있고, 그리고 기술 인터페이스, 언어, 문법, 콘텐츠, 권리, 신청하는 것, 서비스들, 스피드, 크기, 제한들, 장치들 등에서 변경을 수반하면서, 실시될 수 있다.

- [0177] [00106] 본 발명은 많은 실시예들 및 구현들과 관련하여 설명되어 있지만, 본 발명은 그것으로 한정되지는 않고 오히려 첨부된 청구항의 범위에 속하는 다양한 변경과 등가적인 구성을 커버한다.

- [0178] [00107] **첨부**

[0179] 샘플 확장 스키마 (대략/2001/11/compactMusic) :

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Copyright (C) 2001 ContentGuard Holdings, Inc. All rights reserved. "ContentGuard"
is a registered trademark and "XrML", "eXtensible rights Markup Language", the XrML logo,
and the ContentGuard logo are trademarks of ContentGuard Holdings, Inc. All other
trademarks are properties of their respective owners. -->
<!-- NOTE: This is an example extension schema. It is not part of XrML 2.0 Specification. --
>
<xsd:schema targetNamespace="http://www.xrml.org/schema/2001/11/compactMusic"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:sx="http://www.xrml.org/schema/2001/11/xrml2sx"
xmlns:r="http://www.xrml.org/schema/2001/11/xrml2core"
xmlns:cx="http://www.xrml.org/schema/2001/11/xrml2cx"
xmlns:cm="http://www.xrml.org/schema/2001/11/compactMusic"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xsd:import namespace="http://www.xrml.org/schema/2001/11/xrml2cx"
schemaLocation="xrml2cx.xsd"/>
  <!-- - Principals-->
  <xsd:element name="device" substitutionGroup="r:principal">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="r:Principal">
          <xsd:attribute name="cmid" type="xsd:int" use="optional"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
  <!-- - Resources-->
  <xsd:element name="song" substitutionGroup="r:resource">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="r:Resource">
          <xsd:attribute name="cmid" type="xsd:int" use="optional"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
  <!-- - Property Resources-->
  <xsd:element name="myDomain" substitutionGroup="r:resource">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="r:Resource"/>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>
