



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2015-0004877

(43) 공개일자 2015년01월13일

(51) 국제특허분류(Int. Cl.)
G06F 9/06 (2006.01) G06F 9/44 (2006.01)
G06F 15/16 (2006.01)
(21) 출원번호 10-2014-7032812
(22) 출원일자(국제) 2013년04월23일
심사청구일자 없음
(85) 번역문제출일자 2014년11월21일
(86) 국제출원번호 PCT/US2013/037701
(87) 국제공개번호 WO 2013/163122
국제공개일자 2013년10월31일
(30) 우선권주장
13/453,176 2012년04월23일 미국(US)

(71) 출원인
구글 인코포레이티드
미국 캘리포니아 마운틴 뷰 엠피시어터 파크웨이
1600 (우:94043)
(72) 발명자
프로코피오 마이클 제프리
미국 캘리포니아 94043 마운틴 뷰 엠피시어터 파
크웨이 1600
베센 아담
미국 캘리포니아 94043 마운틴 뷰 엠피시어터 파
크웨이 1600
위릭 로버트
미국 캘리포니아 94043 마운틴 뷰 엠피시어터 파
크웨이 1600
(74) 대리인
박장원

전체 청구항 수 : 총 20 항

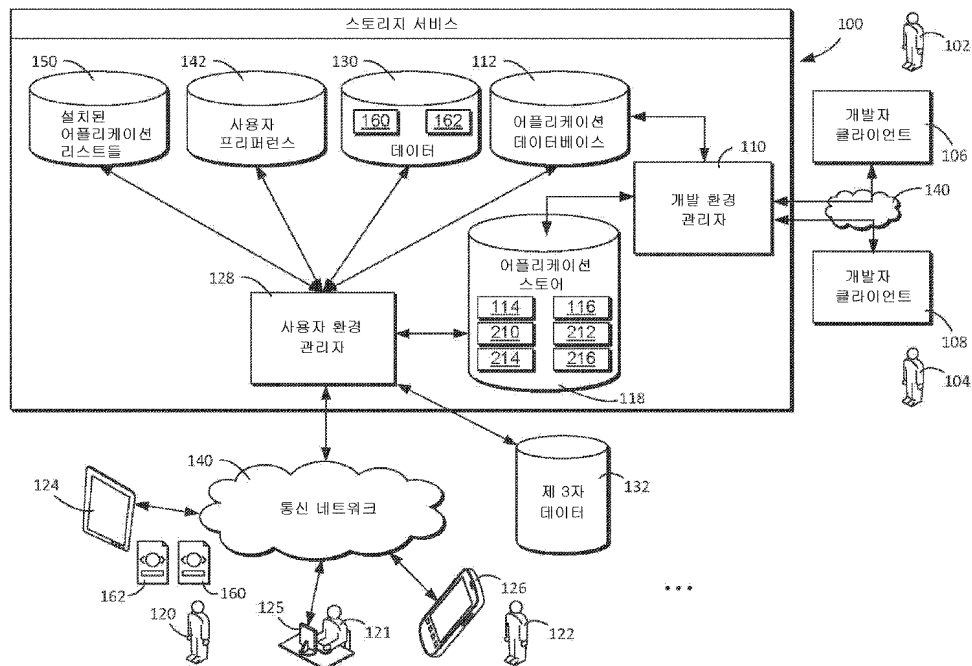
(54) 발명의 명칭 네트워크 스토리지 서비스에서 어플리케이션과 파일타입의 관련

(57) 요약

네트워크 스토리지 서비스에서 파일 타입들을 어플리케이션과 관련짓는 기법들이 기술되었다. 상기 스토리지 서
비스의 사용자 환경 관리자는 컴퓨팅 디바이스를 작동하는 사용자로부터 특정 요청된 파일에 대한 파일 식별자를
수신하고, 상기 파일에 대한 데이터 스토어로부터 상기 파일 식별자와 관련된 파일 포맷 타입 식별자를

(뒷면에 계속)

대표도



획득한다. 상기 사용자 환경 관리자는 사용자 프리퍼런스에 내포된 디폴트 "오픈 위드" 리스트로부터의 어플리케이션 ID (앱 ID)를 상기 파일 포맷 타입을 지원하는 설치 어플리케이션들에 대한 어플리케이션 ID들과 매칭시키는 것에 응답하여, 상기 특정 요청된 파일을 오픈하도록 디폴트 어플리케이션을 결정한다. 결정이 되면, 상기 사용자 환경 관리자는 상기 디폴트 어플리케이션을 활용하여 상기 데이터 스토어로부터 상기 특정 요청된 파일을 오픈한다. 일부 구현들에서, 상기 사용자 환경 관리자는 요청된 파일을 오픈할 수 있는 어플리케이션들의 리스트를 제시한다.

특허청구의 범위

청구항 1

통신 네트워크와 동작적으로 결합되는 컴퓨터로 구현된 서버상에서 동작하는 스토리지 서비스로서,
소프트웨어 어플리케이션 설명들(description)을 포함하는 소프트웨어 어플리케이션 스토어 - 각 소프트웨어 어플리케이션 설명은 어플리케이션 ID, 어플리케이션 타이틀, 및 지원 파일 타입들(supported file types)을 구비하고, 소프트웨어 어플리케이션 설명들은 스토리지 서비스 제공자 또는 제3자 개발자와 관련되며 - 와; 그리고 사용자 데이터 스토어를 포함하며,
상기 사용자 데이터 스토어는:

사용자들과 관련된 파일들, 파일 식별자를 갖는 상기 사용자 데이터 내의 각 파일들, 파일 타입 식별자, 및 파일 타입에 따라 2진 인코딩된 데이터,

사용자를 위해 상기 소프트웨어 어플리케이션 스토어로부터 사용자 설치 소프트웨어 어플리케이션들을 식별하는 어플리케이션 ID들의 리스트, 및

특정한 사용자 설치 소프트웨어 어플리케이션들의 어플리케이션 ID들을 포함하는, 파일들을 오픈하기 위한 디폴트 어플리케이션들을 특정하는 파일 액세스 사용자 프리퍼런스들을 포함하고, 그리고

상기 서비스는:

요청된 특정 파일에 대한 파일 식별자를 컴퓨팅 디바이스를 동작시키는 사용자로부터 상기 통신 네트워크를 통해 상기 스토리지 서비스에서 수신하고;

상기 수신된 파일 식별자와 관련된 상기 파일에 대한 상기 파일 타입 식별자를 상기 사용자 데이터 스토어로부터 획득하고;

상기 획득된 파일 타입을 지원하는 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들의 리스트를 결정하기 위해 상기 획득된 파일 타입 식별자를 상기 소프트웨어 어플리케이션 스토어에 문의(query)하고;

상기 파일 액세스 사용자 프리퍼런스들에 내포된 어플리케이션 ID가 상기 파일 타입을 지원하는 소프트웨어 어플리케이션에 대한 어플리케이션 ID들의 상기 선택된 리스트에 매칭함에 응답하여 상기 요청된 특정 파일을 오픈하기 위한 어플리케이션을 결정하고; 그리고

상기 요청된 특정 파일을 오픈하기 위해 결정된 상기 어플리케이션을 활용하여 상기 사용자 데이터 스토어로부터 상기 요청된 특정 파일을 오픈하도록 동작하는 것을 특징으로 하는 스토리지 서비스.

청구항 2

제1항에 있어서,

상기 서비스는,

상기 소프트웨어 어플리케이션 데이터베이스에서의 상기 문의 동작이 소프트웨어 어플리케이션을 찾지 못함에 응답하여, 시스템 디폴트 내장 어플리케이션을 선택하도록 동작하는 것을 특징으로 하는 스토리지 서비스.

청구항 3

제1항에 있어서,

클라이언트 디바이스를 동작하는 사용자로부터 상기 통신 네트워크를 통해 상기 스토리지 서비스에서 파일 생성 요청을 수신하도록 동작하는 사용자 환경 관리자를 포함하고, 상기 파일 생성 요청은 특정 파일 타입을 포함하고, 상기 사용자 환경 관리자는 (i) 상기 지원 파일 타입을 갖는 소프트웨어 어플리케이션들의 클라이언트 ID들을 선택하기 위해 상기 수신된 특정 파일 타입을 상기 소프트웨어 어플리케이션 데이터베이스에 문의하고, (ii) 상기 선택된 클라이언트 ID들 중 하나와 매칭하는 상기 사용자 선택 파일 액세스 프리퍼런스에 지정된 클라이언트 ID를 갖는 소프트웨어 어플리케이션을 상기 사용자 데이터 스토어로부터 선택함으로써, 상기 특정 파일 타입

을 생성하기 위한 어플리케이션을 결정하도록 동작적으로 구성되고, 상기 사용자 환경 관리자는 상기 선택된 소프트웨어 어플리케이션으로 상기 특정 파일 타입을 갖는 파일을 생성함에 응답하여 상기 컴퓨팅 디바이스에 대한 디스플레이 정보를 생성하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 4

통신 네트워크와 동작적으로 결합된 컴퓨터 구현 서버상에서 동작하는 스토리지 서비스로서,

소프트웨어 어플리케이션들에 대한 메타데이터를 포함하는 소프트웨어 어플리케이션 스토어 - 각 소프트웨어 어플리케이션에 대한 상기 메타데이터는 어플리케이션 식별자(app ID), 어플리케이션 타이틀, 어플리케이션 설명, 및 지원 MIME(Multipurpose Internet Mail Extensions) 타입들을 포함하고, 상기 소프트웨어 어플리케이션은 상기 스토리지 서비스 제공자 또는 제3자 개발자 중 하나와 관련됨-

사용자들과 관련된 파일들을 포함하는 데이터 스토어 - 상기 데이터 스토어 내의 각 파일은 파일 식별자, MIME-타입 식별자, 및 MIME-타입에 따라 2진 인코딩된 데이터를 구비함-

특정 사용자를 위한 상기 소프트웨어 어플리케이션 스토어로부터 설치 소프트웨어 어플리케이션들을 식별하는 어플리케이션 ID들의 설치 어플리케이션 리스트;

파일들을 오픈하기 위해 디폴트로 사용될 특정 사용자 설치 소프트웨어 어플리케이션들의 어플리케이션 ID들을 포함하는 사용자 프리퍼런스 스토어; 및

사용자 인터페이스를 포함하고,

상기 사용자 인터페이스는:

컴퓨팅 디바이스를 동작하는 상기 특정 사용자로부터 상기 통신 네트워크를 통해 상기 스토리지 서비스에서 요청된 특정 파일에 대한 파일 식별자를 수신하고;

상기 수신된 파일 식별자와 관련된 상기 파일을 대한 상기 MIME-타입 식별자를 상기 데이터 스토어로부터 획득하고;

상기 획득된 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션을 위한 어플리케이션 ID들의 리스트를 선택하도록 어플리케이션 ID들의 상기 설치 어플리케이션 리스트 및 상기 획득된 MIME-타입 식별자를 상기 소프트웨어 어플리케이션 스토어에 문의하고;

상기 사용자 프리퍼런스 스토어에 내포된 어플리케이션 ID가 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들을 위한 어플리케이션 ID들의 상기 선택된 리스트와 매칭함에 응답하여 상기 요청된 특정 파일을 오픈하기 위한 디폴트 어플리케이션을 결정하고; 그리고

상기 요청된 특정 파일을 오픈하기 위한 상기 결정된 디폴트 어플리케이션을 활용하여 상기 데이터 스토어로부터 상기 요청된 특정 파일을 오픈하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 5

제4항에 있어서,

사용자 인터페이스는 상기 소프트웨어 어플리케이션 데이터베이스에서의 문의 동작이 상기 MIME-타입을 지원하는 다른 설치 소프트웨어 어플리케이션을 선택하지 못함에 응답하여 임의의 MIME-타입을 갖는 파일들을 프리뷰할 수 있는 범용 디폴트 어플리케이션을 선택하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 6

제4항에 있어서,

사용자 인터페이스는 (i) 상기 획득된 MIME 타입을 지원하는 설치 소프트웨어 어플리케이션들을 위한 어플리케이션 ID들 및 어플리케이션 타이틀들의 리스트를 선택하기 위해 상기 설치 어플리케이션 리스트 및 상기 획득된 MIME-타입 식별자를 상기 소프트웨어 어플리케이션 스토어에 문의하고, (ii) 사용자 동작 컴퓨팅 디바이스로 상기 통신 네트워크를 통해 상기 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들을 위한 어플리케이션 ID들 및 어플리케이션 타이틀들의 상기 선택된 리스트를 보냄으로써, 상기 컴퓨팅 디바이스가 상기 컴퓨팅 디바이스의 디스플레이 상에서 사용자에게 설치 소프트웨어 어플리케이션들에 대한 어플리케이션 타이틀들의 상기 선택

택된 리스트를 디스플레이함에 응답하여, 상기 특정 요청된 파일을 오픈하기 위해 특정 설치 어플리케이션의 선택을 요청하도록 구성되고, 상기 사용자 인터페이스는 또한 상기 요청된 특정 파일을 오픈하기 위해 사용자-선택된 특정 설치 어플리케이션을 활용하여 상기 데이터 스토어로부터 상기 요청된 특정 파일을 오픈하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 7

제4항에 있어서,

상기 사용자 프리퍼런스 스토어는 파일을 오픈하기 위해 디폴트로서 사용될 특정 사용자 설치 소프트웨어 어플리케이션들의 어플리케이션 ID들의 순서 리스트를 포함하고, 상기 사용자 인터페이스는 (i) 상기 MIME 타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들의 선택된 리스트와 매칭하는 상기 사용자 프리퍼런스 스토어에 내포된 어플리케이션 ID들의 순서 리스트를 획득하고, (ii) 상기 요청된 특정 파일을 오픈하기 위한 상기 결정된 디폴트 어플리케이션으로서 적어도 2개의 어플리케이션 ID들의 상기 획득된 순서 리스트에서 가장 최근의 어플리케이션 ID를 선택함에 응답하여, 상기 요청된 특정 파일을 오픈하기 위한 디폴트 어플리케이션을 결정하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 8

