



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2008-0105060  
(43) 공개일자 2008년12월03일

(51) Int. Cl.

G01C 21/00 (2006.01) G01C 21/26 (2006.01)

G01C 21/20 (2006.01)

(21) 출원번호 10-2008-7021697

(22) 출원일자 2008년09월04일

심사청구일자 없음

번역문제출일자 2008년09월04일

(86) 국제출원번호 PCT/EP2007/002078

국제출원일자 2007년03월08일

(87) 국제공개번호 WO 2007/101701

국제공개일자 2007년09월13일

(30) 우선권주장

0604704.7 2006년03월08일 영국(GB)

(뒷면에 계속)

(71) 출원인

톰톰 인터내셔널 비.브이.

네덜란드왕국 암스테르담 엔엘-1017 씨티 램브란트플레인 35번지

(72) 발명자

비스만스 레이비드

네덜란드 엔엘-1071 에이치아이 암스테르담 빌렘스 파크베그 12-3

살테르스 미치엘

네덜란드 멜프트 2624 보스룸 투쌩 플레인 147

테부트 제임스

네덜란드 1018 액스에스 암스테르담 코르테 암스 텔스트라트 12-에프

(74) 대리인

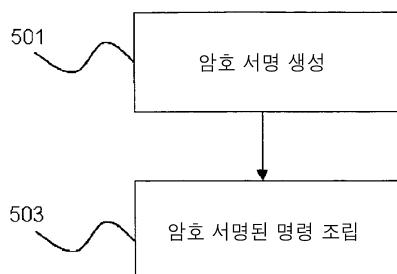
리엔목특허법인

전체 청구항 수 : 총 11 항

(54) 암호 서명된 명령을 제공하는 방법

**(57) 요약**

본 발명은 암호 서명된 명령을 제공하는 방법을 제공하며, 상기 명령은 하나의 명령 이름과 적어도 하나의 명령 인수를 포함하며, 상기 방법은 상기 명령 이름의 암호 서명을 생성하는 단계 (501) 및 상기 명령 이름, 상기 서명 그리고 상기 적어도 하나의 명령 인수를 사용하여 상기 암호 서명된 명령을 조립하는 단계 (503)를 포함한다.

**대 표 도** - 도5

(30) 우선권주장

0604706.2 2006년03월08일 영국(GB)  
0604708.8 2006년03월08일 영국(GB)  
0604709.6 2006년03월08일 영국(GB)  
0604710.4 2006년03월08일 영국(GB)

---

## 특허청구의 범위

### 청구항 1

암호 서명된 명령을 제공하는 방법으로서,  
 상기 명령은 하나의 명령 이름과 적어도 하나의 명령 인수를 포함하며,  
 상기 방법은,  
 상기 명령 이름의 암호 서명을 생성하는 단계 (501); 및  
 상기 명령 이름, 상기 서명 그리고 상기 적어도 하나의 명령 인수를 사용하여 상기 암호 서명된 명령을 조립하는 단계 (503);를 포함하는 방법.

### 청구항 2

제1항에 있어서,  
 상기 암호 서명된 명령을 조립 (503)하기 위해 상기 명령 이름, 상기 명령 이름의 암호 서명 그리고 상기 적어도 하나의 명령 인수를 연결하는 (concatenate) 단계를 포함하는 방법.

### 청구항 3

제1항 또는 제2항에 있어서,  
 비대칭 암호 알고리즘, 특히 RSA (Rivest-Shamir-Adleman) 알고리즘을 기반으로 하여 또는 디지털 서명 알고리즘을 기반으로 하여 상기 명령 이름의 암호 서명을 생성하는 단계 (503);를 포함하는 방법.

### 청구항 4

제1항 내지 제3항 중의 어느 한 항에 있어서,  
 상기 명령 이름은 상기 적어도 하나의 명령 인수에 의해 결정된 목적지로의 내비게이션 경로를 결정하는 것을 나타내거나 또는 상기 적어도 하나의 명령 인수에 의해 나타내지는 주소 데이터에 의해 결정된 내비게이션 지도를 제공하거나 또는 상기 적어도 하나의 명령 인수와 함께 컴퓨터 프로그램을 실행하는 것을 나타내는, 방법.

### 청구항 5

암호 서명된 명령을 기반으로 하여 암호 확인된 명령을 제공하는 방법으로서,  
 상기 암호 서명된 명령은 명령 이름, 상기 명령 이름의 암호 서명 및 적어도 하나의 명령 인수를 포함하며,  
 상기 방법은,  
 제1 암호값을 획득하기 위해 상기 암호 서명을 해독하는 단계;  
 제2 암호값을 획득하기 위해 상기 명령 이름을 처리하는 단계;  
 비교 결과를 얻기 위해 제1 암호값 및 제2 암호값을 비교하는 단계; 및  
 상기 명령 이름 및 상기 적어도 하나의 명령 인수를 상기 비교 결과에 따라 상기 암호 확인된 명령으로서 제공하거나 또는 실행하는 단계;를 포함하는 방법.

### 청구항 6

제5항에 있어서,  
 상기 제1 암호값은 상기 암호 서명의 해시 (hash) 값을 포함하며,  
 상기 제2 암호값은 상기 명령 이름의 해시값을 포함하는, 방법.

### 청구항 7

제5항 또는 제6항에 있어서,

상기 제1 암호값 및 제2 암호값이 동일하면 상기 암호 확인된 명령을 제공하거나 실행하는 단계를 포함하는 방법.

#### 청구항 8

암호 서명된 명령을 제공하는 기기로서,  
상기 명령은 명령 이름 및 적어도 하나의 명령 인수를 포함하며,  
상기 기기는  
상기 명령 이름의 암호 서명을 생성하는 생성기; 및  
상기 명령 이름, 상기 서명 및 상기 적어도 하나의 명령 인수를 사용하여 상기 암호 서명된 명령을 조립하는 조립기;를 포함하는 기기.

#### 청구항 9

암호 서명된 명령을 기반으로 하여 암호 확인된 명령을 제공하는 기기로서,  
상기 암호 서명된 명령은 명령 이름, 상기 명령 이름의 암호 서명 및 적어도 하나의 명령 인수를 포함하며,  
상기 기기는,  
제1 암호값을 획득하기 위해 상기 암호 서명을 해독하는 해독기;  
제2 암호값을 획득하기 위해 상기 명령 이름을 처리하는 프로세서;  
비교값을 얻기 위해 상기 제1 암호값과 제2 암호값을 비교하는 비교기; 및  
상기 명령 이름 및 상기 적어도 하나의 명령 인수를 상기 비교 결과에 따라 암호 확인된 명령으로서 제공하거나 또는 실행하는 제공기;를 포함하는, 기기.

#### 청구항 10

제1항 내지 제7항 중의 어느 한 항에 따른 방법을 실행하도록 구성된, 프로그램 가능하도록 구성된 기기.

#### 청구항 11

컴퓨터에서 작동될 때에 제1항 내지 제7항 중의 어느 한 항에 따른 방법을 실행하는 컴퓨터 프로그램.

### 명세서

#### 기술분야

<1> 본 발명은 특히 이동 내비게이션 분야의 정보 기술에 관련된 것이다.

#### 배경기술

<2> 이동 내비게이션 기기들에 대한 증가하는 요구는, 예를 들면, 내비게이션 콘텐트들 (경로 정보, 교통 정보, 지도 정보 등과 같은 내비게이션 항목들)을 관리하고, 예를 들면, 내비게이션 기기들에 사용자 특유의 내비게이션 콘텐트를 제공하며, 내비게이션 기기들을 동작시키고 또는, 예를 들면, 내비게이션 기기 상에 설치된 실행 가능한 프로그램들을 관리하는 것에 관련되어 여러 가지 제한들을 부과한다.

<3> 그러나, 내비게이션 기기들의 증가하는 복잡성 그리고 정보에 대한 증가되는 요구는 특정 컴퓨터 명령이 신뢰받는 엔티티에 의해 제공되었는가의 여부를 확인하는 것과 연관된 문제들에 영향을 끼친다.

<4> 본 발명의 목적은, 오버헤드를 감소시키면서, 암호 확인 가능한 명령들, 예를 들면, 컴퓨터 명령들을 제공하는 것이다.

<5> 이런 목적은 독립항들의 특징에 의해 달성된다.

<6> 본 발명은, (예를 들면, 실행 가능한) 컴퓨터 명령이 하나의 명령 이름과 하나 또는 그 이상의 명령 인수들 (예를 들면, 명령 인수들의 목록)을 포함하면, 예를 들면, 그 명령 이름에 자동적으로 서명하고 상기 (서명된) 명

령 이름과 그 암호 서명을 추가적인 인수로서 사용하여 확인 가능한 컴퓨터 명령을 조립할 때에, 줄어든 오버헤드를 가진 암호 확인 가능한 컴퓨터 명령이 획득될 수 있을 것이다.