  <!-- - Conditions-->
  <xsd:element name="copyManagement" type="sx:StatefulCondition"
substitutionGroup="r:condition"/>
  <!-- - Issuer Descriptions-->
  <xsd:element name="issuerDevice">
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="cm:IssuerRaw">
          <xsd:sequence minOccurs="0">

```

[0180]

```

        <xsd:element ref="cm:device"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="IssuerRaw">
  <xsd:complexContent>
    <xsd:restriction base="r:Issuer"/>
  </xsd:complexContent>
</xsd:complexType>
<!-- - Service Descriptions-->
<xsd:element name="copyKeeper" type="cm:StateOnDevice"
substitutionGroup="r:licensePart"/>
<xsd:element name="countKeeper" type="cm:StateOnDevice"
substitutionGroup="r:licensePart"/>
<xsd:element name="intervalKeeper" type="cm:StateOnDevice"
substitutionGroup="r:licensePart"/>
<xsd:complexType name="StateOnDevice">
  <xsd:complexContent>
    <xsd:extension base="r:LicensePart">
      <xsd:sequence minOccurs="0">
        <xsd:element ref="cm:device"/>
      </xsd:sequence>
      <xsd:attribute name="key" type="xsd:int" use="optional"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:schema>

```

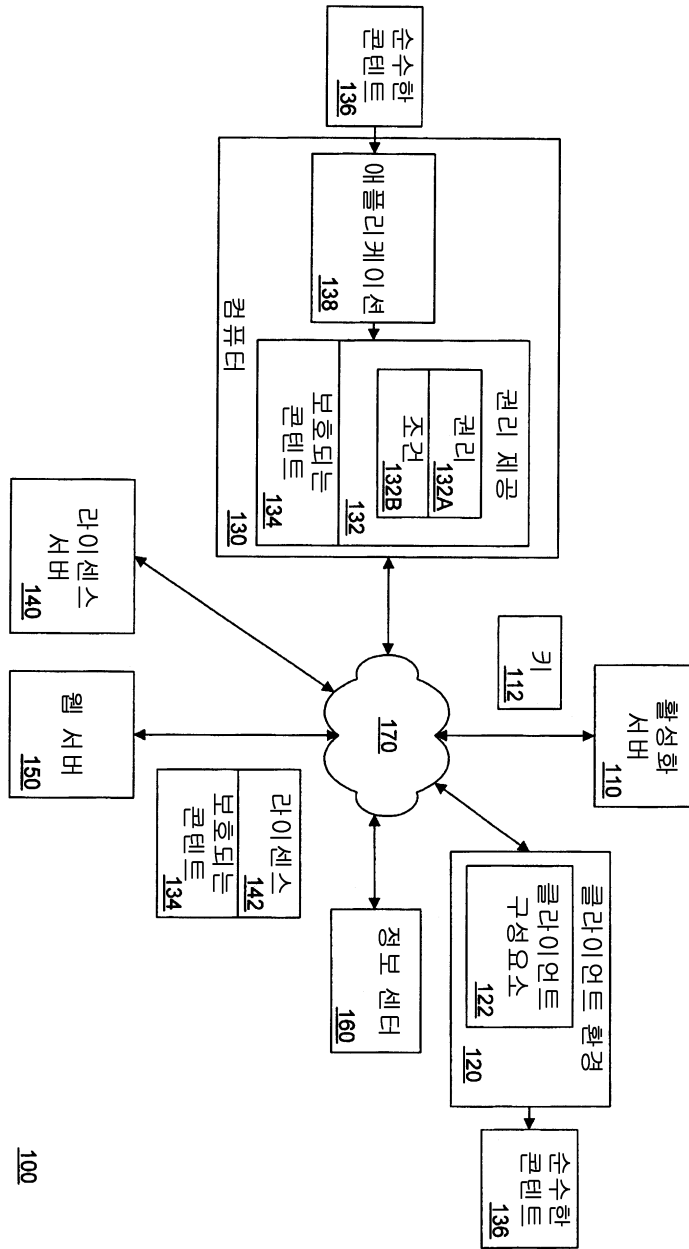
[0181]

도면의 간단한 설명

- [0010] 본 발명은 한정이 아닌 첨부도면의 도면에 있는 실예를 통해 예시적으로 도시되며, 같은 참조번호는 유사한 구성요소들을 가리킨다.
- [0011] 도 1은 본 발명의 여러 가지 실시예들이 실현될 수 있는 전형적인 DRM 시스템을 도시한다.
- [0012] 도 2는 전형적인 권리표현 프로파일링 처리를 도시하고 있는 흐름도이다.
- [0013] 도 3은 도 1의 DRM 시스템내의 자원-제한적인(resource-constrained) 장치들에 관한 전형적인 권리표현 엔코딩 및 디코딩 서브-시스템을 도시한다.
- [0014] 도 4는 도 3의 전형적인 권리표현 엔코딩 및 디코딩 프로세스를 도시하는 흐름도이다.
- [0015] 도 5는 도 1의 DRM 시스템내에 있는 전형적인 권리표현 템플릿 매핑 서브-시스템을 도시한다.
- [0016] 도 6은 도 5의 서브-시스템의 전형적인 템플릿 매핑 프로세스들을 도시하는 흐름도이다.
- [0017] 도 7은 라이선스 서명 값에 의거하여 도 3의 전형적인 권리표현 엔코딩 및 디코딩 서브-시스템을 더 도시한다.
- [0018] 도 8은 라이선스 변수들에 관한 서명값에 의거하여 도 3의 전형적인 권리표현 엔코딩 및 디코딩 서브-시스템을 더 도시한다.