제4항에 있어서,

사용자 프리퍼런스 스토어는 특정 MIME-타입의 파일들을 오픈하기 위해 디폴트로서 사용될 사용자 설치 소프트웨어 어플리케이션의 하나의 어플리케이션 ID에 대한 각 특정 MIME-타입의 맵핑을 포함하고, 상기 사용자 인터페이스는 그 특정 MIME-타입의 파일들을 오픈하기 위해 디폴트로서 사용될 사용자 설치 소프트웨어 어플리케이션의 하나의 어플리케이션 ID를 선택하도록 상기 획득된 MIME-타입 식별자를 구비한 어플리케이션 ID들에 대한 MIME-타입들의 맵핑을 상기 사용자 프리퍼런스 스토어에 문의함에 응답하여, 상기 특정 요청된 파일을 열기 위한 상기 디폴트 어플리케이션을 결정하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 9

제4항에 있어서,

상기 사용자 인터페이스는:

상기 컴퓨팅 디바이스를 동작하는 상기 사용자로부터 상기 통신 네트워크를 통해 상기 스토리지 서비스에 파일 생성 요청을 수신하고, 상기 파일 생성 요청은 특정 MIME-타입을 포함함;

상기 수신된 특정 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션을 위한 어플리케이션 ID를 선택하기 위해 상기 설치 어플리케이션 리스트 및 상기 수신된 특정 MIME-타입을 상기 소프트웨어 어플리케이션 스토어에 문의함으로써 상기 특정 MIME-타입의 파일을 생성하도록 어플리케이션을 결정하고; 그리고

상기 특정 MIME-타입의 파일을 생성하기 위한 상기 결정된 어플리케이션을 활용하여 상기 사용자와 관련된 상기 데이터 스토어에 파일을 생성하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 10

제9항에 있어서,

상기 사용자 인터페이스는 상기 수신된 특정 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 둘 이상의 어플리케이션 ID의 리스트를 야기하도록, 상기 설치 어플리케이션 리스트 및 상기 수신된 특정 MIME-타입을 상기 소프트웨어 어플리케이션 스토어에 문의함에 응답하여 상기 특정 MIME-타입의 파일을 생성하기 위한 특정 설치 어플리케이션의 선택을 요청하도록 구성되며, 상기 사용자 인터페이스는 (i) 상기 수신된 특정 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들 및 어플리케이션 타이틀들의 리스트를 선택하기 위해 상기 설치 어플리케이션 리스트 및 상기 수신된 특정 MIME-타입을 상기 소프트웨어 어플리케이션 스토어에 문의하고, (ii) 상기 수신된 특정 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 상기 어플리케이션 ID들 및 어플리케이션 타이틀들의 리스트를 상기 통신 네트워크를 통해 상기 사용자 동작 컴퓨팅 디바이스로 전송(send)함으로써, 상기 컴퓨팅 디바이스가 상기 컴퓨팅 디바이스의 디스플레이 상에서 상기 사용자에게 설치 소프트웨어 어플리케이션들을 위한 어플리케이션 타이틀들의 리스트를 디스플레이할 수 있

게 하는 것을 특징으로 하는 스토리지 서비스.

청구항 11

제9항에 있어서,

사용자 인터페이스는 상기 특정 MIME-타입의 파일을 생성하기 위해 상기 사용자-선택된 특정 설치 어플리케이션을 활용하여 상기 사용자와 관련된 상기 데이터 스토어에 파일을 생성하도록 구성되는 것을 특징으로 하는 스토리지 서비스.

청구항 12

하나 이상의 서버들 상에서 동작하는 스토리지 서비스에 의해 저장된 특정 요청된 파일을 디폴트 소프트웨어 어플리케이션으로 오픈 가능하게 하는 방법으로서,

특정 사용자에 의해 동작된 컴퓨팅 디바이스로부터 통신 네트워크를 통해 상기 스토리지 서비스에서 상기 특정 요청된 파일에 대한 파일 식별자를 수신하는 단계;

상기 스토리지 서비스에 의해 호스트된(hosted) 데이터 스토어로부터 상기 수신된 파일 식별자와 관련된 상기 특정 요청된 파일에 대한 MIME-타입 식별자를 획득하는 단계;

상기 스토리지 서비스에 의해 호스트된 설치 어플리케이션 스토어로부터 어플리케이션 식별자들(어플리케이션 ID들)의 설치 어플리케이션 리스트를 획득하는 단계, 상기 설치 어플리케이션 리스트는 상기 특정 사용자를 위한 상기 스토리지 서비스에 의해 호스트된 웹-기반 사용자 스토리지 환경에 설치되는 상기 특정 사용자를 위한 소프트웨어 어플리케이션들을 식별함;

상기 획득된 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들의 리스트를 선택하기 위해 상기 설치 어플리케이션 리스트 및 상기 획득된 MIME-타입 식별자를 소프트웨어 어플리케이션 스토어에 문의하는 단계 - 소프트웨어 어플리케이션 각각의 상기 메타데이터는 어플리케이션 ID들, 어플리케이션 타이틀, 어플리케이션 설명(description) 및 지원 MIME 타입들을 포함하고, 상기 소프트웨어 어플리케이션은 상기 스토리지 서비스 제공자 또는 제3자 개발자와 관련됨-;

파일들을 오픈하기 위해 디폴트로서 사용될 특정 사용자 설치 소프트웨어 어플리케이션들의 어플리케이션 ID들을 내포하는 상기 스토리지 서비스에 의해 호스트된 사용자 프리퍼런스 스토어로부터의 어플리케이션 ID가 상기 MIME-타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들의 상기 선택된 리스트에 매칭함에 응답하여, 상기 특정 요청 파일을 오픈하기 위한 디폴트 어플리케이션을 결정하는 단계; 및

상기 특정 사용자를 위한 상기 스토리지 서비스에 의해 호스트된 상기 웹-기반 사용자 스토리지 환경에서 상기 특정 요청된 파일을 오픈하기 위해 상기 결정된 디폴트 어플리케이션을 활용하여 상기 데이터 스토어로부터 상기 특정 요청된 파일을 오픈하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 13

컴퓨터 서버에 의해 실행될 때, 상기 컴퓨터 서버로 하여금 디폴트 소프트웨어 어플리케이션으로 상기 서버상에서 동작하는 스토리지 서비스에 의해 저장된 특정 요청된 파일을 오픈 가능하게 하는 동작들을 수행하게 하는 명령어들을 저장하는 컴퓨터-판독가능 저장 매체로서,

상기 동작들은:

특정 사용자에 의해 동작되는 컴퓨팅 디바이스로부터 통신 네트워크를 통해 상기 스토리지 서비스에서 상기 특정 요청된 파일에 대한 파일 식별자를 수신하는 동작;

상기 수신된 파일 식별자와 관련된 상기 특정 요청된 파일에 대한 상기 스토리지 서비스에 의해 호스트된 데이터 스토어로부터 파일 타입 식별자를 획득하는 동작;

상기 스토리지 서비스에 의해 호스트된 설치 어플리케이션 스토어로부터 어플리케이션 식별자들(어플리케이션 ID들)의 설치 어플리케이션 리스트를 획득하는 동작 - 상기 설치 어플리케이션 리스트는 상기 특정 사용자를 위한 상기 스토리지 서비스에 의해 호스트된 웹-기반 사용자 스토리지 환경에 설치된 상기 특정 사용자를 위한 소프트웨어 어플리케이션들을 식별함 -;

상기 획득된 파일 타입을 지원하는 설치 소프트웨어 어플리케이션들에 대한 어플리케이션들의 리스트를 선택하기 위해 상기 설치 어플리케이션 리스트 및 상기 획득된 파일 타입 식별자를 소프트웨어 어플리케이션 스토어에 문의하는 동작 - 상기 소프트웨어 어플리케이션 스토어는 소프트웨어 어플리케이션들에 대한 메타데이터를 포함하고, 각 소프트웨어 어플리케이션에 대한 상기 메타데이터는 어플리케이션 ID, 어플리케이션 타이틀, 어플리케이션 설명, 지원 파일 타입들을 포함하며, 상기 소프트웨어 어플리케이션은 상기 스토리지 서비스 제공자 또는 제3자 개발자와 관련됨 -;

상기 특정 요청된 파일을 오픈할 수 있는 어플리케이션들의 리스트를 사용자에게 제공하는 동작; 및

상기 사용자-선택 어플리케이션을 활용하여 상기 데이터 스토어로부터 상기 특정 요청된 파일을 오픈하는 동작을 포함하는 것을 특징으로 하는 컴퓨터-판독가능 저장 매체.

청구항 14

제13항에 있어서,

상기 파일 타입은 MIME 타입인 것을 특징으로 하는 컴퓨터 판독가능 저장 매체.

청구항 15

국부적으로 설치된 소프트웨어를 필요로 함이 없이, 각 사용자들이 시스템 제공 어플리케이션들로 파일들을 저장할 수 있으며 그리고 파일들을 작업할 수 있는 네트워크 기반 파일 스토리지 및 네트워크 기반 어플리케이션 시스템을 동작하는 방법으로서,

제3자 어플리케이션 및 상기 제 3자 어플리케이션에 대한 등록 정보를 수신하는 단계;

상기 어플리케이션을 파일 타입들로 맵핑하는 단계;

상기 어플리케이션을 사용자로 하여금 설치할 수 있게 하는 단계;

사용자로부터 파일 오픈 요청에 응답하여, 상기 사용자가 디폴트 어플리케이션을 선택한 경우 사용자 선택 디폴트 어플리케이션으로 파일을 오픈하고, 상기 사용자가 디폴트 어플리케이션을 선택하지 않은 경우 시스템 제공 디폴트 어플리케이션으로 파일을 오픈하는 단계; 및

사용자로부터의 파일 "오픈 위드(open with)" 요청에 응답하여, 설치된 제 3자 어플리케이션들 및 시스템 제공 어플리케이션들의 리스트를 상기 사용자에게 제시하고, 상기 사용자에게 의해서 상기 리스트로부터 선택된 어플리케이션으로 상기 파일을 오픈하는 단계를 포함하는 것을 특징으로 하는 방법.

청구항 16

제15항에 있어서,

상기 맵핑하는 단계는 MIME 타입들을 이용하여 상기 어플리케이션을 파일 타입들로 맵핑하는 것을 특징으로 하는 방법.

청구항 17

제15항에 있어서,

각 사용자들에 대한 설치 어플리케이션들의 각 리스트들을 유지하는 단계를 더 포함하는 것을 특징으로 하는 방법.

청구항 18

제15항에 있어서,

사용자에 의해 설치되지 않았던 어플리케이션들은 사용자로부터의 "오픈 위드" 요청에 응답하여 나타나지 않는 것을 특징으로 하는 방법.

청구항 19

제15항에 있어서,

상기 맵핑하는 단계는 제3자 어플리케이션들을 나타내는 파일 확장자들을 파일 타입들로 맵핑하는 것을 포함하는 것을 특징으로 하는 방법.

청구항 20

제15항에 있어서,

상기 맵핑하는 단계는 제3자 어플리케이션들을 나타내는 파일 확장자들을 파일 타입들로 맵핑하는 것을 포함하며, 상기 각 파일 확장자들은 상기 각각의 제3자 어플리케이션들이 생성한 파일들 내에 포함된 데이터인 것을 특징으로 하는 방법.

명세서

기술 분야

[0001]

[관련 출원의 상호 참조]

[0002]

본 출원은 2012년 4월 23일자로 출원된 미국 출원 번호 제13/453,176호의 우선권을 주장한다. 상기 출원의 콘텐츠는 그 전체가 본 명세서에 참조로서 포함된다.

배경 기술

[0003]

파일 포맷(format)은 파일에 내포된 데이터, 전형적으로 1과 0의 2진 데이터를 인간에게 유용한 정보로 전환시키는 방법을 특정하기 위해 사용된다. 파일의 타입을 식별하는 한 방법은 상기 포맷에 관한 정보를 상기 파일에 자체적으로 저장하는 것이다. 이러한 정보는 파일 내의 특정한 위치들에, 예를 들어 파일의 시작 부분의 파일 헤더에 내포된 메타데이터에 또는 파일의 어느 곳에 위치된다. 파일 타입 정보가 포함된 이 헤더들은 인간 판독 가능하거나 또는 2진일 수 있다.

[0004]

일부 문서와 파일 관리 시스템들에서, 상기 시스템에서의 파일들 또는 아이템들은 표준의 제너릭(jeneric) 타입 정보, 예를 들어 "MIME 타입"과 관련될 수 있다. MIME(Multipurpose Internet Mail Extensions)은 2진 파일 인코딩들을 표준화된 방식으로 특정하는 메커니즘을 기술하는 IETE(Internet Engineering Task Force) 표준이다. MIME 타입들은 원래, 송신자 및 수신자의 운영 체제와는 독립적으로, 근본적으로 이메일에 어떤 타입의 파일이 부착되었는지를 식별하는데 사용되었다. 파일 이름 확장자들(예를 들어, 일반 텍스트(plain text)를 위한 ".txt")은 파일의 포맷을 특정하기 위한 다른 방식이다.