### 발명의 상세한 설명

- <7> 본 발명의 일 실시예는 암호 서명된 명령을 제공하는 방법을 제공하며, 상기 명령은 하나의 명령 이름과 적어도 하나의 명령 인수를 포함한다. 상기 방법은 상기 명령 이름의 암호 서명을 생성하고 그 명령 이름, 상기 서명 그리고 적어도 하나의 명령 인수를 사용하여 상기 암호로 서명된 명령을 조립하는 것을 포함한다.
- <8> 추가의 실시예에 따르면, 상기 방법은 상기 암호로 서명된 명령을 조립하기 위해 상기 명령 이름, 상기 명령 이름의 암호 서명 그리고 적어도 하나의 명령 인수를 연결하는 것을 포함한다.
- <9> 추가적인 실시예에 따르면, 상기 방법은 비대칭 암호 알고리즘, 특히 RSA (Rivest-Shamir-Adleman) 알고리즘을 기반으로 하여 또는 디지털 서명 알고리즘을 기반으로 하여 상기 명령 이름의 암호 서명을 생성하는 것을 포함한다.
- <10> 추가적인 실시예에 따르면, 상기 명령 이름은 상기 적어도 하나의 명령 인수에 의해 결정된 목적지로의 내비게이션 경로를 결정하는 것을 나타내거나 그리고/또는 상기 적어도 하나의 명령 인수에 의해 나타내지는 주소 데이터에 의해 결정된 내비게이션 지도를 제공하는 것 또는 상기 적어도 하나의 명령 인수와 함께 컴퓨터 프로그램을 실행하는 것을 나타낸다.
- <11> 다른 실시예는 암호 서명된 명령을 기반으로 하여 암호 확인된 명령을 제공하는 방법을 제공하며, 상기 암호 서명된 명령은 명령 이름, 상기 명령 이름의 암호 서명 및 적어도 하나의 명령 인수를 포함한다. 상기 방법은 제1 암호값을 획득하기 위해 상기 암호 서명을 해독하며, 제2 암호값을 획득하기 위해 상기 명령 이름을 처리하며, 비교 결과를 얻기 위해 제1 암호값 및 제2 암호값을 비교하며, 그리고, 상기 명령 이름 및 상기 적어도 하나의 명령 인수를 상기 비교 결과에 따라 암호 확인된 명령으로서 제공하거나 또는 실행하는 것을 포함한다.
- <12> 추가의 실시예에 따르면, 상기 제1 암호값은 상기 암호 서명의 해시 (hash) 값을 포함하며 상기 제2 암호값은 상기 명령 이름의 해시값을 포함한다.
- <13> 추가의 실시예에 따르면, 상기 방법은 상기 제1 암호값 및 제2 암호값이 동일하면 상기 암호 확인된 명령을 제공하거나 실행하는 것을 포함한다. 예를 들면, 상기 방법은 상기 해시값들을 비교하는 것을 더 포함한다.
- <14> 다른 실시예는 암호 서명된 명령을 제공하는 기기를 제공하며, 상기 명령은 명령 이름 및 적어도 하나의 명령 인수를 포함한다. 상기 기기는 상기 명령 이름의 암호 서명을 생성하는 생성기와 상기 명령 이름, 상기 서명 및 상기 적어도 하나의 명령 인수를 사용하여 상기 암호 서명된 명령을 조립하는 (예를 들면, 연결시키는) 조립기를 포함한다.
- <15> 바람직하게는, 상기 기기 (또는 그 기기의 엘리먼트들)는 상기 암호 서명된 명령을 제공하기 위해 상기에서 설명된 방법의 단계들을 실행하도록 구성된다.
- <16> 다른 실시예는 암호 서명된 명령을 기반으로 하여 암호 확인된 명령을 제공하는 기기를 제공하며, 상기 암호 서명된 명령은 명령 이름, 상기 명령 이름의 암호 서명 및 적어도 하나의 명령 인수를 포함한다. 상기 기기는 제1 암호값을 획득하기 위해 상기 암호 서명을 해독하는 해독기, 제2 암호값을 획득하기 위해 상기 명령 이름을 처리하는 프로세서, 비교값을 얻기 위해 상기 제1 암호값과 제2 암호값을 비교하는 비교기 및 상기 명령 이름 및 상기 적어도 하나의 명령 인수를 상기 비교 결과에 따라 암호 확인된 명령으로서 제공하거나 또는 실행하는 제공기를 포함한다. 바람직하게는, 상기 기기 (또는 그 기기의 엘리먼트들)는 상기 암호 서명된 명령을 제공하기 위해 상기에서 설명된 방법의 단계들을 실행하도록 구성된다.
- <17> 본 발명의 한 모습에 따르면, 프로그램 가능하도록 구성된 엔티티가 추가의 프로그램 가능하도록 구성된 엔티티를 제어하기 위한 특정 명령을 존재하지 않는, 즉, 더미 (dummy) 네트워크 주소로 전송하거나 또는 전송하도록 요청하여 그 추가의 엔티티를 제어할 수 있다. 전송 또는 전송에 대한 요청은 그 추가의 프로그램 가능한 엔티티에게 특정 명령을 제공하는 추가적인 프로그램 가능하도록 구성된 (예를 들면, 제어) 엔티티에 의해 가로채어질 수 있을 것이다. 그러므로, 그 명령들은 추가적인 복잡성을 줄이도록 간접적으로 배포된다. 이런 점에서, "프로그램 가능하도록 구성된 엔티티"의 용어는 프로그램 가능하도록 구성된 수단 또는 컴퓨터 상에서 실행될 수 있는 컴퓨터 프로그램에 대한 것일 수 있을 것이다.
- <18> 일 실시예는 프로그램 가능하게 구성된 제1 수단, 프로그램 가능하게 구성된 제2 수단 및 프로그램 가능하게 구

성된 제3 수단을 구비한 통신 기기를 제공한다. 프로그램 가능하게 구성된 제1 수단은 프로그램 가능하게 구성된 제2수단에게 특정한 명령을 존재하지 않는 네트워크 주소, 예를 들면 더미 주소로 전송할 것을 요청하도록 구성될 수 있을 것이다. 프로그램 가능하게 구성된 제2 수단은 프로그램 가능하게 구성된 제3 수단에게 특정 명령을 전송할 것을 요청하는 것에 관하여 알리도록 구성될 수 있을 것이다. 그 요청을 나타내는 정보에 응답하여, 프로그램 가능하도록 구성된 제3 수단은 그 특정 명령을 실행할 수 있을 것이다. 그러므로, 프로그램 가능하도록 구성된 제1 수단은 프로그램 가능하도록 구성된 제1 수단이 프로그램 가능하도록 구성된 제3 수단과 직접적으로 상호 작용할 수 없을지라도 프로그램 가능하게 구성된 제3 수단을 간접적으로 제어할 수 있다.

- <19> 추가의 실시예에 따르면, 프로그램 가능하도록 구성된 제2 수단은 예를 들면 다른 프로그램에게 호스트나 어떤 환경을 제공하는 가상 머신 (Virtual Machine)을 제공하는 제2 컴퓨터 프로그램을 실행할 수 있을 것이다. 그러므로, 그 프로그램 가능하도록 구성된 제1 수단은 그 가상 머신 상에서 제1 컴퓨터 프로그램을 실행할 수 있을 것이다. 그러므로, 제1 컴퓨터 프로그램은 제2 컴퓨터 프로그램에 의해 제공된 가상 머신 상에서 실행되어, 제2 컴퓨터 프로그램이 제1 컴퓨터 프로그램을 감독하고 그리고, 예를 들면, 그 요청을 포획할 수 있을 것이다.
- <20> 추가의 실시예에 따르면, 제1 컴퓨터 프로그램은 실행가능하지 않은 스크립트 (script)를 포함할 수 있다. 더 나아가, 제2 컴퓨터 프로그램은, 자신이 실행되면, 네트워크 브라우징 기능, 예를 들면, 웹 브라우저를 제공할 수 있을 것이다. 추가로, 제3 컴퓨터 프로그램이 실행되면 그 제3 컴퓨터 프로그램은 사용자 애플리케이션, 예를 들면 사용자 환경 (예를 들면, 홈 애플리케이션)을 제공할 수 있을 것이다.
- <21> 다른 실시예는 제1 컴퓨터 프로그램, 제2 컴퓨터 프로그램 및 제3 컴퓨터 프로그램을 실행하도록 구성되어 있는 통신 기기를 제공하며, 제1 컴퓨터 프로그램은 제2 컴퓨터 프로그램에게 특정 명령을 존재하지 않는 네트워크 주소로 전송하도록 요청하고, 제2 컴퓨터 프로그램은 제3 컴퓨터 프로그램에게 특정 명령을 전송하라는 요청에 관하여 알리고, 제3 컴퓨터 프로그램은 그 요청을 나타내는 정보에 응답하여 그 특정 명령을 실행한다. 그러므로, 실행되면, 제1 컴퓨터 프로그램은, 예를 들면, 제1 컴퓨터 프로그램인 제3 컴퓨터 프로그램과 직접적으로 상호 작용할 수 있는 실행 가능한 루틴을 포함하고 있지 않더라도 제3 컴퓨터 프로그램을 간접적으로 제어할 수 있다.
- <22> 추가적인 실시예에 따르면, 제2 컴퓨터 프로그램이 실행되면, 그 제2 컴퓨터 프로그램은 가상 머신을 제공할 수 있을 것이며, 그때에 제1 컴퓨터 프로그램은 제2 컴퓨터 프로그램에 의해 제공된 가상 머신 상에서 동작할 수 있을 것이다. 그러므로, 예를 들면 컴퓨터 상에서 실행되면, 제1 컴퓨터 프로그램은 제1 컴퓨터 프로그램을 감독하고 그 요청을 포획하고 그리고 간접적으로 그 특정 명령을 포획하는 것을 가능하게 하는 환경을 그 가상 머신에 의해 제공한다.
- <23> 추가적인 실시예에 따르면, 제1 컴퓨터 프로그램은 실행가능하지 않은 스크립트를 제공할 수 있을 것이며, 제2 컴퓨터 프로그램은 네트워크 브라우징 기능, 예를 들면, 웹 브라우저를 제공할 수 있을 것이며, 그리고 제3 컴퓨터 프로그램은 사용자 애플리케이션, 예를 들면 사용자 환경을 제공할 수 있을 것이다.
- <24> 예를 들면, 그 특정 명령은, 예를 들면, 제3 컴퓨터 프로그램에 의해 통신 네트워크를 경유하여 원격 네트워크 엔티티로의 네트워크 접속을 설립한 것을 나타낸다.
- <25> 다른 실시예는, 존재하지 않는 네트워크 주소로 특정 명령을 제2 컴퓨터 프로그램에 의해 전송하도록 제1 컴퓨터 프로그램에게 요청하는 단계, 제3 컴퓨터 프로그램에게 상기 요청을 나타내는 정보를 제공하는 단계 및 상기 특정 명령을 전송하라는 상기 요청을 나타내는 정보에 응답하여 상기 제3 컴퓨터 프로그램에 의해 그 특정 명령을 실행하는 단계를 포함하는 통신 방법을 제공한다.
- <26> 다른 실시예는 제1 컴퓨터 프로그램, 제2 컴퓨터 프로그램 및 제3 컴퓨터 프로그램을 포함하는 컴퓨터 프로그램을 제공하며, 제1 컴퓨터 프로그램은, 컴퓨터 상에서 동작할 때에, 제2 컴퓨터 프로그램에게 특정 명령을 존재하지 않는 네트워크 주소로 전송하도록 요청하며, 제2 컴퓨터 프로그램은, 컴퓨터 상에서 동작할 때에, 제3 컴퓨터 프로그램에게 그 특정 명령을 전송하도록 하는 요청에 관하여 알리며, 제3 컴퓨터 프로그램은, 컴퓨터 상에서 동작할 때에, 그 요청에 관한 정보에 응답하여 그 특정 명령을 실행한다.
- <27> 본 발명의 다른 모습에 따르면, 다양한 프로그램들 또는 프로그램 가능하도록 구성된 엔티티들에 의해 제공되는 표준의 고유 자원 식별자 (uniform resource identifier, URI) 필드는 컴퓨터 프로그램 또는 프로그램 가능하도록 구성된 엔티티와 간접적으로 상호 작용하는 인터페이스를 형성할 수 있을 것이다. 그러므로, 이미 존재하는 자원들이 효과적으로 이용될 수 있다.
- <28> 일 실시예는 고유 자원 식별자를 나타내는 문자들의 스트링 내에서 특정 컴퓨터 명령을 검출하는 프로세싱 기기

를 제공하며, 그 특정 명령은 미리 정의된 명령 헤더를 포함하며, 복수의 미리 정의된 명령 이름들로부터의 한 명령 이름이 그 명령 헤드 다음에 이어진다. 상기 프로세싱 기기는 문자들의 스트링이 미리 정의된 명령 헤더를 포함하는가의 여부를 판별하는 판별기를 포함하며, 그 판별기는 문자들의 스트링이 미리 정의된 명령 헤더를 포함한다면 그 명령 헤더에 이어지는 문자들의 서브-스트링이 그 명령 이름을 포함하는 가의 여부도 또한 판별할 수 있을 것이다. 상기 프로세싱 기기는 그 명령 헤더가 명령 이름을 포함한다면 미리 정해진 명령 헤더 및 명령 이름을 특정 컴퓨터 명령으로서 제공하는 제공기를 더 포함한다. 미리 정의된 명령 헤더 및 명령 헤더에 이어지는 명령 이름을 포함하는 명령 구조는 고유 자원 식별자 내의 특정 명령을 낮은 복잡도로 식별하는 것을 가능하게 한다.