- [0019] 도 9는 도 1의 DRM 시스템에서 자원-제한적인 장치들에 관한 전형적인 템플릿 공간을 도시한다.
- [0020] 도 10은 도 1의 DRM 시스템에 의해 생성될 수 있는 프로파일 또는 템플릿을 포함하는 권리표현의 전형적인 상징적인 대표자를 도시한다.

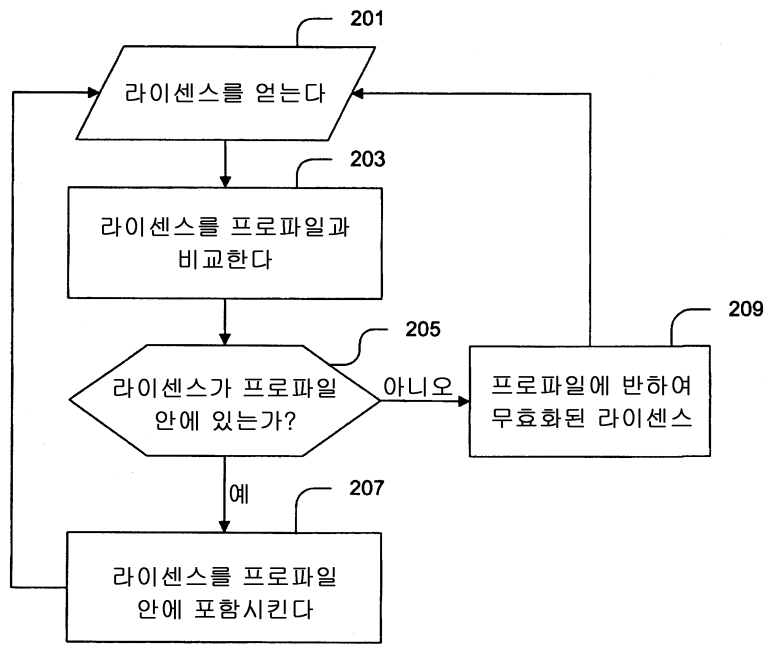