발명의 내용

해결하려는 과제

[0005]

네트워크 스토리지 서비스에서 파일 타입을 어플리케이션과 관련짓기 위한 기법들이 기술된다. 스토리지 서비스에서의 사용자 환경 관리자는 컴퓨팅 디바이스를 작동시키는 사용자로부터 특정 요청된 파일을 위한 파일 식별자를 수신하고, 상기 파일을 위해 데이터 스토어로부터 상기 파일 식별자와 관련된 파일 포맷 타입 식별자를 획득한다. 사용자 환경 관리자는 파일 포맷 타입을 지원하는 설치된 어플리케이션들을 위한 어플리케이션 ID들의 리스트를 갖는 사용자 프리퍼런스 스토리지에 내포된 리스트와 디폴트 오픈(default open)으로부터의 어플리케이션 ID(app ID)를 매칭(match)시키는 것에 응답하여 특정 요청 파일을 오픈(open)하도록 디폴트 어플리케이션을 결정한다. 결정되면, 사용자 환경 관리자는 디폴트 어플리케이션을 활용하여 데이터 스토리지로부터 특정 요청 파일을 오픈한다. 몇몇 구현들에서, 사용자 환경 관리자는 요청 파일을 오픈할 수 있는 어플리케이션들의 리스트를 제시한다.

[0006]

몇몇 구현들에서, 제3자 어플리케이션 및 등록 정보가 수신된다. 이 어플리케이션은 파일 타입들, 예를 들어, MIME 타입들(또는 파일들 자체에 저장된 다른 파일 타입 정보)에 맵핑된다. 이 어플리케이션은 사용자로 하여금 설치가능하게 만들어진다. "오픈(Open)" 요청에 응답하여, 사용자 디폴트 어플리케이션이 선택된 경우에 파일은 사용자 디폴트 어플리케이션으로 오픈된다. 사용자 디폴트 어플리케이션이 선택되지 않은 경우에, 상기 파일은 시스템 디폴트 어플리케이션으로 오픈된다. "오픈 위드(Open with)" 요청에 응답하여, 사용자는 설치된 제3자 및 시스템 어플리케이션들의 리스트를 제공받는다. 파일은 상기 리스트로부터 선택된 어플리케이션으로 오픈된다.

[0007] 본 개요는 청구항의 범위와 의미를 해석하거나 제한하는데 사용되지 않는다는 이해와 함께 제공된다. 본 개요는 청구 대상의 주요 특징들 또는 본질적 특징들을 식별하도록 의도되지 않으며, 청구 대상의 범위를 결정하는데 보조물으로써 사용되도록 의도되지도 않는다.

도면의 간단한 설명

[0008] 도 1은 어플리케이션과 파일 타입을 관련짓는 시스템의 개략도이다.

도 2는 본 발명의 하나 이상의 구현들에 따라 이용된 사용자 인터페이스를 보여주는 모의 스크린도이다.

도 3은 도 2에 도시된 것과 다른 스테이지(stage)에서, 본 발명의 하나 이상의 구현들에 따라 이용된 사용자 인터페이스를 보여주는 모의 스크린도이다.

도 4는 본 발명의 하나 이상의 구현들에 따라 이용된 기법들의 흐름도이다.

도 5는 본 발명의 하나 이상의 구현들에 따라 이용된 기법들의 흐름도이다.

도 6은 본 발명의 하나 이상의 구현들에 따라 이용된 예시적인 컴퓨터 시스템의 개략도이다.

구현들은 첨부된 도면들을 참조하여 기술된다. 상기 도면들에서, 동일한 참조번호들은 동일하거나 기능적으로 유사한 요소들을 표시할 수 있다. 요소가 첫 번째로 나타나는 도면은 일반적으로 해당 참조 번호에서 가장 왼쪽 디지털(digit)에 의해 표시된다.

발명을 실시하기 위한 구체적인 내용

[0009] 클라우드 스토리지(cloud storage)의 가용성이 크게 확대되어 왔다. 클라우드 스토리지는 인터넷을 통해 이용가능한 네트워크 스토리지의 한 형태이다. 본원에 사용되는 바와 같이, 네트워크 스토리지는, 예를 들어 인터넷을 포함하거나 포함하지 않을 수 있는 네트워크 또는 인터넷과 연결된 네트워크를 통해 이용가능한 스토리지를 포함할 수 있다. 네트워크 저장소에 저장된 데이터는 일반적으로 네트워크를 통해 공유되는 데이터 베이스 콘텐츠, 사용자 파일들, 또는 다른 데이터 콘텐츠의 형태를 취할 수 있다. 네트워크 스토리지의 호스팅(hosting)은 대용량 데이터 센터들에서 전형적으로 일어나며, 고객들은 이러한 네트워크 스토리지를 임대 또는 사용한다. 데이터 센터 오퍼레이터(operator)는, 백그라운드(background)에서, 고객의 요청사항들에 따라 자원들을 가상화하고 그들을 스토리지 풀(storage pool)로 노출시킴으로써, 고객들은 파일 또는 데이터 객체들을 저장하도록 저장 풀을 사용할 수 있다. 물리적으로, 자원은 다양한 위치들에 있는 다양한 서버들에 걸쳐있을 수 있으며, 고객 또는 그들의 최종 사용자들에 근접한 각종 데이터 센터들에서 복제될 수 있다. 이러한 클라우드 스토리지 서비스들은 웹 서비스 API(application programming interface), 또는 웹-기반 사용자 인터페이스를 통해 액세스될 수 있다.

[0010] 이 클라우드 또는 네트워크 스토리지의 서브세트(subset)는 개인 스토리지이다. 저장소 서비스들은 문서(document)들, 사진들, 및 음악 파일들을 포함하는 모든 종류의 파일들을 네트워크상에 저장하는 쉽고 편리한 방법을 네트워크-기반 저장 서비스와 함께 제공한다. 파일들은 네트워크 연결(예를 들어, 클라우드 스토리지의 경우 인터넷 연결)이 있는 곳, 즉, 어디든지 그들이 실질적으로 액세스가 가능한 네트워크상에 저장된다. 만약 원본 파일이 사용자의 로컬 하드 드라이브(local hard drive)로부터 손상 또는 손실된다면, 파일은 스토리지 서비스에 의해 쉽게 복원될 수 있다. 몇몇 스토리지 서비스들은 또한 전형적으로 공유될 파일들을 갖는 폴더로의 링크를 통해 또는 파일로의 직접 링크를 통해, 사용자로 하여금 다른 파일과 함께 파일들을 공유하도록 한다.

[0011] 파일들은 전형적으로 컴퓨팅 디바이스들의 다양한 타입들로부터 액세스될 수 있는데, 상기 컴퓨팅 디바이스들은 예를 들어, 컴퓨터 워크스테이션(workstation), 모바일 폰들, 비디오 게임 콘솔(consoles)들, 랩탑(laptop)들, 태블릿 컴퓨터(tablet computer)들 또는 인터넷을 액세스할 수 있는 임의의 다른 디바이스를 포함한다. 몇몇 네트워크 스토리지 서비스들은 독점적인 인터페이스(proprietary interface)를 갖는 웹 브라우저 인터페이스(web browser interface) 및 기타의 인터페이스를 통해 액세스되고, 상기 독점적인 인터페이스는 업로드, 액세스 및 공유 파일들로 하여금 더 사용되기 쉽도록 설계된 특징들을 제공하는 동시에, 또한 사용자를 위한 백업 스케줄링 또는 하위-계정들(sub-accounts)의 생성과 같은 관리 기능들에 액세스를 제공한다.

[0012] 이러한 파일 또는 데이터 공유 서비스들(즉, 클라우드 스토리지 및 개인 네트워크 스토리지)의 경우, 한가지 문제는 파일을 저장하는 방법이다. 파일을 저장하는 한가지 방법은 파일을 지속적(persistent)(예를 들어, 전자적 또는 전자-기계적) 스토리지 디바이스에 2진 인코딩 포맷으로 기입하는 것이다. 파일 포맷은 2진 파일에 내포된

데이터를 해석하는 방법을 특정하는데 사용된다. 다시 말해, 파일 포맷은 2진 데이터 1들과 0들을 인간들에게 유용한 정보로 만드는 방법을 특정하는데 사용된다. 파일을 해석하는 방법을 결정할 적어도 3개의 구현들이 본원에 기술된다. 파일의 타입을 식별하는 한가지 방법은 포맷에 관한 정보를 파일 자체에 저장하는 것이다. 이러한 정보는 파일 내의 특정 위치들에, 예를 들어, 파일 헤더에 내포된 메타데이터에 배치된다. 용어 헤더는 반드시 파일의 시작 부분을 의미하는 것은 아니다. 파일 헤더는 상기 파일에서의 다른 곳에 있을 수 있다. 파일 타입 정보를 포함하는 헤더는 인간이 판독가능하거나 또는 2진수일 수 있다. 대안적인 구현에서, 파일 포맷에 관한 정보를 포함하는 파일 메타데이터는 파일 자신에 저장되지 않고, 개별 데이터베이스에 저장된다.

[0013]

MIME는 파일 타입들을 식별하는 방법을 특정하는 다수의 가능한 표준들 중 단지 하나의 표준이다. 인터넷 미디어 타입으로도 알려진 MIME는 표준화된 방식으로 2진 파일 인코딩들을 특정하는 매커니즘을 기술하는 IETF 표준이다. MIME의 양상들은 몇몇의 IETF 표준 문서들, 예를 들어, RFC2045, RFC2046, RFC4288, 및 다른 것들에서 기술된다. MIME의 경우, 2진 파일 타입을 기술하는 스트링(string)은 보통 "MIME 타입"으로 지칭된다. 2진 파일들의 경우, 파일에 있는 2진 데이터가 어떻게 해석되어야만 하는지를 결정하는 잘 정의된 인코딩 방식이 사용된다. 예를 들어, JPEG로 알려진 이미지 인코딩 표준은 이미지 파일에서의 2진 데이터와 그 이미지의 비주얼 렌더링(visual rendering) 사이의 맵핑을 기술한다. JPEG용 MIME는 image/jpeg 이다. 몇 가지 다른 MIME 타입들은 공통 표준화된 파일 타입들, 예를 들어, MPEG 오디오 레이어(audio layer) III 또는 MP3로 알려진 음악 파일 타입, MPEG-4 Part 14 또는 MP4로 알려진 음악 파일 타입, 그래픽 인터체인지 포맷(Graphics Interchange Format) 또는 GIF로 알려진 이미지 파일, 포터블 네트워크 그래픽(portable network graphic) 또는 PNG로 알려진 그래픽 파일 타입, 태그된 이미지 파일 포맷(tagged image file format) 또는 TIFF로 알려진 이미지 파일 타입, 일반 텍스트(plain text), 하이퍼텍스트 마크업 언어(hypertext markup language) 또는 HTML로 알려진 포맷의 파일, 확장성 마크업 언어(extensible markup language) 또는 XML로 알려진 포맷의 파일, 포터블 문서 포맷(portable document format) 또는 PDF로 알려진 파일 포맷, 등을 위해 존재한다.

[0014]

몇몇 클라우드 또는 네트워크 스토리지 시스템들은 사용자들로 하여금 오직 웹 브라우저만을 사용하여, 소프트웨어 프로그램의 풀 버전(full version)이 그들의 로컬 머신(local machine) 상에 설치되게 할 필요없이, 네트워크 스토리지 상에 저장된 문서들과 인터랙트하도록 어플리케이션들을 제공한다. 본원에 사용된 바와 같이, "어플리케이션"은 API들 또는 어플리케이션 프로그래밍 인터페이스(Application Programming Interface)들을 통해 스토리지 시스템과 통합되는 소프트웨어 프로그램을 포함할 수 있다. 예를 들어, 몇몇 네트워크 스토리지 서비스들은 문서들로 하여금 네트워크 스토리지 서비스에 의해 제공된 워드 프로세서를 이용하여 오픈되거나 편집되게 한다. 사용자는 그의 하드 드라이브 상에서 워드 프로세서의 통상적인 풀 버전을 소유하거나 설치할 필요가 없고, 대신에 오직 그의 브라우저만을 사용하여 파일들을 편집할 수 있다.

[0015]

이러한 어플리케이션들은 전형적으로 네트워크 스토리지 서비스 자체에 의해 제공되는 반면에, 본원에 기술된 다양한 구현들은 제3의 개발자들(third-part developers)로 하여금 그들의 어플리케이션을 스토리지 서비스와 통합시키도록 할 수 있거나, 그들의 어플리케이션들로 하여금 네트워크 스토리지 서비스와 함께 사용되도록 할 수 있다. 본원에 사용된 바와 같이, "제3의 개발자"는 스토리지 서비스의 소유자 또는 운영자와 동일하지 않은 소프트웨어의 개발자 또는 공급 업체를 포함할 수 있다.

[0016]

따라서, 이러한 파일 또는 데이터 네트워크 스토리지 서비스들의 기술적 문제는 어떤 소프트웨어 어플리케이션들이 상기 서비스로 저장된 상이한 파일 타입들의 파일들/데이터들을 오픈 또는 생성할 수 있는지를 관리하는 방법이다. 이러한 소프트웨어 어플리케이션들은 상기 시스템에서 특정 아이템들을 오픈, 프리뷰(preview), 뷰(view), 편집, 렌더 또는 프로세스할 수 있다. 이러한 성능(capacity)은 특정 아이템의 관련된 파일 타입의 매칭 또는 대응성(correspondence) 및 소프트웨어 어플리케이션과 관련된 하나 이상의 파일 타입들(예를 들어, MIME 또는 다른 파일 타입 식별 시스템)의 리스트에서 상기 관련된 파일 타입의 프리젠텔에 근거하여 결정될 수 있다. 그러한 매칭은 소프트웨어 어플리케이션이 매칭 타입을 갖는 임의의 파일을 오픈, 프리뷰, 렌더 또는 프로세스하는데 사용될 수 있다는 것을 나타낼 수 있다. 보안상의 이유로, 제3의 어플리케이션 제공자들이 그들의 어플리케이션이 다루도록 하여되지 않는 타입들을 갖는 사용자 파일들을 액세스하는 것을 방해하는 것이 일반적인 것으로 바람직하다.