- <29> 추가의 실시예에 따르면, 상기 판별기는, 명령 헤더에 이어지는 문자들의 서브-스트링이 명령 이름을 포함하면, 명령 이름에 이어지는 문자들의 추가적인 서브-스트링이 적어도 하나의 명령 인수 (argument)를 포함하는 가의 여부를 판별할 것이다. 더 나아가, 상기 판별기는 그 적어도 하나의 명령 인수가 그 특정 명령의 미리 결정된 파라미터에 대응하는가의 여부를 판별할 수 있을 것이다. 상기 제공기는, 상기 적어도 하나의 명령 인수가 상기 미리 결정된 파라미터에 대응하면, 상기 미리 정의된 명령 헤더, 명령 이름 및 상기 적어도 하나의 명령 인수를 그 특정 컴퓨터 명령으로서 제공할 수 있을 것이다. 그러므로, 그 특정 컴퓨터 명령은 고유 자원 식별자 내의 검출된 명령 부분들을 기반으로 구성되며, 예를 들면 프로세서에 의해 실행될 수 있을 것이다.
- <30> 추가적인 일 실시예에 따르면, 상기 프로세싱 기기는, 예를 들면, 그 특정 명령을 실행하기 위해 프로그램 가능하도록 구성된 프로세서를 포함할 수 있을 것이다.
- <31> 다른 실시예는 고유 자원 식별자를 나타내는 문자들의 스트링 내의 특정 컴퓨터 명령을 검출하는 방법을 제공하며, 그 특정 명령은 미리 정의된 명령 헤더를 포함하며, 복수의 미리 정의된 명령 이름들로부터의 한 명령 이름이 그 명령 헤더에 이어진다. 상기 방법은 상기 문자들의 스트링이 미리 정의된 명령 헤더를 포함하는 가의 여부를 판별하고, 문자들의 스트링이 미리 정의된 명령 헤더를 포함한다면 상기 명령 헤더에 이어지는 문자들의 서브-스트링이 명령 이름을 포함하는가의 여부를 판별하고, 명령 헤더가 명령 이름을 포함하면 미리 정의된 명령 헤더와 명령 이름을 상기 특정 컴퓨터 명령으로서 제공하는 것을 포함한다.
- <32> 추가적인 실시예에 따르면, 하나 또는 그 이상의 명령 인수들은 상기 명령 이름에 이어지며 방법은 명령 헤더에 이어진 서브-스트링이 명령 이름을 포함하면 상기 명령 이름을 따르는 문자들의 추가적인 서브-스트링이 적어도 하나의 명령 인수를 포함하는 가의 여부를 판별하며, 명령 이름을 따르는 상기 문자들의 추가적인 서브-스트링이 적어도 하나의 명령 인수를 포함한다면 미리 정의된 명령 헤더, 명령 이름 및 그 적어도 하나의 명령 인수를 특정 컴퓨터 명령으로서 제공하는 것을 포함한다.
- <33> 추가적인 실시예에 따르면, 명령 헤더를 따르는 문자들의 서브-스트링이 명령 이름을 포함하면 상기 방법은 상기 명령 이름을 따르는 문자들의 추가적인 서브-스트링이 적어도 하나의 명령 인수를 포함하는지의 여부를 판별하고, 상기 적어도 하나의 명령 인수가 상기 특정 명령의 미리 결정된 파라미터에 대응하는지 여부를 판별하고, 상기 적어도 하나의 명령 인수가 상기 미리 결정된 파라미터에 대응하면 상기 미리 정의된 명령 헤더, 명령 이름 및 상기 적어도 하나의 명령 인수를 상기 특정 컴퓨터 명령으로서 제공하는 것을 포함한다.
- <34> 추가적인 실시예에 따르면, 상기 미리 결정된 파라미터는 상기 명령 이름과 추가적인 명령 인수를 분리하며, 상기 방법은 상기 미리 정의된 명령 헤더, 명령 이름 및 적어도 하나의 명령 인수 그리고 상기 추가적인 명령 인수를 상기 특정 컴퓨터 명령으로서 제공하는 단계를 더 포함한다.
- <35> 추가적인 실시예에 따르면, 상기 특정 명령은 사용자 특유 정보를 획득하기 위해 통신 네트워크를 경유하여 원격 네트워크 엔티티로의 네트워크 접속을 설립한 것을 나타낸다.
- <36> 추가적인 실시예에 따르면, 상기 방법은 상기 특정 컴퓨터 프로그램을 프로그램 가능하도록 실행하는 것을 포함한다.
- <37> 본 발명의 한 모습에 의하면, 내비게이션 기기의 동작은 그 내비게이션 기기 상에 장착된 기기 펌웨어를 다른 프로그램이 대리 실행 (emulate)하면 그 다른 프로그램을 사용하여 대리 실행될 수 있을 것이다. 그러므로, 그 다른 프로그램은, 예를 들면, 다른 기기 상에서의 내장되어 그 내비게이션 기기를 대리 실행하기 위해 실행되는 대리 실행기 (emulator)로서 장착될 수 있을 것이다. 그러나, 그 내비게이션 기기를 대리 실행하기 전에, 그 다른 프로그램이 기기 펌웨어에 대응한다는 것을, 예를 들면, 그 다른 프로그램의 소스 코드가 기기 펌웨어의 소스 코드에 대응한다는 것이 보장되어야 한다. 그러므로, 내비게이션 기기가 스위치 오프 되더라도 사용자는 그 대리 실행 프로그램을 이용하여 그 내비게이션 기기와 함께 동작시킬 수 있다.

- <38> 일 실시예는 내비게이션 기기 상에 장착된 기기 펌웨어 프로그램을 실행하는 것에 응답하여 그 내비게이션 기기의 동작을 대리 실행하기 위한 통신 기기를 제공한다. 바람직하게는, 그 통신 기기는 기기 펌웨어 프로그램에 관련된 정보를 제공하는 제공기, 상기 통신 기기 상에 장착된 현재의 펌웨어 프로그램이 상기 내비게이션 기기 상에 장착된 기기 펌웨어 프로그램에 대응하는지의 여부를 판별하는 판별기 및 현재의 펌웨어 프로그램이 기기 펌웨어와 대응하면 상기 내비게이션 기기의 동작을 대리 실행하기 위해 상기 통신 기기 상의 현재 펌웨어 프로그램을 실행하는 프로세서를 포함한다.
- <39> 추가적인 실시예에 따르면, 상기 프로세서는 현재의 펌웨어 프로그램이 기기 펌웨어 프로그램에 대응하지 않는다면 현재의 펌웨어 프로그램의 개신된 버전을 획득하기 위해 통신 네트워크를 경유하여 원격 네트워크 엔티티로의 네트워크 접속을 설립할 수 있을 것이다. 더 나아가, 상기 프로세서는 상기 내비게이션 기기의 동작을 대리 실행하기 위해 현재 펌웨어 프로그램의 개신된 버전을 실행할 수 있을 것이다.
- <40> 추가적인 실시예에 따르면, 상기 제공기는 기기 펌웨어 프로그램을 나타내는 인출 정보를 위해 상기 내비게이션 기기에 접속할 수 있을 것이다.
- <41> 추가적인 실시예에 의하면, 상기 판별자는 기기 펌웨어 프로그램을 나타내는 정보를 포함하는 콘텐츠 파일의 테이블을 기반으로 하여 또는 현재의 펌웨어 프로그램을 나타내는 정보를 포함하는 콘텐츠 파일의 테이블을 기반으로 하여 통신 기기 상에 설치된 현재의 펌웨어 프로그램이 기기 펌웨어에 대응하는지의 여부를 판별한다.
- <42> 추가적인 실시예에 따르면, 상기 프로세서는 상기 내비게이션 기기의 동작을 대리 실행하는 것을 기반으로 하여 상기 내비게이션 기기를 제어할 수 있을 것이다.
- <43> 다른 실시예는 상기 내비게이션 기기 상에 설치된 기기 펌웨어 프로그램을 실행하는 것에 응답하여 내비게이션 기기의 동작을 대리 실행하는 방법을 제공한다. 상기 방법은 상기 내비게이션 기기로부터 기기 펌웨어 프로그램에 관련된 정보를 제공하며, 현재의 펌웨어 프로그램이 상기 내비게이션 기기 상에 설치된 기기 펌웨어에 대응하는지의 여부를 판별하며, 현재 펌웨어 프로그램이 기기 펌웨어에 대응하면 상기 내비게이션 기기의 동작을 대리 실행하기 위해 통신 기기 상에서 현재의 펌웨어 프로그램을 실행하는 것을 포함한다.
- <44> 다른 실시예에 따르면, 상기 방법은 현재 펌웨어 프로그램이 기기 펌웨어와 대응하지 않으면 현재 펌웨어 프로그램의 개신된 버전을 획득하기 위해 통신 네트워크를 경유하여 원격 네트워크 엔티티로의 네트워크 접속을 설립하고, 상기 내비게이션 기기의 동작을 대리 실행하기 위해 상기 현재 펌웨어 프로그램의 개신된 버전을 실행하는 것을 포함한다.
- <45> 다른 실시예가 제공하는 것에 따르면, 상기 방법은 상기 기기 펌웨어 프로그램을 나타내는 정보를 인출하기 위해 상기 내비게이션 기기에 접속하는 것을 포함한다.
- <46> 다른 실시예에 따르면, 상기 방법은, 상기 기기 펌웨어 프로그램을 나타내는 정보를 포함하는 콘텐츠 파일의 테이블을 기반으로 하여 또는 현재 펌웨어 프로그램을 나타내는 정보를 포함하는 콘텐츠 파일의 테이블을 기반으로 하여, 현재의 펌웨어 프로그램이 상기 기기 펌웨어 프로그램에 대응하는지의 여부를 판별하는 것을 포함한다.
- <47> 다른 실시예는 내비게이션 기기 상에 설치된 기기 펌웨어 프로그램을 실행한 것에 응답하여 상기 내비게이션 기기의 동작을 대리 실행하는 프로그램 가능하도록 구성된 통신 기기를 제공하는 것이다. 그 프로그램 가능하도록 구성된 통신 기기는 기기 펌웨어 프로그램 관련된 정보를 제공하기 위해 제1 컴퓨터 프로그램을 실행하며, 상기 프로그램 가능하도록 구성된 통신 기기에 이용 가능한 현재의 펌웨어 프로그램이 상기 내비게이션 기기 상에 설치된 기기 펌웨어 프로그램에 대응하는지의 여부를 결정하기 위해 제2 컴퓨터 프로그램을 실행하며, 그리고, 현재의 펌웨어 프로그램이 상기 기기 펌웨어 프로그램에 대응하면 상기 내비게이션 기기의 동작을 대리 실행하기 위해 상기 통신 기기 상에서 현재의 펌웨어 프로그램을 실행할 수 있을 것이다.
- <48> 바람직하게는, 상기 프로그램 가능하도록 구성된 통신 기기는 상기에서 설명된 방법의 단계들을 실행하도록 구성된다.
- <49> 본 발명의 다른 모습에 따르면, 특정 정보는, 원격 서버에 제공된 그 특정 정보를 위한 요청에 대해 응답하여, 상기 특정 정보를 전송하는 것 대신에, 그 특정 정보를 획득하기 위해 상기 원격 서버에 다시 접속하는 순간의 시각을 나타내는 시각 정보가 제공되면, 시기 적절하게 제공될 수 있다. 그러므로, 쓸모없게 된 정보를, 예를 들면, 사용자에 의해 관리하거나 평가하는 것과 연관된 문제가 피해지게 된다.
- <50> 본 발명의 일 실시예는 특정한 정보를 추가의 네트워크 엔티티 (entity) (예를 들면, 네트워크 서버)로부터 네

트워크 엔티티 (예를 들면, 통신 기기)에게 제공하는 방법을 제공한다. 상기 방법은, 통신 네트워크를 경유하여 상기 네트워크 엔티티로부터 상기 추가의 네트워크 엔티티로 상기 특정 정보에 대한 요청을 전송하는 단계 및 상기 통신 네트워크를 경유하여 상기 추가의 네트워크 엔티티로부터 상기 네트워크 엔티티로 시각 정보를 전송하는 단계를 포함하며, 상기 시각 정보는 상기 특정 정보를 획득하기 위해 상기 추가의 네트워크 엔티티로 접속하는 순간의 시각을 나타낸다.