도면

도면1

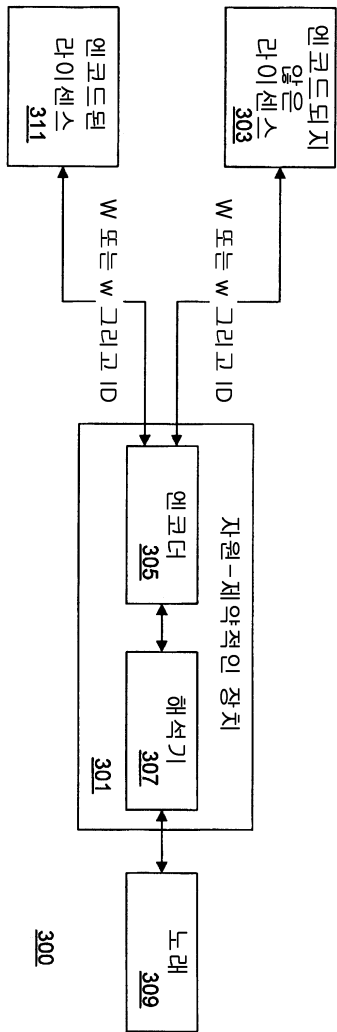


100

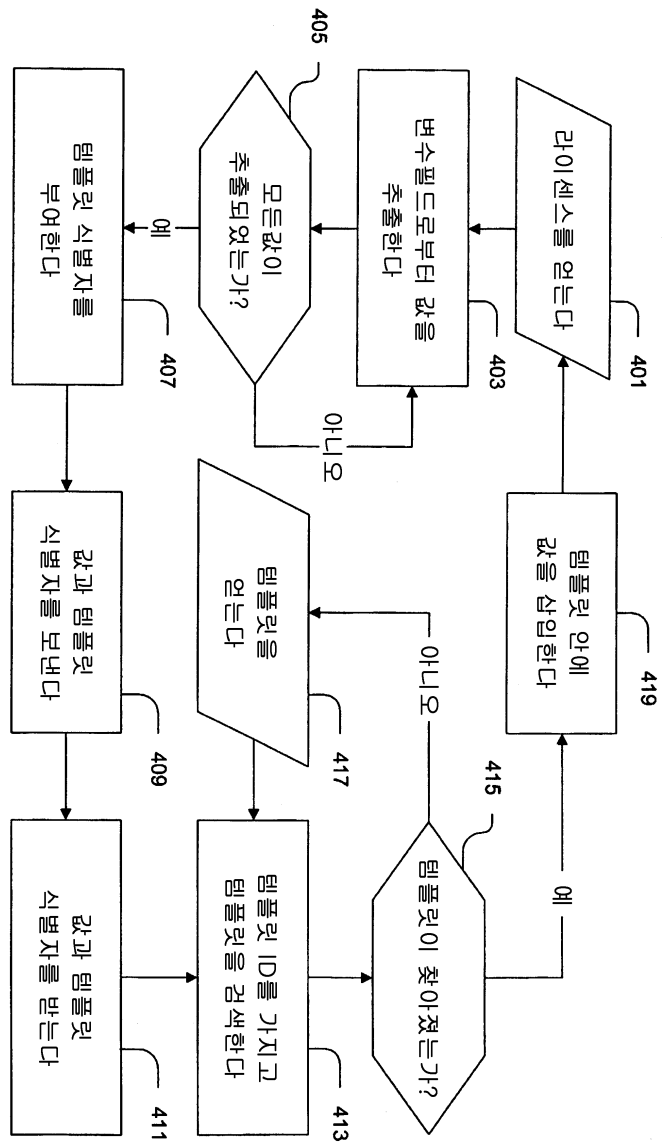
도면2



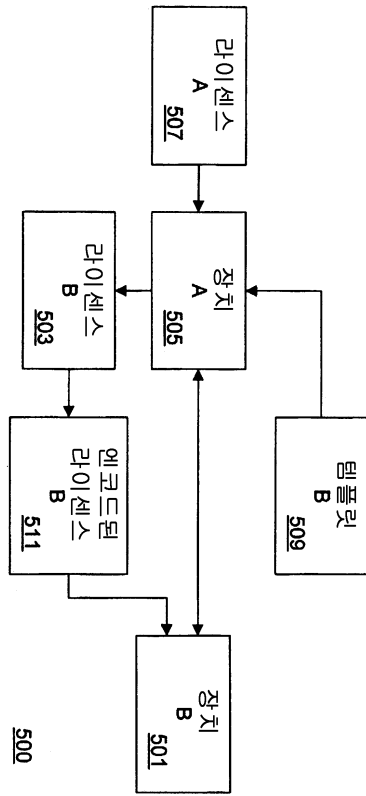
도면3



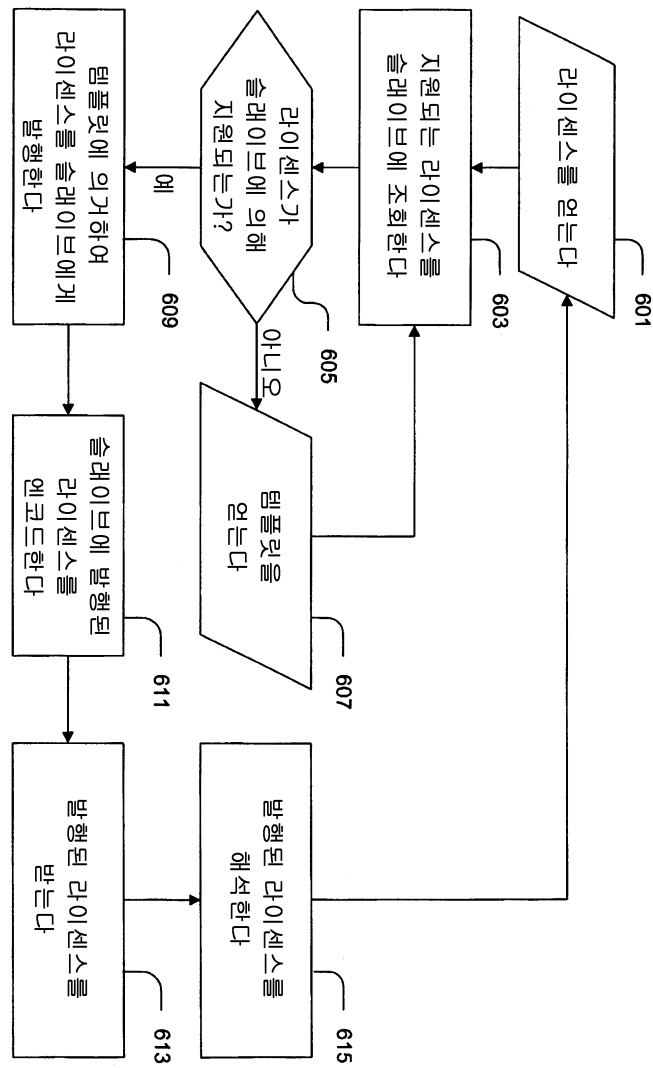