[0017]

제3의 어플리케이션들

- [0018] 특정 파일 타입에 대한, 파일 또는 데이터 스토리지 또는 공유 서비스 또는 시스템일 수 있는 스토리지 서비스는 그 타입의 파일들을 오픈할 수 있는 다중 소프트웨어 어플리케이션이 있을 수 있다. 이러한 소프트웨어 어플리케이션들은 예를 들어, 스토리지 서비스 소유자 또는 운영자, 제 3자들, 또는 둘 다에 의해 제작되거나 제공되는 어플리케이션들의 조합을 포함할 수 있다. 상기 어플리케이션들은 "유니버설 디폴트(universal default)" 소프트웨어 어플리케이션을 포함할 수 있는데, 이것은 상시 설치되고 파일 또는 데이터 공유 서비스 시스템에서 모든 타입들의 파일들을 몇몇 기본적 기능 레벨(basic level)로 오픈 또는 프리뷰할 수 있다. 예를 들어, 포터블 네트워크 그래픽 또는 PNG 이미지 파일을 고려해보자. 상기 시스템 어플리케이션은 본 파일을 프리뷰할 수 있다. 그러나, 제 3자 어플리케이션들은 더 풍부한 프리뷰 경험을 허가할 수 있다. 몇몇의 어플리케이션들은 심지어 편집 경험을 허가할 수 있다.
- [0019] 도 1은 어플리케이션과 파일 타입을 관련짓는 기술된 기법들을 구현할 수 있는 예시적인 네트워크 스토리지 서비스(100)를 도시한다. 스토리지 서비스(100)는 제 3의 개발자로 하여금 스토리지 서비스의 사용자들에 의한 사용을 위해 웹-기반 소프트웨어 어플리케이션들을 지원하게 한다.
- [0020] 상기 어플리케이션들은 일반적으로 하나 이상의 파일 타입들의 파일을 오픈, 프로세스, 뷰, 또는 편집할 수 있다. 제 3의 개발자들(102 및 104)은 클라이언트 디바이스들(106 및 108)을 이용하여 스토리지 서비스(100) 내로 그들의 어플리케이션들을 통합 또는 등록할 수 있는데, 이를 위해 이러한 통합을 위해 제공되고 서비스(100)의 개발 환경 관리자(110)에 의해 공급되는 인터페이스를 사용한다. 등록 또는 통합은 구성 정보를 서비스(100)로, 예를 들어, 서비스(100)에 의해 유지되거나 액세스 가능한 제3자 어플리케이션 데이터베이스(112) 내로 공급하는 것을 포함한다. 상기 구성 정보는 예를 들어, 어플리케이션이 오픈하려고 하는 파일 타입들의 리스트를 포함한다. 따라서, 개발자가 제3자 어플리케이션을 등록할 때, 그들은 또한 어플리케이션이 동작 가능한 파일 타입들의 리스트를 등록한다. 몇몇 구현들에서, 개발자는 MIME 타입, 파일 확장자 또는 이 둘 다를 등록함으로써 상기 리스트 등록을 실행한다. 이것들은 어플리케이션-대 MIME 타입(또는 파일 타입) 맵핑들, 전형적으로 일-대-다의 관계이다.
- [0021] 일부 제3자 어플리케이션들은 다수의 파일 타입들 또는 MIME 타입들 상에서 동작할 수 있다. 예를 들어, 포토 편집자는 image/jpeg, image/png, 또는 다른 파일 타입들 상에서 동작할 수 있다. 반면에, 제3자 프로젝트 관리 어플리케이션은 아마도 오직 사용자 지정 파일 확장자 또는 MIME 타입과 관련된 파일들 상에서만 동작할 수 있을 것이다. 이러한 파일들은 예를 들어, 제3자 어플리케이션으로 또는 제3자 어플리케이션을 위해 만들어진 파일들일 수 있다.
- [0022] 모든 제3자 어플리케이션은 전형적으로 그 자신의 파일 타입 또는 제3자 어플리케이션의 고유 파일 포맷에 관련된 MIME 타입을 수신한다. 일부의 구현들에서, 제3자 개발자들이 그들의 어플리케이션들이 처리할 수 있는 MIME 타입들과 파일 확장자들 둘 다를 등록한다. 유사하게, 이러한 제3자 어플리케이션들 중 하나로 생성된 문서들 또는 파일들은 파일을 생성하기 위해 사용된 어플리케이션의 파일 타입 또는 MIME 타입을 가질 것이다. MIME 타입들의 디폴트 파일 타입들뿐만 아니라 제3자 어플리케이션들은 다양한 파일 타입들 또는 MIME 타입들 중 어느 하나로 익스포트(export)를 허용할 수 있다. 만약 사용자(120, 121, 또는 122)가 어플리케이션의 상이한 타입으로 파일을 사용하고 싶다면, 이것은 유용할 수 있다.
- [0023] 개발자들(102 및 104)은 또한 그들의 클라이언트 디바이스들(106 및 108)을 사용하여, 예를 들어, 개발 환경 관리자(110)를 통해, 그들의 어플리케이션들(114 및 116)을 어플리케이션 스토어(118)로 업로드한다. 어플리케이션 스토어(118)는 서비스(100)에 의해 호스트(host)되고, 관리되고, 제어될 수 있거나 또는 독립적일 수 있다. 본원에 사용된 바와 같이, 용어 "어플리케이션 스토어"는 예를 들어, 사용자에게 시장 또는 점포를 제공하는 인터페이스뿐만 아니라 데이터베이스도 포함할 수 있다. 어플리케이션들(114 및 116)은 이후 광고될 수 있거나 그 반대로 사용자가 이용가능하게 하고 사용자에게 의해 사용될 수 있다. 어플리케이션들은 무료 또는 유료일 수 있다.
- [0024] **시스템 개요**
- [0025] 사용자(120, 121, 122 등)들은 다양한 디스커버리 채널(discovery channel)들, 예를 들어, 어플리케이션 스토어(118), 광고, 개발자 웹 사이트, 또는 다른 사용자로부터의 참조(reference)를 통해 어플리케이션(114 또는 116)의 존재를 발견 또는 학습할 수 있다. 사용자(120, 121, 또는 122)는 이러한 목적을 위해 의도된 매커니즘

을 통해, 예를 들어, 상기 어플리케이션과 서비스(100)의 사용자의 뷰와 통합시키는 사용자 환경 관리자(128)를 통해 그의 디바이스(124, 125, 또는 126) 상에 어플리케이션(114)을 설치한다.

[0026] 어플리케이션을 설치하는 것은 다수의 파일들이 사용자의 하드 드라이브 또는 솔리드 스테이트 드라이브(solid state drive)로 복사되고 소프트웨어가 삭제될때까지 그곳에 상주하는 소프트웨어의 전형적인 설치와 혼동되지 않아야한다. 대신에, 설치된 어플리케이션은 설치된(등록된) 어플리케이션들과 사용자들을 연관짓는 데이터베이스에 등록된다. 설치된 또는 등록된 어플리케이션들의 리스트는 사용자들이 서비스(100)를 사용할 때 그들에게 보인다.

[0027] 제3자 어플리케이션을 "설치"하는 행위는 사용자와 제3자 어플리케이션 사이의 관계를 등록하는 것을 구성하고, 제3자 어플리케이션으로 하여금 서비스(100) 상에서 사용자의 자원들을 액세스하게 하는 하나의 단계로써 역할을 한다.

[0028] 예를 들어, 설치된 어플리케이션(114)은 사용자의 브라우저 또는 다른 사용자 인터페이스에 의해 제어될 수 있고, 사용자의 브라우저(예를 들어, Java™ 또는 다른 플랫폼-독립 가상 머신(platform-independent virtual machine)상에서)에 의해 구동되는 일시적인 스크립트(temporary script) 또는 애플릿(applet)일 수 있으며, 사용자의 브라우저로 플러그-인(plug in)일 수 있고, 원격 컴퓨터상에 존재하며 사용자의 클라이언트 기계(124, 125, 및 126)에 의해 원격 제어될 수 있고, 또는 그와 달리 액세스되거나 제어될 수 있다.

[0029] 어플리케이션(114)은 사용자가 서비스(100)에서 해당되는 타입들의 파일들을 오픈하는데 사용될 수 있다. 사용자(120, 121, 및 122)들은 서비스(100) 내의 데이터 스토리지(130)내의 또는 서비스(100)의 외부의 제3자 데이터(132)내의 데이터들을 전형적으로 제어, 프로세스, 또는 액세스할 것이고, 통신 네트워크(140), 예를 들어, 인터넷 또는 다른 네트워크를 통해 그들의 각 클라이언트 디바이스(124, 125 및 126)들을 이용하여 이러한 파일들을 액세스할 수 있다. 데이터 스토리지(130)내의 파일들은 MIME 타입들 또는 확장자 또는 둘 다 와 같은 파일 타입들을 가진다.

[0030] 일부 구현들에서, 서비스(100)는 파일을 해석하는 방법을 결정하는 적어도 3개의 방법들 중 하나 이상을 사용한다. 상기 3개의 방법은:

[0031] 1. 헤더의 정보. 본 구현에서, 서비스는 헤더의 위치를 찾고, 그것을 파스(parse)하고, 및 그것의 구조와 포맷을 알려진 포맷들의 데이터베이스와 비교하기 위해 파일의 콘텐츠를 조사한다. 예를 들어, 실행가능(executable) 또는 .exe 파일들은 그들의 실행가능성을 표시하는 특정 코드로 시작한다.

[0032] 2. 파일 확장자, 파일의 이름의 특성.

[0033] 3. MIME 타입, (헤더에 나타나지 않으면) 파일의 콘텐츠의 직접적인 부분이 아닌 또는 파일명의 부분이 아닌 외부 메타데이터를 수반하는 부분.

[0034] 클라이언트 디바이스(124, 125 및 126)들은 워크스테이션들, 랩탑(laptop) 컴퓨터들, 노트북 컴퓨터들, PDA(personal digital assistant)들, 태블릿 컴퓨터들, 모바일 폰들, 스마트 티비(TV)들, 스마트 디비디(DVD) 또는 다른 미디어 플레이어들, 스마트 가전제품들, 또는 네트워킹될 수 있는 인간이 이용가능한 입력 및 출력 인터페이스들을 구비하며, 파일들을 오픈할 수 있는 디바이스들의 다른 타입들일 수 있다.

[0035] 서비스(100)는 또한 사용자 프리퍼런스 스토어 또는 데이터베이스(142)에서 사용자 프리퍼런스들을 저장할 수 있다. 서비스(100)는 또한 설치된 어플리케이션 리스트들(150)에서 각 사용자(120, 121 및 122)들을 위한 설치된 어플리케이션들의 리스트를 유지할 수 있다. 만약 어플리케이션 스토어(118)가 서비스(100)에 의해 유지된다면, 다수의 사용자들을 위한 설치된 어플리케이션들은 어플리케이션 스토어(118)에서 전형적으로 유지된다. 대안적으로, 어플리케이션들은 서비스(100)에 의해 유지될 수 있다. 대안적으로, 어플리케이션들은 서비스(100)의 외부에 또는 내부 및 외부 둘 다의 조합에 의해 유지될 수 있다. 어플리케이션들의 사례들은 서비스(100)로부터 또는 서비스(100)에 의해 사용자(120, 121 및 122)들에게 효율적으로 사용된다.

[0036] 사용자 인터페이스

[0037] 도 2는 기술된 기법들의 적어도 하나의 구현에 따른 예시적인 사용자 인터페이스(200)을 도시한다. 설치된 외부의 제3자 어플리케이션(114)을 이용하여 서비스(100)에서 파일을 오픈하기 위해, 사용자(102, 121, 및 122)는 우선 파일을 선택한다. 파일 동작 다이얼로그, 예를 들어, 사용자 인터페이스(200)의 다이얼로그(202)로부터,

사용자(120, 121 및 122)는 사용자가 파일을, 예를 들어, "오픈(Open)"(204) 또는 "오픈 위드(Open with)"(206) 메뉴 옵션으로부터 어떻게 오픈하고 싶은지를 선택한다. 그리고 나서, 파일을 오픈하는 것이 가능한 해당되는 어플리케이션(210, 212, 214, 및 216)들의 리스트(208)는 사용자 인터페이스(200)의 하위 메뉴, 보조 메뉴 또는 하위-다이얼로그(220)에 렌더링된다. 오픈 위드" 메뉴 옵션(206)은 사용자(120, 121, 또는 122)들 중 한 명이 "오픈" 메뉴 옵션(206) 상에서 오른쪽 클릭하면, 나타날 수 있고, 또는 오픈 옵션(204)와 함께 다이얼로그(202)에 상시 나타날 수 있다.

[0038] 대안적으로, 사용자 인터페이스(200)는 특정 파일을 선택한 결과로서, 사용자 인터페이스의 어딘가에 나타나는 클릭 가능한 링크들 또는 버튼들의 리스트를 디스플레이할 수 있다.

[0039] 이러한 어플리케이션의 선택 이후에, 선택된 어플리케이션, 예를 들어, 어플리케이션(114)은 파일을 오픈하는데 사용된다.

[0040] 그러한 선택을 하지 않는다면, 사용할 어플리케이션 리스트를 먼저 제시하지 않는 어떤 디폴트 오픈 플로우(default open flow)를 호출함으로써, 예를 들어, 어플리케이션(116)일 수 있는 디폴트 어플리케이션이 파일을 오픈 또는 프리뷰하는데 사용될 것이다.

[0041] 디폴트 어플리케이션

[0042] 만일, 예를 들어, 사용자가 "오픈 위드" 옵션(206) 대신에 "오픈" 메뉴 옵션(204)을 선택하면, 디폴트 어플리케이션(116)이 사용될 수 있다. 디폴트 어플리케이션은 파일을 오픈하기 위해 존재할 수 있거나 존재하지 않을 수 있다. 시스템(100)에 내장되어(예를 들어, 다수의 사용자가 어플리케이션을 설치해야함 없이 다수의 사용자들이 이용가능하며, 전형적으로 제3자 어플리케이션이 아님)있고, 많은 타입들의 파일들을 오픈할 수 있는 시스템 디폴트 어플리케이션(116)이 있을 수 있다. 일부 구현들에서, 디폴트 어플리케이션(116)은 결과적인 프리뷰의 효과가 디폴트 어플리케이션(116)이 파일을 얼마나 잘 렌더링할 수 있느냐에 달려있겠지만, 모든 파일 타입들을 오픈할 수 있다.