- <51> 추가적인 실시예에 따르면, 상기 방법은, 상기 순간의 시각에서 상기 네트워크 엔티티와 상기 추가의 네트워크 엔티티 사이의 네트워크 접속을 수립하는 단계 및 상기 특정 정보를 상기 추가의 네트워크 엔티티로부터 상기 네트워크 엔티티로 상기 통신 네트워크를 경유하여 전송하는 단계를 포함한다.
- <52> 추가적인 실시예에 따르면, 상기 방법은, 상기 순간의 시각에서 상기 네트워크 엔티티와 상기 추가의 네트워크 엔티티 사이의 접속을 수립하는 단계, 상기 특정 정보를 상기 추가의 제2 네트워크 엔티티로부터 상기 네트워크 엔티티로 상기 통신 네트워크를 경유하여 전송하는 단계 및 상기 네트워크 엔티티에 의해 상기 특정 정보를 디스플레이하는 단계를 포함한다.
- <53> 추가적인 실시예에 따르면, 상기 방법은 상기 특정 정보에 대한 요청을 상기 네트워크 엔티티에 의해 수신하는 단계를 더 포함한다.
- <54> 추가적인 실시예에 따르면, 상기 특정 정보는 사용자-특유의 내비게이션 경로에 관련된 교통 정보를 포함한다.
- <55> 추가적인 실시예에 따르면, 상기 방법은 상기 요청을 상기 추가의 네트워크 엔티티 내에 저장하는 단계를 포함한다.
- <56> 다른 실시예는 특정 정보를 통신 네트워크를 경유하여 추가의 네트워크 엔티티로부터 획득하는 네트워크 엔티티를 제공한다. 상기 네트워크 엔티티는 상기 통신 네트워크를 경유하여 상기 추가의 네트워크 엔티티로 상기 특정 정보에 대한 요청을 전송하고 상기 통신 네트워크를 경유하여 상기 추가의 네트워크 엔티티로부터 시각 정보를 수신할 수 있을 것이며, 상기 시각 정보는 상기 특정 정보를 획득하기 위해 상기 추가의 네트워크 엔티티로 접속하는 순간의 시각을 나타낸다.
- <57> 추가의 실시예에 따르면, 상기 네트워크 엔티티는 상기 특정 정보를 획득하기 위해 상기 순간의 시각에 상기 추가의 네트워크 엔티티에 접속할 수 있을 것이다.
- <58> 다른 실시예는 통신 네트워크를 경유하여 추가의 통신 엔티티에 의한 전송 가능한 특정 정보에 대한 요청에 응답하여 상기 통신 네트워크를 경유하여 상기 추가의 네트워크 엔티티에 상기 특정 정보를 제공하는 네트워크 엔티티를 제공하며, 상기 네트워크 엔티티는 상기 통신 네트워크를 경유하여 시각 정보를 전송하도록 구성되며, 상기 시각 정보는 상기 특정 정보를 획득하기 위해 상기 네트워크 엔티티로 접속하는 순간의 시각을 나타낸다.
- <59> 추가적인 실시예에 따르면, 상기 추가의 네트워크 엔티티가 상기 네트워크 엔티티로의 네트워크 접속을 설립하면 상기 네트워크 엔티티는 상기 특정 순간의 시각에 상기 추가의 네트워크 엔티티로 상기 특정 정보를 전송할 수 있을 것이다.
- <60> 본 발명의 한 모습에 따르면, (사용자-특유일 수 있는) 내비게이션 콘텐츠는 통신 네트워크를 경유하여 액세스 가능한 원격 서버 상에서 효과적으로 관리될 수 있다. 바람직하게는, 원격 서버는, 예를 들면 상기 내비게이션 기기 상에 현재 설치된 내비게이션 콘텐츠 또는 기기 자격 또는 다른 파라미터들에 따라서 내비게이션 콘텐츠를 제공한다. 그러므로, 개신된 내비게이션 콘텐츠는 원격 서버에서 맞춤으로 재단될 수 있으며, 상기 내비게이션 기기나 그 내비게이션 기기를 관리하는 컴퓨터 프로그램의 복잡성을 줄게 한다.
- <61> 일 실시예는 내비게이션 기기 내에 저장된 현재의 내비게이션 콘텐츠를 개신하기 위한 통신 기기를 제공한다. 그 통신 기기는 내비게이션 기기로부터 현재의 내비게이션 콘텐츠를 나타내는 정보를 인출하기 위한 인출 엘리먼트, 그 현재의 내비게이션 콘텐츠를 나타내는 정보를 통신 네트워크를 경유하여 원격 네트워크 기기로 전송하기 위한 송신기, 상기 원격 통신 기기로부터의 개신된 내비게이션 콘텐트를 수신하는 수신기, 그리고 상기 개신된 내비게이션 콘텐츠에 연관된 정보를 디스플레이하거나 상기 내비게이션 기기 상에 상기 개신된 내비게이션 콘텐츠를 설치하기 위한 프로세서를 포함한다.
- <62> 추가적인 실시예에 따르면, 상기 인출 엘리먼트는 상기 내비게이션 기기로부터 기기 식별을 나타내는 정보 또는 사용자 식별을 나타내는 정보를 인출할 수 있을 것이며, 이때에 상기 송신기는 사용자 식별의 기기를 원격 통신 기기로 전송할 수 있을 것이다.

- <63> 추가적인 실시예에 따르면, 상기 송신기는 현재 내비게이션 콘텐츠를 나타내는 정보를 메타 데이터로서 또는 콘텐츠 파일의 테이블로서 전송할 수 있을 것이다.
- <64> 추가적인 실시예에 따르면, 상기 인출 엘리먼트는 내비게이션 기기로부터 현재의 내비게이션 콘텐츠를 획득하기 위해 상기 내비게이션 기기로 접속할 수 있을 것이다. 예를 들면, 상기 인출 엘리먼트 또는 통신 기기는 상기 내비게이션 기기로 무선으로 접속할 수 있을 것이다. 다른 실시예에 따르면, 상기 내비게이션 기기 또는 통신 기기는 유선을 경유하여 상기 인출 엘리먼트로 접속 가능할 수 있을 것이며, 그래서 상기 인출 엘리먼트는 상기 내비게이션 기기로 유선 접속을 시작할 수 있을 것이다 (또는 그 반대로 동일).
- <65> 다른 실시예는 내비게이션 기기 내에 저장된 현재의 내비게이션 콘텐츠를 갱신하기 위한 방법을 제공한다. 바람직하게는, 상기 방법은 상기 내비게이션 기기로부터 현재의 내비게이션 콘텐츠를 나타내는 정보를 인출하고, 그 현재의 내비게이션 콘텐츠를 나타내는 상기 정보를 통신 네트워크를 경유하여 원격 네트워크 기기로 전송하며, 상기 원격 통신 기기로부터 갱신된 내비게이션 콘텐츠를 수신하며, 그리고, 그 갱신된 내비게이션 콘텐츠에 관련된 상기 정보를 디스플레이하거나 또는 상기 내비게이션 기기 상에 그 갱신된 내비게이션 콘텐츠를 설치하는 것을 포함한다.
- <66> 다른 실시예에 따르면, 상기 방법은 상기 내비게이션 기기로부터 기기 식별 또는 식별을 나타내는 정보를 인출하고 상기 원격 통신 기기로 사용자 식별의 상기 기기를 전송하는 것을 포함한다.
- <67> 다른 실시예에 따르면, 상기 방법은 상기 현재 내비게이션 콘텐츠를 나타내는 상기 정보를 메타 데이터 또는 콘텐츠 파일의 테이블로서 전송하는 것을 포함한다.
- <68> 다른 실시예에 따르면, 상기 방법은 특정한 갱신된 내비게이션 콘텐츠를 획득하기 위해 상기 내비게이션 기기와 연관된 사용자의 권리를 확인하거나 또는 내비게이션 기기의 권리를 확인하는 것을 포함한다.
- <69> 다른 실시예에 따르면, 상기 내비게이션 콘텐츠 또는 갱신된 내비게이션 콘텐츠는 경로 정보 또는 갱신된 펌웨어 또는 지도 정보를 나타내는 정보를 포함한다.
- <70> 다른 실시예에 따르면, 상기 갱신된 내비게이션 콘텐츠를 나타내는 정보는 어떤 내비게이션 콘텐트가 다른 내비게이션 콘텐트보다 더 높은 우선 순위를 가졌다는 것을 나타내는 우선권 정보를 포함한다.
- <71> 다른 실시예는 컴퓨터 상에서 동작할 때에 상기 특허성이 있는 방법들의 적어도 하나를 실행하는 컴퓨터 프로그램을 제공한다.