도면4



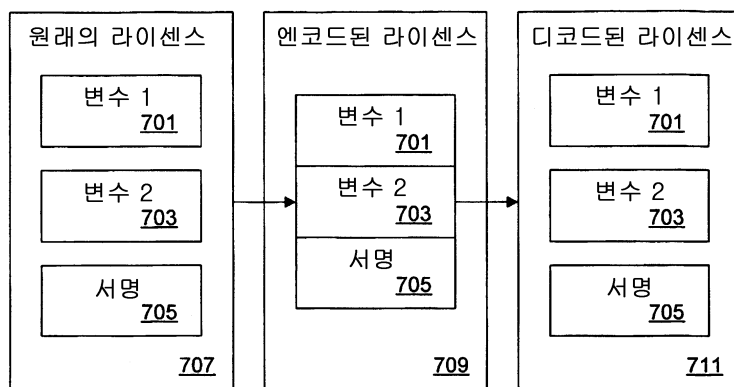
도면5



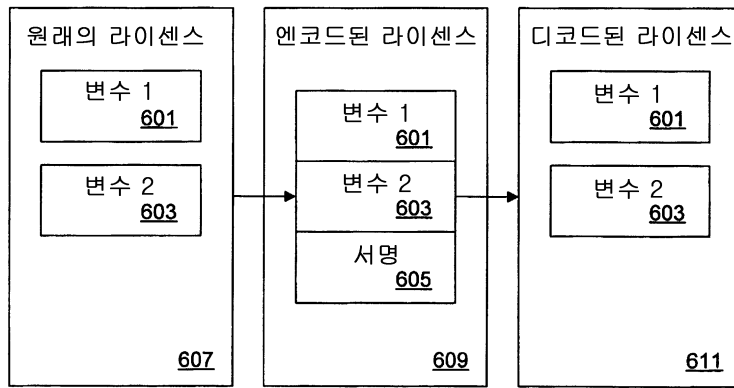
도면6



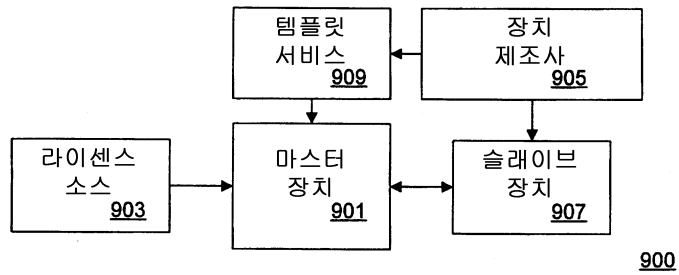
도면7



도면8



도면9



도면10