[0043] 사용자 어플리케이션 프리퍼런스

[0044] 사용자(120)는 시스템 디폴트 어플리케이션 대신에 또는 다수의 가능 어플리케이션들의 리스트로부터 선택해야 하는 대신에 지정된 타입의 파일들을 오픈하기 위해 디폴트에 의한 외부 제3자 어플리케이션(114)을 사용하고 싶을 수 있다. 이러한 프리퍼런스는 단지 사용자(120)를 위한, 디폴트 "파일 타입으로 적용(application to file type)" 맵핑이다. 제3자 어플리케이션(114)은, 예를 들어, 새롭게 이용가능하게 되며, 사용자(120)는 디폴트 어플리케이션(116)보다 그것을 선호할 수 있다. 대안적으로, 사용자(120)는 디폴트 어플리케이션(116)보다 제3자 어플리케이션(114)을 선호할 수 있다. 왜냐하면, 제3자 어플리케이션(114)은 더 풍부한 프리뷰 또는 편집 경험을 제공하기 때문이다.

[0045] 사용자(102, 121, 및 122)는 디폴트에 의해 그들이 특정 파일 타입(예를 들어, MIME 타입)의 모든 파일들을 오픈하는데 사용하려는 어플리케이션은 어느것이든 선택할 수 있다.

[0046] 하나의 특정 예에서, 상기 오픈 메뉴 선택(choice)(204) 상의 왼쪽-클릭(디폴트 오픈)은 상기 지정된 어플리케이션을 호출할 것이다. 일부 구현들에서, 파일을 왼쪽-클릭하는 것은 "오픈" 옵션을 선택할 필요 없이, 상기 파일을 선택하여 상기 디폴트 오픈 액션(action)을 호출한다. "오픈 위드" 하위 메뉴(220)를 액세스하기 위해 오픈 메뉴 선택(204)을 오른쪽 클릭할 필요는 없다. 그러나, 오른쪽-클릭 및 "오픈 위드" 메뉴 선택(206)은 디폴트 설정을 중단시키고 상이한 어플리케이션을 선택하는데 여전히 사용될 수 있다.

[0047] 예를 들어, 사용자(120)가 2개의 개별 포토 에디터(editor) 어플리케이션(116 및 114)들을 설치했을 때를 고려해보자. 그들은 디폴트에 의한 어플리케이션(116)을 JPEG 이미지들과 연결짓는 것을 선택한다. 상기 사용자가 JPEG 이미지를 선택할 때, 그것은 그들의 프리퍼런스에 따라 어플리케이션(116)으로 오픈된다. 그러나, 사용자가 동일한 파일을 오른쪽-클릭하고, 어플리케이션(114)으로 오픈하거나 그들의 디폴트 선택을 중단할 수 있다.

- [0048] 하나의 특정 예에서, 디폴트 사용자 어플리케이션 프리퍼런스는 시스템 디폴트 어플리케이션 또는 뷰어(viewer)가 파일을 오픈하는데 사용되고 있는 경우를 제외하고, 하위메뉴(220), 예를 들어, 어플리케이션(210)의 위치(222)에 표시된다. 상기 어플리케이션의 명칭은 다이얼로그(202)에서 오픈 아이템 바로 옆에 "오픈 위드" 다음 어플리케이션 명칭으로 나타날 것이다. 만일 디폴트가 시스템 디폴트이면, 다이얼로그는 단순히 "오픈"을 말한다.
- [0049] 사용자 어플리케이션 프리퍼런스는, 서비스(100)에, 예를 들어, 사용자 데이터 스토리지(130) 내의 사용자-특정 기록에 저장될 수 있다. 이 프리퍼런스는 파일 관리 시스템의 뷰가 사용자 환경 관리자(128)에 의해 사용자를 위해 렌더링될 때, 서비스(100) 또는 클라이언트 디바이스(124, 125, 및 126)에 의해 사용된다. 어플리케이션과 파일 타입 사이에서 서비스(100)에 의해 연관성이 생성됨으로써, 파일(예를 들어, MIME 타입)의 타입 및 선호되는 어플리케이션 사이의 맵핑을 일으킨다. 만일 파일 타입 또는 MIME 타입을 위한 엔트리(entry)가 없다면, 시스템 디폴트 어플리케이션 또는 뷰어가 일부 구현들에서 사용된다.
- [0050] 사용자들(120, 121, 또는 122) 중 한 명이 서비스(100)를 사용할 때, 상기 프리퍼런스는 시스템(100)으로부터 사용자 클라이언트 머신(124, 125, 또는 126)으로 옮겨질 수 있고, 예를 들어, 어플리케이션 내부 또는 웹 브라우저 또는 다른 인터페이스 내부의 캐시 데이터 구조 근처에 저장될 수 있다. 이러한 디폴트 프리퍼런스는 임의의 디폴트 프리퍼런스가 사용자에게 의해 미리 설정되거나 표현되었는지를 결정하기 위해 고려되어 지거나, 사용자가 서버(100)상에서 파일을 오픈함에 관심을 표현할 때에 참조된다. 이러한 디폴트 프리퍼런스 조사 결과는 어떤 어플리케이션이 상기 지정된 파일을 오픈하는데 사용되는지를 결정할 것이다.
- [0051] **아이콘**
- [0052] 일부 예들에서, 각 어플리케이션(210, 212, 214, 및 216)들은 그들과 관련되고 그들 바로 옆에 보여지는 아이콘(224, 226, 228 및 230)들을 가진다.
- [0053] **관리 어플리케이션**
- [0054]
- [0055] "관리 어플리케이션(Manage apps)" 메뉴 옵션(232)은 "오픈 위드" 하위 메뉴 또는 개별 설정 메뉴에서 나타날 수 있다. 요망되면, 하나 이상의 메뉴 아이템 세퍼레이터(separator)(236)가 포함될 수 있다. 일부 예들에서, 이 메뉴 아이템은 디폴트 "오픈 위드"가 활성화된 경우에만 나타난다.
- [0056] 도 3은 상기 기술된 기법들의 적어도 하나의 구현에 따른 사용자 인터페이스(200)의 변형을 도시한다. "관리 어플리케이션" 메뉴 옵션(232)이 클릭되거나 작동될 때, "관리 어플리케이션" 다이얼로그(234)는 시작된다. 다이얼로그(234)를 이용하여, 사용자(120, 121, 및 122)는 디폴트 "오픈 위드" 맵핑을 특정할 수 있다.
- [0057] 일부 예들에서, 사용자들 중 한 명은 관리 어플리케이션 다이얼로그 또는 기타 다이얼로그(비도시)를 사용하여, 어플리케이션이 오픈할 수 있는 파일들을 오픈하도록 디폴트로서 어플리케이션을 사용하는 것을 선택할 수 있다. 다이얼로그 상에서 보여지는 실제 텍스트(actual text)는, 예를 들어, "디폴트에 로서 사용(use by default)" 훨씬 더 간단할 수 있다. 이 텍스트는, 예를 들어, 어플리케이션(238, 240, 242, 또는 244)이 선택된 후에 나타날 것이다.
- [0058] 상기 다이얼로그(234)는 또한 클릭되거나 동작될 때, 추가 어플리케이션(248, 250, 252, 254, 및 256)들로하여금 다이얼로그(258) 상의 어플리케이션(238, 240, 242, 및 244)들 옆에 드러나도록 하는 "모어(more)" 메뉴 옵션(246)을 포함한다. 이러한 어플리케이션들 중 임의의 것이 디폴트 어플리케이션으로서 선택될 수 있다. 메뉴 옵션(260)은 클릭되거나 동작될 때, 훨씬 더 많은 어플리케이션들이 드러나게 되는 다이얼로그(258) 상에서 제공될 수 있다.
- [0059] **디폴트 오픈 충돌(conflict) 해결**

- [0060] 일부 예들은 사용자가 "디폴트에 의해" 사용하고자 하는 어플리케이션 ID들의 간단한 리스트를 사용한다. 특별한 주의를 요구하는 특정 시나리오(scenario)는 파일 타입들 또는 MIME 타입들을 오버랩핑(overlapping)하는 경우: 2개의 어플리케이션들이 동일한 파일 타입 또는 MIME 타입을 위한 "디폴트"로서 사용자에게 의해 선택될 수 있는 경우이다. 예를 들어, 어플리케이션들 포토(Photo) 및 픽스(Pix)는 디폴트로서 확인될 수 있고 이미지들, 예를 들어, MIME 타입 "image/jpeg"을 가진 파일들을 "획득할 가능성"이 있다.
- [0061] 오직 하나의 어플리케이션만이 이러한 경우에서 남을 수 있다. 오히려 알파벳 순서로 또는 임의의 결정을 내리기보다, 상기 서비스는 디폴트 어플리케이션 ID들의 순서 리스트를 유지하고, 상기 리스트에 나타나는 적어도 하나가 남는다. 도 2 내지 3의 사용자 인터페이스에서, 사용자(120, 121, 또는 122)가 디폴트 어플리케이션들을 선택하는 순서는 어플리케이션들이 상기 리스트에 나타나는 순서를 표시한다. 그러므로, 사용자는 어플리케이션을 선택하고 그것을 선택하지 않음으로써 어플리케이션을 리스트의 상부로 "이동(bump)"시킬 수 있다.
- [0062] 대안적으로, 이 리스트는 "상부로 이동" 동작이 존재하는 큐(queue)로서 유지될 수 있거나, 또는 완전히 순서 가능한 리스트일 수 있다.
- [0063] **사용 사례**
- [0064] 사용자(120)가 JPEG 타입의 이미지 파일(160)(도 1 참조)을 서비스(100)로 업로드한다고 가정해보자. 사용자(120)는 뷰-전용 컨텍스트(context)에서 상기 이미지를 렌더링 또는 뷰잉하기 위해 내장된 뷰어 또는 프리뷰어 어플리케이션(210)(도 2 참조)을 사용할 수 있지만, 그것을 편집할 수는 없다. 사용자(120)는 이미지들의 편집을 가능하게 하는 향상된 기능을 구비하고 싶어한다. 사용자는 2개의 이러한 포토 어플리케이션(212 및 214)들을 검색하여, 발견한 후에 설치한다(추가한다). 이 때, 어플리케이션(212)도 어플리케이션(214)도 디폴트로서 설정되지 않았으므로, 일부 구현들에서, 어플리케이션(212 및 214)들은 "오픈 위드"를 통해 액세스만 가능하다. 포토 어플리케이션(212 및 214)들은 둘 다 MIME 타입 JPEG의 파일들을 편집할 수 있다. 디폴트 뷰어 어플리케이션(210)을 사용하는 대신에, 사용자(120)는 이제 JPEG 포토 파일(160)을 오픈하기 위해 어플리케이션들(212 및 214) 중 하나를 사용하고 싶어한다. 사용자는 파일(160)을 선택하고, 이 파일(160)은 그 다음에 사용자 환경 관리자(128)에 의해 그들의 컴퓨팅 디바이스(124) 상에 선택된 파일, 즉, JPEG의 타입의 파일들을 오픈할 수 있는 어플리케이션들의 리스트와 함께 제시된다. 상기 리스트는 내장된 시스템 디폴트 뷰어 어플리케이션(210)과, JPEG 타입의 파일들을 오픈할 수 있는 최근에 설치된 어플리케이션(212 및 214)들(도 2 참조)을 포함한다. 상기 리스트는 임의의 비주얼 매커니즘을 통해, 리스트에서 어떤 어플리케이션이 현재 텍스트, 홀딩(holding), 체크마크(checkmark), 또는 상기 리스트의 순서를 포함하는 디폴트인지를 표시한다. 예를 들어, 디폴트 어플리케이션은 리스트의 상부에 나타날 수 있다. 사용자는 리스트에서 그들이 사용하고자 하는 어플리케이션이 어떤 어플리케이션인지를 선택한다. 그들은 첫 번째로 최근에 설치된 어플리케이션(212)을 선택한다. 상기 이미지는 어플리케이션(212)으로 오픈된다.
- [0065] 사용자(120)는 이제 두 번째 어플리케이션(214)을 시도하려고 한다. 그들은 상기 플로우(flow)를 반복하고, 이번에는 두 번째로 최근에 설치된 어플리케이션(214)을 선택한다. 두 번째 어플리케이션(214)은 이제 JPEG 포토를 오픈하는데 사용된다. 두 번째 어플리케이션(214)에 만족하여, 사용자(120)는 JPEG 타입의 파일들을 오픈하기 위해 항상 디폴트로서 두 번째로 최근에 설치된 어플리케이션(214)을 사용하려고 한다. 그들은 프롬프트(prompt)되려고 하지 않는다. 사용자(120)는 임의의 사용자 인터페이스 컴포넌트, 예를 들어 관리 어플리케이션 관리 다이얼로그에 따라 이 프리퍼런스를 설정하고, 프리퍼런스는 서비스(100), 예를 들어, 사용자 프리퍼런스 스토어(142)에 기록된다. 후에, 사용자(120)가 첫 번째 포토(160)와 다른 두 번째 JPEG 포토(162)를 오픈하려고 한다. 그들은 상기 디폴트 오픈 프로세스를 호출한다. 그들은 이전과 같이, 파일을 오픈할 수 있는 어플리케이션들의 리스트로 프롬프트되지 않는다. 그들이 디폴트 오픈 프리퍼런스를 설정했기 때문에, JPEG 타입의 어플리케이션(214)으로의 맵핑이 존재한다. 그러므로, 이전에 저장된 프리퍼런스가 동적으로 참조되었기 때문에, 어플리케이션(214)이 JPEG 포토(162)를 오픈하는데 자동으로 사용된다. 사용자는 하나의 특정 목적을 위해, 일회성으로 어플리케이션(212)에서 파일(162)을 오픈하고자 함을 깨닫는다. 그러나, 그들은 더 일반적으로 그들의 이전에 지정한 디폴트 프리퍼런스를 방해하고자 하지 않는다. 따라서, 그들은 일회성으로(예를 들어, 문서 리스트에서 파일 엔트리를 오른쪽 클릭하거나 또는 "오픈" 메뉴 옵션(204)을 오른쪽 클릭함으로써) 포토를 오픈할 수 있는 적용가능한 어플리케이션들의 리스트로 프롬프트되도록 선택할 수 있다. 이런 하나의 사례에서, 그들은 첫 번째 설치된 어플리케이션(212)을 사용하는 것을 선택하고, 따라서 파일(162)은 어플리케이션(212)에서 오픈한

다.