### 실 시 예

- <80> 도 1은 프로그램 가능하게 구성된 제1 수단 (101), 프로그램 가능하게 구성된 제1 수단 (101)에 연결된 프로그램 가능하게 구성된 제2 수단 (103) 및 프로그램 가능하게 구성된 제1 수단 (105)에 연결된 프로그램 가능하게 구성된 제3 수단 (105)을 포함하는 통신 기기의 블록도를 보여준다.
- <81> 프로그램 가능하게 구성된 제1 수단 (101)은 프로그램 가능하게 구성된 제2 수단에게 존재하지 않는 네트워크 주소, 즉, 존재하지 않는 HTTP 주소로 특정 명령을 전송하도록 요청한다. 그에 대한 응답으로, 프로그램 가능하게 구성된 제2 수단 (103)은 프로그램 가능하게 구성된 제3 수단 (105)에게 상기 특정 명령을 전송하라는 요청에 관하여 알린다. 상기 요청을 나타내는 정보에 대한 응답으로, 프로그램 가능하게 구성된 제3 수단 (105)은 그 요청을 나타내는 정보에 응답하여 상기 특정 명령을 실행한다.
- <82> 프로그램 가능하게 구성된 제1 수단 (101)은 프로그램 가능하게 구성된 제2 수단 (103)에 의한 실행 가능한 제2 컴퓨터 프로그램 (예를 들면, 웹 브라우저)에 의해 제공되는 환경 내에서 실행 가능하지 않은 스크립트, 예를 들면 자바 (Java) 스크립트를 포함하는 제1 프로그램을 실행할 수 있을 것이다.
- <83> 프로그램 가능하게 구성된 제3 수단 (105)은 제3 컴퓨터 프로그램, 예를 들면 사용자 애플리케이션 (홈 애플리케이션)을 바람직하게 실행하며, 제1 컴퓨터 프로그램이 그 내부에서 자신이 (상기 웹 브라우저 및 과도적으로는 상기 사용자 애플리케이션을) 동작시키는 환경에게 상기 명령을 자신을 대신해서 송신하도록 요청하기 때문에 상기 제3 컴퓨터 프로그램은 상기 명령을 수신한다. 자바 스크립트가 웹페이지 상에 존재하고 제3 컴퓨터 프로그램과는 직접적으로 통신할 수 없기 때문에 상기 웹 브라우저 (제2 컴퓨터 프로그램)가 제공된다.
- <84> 예를 들면, 상기 개념은, 예를 들면, 경로 정보 또는 경로 지도들과 같은 내비게이션 콘텐츠를 관리하기 위해 채택될 수 있을 것이다. 예를 들면, 제1 컴퓨터 프로그램은 예를 들면 (예를 들면, 액티브 페이지를 포함하는)

원격 서버와 (홈 클라이언트 애플리케이션을 제공하는) 제3 컴퓨터 프로그램 사이의 통신을 개시시킬 수 있을 것이다. 예를 들면, 상기 홈 애플리케이션 (또는 원격 서버)은 보이스 (voice), 지도 등을 판매하는 온라인 (웹) 상점을 포함한다. 이 웹 상점은 바람직하게는 통합된 웹 브라우저를 이용하여 구현된다. 내장된 브라우저는 상기 상점을 나타내는 웹 페이지들을 다운로드하라는 지시를 상기 홈 애플리케이션에게 받는다. 더 나아가, 상기 웹 상점 페이지들은 자바 스크립트를 포함할 수 있을 것이며, 그래서 다운로드된 페이지들은, 예를 들면 소위 AJAX 기술을 사용하여 원격 서버와 통신할 수 있다. 그러나, 어떤 포인트에서, 상기 페이지 상의 상기 자바 스크립트는 상기 홈 애플리케이션 클라이언트에게 어떤 명령들을 주어야 한다. 명령의 한 예는 사용자가 웹 상점에서 보이스를 구입한 후에 '그 보이스를 다운로드 하라'는 것이다. 상기 자바 스크립트가 신용 카드 처리를 구현할 수 있지만, 그 자바 스크립트는 실제로 다운로드 해서, 예를 들면, 내비게이션 기기에 설치할 수 없으며, 그래서 상기 자바 스크립트는 그 명령을 송신하거나 또는 그 명령을 송신하라고 요청한다.

<85> 상기 명령은 처음에는 정교하게 (예를 들면, 상기 상점 내의 웹페이지 상에서 자바스크립트 내에서) 특별한 요청을 하고, 상기 요청을 존재하지 않는 주소 (예를 들면, <http://ttds>)로 송신함으로써 홈 애플리케이션 클라이언트에게 송신된다. (예를 들면, 내장된) 웹 브라우저는 자신의 호스트인 홈 클라이언트 애플리케이션에게 그 요청에 대해 알린다. 상기 홈 클라이언트는 <http://ttds> 와 같은 그런 주소가 없다는 것을 알고 있으며, 그래서 그 홈 클라이언트는, 예를 들면 그 명령(들)을 추출하여 실행한다. 추가적인 프로세싱을 빠르게 하기 위해, <http://ttds> 로의 더미 요청이 취소될 것이다.

<86> 예를 들면, 지도들 및 보이스들과 같은 내비게이션 콘텐츠를 다운로드하는 것에 추가하여, (제3 컴퓨터 프로그램에 의해 제공되는) 홈 클라이언트는, 예로서, 더 많은 웹 페이지들 (예를 들면, 두 번째 상점 페이지)을 열거나 또는 그 페이지들을 닫으며, 팝업 메시지를 보여주며, 홈 애플리케이션 내부의 어떤 윈도우 (예를 들면, 보이스를 다운로드한 후에 유용한 기기로-설치 패널)로 스위치하고, 새로운 홈 애플리케이션 버전이 있는가 검사하고, 기기 펌웨어로의 갱신이 있는가 검사하며, 내비게이션 콘텐트로의 갱신이 있는가 검사하고, 내비게이션 기기를 대리 실행하는 대리 실행기에 대한 갱신이 있는가 검사하고, 또는 사용자 자격 증명을 요청하고 송신하도록 또한 지시받을 수 있다.

<87> 도 2는 URI (uniform resource identifier)를 나타내는 문자들의 스트링 내의 특정 명령을 검출하기 위한 프로세싱 기기의 블록도이다. 그 특정 명령은 복수의 미리 정의된 명령 이름들로부터의 명령 이름 그리고 선택적인 하나 또는 그 이상의 명령 인수들이 뒤따르는 미리 정의된 명령 헤더를 포함할 수 있을 것이다. 상기 프로세싱 기기는 상기 문자들의 스트링이 상기 미리 정의된 명령 헤더를 포함하는가의 여부를 판별하기 위한 판별기 (201)를 포함하며, 상기 판별기는, 상기 문자들의 스트링이 상기 미리 정의된 명령 헤더를 포함하면, 상기 명령 헤더에 이어지는 문자들의 서브-스트링이 상기 명령 이름을 포함하는지의 여부를 판별하도록 더 구성되며, 그리고 상기 프로세싱 기기는 상기 판별기 (201)에 연결된 제공기 (203)를 포함하며, 상기 제공기 (203)는, 상기 명령 헤더가 명령 이름을 상기 특정 컴퓨터 명령으로서 포함하면, 상기 미리 정의된 명령 헤더 및 명령 이름을 제공한다. 도 2에 도시된 개념은 예를 들면 (실행 가능한) 컴퓨터 프로그램들을 이용하여 또한 구현될 수 있을 것이다. 더 나아가, URI에 관련된 설명은 URL (uniform resource locator)에도 역시 적용된다.

<88> URI에 관해서, 동일한 URI가 웹페이지 상의 또는 이메일 내의 하이퍼링크의 "목적물"로서 또한 사용될 수 있다. 그런 경우에, URI는, 예를 들면 그러한 하이퍼링크에 클릭함으로써 활성화될 수 있을 것이다. 웹페이지 상에 내장된 자바 스크립트 프로그램으로부터의 URI를 이용하는 것 또한 가능하다. 그러나, 예를 들면, 컴퓨터 상에 국부적으로 설치된 다른 프로그램 또한 그런 URI를 사용할 수 있다. 예를 들면, 이메일 주소록 플러그인은 이메일 프로그램으로부터, 예를 들면 PC 상에 설치된 홈 애플리케이션 프로그램으로 주소를 송신하기 위해 이 URI를 사용할 수 있다.

<89> 예를 들면, (예를 들면, 내비게이션 콘텐츠를 관리하는) 홈 애플리케이션은 그 자신의 UI를 제공한다. 그러나, 추가의 콘텐츠는 상기 홈 애플리케이션의 외부에 저장된 내비게이션 목적을 위해 사용될 수 있을 것이다. 예를 들면, 이메일 프로그램은, 월드 와이드 웹과 같은 복수의 주소들을 포함하는 주소록을 구비한다. 홈 클라이언트가 이런 것들을 이용할 수 있다면 유용할 것이다. 다른 측들은, 예를 들면, 그런 액세스를 허용하는 관습적인 <header>:/// URI 방식을 구현할 수 있을 것인, 홈 애플리케이션에 의한 설치를 허용하는 형식으로 월드 와이드 웹 상에 내비게이션 콘텐츠를 제공하는 것에 흥미가 있을 수 있다.

<90> RFC 3986에 의해 정의된 상기 URI 방식은 내부에서 다중 프로토콜들이 구현될 수 있는 프레임워크를 정의한다. 현재의 운영 시스템들은 주어진 프로토콜을 가진 지명된 처리기 (handler)로서 스스로를 등록시키는 애플리케이션들을 허용한다. 예를 들면, 상기에서 설명된 컴퓨터 프로그램에 의해 제공된 홈 애플리케이션 (홈

클라이언트)은 스스로를 <header>:/// 로 시작하는 URI들을 위한 처리기로서 등록한다.

<91> URI의 나머지는 홈 클라이언트에서 명령 목록으로서 사용된다. 이런 접근 방법을 사용하여, "Amsterdam, Rembrandtplein 35"과 같은 주소는, 예를 들면 <header>://Address?Amsterdam&Rembrandtplein&35 로서 인코드될 수 있다. 이런 점에서, "?" 그리고 "&"의 문자들은 선택적인 것이며 다른 미리 정의된 문자들 또는 문자들의 스트링들로 교체될 수 있다.

<92> URI를 지원하는 홈 클라이언트가 아닌 어떤 애플리케이션도, 이런 URI와 마주치게 되면, 운영 시스템 (operating system, OS)에게 그 URI를 해결해달라고 요청할 것이다. 차례로 OS는 홈 클라이언트에게 상기 사용자가 "Address?Amsterdam&Rembrandtplein&35"를 사용할 의도라는 것을 알릴 것이다. 그러면 홈 클라이언트는 그 주소를 가지고 어떤 것을 할 것인가의 선택 목록을 사용자에게 제공한다. 그러나, 주소가 사용될 수 있는 복수의 내비게이션 작업들이 있다.

<93> 동일한 접근 방법을 사용하고, "http://shop.<web address>%voices/dutch/bram.toc"를 네덜란드 보이스로 가정하면, 다음의 URI를 생성하는 것이 가능하다:

<94> <header>://install?http%3a%2f%2fshop.<web

<95> address%2fvoices%2fdutch%2fbram.toc

<96> 그런 URI가 웹페이지에 나타나지 않는다면, 그리고 사용자는 그 위에 클릭한다면, 웹 브라우저는 OS에게 그것을 포워드하라고 요청할 수 있을 것이다. OS는 홈 클라이언트에게, 이 경우에는 주어진 주소로부터 bram.toc (table of contents; 콘텐츠 테이블)을 다운로드하는 것인,

<97> install?http%3a%2f%2fshop.<web address>%2fvoices%2fdutch%2fbram.toc

<98>에 의해 함축된 동작을 취하라고 요청할 것이다.

<99> 다음에, 명령들의 형식 규격이 설명될 것이다.