- [0066] 사용자(120)는 후에 어플리케이션(212)을 삭제한다. 사용자가 JPEG 포트를 위한 비-디폴트 오픈 플로우를 실행하고, 모든 적용가능한 설치된 어플리케이션(210 및 214)들의 리스트로 프롬프트되고자 할지라도, 상기 리스트는 삭제된 첫 번째 설치된 어플리케이션(212)을 내포하지 않을 것이다. 현재 디폴트 프리퍼런스인 어플리케이션이 삭제될 때, 이 (디폴트이었을 수 있지만 겹쳐쓰여졌던)디폴트 프리퍼런스는 이전에 등록된 디폴트 어플리케이션으로 되돌아 갈 것이다. 만일 이러한 어플리케이션이 없으면, 디폴트 프리퍼런스는 모든 파일들을 프리뷰할 수 있는 시스템 디폴트 어플리케이션(210)으로 되돌아갈 것이다.
- [0067] 도 4는 더 일반화된 구현들에 따른 예시적인 워크플로우(workflow)(400)를 도시한다. 402에서, 서비스(100)는 제3자 어플리케이션 및 등록 정보를 수신한다. 404에서, 서비스(100)는 어플리케이션이 오픈할 수 있는 파일 타입들(예를 들어, MIME 타입들 또는 다른 파일 타입들)로 제3자 어플리케이션을 맵핑하는데, 여기서 파일 타입 데이터는 파일들 그 자체(예를 들어, 헤더 또는 메타데이터)에 저장된다. 406에서, 서비스(100)는 어플리케이션으로 하여금 설치할 사용자에게 이용가능하게(예를 들어, 그들의 "오픈 위드" 리스트들에서 보이도록) 한다. 408에서, "오픈" 요청에 응답하여, 서비스(100)는 (서비스(100)에 의해 제공되는 어플리케이션을 이용하여) 만일 사용자 디폴트 어플리케이션이 사용자에게 의해 선택되었거나 설정되면, 사용자 디폴트 어플리케이션으로 파일을 오픈한다. 그리고 만일 사용자 디폴트 어플리케이션이 사용자에게 의해 선택되었거나 설정되지 않으면, 서비스(100)는 시스템 디폴트 어플리케이션을 이용한다. 410에서, "오픈 위드" 요청에 응답하여, 서비스(100)는 사용자에게 설치된 제3자 및 시스템 어플리케이션들의 리스트를 제시하고 사용자가 리스트로부터 선택한 어플리케이션으로 파일을 오픈한다.
- [0068] 도 5는 더 일반화된 구현들에 따른 간단한 워크플로우(500)를 도시한다. 502에서, 소프트웨어 어플리케이션 설명들을 포함하는 소프트웨어 어플리케이션 스토어가 제공되는데, 각 소프트웨어 어플리케이션 설명은 어플리케이션 ID, 어플리케이션 타이틀, 및 지원된 MIME 타입들 또는 파일 타입들을 가진다. 본원에 사용되는 바와 같은, 용어 "소프트웨어 어플리케이션 설명들"은 "설명"이라고 불리는 메타데이터 필드와 혼동되어서는 않된다. 소프트웨어 어플리케이션 설명들은 서비스(100)의 제공자 또는 제3의 개발자 중 한 명과 관련된다.
- [0069] 504에서, 사용자와 관련된 파일들을 포함하는 사용자 데이터 스토어가 제공되는데, 사용자 데이터 스토어(130) 내의 각 파일들은 파일 식별자, MIME-타입 식별자(또는 파일 타입 식별자), MIME-타입(또는 파일 타입)에 따라 2진 인코딩된 데이터를 가진다.
- [0070] 사용자 데이터 스토어는 소프트웨어 어플리케이션 스토어로부터 사용자 설치 소프트웨어 어플리케이션들을 식별하는 어플리케이션 ID들의 리스트를 포함한다. 사용자 데이터 스토어(130)는 특정 사용자 설치 소프트웨어 어플리케이션들의 어플리케이션 ID들을 포함하는, 파일들을 오픈하기 위해 디폴트 어플리케이션들을 특정하는 파일 액세스 사용자 프리퍼런스들을 더 포함한다.
- [0071] 506에서, 컴퓨팅 디바이스(124, 125, 또는 126)를 동작시키는 사용자(120, 121, 또는 122)로부터 통신 네트워크를 통해 서비스(100)에서 특정 요청된 파일에 대한 파일 식별자가 수신된다.
- [0072] 508에서, 상기 수신된 파일 식별자와 관련된 파일에 대한 파일 타입 식별자가 사용자 데이터 스토어(130)로부터 획득된다.
- [0073] 510에서, 소프트웨어 어플리케이션 스토어는 획득된 MIME 타입(또는 파일 타입)을 지원하는 소프트웨어 어플리케이션들에 대한 어플리케이션 ID들의 리스트를 결정하기 위해 획득된 MIME-타입(또는 파일 타입) 식별자로 문의된다.
- [0074] 512에서, 특정 요청된 파일을 오픈하는 디폴트 어플리케이션이 파일 액세스 사용자 프리퍼런스들에 내포된 어플리케이션 ID를 MIME 타입(또는 파일 타입)을 지원하는 소프트웨어 어플리케이션에 대한 선택된 어플리케이션 ID들의 리스트와 매칭함에 응답하여, 결정된다.
- [0075] 514에서, 특정 요청된 파일은 특정 요청된 파일을 오픈하도록 상기 결정된 디폴트 어플리케이션을 활용하여 사용자 데이터 스토어(130)로부터 획득된다.
- [0076] 다른 대안들

- [0077] 일부 구현들에서, 제3자 어플리케이션이 오픈할 수 있는 파일들의 타입을 표시하기 위해 MIME 타입 또는 파일 타입 대신에 파일 확장자 정보가 사용된다. 확장자와 MIME 타입은 둘 다 제3자 어플리케이션에 의해 생성된 파일상의 메타데이터로 저장된다. 일부 구현들에서, 파일 확장자 타입 및 MIME 타입 모두 사용된다. 이는 일부 파일들이 다양한 이유로 설정된 그들의 MIME 타입을 구비하지 않고, 파일 타입을 표시하는 유효 확장자를 구비하기 때문이다. 대안적으로, 일부 파일들은 그들의 파일 확장자가 사용자에게 의해 제공되게 할 수 있지만, 서비스(100)는 MIME 타입 정보가 존재하고 사용될 수 있기 때문에 여전히 그들을 이용하여 무엇을 해야할지 알고 있다.
- [0078] 제3자 개발자들은 그들이 MIME 타입들의 리스트를 등록하는 것과 같이, 또한 확장자들의 리스트를 등록한다.
- [0079] 일부 구현들에서, 두 플로우들은 독립적으로 실행되며, 결과적인 "오픈 워드" 리스트들은 2 개의 리스트들을 함께 조합(unioning)하고 사용자에게 디스플레이된 "오픈 워드" 리스트에 대한 중복들을 제거함으로써 조정된다. "조합"이 의미하는 것은 제거된 사본(duplicate)들과 함께 유니언 세트(union set), 즉, 리스트들의 결합을 취하는 것이다.
- [0080] 디폴트 "오픈 워드"는 MIME 타입 또는 확장자 기초(extension basis) 둘 중의 하나에 설정될 수 있다.
- [0081] 다른 구현들에서, 서비스(100)는 사용자로 하여금 디폴트 "오픈 워드" 프리퍼런스를 표현하게 하는 특정 매커니즘을 경험하게 함 없이, 사용자를 상시 프롬프트하는 부분적인 "오픈 워드" 기능을 제공한다.
- [0082] **예시적인 구현**
- [0083] 본원에 기술된 접근 방법들과 시스템들은 클라이언트 디바이스들 및/또는 웹-기반 스토리지 시스템과 관련된 서버 또는 서버들에 내포된 하나 이상의 프로세서들 상에서의 컴퓨터 소프트웨어, 프로그램 코드들, 및 또는 명령어들을 실행하는 머신을 통해 부분적으로 또는 전체로 사용될 수 있다.
- [0084] 본 발명의 구현들은 머신상의 방법, 상기 머신의 일부로서 또는 머신에 관한 시스템 또는 장치, 상기 머신들 중 하나 이상에서 실행되는 컴퓨터 판독가능 매체를 구현하는 컴퓨터 프로그램 제품을 포함할 수 있다. 하나 이상의 프로세서들은 서버, 클라이언트, 네트워크 인프라구조, 모바일 컴퓨팅 플랫폼, 고정(stationary) 컴퓨팅 플랫폼, 또는 다른 컴퓨팅 플랫폼 중 일부일 수 있다.
- [0085] 프로세서는 프로그램 명령어들, 코드들, 2진 명령어들, 등을 실행할 수 있는 컴퓨터의 또는 프로세싱 디바이스의 어떤 종류일 수 있다. 이 프로세서는 본원 및 그 밖의 곳에서 기술된 바와 같이 방법들, 코드들, 명령어들, 및 프로그램들을 저장하는 메모리를 포함할 수 있다. 이 프로세서는 본원 또는 다른 곳에서 기술된 바와 같이 방법들, 코드들, 명령어들을 저장하는 인터페이스를 통해 스토리지 매체를 액세스할 수 있다. 방법들, 프로그램들, 코드들, 프로그램 명령어들, 또는 컴퓨팅 또는 프로세싱 디바이스에 의해 실행될 수 있는 명령어들의 다른 타입을 저장하는 프로세서와 관련된 스토리지 매체는 CD-ROM, DVD, 메모리, 하드 디스크, 플래시 드라이브, RAM, ROM, 캐시, 등 중 하나 이상을 포함할 수 있으나, 이에 한정되는 것은 아니다.
- [0086] 도 6은 파일 타입을 도 1 내지 2의 어플리케이션과 관련짓는 시스템이 구현될 수 있는 예시적인 컴퓨터 시스템(600)을 도시한 블럭도이다. 특정 양상들에서, 컴퓨터 시스템(600)은 하드웨어 또는 소프트웨어와 하드웨어의 조합을 이용해서 전용서버에서 또는 다른 엔티티로 통합되거나 다수의 엔티티들에 걸쳐 분포되어 전용서버로 구현될 수 있다.
- [0087] 컴퓨터 시스템(600)은 버스(608) 또는 정보를 통신하는 다른 통신 매커니즘, 및 정보를 프로세싱하는 버스(608)와 연결된 프로세서(602)를 포함한다. 한 예로서, 컴퓨터 시스템(600)은 하나 이상의 프로세서들(602)로 구현될 수 있다.
- [0088] 컴퓨터 시스템(600)은 하드웨어 이외에, 당해 컴퓨터 프로그램에 대한 실행 환경을 생성하는 코드, 예를 들어, 프로세스 펌웨어, 프로토콜 스택, 데이터베이스 관리 시스템, 운영 체제, 또는 프로세서(602)에 의해 실행될 정보 및 명령어들을 저장하는 버스(608)와 연결된 포함된 메모리(604)에 저장된 그들 중 하나 이상의 조합을 구성하는 코드를 포함할 수 있는데, 여기서 상기 포함된 메모리(604)는, 예를 들어, RAM, 플래시 메모리, ROM, PROM, EPROM, 레지스터들, 하드 디스크, 이동식 디스크, CD-ROM, DVD, 또는 임의의 다른 적합한 스토리지 디바이스이다. 프로세서(602) 및 메모리(604)는 논리 회로에 의해 보충되거나 논리 회로에 포함될 수 있다.

- [0089] 본원에 기술된 방법들 및 시스템들은 서버, 클라이언트, 펌웨어, 게이트웨이, 허브(hub), 라우터 또는 다른 이러한 컴퓨터 및/또는 네트워킹 하드웨어 상의 컴퓨터 소프트웨어를 실행하는 머신을 통해 부분적으로 또는 전체로 사용될 수 있다. 소프트웨어 프로그램은 파일 서버, 프린트 서버, 도메인 서버, 인터넷 서버, 인트라넷 서버, 및 부 서버(secondary server), 호스트 서버, 분산 서버(distributed server) 등과 같은 다른 변형들(variants)을 포함하는 서버와 관련될 수 있다. 상기 서버는 메모리들, 프로세서들, 컴퓨터 관독가능 매체, 스토리지 매체, (물리적 및 추상적)포트들, 통신 디바이스들, 및 다른 서버들, 클라이언트들, 머신들 및 디바이스들을 유/무선 매체 등을 통해 액세스할 수 있는 인터페이스들을 포함할 수 있다. 본원 및 다른 곳에 기술된 바와 같이 방법들, 프로그램들 또는 코드들은 서버에 의해 실행될 수 있다. 추가로, 본원에 기술된 바와 같은 방법들의 실행을 위해 요청된 다른 디바이스들은 서버와 관련된 인프라구조의 일 부분으로 고려될 수 있다.
- [0090] 서버는 클라이언트들, 다른 서버들, 프린터들, 데이터베이스 서버들, 프린트 서버들, 파일 서버들, 통신 서버들, 분산 서버들, 등을 제한 없이 포함하는 다른 디바이스들에게 인터페이스를 제공할 수 있다. 또한, 이러한 결합 및/또는 연결은 네트워킹을 통해 프로그램들의 원격 실행을 용이하게 할 수 있다. 이러한 디바이스들의 일부 또는 전체의 네트워킹은 개시된 청구 대상의 범위를 벗어나지 않고 하나 이상의 위치에서 프로그램 또는 방법의 병렬 처리를 용이하게 할 수 있다. 또한, 인터페이스를 통해 서버에 연결된 디바이스들 중 어느 것이라도 방법들, 프로그램들 코드 및/또는 명령어들을 저장할 수 있는 적어도 하나의 스토리지 매체를 포함할 수 있다. 중앙 저장소는 상이한 디바이스들 상에서 실행될 프로그램 명령어들을 제공할 수 있다. 이러한 구현에서, 원격 저장소는 프로그램 코드, 명령어들 및 프로그램들에 대한 스토리지 매체로 역할을 할 수 있다.
- [0091] 본원에 기술된 방법들 및 시스템들은 네트워크 인프라구조들을 통해 부분적으로 또는 전체로 사용될 수 있다. 네트워크 인프라구조는, 예를 들어, 컴퓨팅 디바이스들, 서버들, 라우터들, 허브들, 방화벽(firewall)들, 클라이언트들, 개인 컴퓨터들, 통신 디바이스들, 라우팅 디바이스들 및 다른 능동적 및 수동적인 디바이스들, 본 기술 분야에 알려진 바와 같은 모듈들 및/또는 컴포넌트들과 같은 요소를 포함할 수 있다. 네트워크 인프라구조와 관련된 컴퓨팅 및/또는 비-컴퓨팅 디바이스(들)는 다른 컴포넌트들 이외에, 스토리지 매체, 예를 들어, 플래시 메모리, 버퍼, 스택, RAM, ROM, 등을 포함할 수 있다. 본원 및 그 밖의 곳에 기술된 상기 프로세스들, 방법들, 프로그램 코드들, 명령어들은 네트워크 인프라구조의 요소들 중 하나 이상에 의해 실행될 수 있다.
- [0092] 컴퓨터 소프트웨어, 프로그램 코드들 및/또는 명령어들은 머신 관독가능 매체 상에서 저장 및/또는 액세스될 수 있는데, 이 머신 관독가능 매체는: 일정 시간 간격 동안 컴퓨팅하는데 사용된 디지털 데이터를 구비하는 컴퓨터 컴포넌트들, 디바이스들 및 기록 매체; RAM로서 알려진 반도체 스토리지; 예를 들어, 광 디스크들과 같은 전형적으로 좀 더 영구적인 스토리지, 하드 디스크들, 테이프들, 드럼(drum)들, 카드들 및 다른 타입들과 같은 자기 스토리지의 형태를 위한 대용량 스토리지(mass storage); 프로세스 레지스터들, 캐시 메모리, 휘발성 메모리, 비-휘발성 메모리; CD 및 DVD와 같은 광 스토리지; 플래시 메모리(예를 들어, USB 스틱들 또는 키들), 플로피 디스크들, 자기 테이프, 페이퍼 테이프, 펀치 카드들, 독립형 RAM 디스크들, 집 드라이브들, 착탈가능한 대용량 스토리지, 오프라인 등과 같은 착탈가능 매체; 동적 메모리, 정적 메모리, 관독/기록 스토리지(read/write storage), 가변 스토리지(mutable storage), 읽기 전용(read only), 랜덤 액세스, 순차 액세스(sequential access), 위치 어드레싱가능(location addressable), 파일 어드레싱가능(file addressable), 콘텐츠 어드레싱가능(content addressable), 네트워크 접속형 스토리지(network attached storage), 스토리지 영역 네트워크, 바 코드들, 자기 잉크(magnetic ink), 등과 같은 다른 컴퓨터 메모리를 포함할 수 있다.
- [0093] 도면들을 통한 흐름도들과 블록도들에 포함된 본원에 기술된 및 도시된 요소들은 상기 요소들 사이의 논리적 관계들을 의미한다. 그러나, 소프트웨어 또는 하드웨어 엔지니어링 실시들에 따르면, 도시된 요소들과 그 기능들은 모놀리식 소프트웨어 구조, 독립형 소프트웨어 모듈들 또는 외부 루틴들, 코드, 서비스들, 등 또는 이들의 임의의 조합을 이용하는, 모듈로서 저장된 프로그램 명령어들을 실행할 수 있는 프로세서를 구비하는 컴퓨터 실행가능 매체를 통해 머신들 상에서 구현될 수 있으며, 이러한 모든 구현들은 본 발명의 범위 내에 존재할 수 있다.
- [0094] 따라서, 선행 도면들과 설명들은 본 개시된 시스템들의 기능적 양상들을 설명하는 한편, 이러한 기능적 양상들을 구현하기 위한 소프트웨어의 특정 배열이 특정적으로 언급되거나 컨텍스트로부터 분명치 않는 한, 이러한 설명들로부터 추론되어서는 않된다. 유사하게, 상기에서 식별되고 기술된 다양한 기법들이 변형될 수 있으며, 기법들의 순서(order)는 본원에 개시된 기법들의 특정 어플리케이션들에 적응될 수 있음이 인식될 것이다. 모든 이러한 변형들 및 수정들은 본 발명의 범위 내에 속하는 것으로 의도된다. 다양한 기법들에 대한 순서의 묘사 및/또는 설명은 특정 어플리케이션에 의해 요청되거나, 특정적으로 언급되거나 컨텍스트로부터 명백하지

않다면, 그러한 기법들을 위한 실행의 특정 순서를 요구하는 것으로 이해되어서는 않된다.

[0095]

상기에 기술된 방법들 및/또는 프로세스들 및 그 기법들은 하드웨어 또는 특정 어플리케이션에 적합한 하드웨어와 소프트웨어의 임의의 조합으로 구현될 수 있다. 하드웨어는 범용 컴퓨터 및/또는 전용 컴퓨팅 디바이스, 또는 특정 컴퓨팅 디바이스 또는 특정 컴퓨팅 디바이스의 특정 양상 또는 컴포넌트를 포함할 수 있다. 프로세스들은 내부 및/또는 외부 메모리와 함께, 하나 이상의 마이크로프로세서들, 마이크로컨트롤러들, 내장된 마이크로컨트롤러들, 프로그래머블 디지털 신호 프로세서들, 또는 다른 프로그래머블 디바이스로 구현될 수 있다. 프로세스들은 또는 대신에, 어플리케이션 특정 집적 회로, 프로그래머블 게이트 어레이(programmable gate array), 프로그래머블 어레이 로직, 또는 전자 신호들을 처리하도록 구성될 수 있는 임의의 다른 디바이스 또는 디바이스들의 조합으로 구현될 수 있다. 프로세서들 중 하나 이상이 머신 판독가능 매체 상에서 실행될 수 있는 컴퓨터 실행가능 코드로 구현될 수 있다는 것을 더 인식될 수 있다.

[0096]

명령어들은 메모리(604)에 저장될 수 있고, 하나 이상의 컴퓨터 프로그램 제품들, 즉, 서버(100)에 의한 실행을 위해 또는 서비스(100)의 동작을 제어하기 위해 컴퓨터 판독가능 매체 상에서 인코딩되고, 본 기술 분야의 당업자에게 잘 알려진 임의의 방법에 따라, 데이터-지향 언어들(예를 들어, SQL, dBase), 시스템 언어들(예를 들어, C, Objective-C, C++, 어셈블리), 아키텍처 언어들(예를 들어, 자바(Java), NET), 및 어플리케이션 언어들(예를 들어, PHP, Ruby, Perl, Python)과 같은 컴퓨터 언어들을 포함하지만 이에 한정되지 않는 컴퓨터 프로그램 명령어들의 하나 이상의 모듈들로 구현될 수 있다.

[0097]

본원에 논의된 컴퓨터 프로그램은 파일 시스템 내의 파일에 반드시 대응하지는 않는다. 프로그램은 다른 프로그램들 또는 데이터(예를 들어, 마크업(markup) 언어 문서에 저장된 하나 이상의 스크립트들)를 홀드하는 파일의 일 부분, 당해 프로그램에 전용되는 단일 파일, 또는 다수의 조직화된 파일들(예를 들어, 하나 이상의 모듈들, 서브프로그램들, 또는 코드의 부분들을 저장하는 파일들)에 저장될 수 있다. 컴퓨터 프로그램은 하나의 컴퓨터 상에서 또는 하나의 사이트에 위치되거나 또는 다수의 사이트들에 걸쳐 분포되고 통신 네트워크에 의해 서로 연결되는 다수의 컴퓨터들 상에서 실행되도록 사용될 수 있다. 본 명세서에 기술된 프로세스들 및 로직 플로우들은 입력 데이터 상에서 동작하고 출력을 발생시킴으로써 기능들을 수행하도록 하나 이상의 컴퓨터 프로그램들을 실행하는 하나 이상의 프로그래머블 프로세서들에 의해 수행될 수 있다.

[0098]

컴퓨터 시스템(600)은 정보 및 명령어들의 저장을 위해 버스(608)로 결합되는 자기 디스크 또는 광 디스크와 같은 데이터 스토리지 디바이스(606)를 더 포함한다. 컴퓨터 시스템(600)은 입/출력 모듈(610)을 통해 다양한 디바이스들로 결합될 수 있다. 입/출력 모듈(610)은 임의의 입/출력 모듈일 수 있다. 예시적인 입/출력 모듈(610)은 USB 포트들과 같은 데이터 포트들을 포함한다. 이 입/출력 모듈(610)은 통신 모듈(612)과 연결되도록 구성된다. 예시적인 통신 모듈(612)들은 네트워킹 인터페이스 카드, 예를 들어, 이더넷(Ethernet) 카드들 및 모듈들을 포함한다. 특정 양상들에서, 입/출력 모듈(610)은 다수의 디바이스들, 예를 들어, 입력 디바이스(614) 및/또는 출력 디바이스(616)와 연결되도록 구성된다. 예시적인 입력 디바이스(614)들은 사용자가 입력을 컴퓨터 시스템(600)으로 제공할 수 있는 키보드 및 포인팅 디바이스(pointing device), 예를 들어, 마우스 또는 트랙볼(trackball)을 포함한다. 입력 디바이스(614)들의 다른 종류들, 예를 들어, 촉각 입력 장치, 비주얼 입력 장치, 오디오 입력 장치, 또는 브레인-컴퓨터 인터페이스 장치는 또한 사용자와의 인터랙트를 제공하는데 사용될 수 있다. 예를 들어, 사용자에게 제공되는 피드백은 감각적 피드백, 예를 들어, 비주얼 피드백, 청각 피드백, 또는 촉각 피드백의 어떤 형태일 수 있으며, 청각, 음성, 촉각, 또는 브레인 웨이브(brain wave) 입력을 포함하는 사용자로부터의 입력은 임의의 형태로 수신될 수 있다. 예시적인 출력 디바이스(616)들은 사용자에게 정보를 디스플레이하기 위해, 디스플레이 디바이스들, 예를 들어, CRT(cathode ray tube) 또는 LCD(liquid crystal display) 모니터를 포함한다.

[0099]

본 발명의 한 양상에 따르면, 도 1 내지 2에 도시된 바와 같이, 파일 타입을 어플리케이션과 관련짓는 시스템은 메모리(604)에 내포된 하나 이상의 명령어들의 하나 이상의 시퀀스들을 실행하는 프로세서(602)에 응답하여 컴퓨터 시스템(600)을 사용하여 구현될 수 있다. 이러한 명령어들은 데이터 스토리지 디바이스(606)와 같은 다른 머신-판독 가능 매체로부터 메모리(604)내로 읽혀질 수 있다. 메인 메모리(604)에 내포된 명령어들의 시퀀스들의 실행은 프로세서(602)로 하여금 본원에 기술된 프로세스들을 수행하게 한다. 다중-프로세싱 배열의 하나 이상의 프로세서들은 또한 메모리(604)에 내포된 명령어들의 시퀀스들을 실행하기 위해 이용될 수 있다. 대안적인 양상들에서, 하드-와이어 회로(hard-wired circuitry)는 본 발명의 다양한 양상들을 구현하기 위해 소프트웨어 명령어들을 대신에 또는 상기 소프트웨어 명령어들과 조합하여 사용될 수 있다. 따라서, 본 발명의 양상들은 하드웨어 회로와 소프트웨어의 임의의 특정 조합에 제한되지 않는다.