<100> (RFC 3986-준수하는) 일반적인 URL 형식을 언급하면, 명령은 다음의 구조를 가질 수 있을 것이다:

<101> <header>://command-list

<102> 이 때에,

<103> 명령-목록 (command-list) = command [ '#' command ]+

<104> 명령 (command) = command-name '?' argument-list

<105> 명령-이름 (command-name) = (지원되는 명령의 이름, 아래 참조)

<106> 인수-목록 (argument-list) = UrlEncode(argument) [ '&' argument-list ]+

<107> 인수(argument) . = UTF8-string

<108> 레거시 (legacy) URL 형식을 참조하면, 명령은 다음의 구조를 가져야 한다:

<109> <header>://legacy-command-list

<110> 이 때에 :

<111> legacy-command-list = legacy-command [ '&' legacy-command ]+

<112> legacy-command = command-name '(' legacy-argument-list ')'

<113> legacy-argument-list = UrlEncode(argument) [ ',' legacy-argument-list ]+

<114> 올바른 URL들이 항상 "?" 문자를 포함하는 것에 반하여 레거시 URL들은 "?" 문자를 포함할 수 없다. 이는 그것들을 구분하는 것을 가능하게 한다. 이런 URL들은 RFC를 준수하지 않는다는 것에 유의한다. URL-처리하는 많은 엔티티들은 처리 도중에 그것들을 변경한다. 특히, 많은 윈도우 웹 브라우저들은 URL 일치를 만들기 위한 시도로 추가의 "/"을 더할 것이다. 일부 웹 브라우저들은 그것들을 전혀 지원하지 않는다. 홈 애플리케이션은 잘못-형성된 URL들을 변역하려고 시도하여, 다양한 종류의 URL-가능하게 된 애플리케이션에 대처하려고 하지만, 레거시 URL 형식을 사용하는 것은 그런 접근 방법을 더욱 복잡하게 만든다.

- <115> 명령들은 다음의 구조와 인수들을 가질 수 있을 것이다:
- <116> 구조 : EMailContact?folder-ID&entry-ID
- <117> 인수들 : 이메일 접속 ID들 (EMail contact IDs).
- <118> 홈 애플리케이션은 이메일 프로그램으로부터 접속 데이터를 인출하여 사용자에게 그것을 가지고 무엇을 할 것인가 (지도상에 표시, 그곳으로 내비게이트, 즐겨찾기로서 추가 등)를 문의할 것이다.
- <119> 구조 : Address?city&street&number
- <120> 인수들: 주소 데이터 (address data).
- <121> 홈 애플리케이션은 사용자에게 주소를 가지고 무엇을 할 것인가 (지도상에 표시, 그곳으로 내비게이트, 즐겨찾기로서 추가 등)를 문의할 것이다.
- <122> 구조: ShowOnMap?city&street&number
- <123> 인수들: 주소 데이터.
- <124> 적어도 하나의 도시 또는 거리는 비어있지 않아야 한다.
- <125> 구조: NavigateTo?city&street&number
- <126> 인수들: 주소 데이터.
- <127> 적어도 하나의 도시 또는 거리는 비어있지 않아야 한다.
- <128> 구조: AddFavourite?favourite-name&city&street&number
- <129> 인수들: 즐겨찾기의 이름과 주소 데이터 (name of the favourite and address data).
- <130> 적어도 하나의 도시나 거리는 비어있지 않아야 한다.
- <131> 구조: Install?URL
- <132> 인수: 다운로드될 아이템에 대한 TOC (table of contents)의 HTTP URL, 그리고 기기가 연결되면 설치
- <133> 상기 URL에 의해 표시된 리소스는 MIME-유형 "text/xml"을 구비한다.
- <134> TOC (table of contents) 형식은 "TOC format.xsd" 파일 내에서, 예를 들면, XML 방식에 의해 특정될 수 있을 것이다.
- <135> 다음 유형의 아이템들은 사용자 (예를 들면, 서드 파티들)에 의해 설치될 수 있다 : POI (points of interest) 데이터 세트들, 보이스, 색상 방식들.
- <136> 구조: Execute?URL
- <137> 인수들: 실제 명령-목록을 포함하는 텍스트파일의 HTTP URL
- <138> 상기 URL에 의해 표시된 리소스는 MIME-유형 "text/plain"을 구비한다.
- <139> 예를 들어 상기 명령들을 URL의 일부로서 통과시키는 것이 너무 긴 URL을 생기게 하면 상기 명령이 사용될 수 있다.
- <140> Execute( ) 명령-이름이 래거시-명령 내에서 사용되면, URL은 그에 대한 응답으로 래거시-명령-목록을 제공할 수 있을 것이다.
- <141> 구조: SwitchTab?Tabname
- <142> 인수들: 서버-생성된 탭 (tab)의 이름 (name of a server-created tab).
- <143> 홈 애플리케이션은 이 탭이 존재한다면 이 탭으로 전환할 것이다. 이 탭이 존재하지 않는다면, 홈 애플리케이션은 그 이름을 기억할 것이며 일단 서버가 그 탭을 생성하면 그 탭으로 전환할 것이다. 홈 애플리케이션은 단지 하나의 이름만을 기억할 것이며; 두 번째 SwitchTab 명령이 첫 번째를 대체할 것이다.
- <144> 구조: <user or host name>Page?Tabname&hostname&URL-path&anchor

- <145> 인수들: 새롭게 생성된 템의 템이름 (tabname), 호스트이름 (hostname), 호스트이름에 대한 (경로) 및 URL 앵커 (anchor).
- <146> 홈 애플리케이션은 주어진 템이름을 가진 템이 존재하는가를 먼저 검사하고, 존재하지 않으면 하나를 생성한다. 그러면 이 템은 자신의 URL을
- <147> `http://hostname.<web address>/URL-path#anchor` 로 설정하도록 한다.
- <148> 홈 애플리케이션은 이름 주입을 방해하는 비-알파뉴메릭 문자들을 가지는 호스트이름들을 거절할 것이다. 예를 들면, 호스트 이름 "badguys.com/ignore" 은 거절될 것이다. 왜냐하면 "." 그리고 "/"은 올바르지 않기 때문이다.
- <149> 이하에서 다루어질 것과 같이, 명령들은 그 명령을 확인하는 것을 허용하는 암호 서명과 함께 제공될 수 있을 것이다. 예를 들면, 다음의 명령 구조들이 사용될 수 있다:
- <150> 구조: <PublicKeyIdentifier>?command-list&signature
- <151> 인수들: 명령-목록 및 서명
- <152> 이 명령은 사용자에 의해서만 실행될 수 있는 명령들의 목록들을 실행하기 위해 사용된다. 상기 서명은 명령-목록의 확실성을 확인한다.
- <153> 서명 (signature) = `Sign<PrivateKey>(명령-목록)`
- <154> 누구도 (공중 키 (public key)를 가지고) 서명을 확인할 수 있다는 것에 유의한다.
- <155> `Sign<PrivateKey>` (명령-목록)은 다음과 같이 구현될 수 있을 것이다:
- <156> 명령 목록은 UTF-8 스트링이며 그래서 바이트-순서로서 표현될 수 있다. 명령 목록은 `Sign<PrivateKey>` 의 입력으로서 사용될 때에는 URL-인코드되지 않을 것이다.
- <157> 인코딩 방식을 참조하면, 이 바이트-순서의 SHA-26 해시 H가 계산된다 (256 비트). 더 나아가, 비밀 키 (private key) ( $Q, M$ )를 사용하여, 서명 값 ( $Q^H \bmod M$ )이 계산된다. 상기 서명은 바이트들의 베이스-64 (Base-64) 인코드된 리틀 인디안 순서 (little-endian sequence)로서 저장된다 (URL을 짧게 유지시키기 위해 채워넣는 것이 필요하지도 않고 추가되지도 않는다). 모든 인수들과 같이, 명령-목록 및 서명 인수들은 URL-인코드될 것이다. <PublicKeyIdentifier>() 명령-이름이 래거시-명령에 사용될 때에, 첫 번째 인수는 바람직하게 래거시-명령-목록이어야 한다.
- <158> 해독은 역의 프로세스이다: 홈 애플리케이션은 공용 (public) 키 ( $P, M$ )을 유지하다. 명령-목록의 SHA-256 해시 H가 계산된다 (256 비트). 공용 키 ( $P, M$ )을 사용하여, 대응 서명 (countersignature) ( $P^H \bmod M$ )이 계산된다. 서명은 ( $Q^H \bmod M$ )을 얻기 위해 기반-64 디코드되며, 이 때에  $(P^H \cdot Q^H) \bmod M$  이 계산된다. 상기 서명은 그 서명이 상기 대응 서명과 매치하면 받아들여지며, 이는  $(P^H \cdot Q^H) \bmod M$  이 1과 같다는 것을 의미한다.
- <159> 상기의 개념은 명령 인수들이 존재하지 않으면 또한 적용할 수 있다. 더 나아가, 상기 명령 이름은 인수들에 의미론 (semantics)을 부여하는 수많은 파라미터들을 포함할 수 있을 것이다. 예를 들면, 모든 인수는, 프로그래밍 언어에서 함수 호출 (function call) 구문과 유사한, 이하에서 위치상의 접근 (positional approach)과 명명된 접근 (named approach)으로서 언급되는 두 가지 가능한 방법에서의 파라미터와 매치될 수 있을 것이다.
- <160> 위치상의 구문을 참조하면, 명령 이름이, 예를 들면, 세 개의 파라미터들을 구비하면, 세 개의 인수들이 제공되며, 그러면, 첫 번째 파라미터는 첫 번째 인수와 매치되며, 이하 마찬가지이다. 예를 들어, 주소 명령은 세 가지 파라미터들을 구비한다: 도시, 거리 및 집번호(HouseNumber). 그러면 실제의 URI는 세 인수들을 포함할 것이며, 그 때에 첫 번째 인수는 도시 이름으로 번역되며, 이하 마찬가지이다.
- <161> 대안으로, 다음의 구문이 지원될 수 있을 것이다:
- <162> `Address?Street=RembrandtPlein&City=Amsterdam.`
- <163> 이런 경우에, 모든 인수는 파라미터 이름을 포함한다. 그러므로, 이들은 "명명된 (named)" 인수들로서

알려진다.

<164> 더 나아가, 상기에 언급된 파라미터들은 필수적인 파라미터들인 것으로서 지정될 수 있다. 이는 매치하는 인수가 있어야 한다는 것을 의미한다. 예를 들면, "?" 대신에 다른 파라미터들이 명령 이름과 명령 인수들 사이의 분리를 나타내기 위해 사용될 수 있을 것이다. 더 나아가, 고정된 길이의 명령 이름을 요구하는 것과 같이 (이는 그 위치 이후의 첫 번째 글자가 첫 번째 명령 인수를 시작하게 한다는 것을 의미한다) 다른 메카니즘이 채택될 수 있을 것이다.

<165> 일반적인 <header>:/// URI 는 하나 또는 그 이상의 명령들을 포함할 수 있을 것이다. 그러므로, 문자 "#" (또는 어떤 다른 문자, 예를 들면 "and")가 명령들을 연결시키기 위해 사용될 수 있다.

<166> 도 3은 내비게이션 기기 상에 설치된 기기 펌웨어 프로그램을 실행하는 것에 응답하여 상기 내비게이션 기기의 동작을 대리 실행하기 위한 통신 기기의 블록도를 보여준다. 상기 통신 기기는 기기 펌웨어 프로그램에 관련된 정보를 제공하는 제공기 (301), 상기 제공기 (301)에 연결되어 상기 통신 기기 상에 설치된 현재 펌웨어 프로그램이 상기 내비게이션 기기 상에 설치된 기기 펌웨어 프로그램과 대응하는지의 여부를 판별하는 판별기 (303) 및 상기 판별기 (303)에 연결되어, 현재 펌웨어 프로그램이 기기 펌웨어에 대응하면, 상기 내비게이션 기기의 동작을 대리 실행하기 위해 통신 기기 상의 현재의 펌웨어 프로그램을 실행하는 프로세서 (305)를 포함한다.

<167> 상기 통신 기기는 프로그램 가능하게 구성되고 상기 대리 실행 프로세스를 제어할 수 있는 상기에서 언급된 홈 애플리케이션을 실행하도록 구성될 수 있을 것이다.

<168> 대리 실행의 개념은 상기 내비게이션 기기의 휴대성과 가용성을 또한 지원한다. 보통, (휴대용) 내비게이션 기기는, 예를 들면, 그 기기의 입력부로서 또한 동작하는 작은 스크린을 구비한다. 대조적으로, 예를 들면, (데스크탑) 컴퓨터 (예를 들면, PC)는 아주 더 큰 스크린과 더 진보된 입력 방법들을 구비한다. 그러므로, 내비게이션 기기에 연결될 수 있는 컴퓨터에 의해 그 내비게이션 기기가 제어되면, 그 내비게이션 기기 상에서 실행될 수 있는 많은 작업들은 더 쉽게 행해진다. 게다가, 컴퓨터의 프로세서 (예를 들면, CPU)는 더 빠르다.

<169> 더 나아가, 대리 실행 개념은 프로그램 가능하게 구현될 수 있을 것이며, 그리고, 예를 들면, (상기 통신 기기의 일 실시예를 대표하는) 컴퓨터가 더 큰 용량의 컴퓨터 자원들을 이용하여 내비게이션 기기를 제어하는 것을 효과적으로 가능하게 하기 위해 상기 홈 애플리케이션으로 내장될 수도 있을 것이다.

<170> 상기 내비게이션 기기가 홈 클라이언트에 (또는 상기 통신 기기에) 연결될 때에, 상기 내비게이션 기기 상에 설치된 펌웨어 버전이 판별된다. 홈 클라이언트가 펌웨어의 버전과 동등한 컴퓨터 버전을 구비한다면, 그 펌웨어는 시작된다. 동등한 버전이 이용가능하지 않으나, 네트워크 접속 (예를 들면, 인터넷 접속)이 이용 가능하다면, 상기 홈 클라이언트는 홈 서버에게 동등한 버전이 어떤 것이며, 그리고 그것이 다운로드 가능한가를 문의할 것이다.

<171> 도 4는 네트워크 엔티티에 추가의 네트워크 엔티티로부터 특정 정보를 제공하는 방법을 보여준다. 상기 방법은 상기 네트워크 엔티티로부터 통신 네트워크를 경유하여 상기 추가의 네트워크 엔티티로 특정 정보를 찾는 요청을 전송하고 (401), 상기 추가의 네트워크 엔티티로부터 통신 네트워크를 경유하여 상기 네트워크 엔티티로 시각 정보를 전송하는 (403) 것을 포함하며, 상기 시각 정보는 상기 특정 정보를 획득하기 위해 상기 추가의 네트워크 엔티티에 접속하는 어떤 순간의 시각을 나타낸다. 상기 방법은, 예를 들면, 실행되면 상기 방법의 단계들을 실행하는 소프트웨어 내에서 (예를 들면, 홈 애플리케이션으로서) 구현될 수 있을 것이다.

<172> 예를 들면, 사용자는 홈 애플리케이션에게 어떤 도로 상의 (특정한 그리고 사용자-특유의 정보인) 매일의 교통 보고를 요청할 수 있다. 그러면, 그 홈 애플리케이션은, 예를 들면 매일 요청된 시각에 그 요청된 도로 상에서의 교통 혼잡과 사고들에 관한 보고를 보여줄 것이다. 그 교통 보고 요청이 그 홈 클라이언트 내에서 만들어질 때에, 그 클라이언트는, 예를 들면, 즉각적으로 그 요청을 서버로 포워드한다. 그 요청 데이터는 주어진 사용자를 위해 서버 내에 저장된다. 그 요청에 대한 응답으로, 서버는 홈 클라이언트가 그 홈 서버에 폴링(poll)해야 하는 시각을 회답하여 송신한다. 또한, 동일한 사용자가 다시 로그할 때마다, 홈 서버는 홈 클라이언트에게 서버에 폴링해야 하는 시각을 알릴 것이다. 실제의 교통 보고는 그 폴링 보고에 대한 반응으로서 서버에 의해 송신된다. 홈 애플리케이션은, 예를 들면, 그 교통 보고를 디스플레이 할 목적으로 형식화한다. 결과는 모든 데이터가 서버 측에서 저장된다는 것이며, 그러나 그럼에도 불구하고 상기 접속은 클라이언트에 의해 개시된다. 이는 홈 클라이언트가, 컴퓨터 상에서의 소프트웨어 프로그램으로서 실행될 때에, 방화벽 또는 팝업 차단기에도 불구하고 교통 경보를 보여주도록 한다.

<173> 예를 들면, 사용자 특유의 정보를 획득하려고 사용자 정보를 판별하기 위해, 사용자에 의해 수동으로 입력되는

로그인 증명들이 이용될 수 있을 것이다. 대안으로, 홈 클라이언트는 그 로그인 증명들을 미리 저장할 수 있을 것이다. 어느 방법이든, 홈 클라이언트가 로그인 증명들을 구비하면, 소프트웨어 프로그램에 의해 제공된 그 홈 클라이언트가 실행될 때마다 상기 홈 클라이언트는 그 증명들을 자동적으로 송신할 수 있을 것이다. 홈 클라이언트가 아직 그 증명들을 구비하지 않는다면, 사용자는 그 로그인을 수동으로 개시할 수 있다. 그런 경우에, 홈 클라이언트는 증명들을 요구할 것이며, 그것들을 즉각 원격 서버로 송신하고 또한 나중에 사용하기 위해 그것들을 저장할 것이다.

<174> 도 5는 암호 서명된 명령을 제공하는 방법을 나타내며, 상기 명령은 명령 이름과 적어도 하나의 명령 인수를 포함한다. 상기 방법은 상기 명령 이름의 암호 서명을 생성하고 (501) 상기 명령 이름, 서명 및 적어도 하나의 명령 인수를 사용하여 암호 서명된 명령을 조립하는 (예를 들면, 연결시킴) 것 (503)을 포함한다.

<175> 예를 들면, 도 2의 실시예와 연결하여 설명된 상기 암호화 및 해독화 방식들은 디지털 서명을 제공하고 확인하기 위해 채택될 수 있다. 예를 들면, 인증된 URI는 두 인수들을 가진 명령을 포함할 수 있을 것이다. 그 첫 번째 인수는 명령인 문자들의 스트링이다. 두 번째 인수는 첫 번째 인수의 "디지털 서명"이다. 바람직하게는, 디지털 서명을 제공하는 비밀 키를 사용하여 첫 번째 인수를 서명하기 위해 표준 공용 키 암호가 채택된다. 예를 들면, 매치하는 공용 키를 구비한 홈 클라이언트는 첫 번째 인수를 대응 서명한다 (countersign). 그 키들이 같을 때 그리고 같을 때에만 그 서명 및 그 대응 서명은 상보적이다. 경우가 그러하다면, 홈 클라이언트는 첫 번째 명령을 신뢰하며, 첫 번째 인수 내의 명령이 위험하다고 간주되더라도 그 첫 번째 명령을 실행할 것이다.

<176> 상기 인증된 URI 메카니즘은 내비게이션 기기 상의 펌웨어를 갱신하기 위해서도 또한 사용될 수 있을 것이다. 바람직하게는, 펌웨어를 변경하지 않으면서, 명령을 복제하거나 또는 갱신하는 것이 허용될 것이다. 그러므로, 예를 들면, 틀린 펌웨어로 내비게이션 기기를 갱신하는 것은 그 내비게이션 기기에 돌이킬 수 없는 손상을 가져올 수 있을 것이다.

<177> 일 실시예에 따르면, 그러면, 서명된 명령은 다른 명령 (암호 서명된 명령) 내의 첫 번째 인수로서 사용되며, 그 경우 상기 서명은, 예를 들면 두 번째 인수로서 사용된다. 암호 서명된 명령의 구조는 다음과 같을 수 있을 것이다:

<178> verify\_command(명령이름1 (인수1, 인수2, 인수3), 서명)

<179> (verify\_command(commandname1(argument1,argument2,argument3),signature))

<180> 상기 서명 (외부 명령에 대해서는 두 번째 인수)은, 예를 들면 비밀 (private (secret)) 키를 인증받아야 하는 명령에 적용하여 생성될 수 있을 것이다. 즉, 명령이름1(인수1, 인수2, 인수3)+비밀\_키=대응 서명 (commandname1 (argument1,argument2,argument3)+private\_key = signature).

<181> 대응 서명을 생성하기 위해, 유사한 알고리즘이 사용될 수 있을 것이다:

<182> 명령이름1(인수1, 인수2, 인수3)+비밀\_키=대응 서명 (commandname1 (argument1, argument2, argument3)+private\_key = countersignature)

<183> 서명과 대응 서명이 서로 대응하면 그 명령은 검증된 것이다. 바람직하게는, 예를 들면, RSA 또는 DSA 알고리즘과 같은 표준의 공용 키 암호가 서명 목적으로 채택될 수 있을 것이다.

<184> 예를 들면, 다음의 명령

<185> <header>://<PublicIdentifier>?Install%3fhttp%3a%2f%2fintranet%2fplus%2fdocuments%2ftemp%2fnavcore\_6.522.7709.go510-go710.toc&JYrSIQ

<186> 을 고려하며, 이 경우 <PublicIdentifier> 는 사용자에 의해 인증되는 명령들을 위해 사용되는 실제의 명령-이름이다. 첫 번째 인수는

<187> Install%3fhttp%3a%2f%2fintranet%2fplus%2fdocuments%2ftemp%2fnavcore\_6.5 22.7709.go510-go710.toc

<188> 이다.

<189> 이는 내장된 명령이다. 실제의 명령 이름은 설치(install)이며, "?"은 "%3f"로서 인코드되며, 나머지는 설치 명령의 인수이다. 사용자는 비밀 키를 사용하여 상기 설치 명령에 서명하며, 나타나는 서명은 JYrSIQ 이다. 상기 명령을 검증하기 위해, 대응 서명이 생성될 수 있을 것이다. 상기 명령이 올바르게 서명되었다면 그 대응 서명이 매치될 것이다.

- <190> 다른 사용자가 그 명령을, 예를 들면, 다음과 같이 편집하려고 하면,
- <191> <header>://<PublicIdentifier>?Install%3fhttp%3a%2f%evil.com%2fnavcore\_6. 522.7709.go510-go710.toc&JYrSIQ,
- <192> 상기 서명과는 매치하지 않는 다른 대응 서명이 계산될 것이다. 그러므로, evil.com 으로부터의 콘텐트는 거부될 것이다.
- <193> 도 6은 내비게이션 기기 내에 저장된 현재 내비게이션 콘텐츠를 생성하는 통신 기기를 보여준다. 상기 통신 기기는 내비게이션 기기로부터 현재의 내비게이션 콘텐츠를 나타내는 정보를 인출하기 위한 인출 엘리먼트 (601), 상기 인출 엘리먼트에 연결되어 상기 현재의 내비게이션 콘텐츠를 나타내는 정보를 통신 네트워크를 경유하여 원격 네트워크 기기로 송신하는 송신기 (603), 생성된 내비게이션 콘텐츠를 상기 원격 통신 기기로부터 수신하는 수신기 (605) 및 상기 생성된 내비게이션 콘텐츠에 관련된 정보를 디스플레이하거나 또는 상기 생성된 내비게이션 콘텐트를 내비게이션 기기 상에 설치하는 프로세서 (607)를 포함한다.
- <194> 현재의 또는 생성된 내비게이션 콘텐트를 나타내는 정보는, 예를 들면, 콘텐트 유형, 콘텐트 이름, 콘텐트의 크기 또는 콘텐트 버전을 나타내는 콘텐트 파일 테이블로서 전송될 수 있을 것이다. 더 나아가, 기기 식별 또는 사용자 식별에 관련된 정보가 전송될 수 있을 것이다.
- <195> 원격 통신 기기 (예를 들면, 서버)는, 예를 들면, 이용 가능한 내비게이션 아이템들 (예를 들면, 지도, 경로 정보, GPS 상태 (fixes) 등)의 목록을 저장하는 데이터베이스를 포함할 수 있을 것이다. 더 나아가, 원격 서버는 생성된 콘텐트를 전송하기 전에, 예를 들면, 기기 식별에 기인한 내비게이션 기기의 가입 상태 또는 자격으로 인한 사용자의 자격을 검사할 수 있을 것이다.
- <196> 이전에서 언급된 것과 같이, 콘텐트 정보는 TOC (콘텐트 테이블; tale of contents)로 요약될 수 있을 것이다. 예시적인 TOC 파일은 아래와 같다:
- <197> <Program> <1d>13214</1d>
- <198> <Version>1.2<[Lambda]/ersion>
- <199> <Target>target</Target>
- <200> <Name>NameOfItem</Name>
- <201> <Preview>http://download.<download\_address>/somepreview.exe</Preview>
- <202> <Location>http://download.<web address>/someitem.cab</Location>
- <203> <Size>546484</Size>
- <204> </Program>
- <205> <Map>
- <206> <1d>8977</1d>
- <207> <Version>1.2</Version>
- <208> <Target>This target</Target>
- <209> <Name>Benelux</Name><Category>Category</Category>
- <210> <Preview>http:// download. <download\_address> /mappreview.png</Preview>
- <211> <Location>http://download.<download\_address>/mapdownload.cab</Location>
- <212> <Icon>http://download.<download\_address>/mapicon.png</Icon>
- <213> <Size>46464987</Size>
- <214> <InstallRestrictions DeviceId="35265252" />
- <215> </Map>

<216> <Voice language="en" gender="male">

<217> <Id>4657</Id> <Version>1.1<[Lambda]/ersion>

<218> <Name>user\_name</Name>

<219> <Preview>http://download.<download\_address>/voicepreview.mp3</Preview>

<220> <Location>http://download.<download\_address>/voicedownload.cab</Location>

<221> <Size>46879794623K</Size>

<222> <Picture>http://<download\_address>/voicepicture.png</Picture>

<223> <lcon>http://<download\_address>/voiceicon.png</lcon>

<224> </Voice>

<225> <Colorscheme> <Id>98797</Id>

<226> <Version>1.1 <[Lambda]/ersion>

<227> <Name>Bright Black</Name>

<228> <Location>http://download.<download\_address>/colordownload.cab</Location>

<229> <Size>6464</Size>

<230> </Colorscheme>

<231> <Poi>

<232> <Id>65446</Id>

<233> <Version>1.3<[Lambda]/ersion>

<234> <Name>Wifi Hotspots</Name>

<235> <Category>Connectivity</Category>

<236> <Location>http://download.<download\_address>/poidownload.cab</Location>

<237> <Size>54567</Size>

<238> </Poi>

<239> </Toc>

<240> 필드 <InstallRestrictions DeviceId="35265252"/> 는 (<Target> 엘리먼트와는 별도로) 이 아이템의 설치 가능성 상의 제한을 상술한다 그것은 아무 콘텐트도 가지지 않는다. 선택적인 속성들은, 아이템이 이 기기 ID를 가지고 기기 상에 설치되어야 한다는 것을 의미하는 deviceId (스트링 (string)) 그리고 이 아이템을 위해서 꼭 설치되어야 하는 내비게이션 프로그램의 최소 버전을 의미하는 minNavigatorVersion (스트링) 이다.

<241> 도 7은, 예를 들면 실행 가능한 애플리케이션 프로그램 (701) (스위트 애플리케이션, 홈 애플리케이션)과 함께 하는 통신 시나리오의 블록도를 보여주며, 상기 애플리케이션 프로그램은, 예를 들면, 도 7에 도시된 것과 같이 원격 서버 (705)와 정보를 교환하기 위해 통신 네트워크 (703)를 경유하여 통신하는 컴퓨터 상에서 동작한다. 서버 (707)는, 예를 들면, 애플리케이션 프로그램 (701)에 의해 제공된 벌룬 메시지들 (balloon messages) (707) 내에 또는 브라우저 컴포넌트들 내의 어느 하나에서 표시되는 메시지들을 제공할 수 있을 것이다.

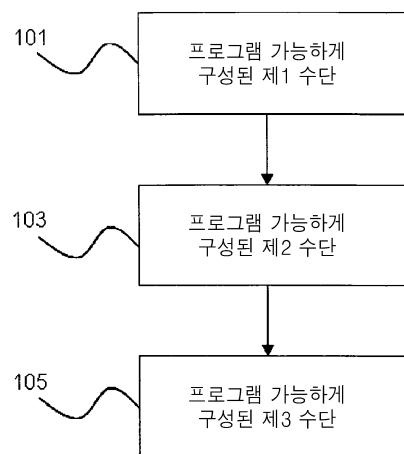
<242> 상기에서 설명된 실시예들 내의 엘리먼트들과 특징들의 특정한 결합들은 단지 예시적일 뿐이며, 이런 교시들을 참조로서 편입된 특허들/애플리케이션들 내에서 다른 교시들과 바꾸거나 대체하는 것 역시 명백하게 예측된다. 본 발명이 속한 기술 분야의 통상의 지식을 가진 당업자는 여기에서 설명된 것이 청구된 본 발명의 사상과 범위에서 벗어나지 않고서도 변형되고 수정되며 다른 실시예들이 발생할 수 있다는 것을 인식할 것이다. 따라서, 전술한 설명은 예일 뿐이며 제한하려는 의도는 아니다. 본 발명의 범위는 이어지는 청구항들 및 그들의 등가물 내에서 정의된다. 더 나아가, 설명과 청구항들에서 사용된 참조 기호들은 청구된 발명의 범위를 한정하지 않는다.

## 도면의 간단한 설명

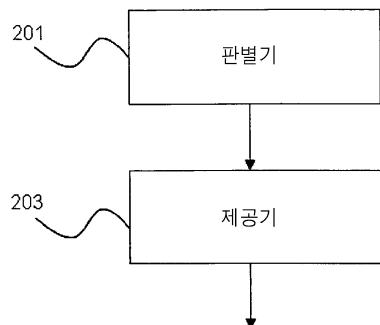
- <72> 본 발명의 추가의 실시예들은 다음의 도면들에 관해 설명될 것이다.
- <73> 도 1은 통신 기기의 블록 도면을 보여준다.
- <74> 도 2는 프로세싱 기기의 블록 도면을 보여준다.
- <75> 도 3은 통신 기기의 블록 도면을 보여준다.
- <76> 도 4는 특정 정보를 제공하는 방법을 보여준다.
- <77> 도 5는 암호 서명된 명령을 제공하는 방법을 보여준다.
- <78> 도 6은 내비게이션 콘텐트를 생성하는 통신 기기의 블록도를 보여준다.
- <79> 도 7은 통신 시나리오의 블록도를 보여준다.

## 도면

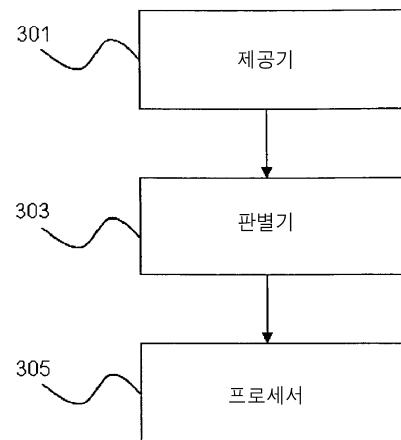
### 도면1



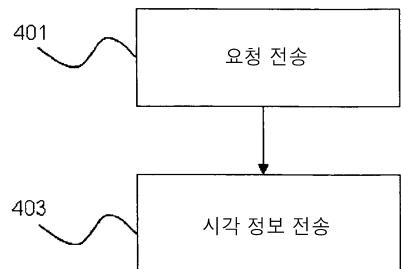
### 도면2



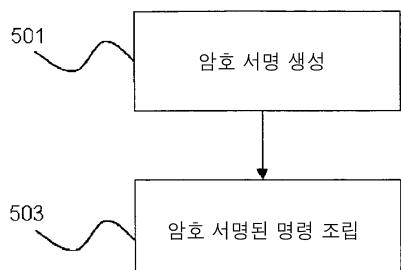
도면3

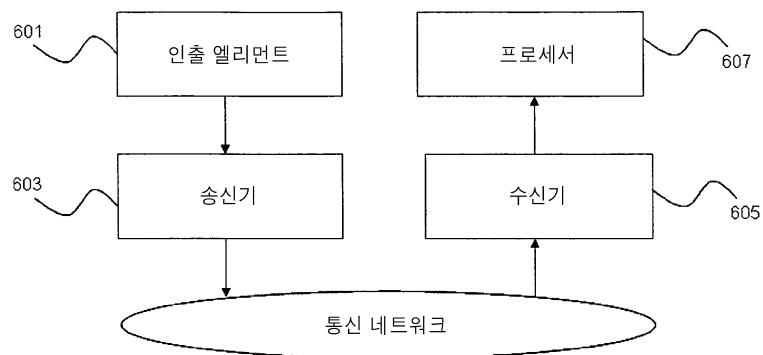


도면4



도면5



**도면6****도면7**