- [0100] 본 명세서에 기술된 주제의 다양한 양상들은 컴퓨팅 시스템으로 구현될 수 있으며, 상기 컴퓨팅 시스템은 예를 들어, 데이터 서버로서, 백 엔드 컴포넌트(back end componet)를 포함하거나, 미들웨어(middleware) 컴포넌트, 예를 들어, 어플리케이션 서버를 포함하거나, 프론트 엔드(front end) 컴포넌트, 예를 들어, 사용자가 본 명세서에 기술된 주제의 구현시 인터랙트할 수 있는 그래픽 사용자 인터페이스 또는 웹 브라우저를 구비하는 클라이언트 컴퓨터를 포함하거나 또는 하나 이상의 이러한 백 엔드, 미들웨어, 프론트 엔드 컴포넌트들 중 임의의 조합을 포함한다. 시스템의 컴포넌트들은 디지털 데이터 통신, 임의의 형태 또는 매체, 예를 들어, 통신 네트워크에 의해 서로 연결될 수 있다. 통신 네트워크는 예를 들어, PAN(personal area network), LAN(local area network), CAN(campus area network), MAN(metropolitan area network), WAN(Wide area network), BBN(broadband network), 인터넷, 등 중 임의의 하나 이상을 포함할 수 있다. 또한, 통신 네트워크들은, 예를 들어, 하기의 네트워크 토폴로지(topology)들 중 임의의 하나 이상을 포함할 수 있으나 이에 한정되지 않는데, 이러한 네트워크 토폴로지들은 버스 네트워크, 스타 네트워크, 링 네트워크, 메쉬(mesh) 네트워크, 스타-버스 네트워크, 트리, 또는 계층적(hierarchical) 네트워크, 등을 포함한다. 통신 모듈들은, 예를 들어, 모듈들 또는 이더넷 카드들일 수 있다.
- [0101] 상기 논의된 바와 같이, 컴퓨팅 시스템(600)은 클라이언트들 및 서버들을 포함할 수 있다. 클라이언트 및 서버는 일반적으로 서로 멀리 떨어져 있고, 통신 네트워크를 통해 전형적으로 인터랙트한다. 클라이언트와 서버의 관계는 각 컴퓨터들 상에서 동작하고 서로에 대해 클라이언트-서버 관계를 구비하는 컴퓨터 프로그램들로 인해 발생된다. 컴퓨터 시스템(600)은, 예를 들어, 제한 없이 엔터프라이즈 서버(enterprise server) 또는 서버들의 그룹, 하나 이상의 데스크톱 컴퓨터들, 하나 이상의 랩탑 컴퓨터들 등일 수 있다. 컴퓨터 시스템(600)은, 다른 디바이스 예를 들어, 제한 없이 휴대폰, PDA(personal digital assistant), 모바일 오디오 플레이어, GPS(Global Positioning System) 수신기, 비디오 게임 콘솔, 및/또는 텔레비전 세트 탑 박스(television set top box)에 삽입될 수 있다.
- [0102] 본원에 사용된 바와 같은 용어 "머신-판독 가능 스토리지 매체" 또는 "컴퓨터 판독가능 매체"는 실행을 위해 명령어들을 프로세서(602)로 제공하는데 기여하는 임의의 매체 또는 매체들을 의미한다. 이러한 매체로 비-휘발성 매체, 휘발성 매체, 및 전송 매체를 포함하지만 이에 한정되지 않는 여러 형태들을 취할 수 있다. 비-휘발성 매체는, 예를 들어, 데이터 스토리지 디바이스(606)과 같은 광 또는 자기 디스크들을 포함한다. 휘발성 매체는 메모리(604)와 같은 동적 메모리를 포함한다. 전송 매체는 버스(608)를 구성하는 와이어들을 포함하는, 동축 케이블들, 구리 와이어, 또는 광 섬유들을 포함한다. 머신-판독 가능 매체의 공통적인 형태들은, 예를 들어, 플로피 디스크, 플렉시블 디스크, 하드 디스크, 자기 테이프, 임의의 다른 자기 매체, CD-ROM, DVD, 임의의 다른 광 매체, 펀치 카드, 종이 테이프, 홀을 갖는 임의의 다른 물리적 매체, RAM, PRAM, EPROM, 플래시 EPROM, 임의의 다른 메모리 칩 또는 카트리지(cartridge), 또는 컴퓨터가 판독 가능한 임의의 다른 매체를 포함한다. 머신-판독 가능 스토리지 매체는 머신-판독가능 스토리지 디바이스, 머신-판독가능 스토리지 기관, 메모리 디바이스, 머신-판독가능 전파 신호에 영향을 미치는 물질의 조성물(composition), 또는 그들 중 하나 이상의 조합일 수 있다.
- [0103] 본 명세서가 많은 세부 사항을 내포하긴 하지만, 이것들은 청구될 수 있는 것의 범위에 대한 제한들로서 해석되어서는 안되며, 오히려 주제의 특정 구현들의 설명들로서 해석되어야 한다. 개별 구현들의 맥락에서 본 명세서에 기술된 일정 특징들은 또한 단일 구현에서 조합으로 구현될 수 있다. 반대로, 단일 구현의 맥락에서 기술된 다양한 특징들은 또한 다수의 구현들로 개별적으로 또는 임의의 적절한 하위조합으로 구현될 수 있다. 더욱이, 비록 특징들이 특정 조합들로 행해지고 심지어 최초로 청구되는 것처럼 상기에 기술될지라도, 청구된 조합으로부터 하나 이상의 특징들은 어떤 경우에는 상기 조합으로부터 제거될 수 있고, 상기 청구된 조합은 하위조합 또는 하위조합의 변형에 관련될 수 있다.
- [0104] 동작들이 도면에 특정 순서로 도시되어 지는데, 이것은 그러한 동작들이 요망되는 결과들을 달성하기 위해 도시된 특정 순서로 또는 순차적인 순서로 수행되거나 모든 예시적인 동작들이 실행되는 것을 요구하는 것으로 이해되지 않아야 한다. 특정 상황들에서, 멀티태스킹 및 병렬 프로세싱은 유리할 수 있다. 더욱이, 상기에 기술된 양상들에서 다양한 시스템 컴포넌트들의 분리는 모든 양상들에서 이러한 분리를 요구하는 것으로 이해되지 않아야 하며, 상기 기술된 프로그램 컴포넌트들 및 시스템들은 일반적으로 단일 소프트웨어 제품에 함께 통합되거나 또는 다수의 소프트웨어 제품내로 패키징(package)될 수 있다.
- [0105] 본 명세서의 주제는 특정 양상들에 대해 기술되었지만, 다른 양상들은 하기의 청구항들의 범위 내에서 구현될 수 있으며 존재할 수 있다. 예를 들어, 청구항들에 기재된 동작(action)들은 상이한 순서로 수행될 수 있고 여전히 요망되는 결과들을 얻을 수 있다. 하나의 예로서, 첨부된 도면들에 도시된 프로세스들은 요망되는 결과들을 획득하기 위해 도시된 특정 순서, 또는 순차적인 순서를 반드시 요구하지는 않는다. 특정 구현들에서, 멀티

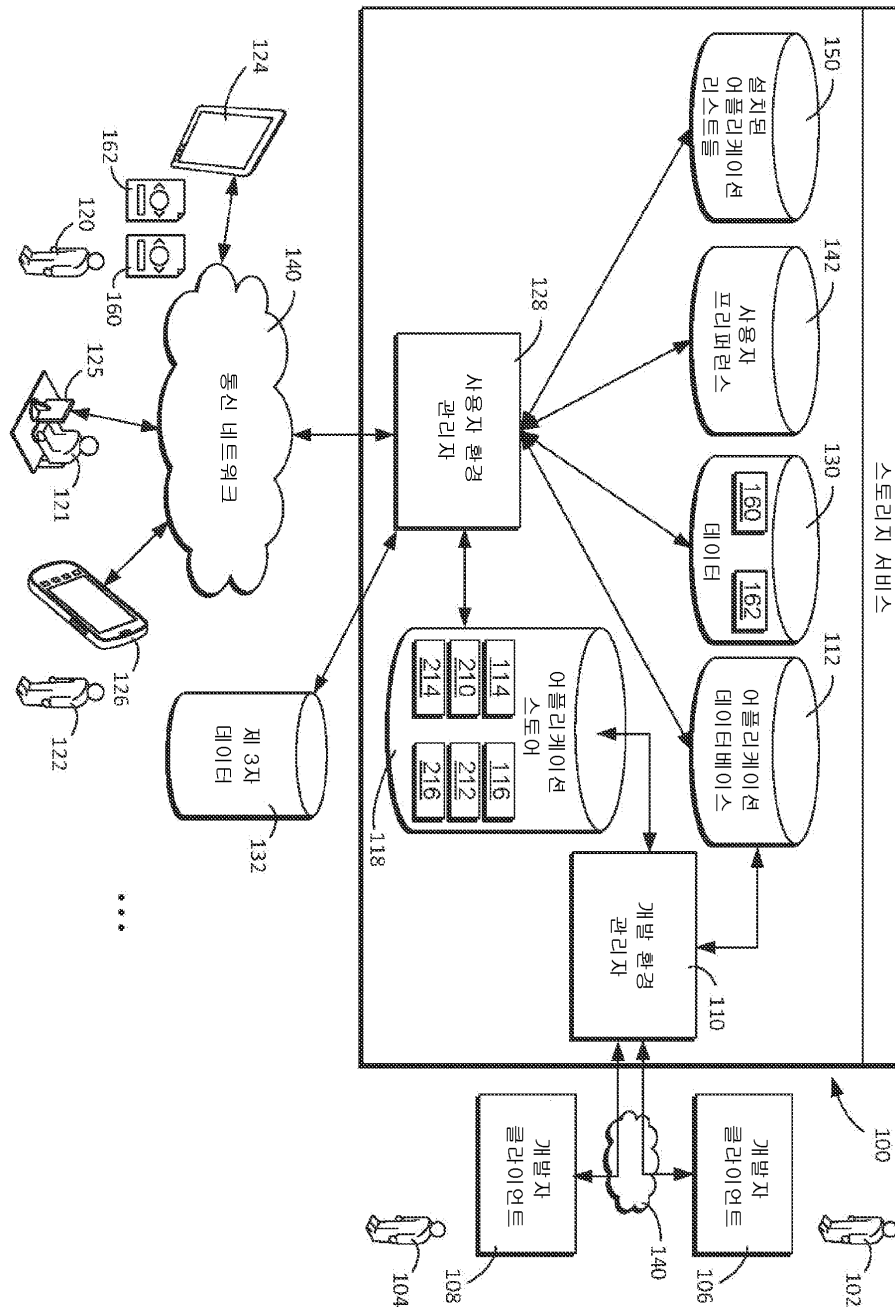
테스킹 및 병렬 프로세싱이 유리할 수 있다. 다른 변형들은 하기의 청구항들의 범위 내에 존재한다.

[0106]

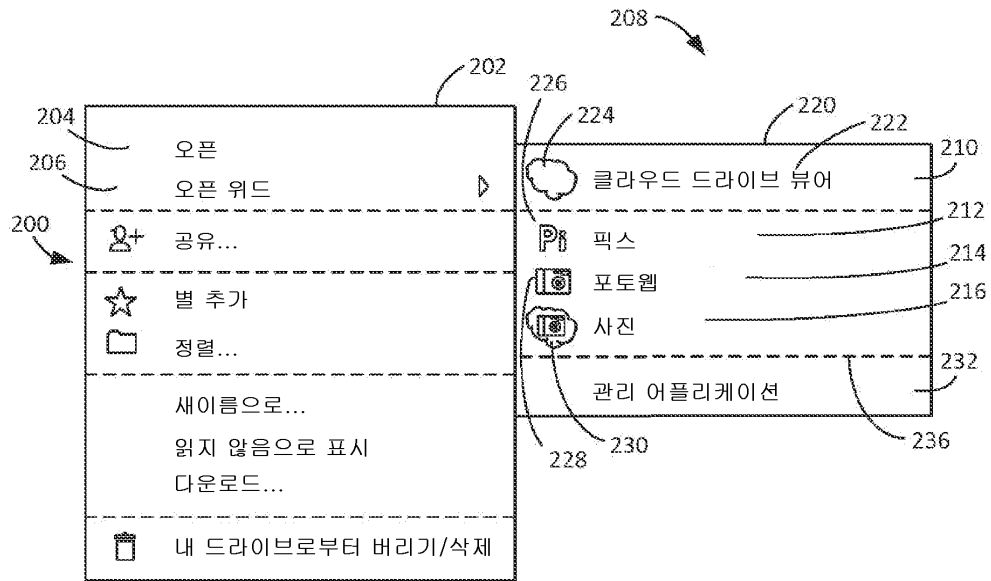
본원에 첨부된 청구항들에서, 단어 "~하는 수단" 또는 "~하는 단계"가 본 청구항에 사용될 때만 발명자들은 35 U.S.C. § 112, 단락 6을 적용한다. 만일 이러한 단어들이 청구항에서 사용되지 않으면, 발명자들은 35 U.S.C. § 112, 단락 6에 따라 청구항이 본원에 기술된 해당 구조, 재료, 또는 동작(act)(및 이들의 등가물들)을 커버하는 것으로 해석되게 하지 않는다.

도면

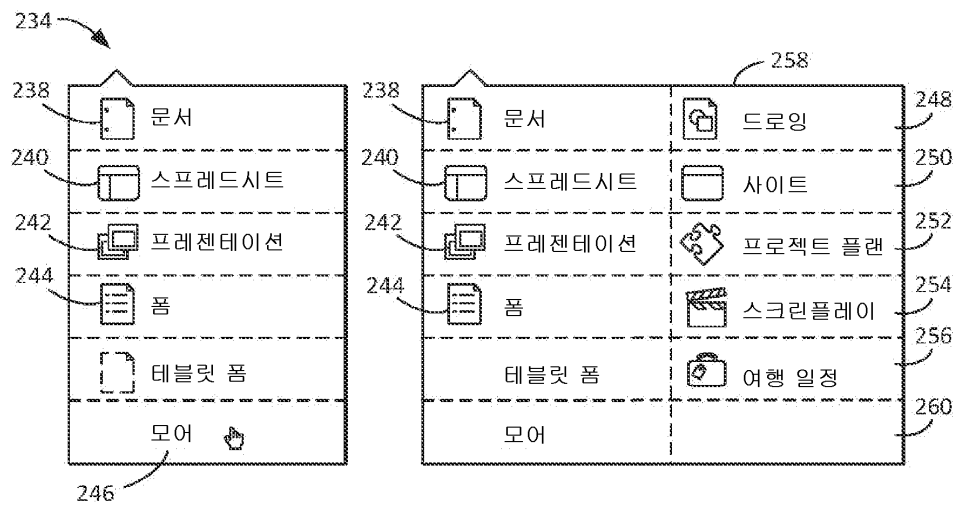
도면1



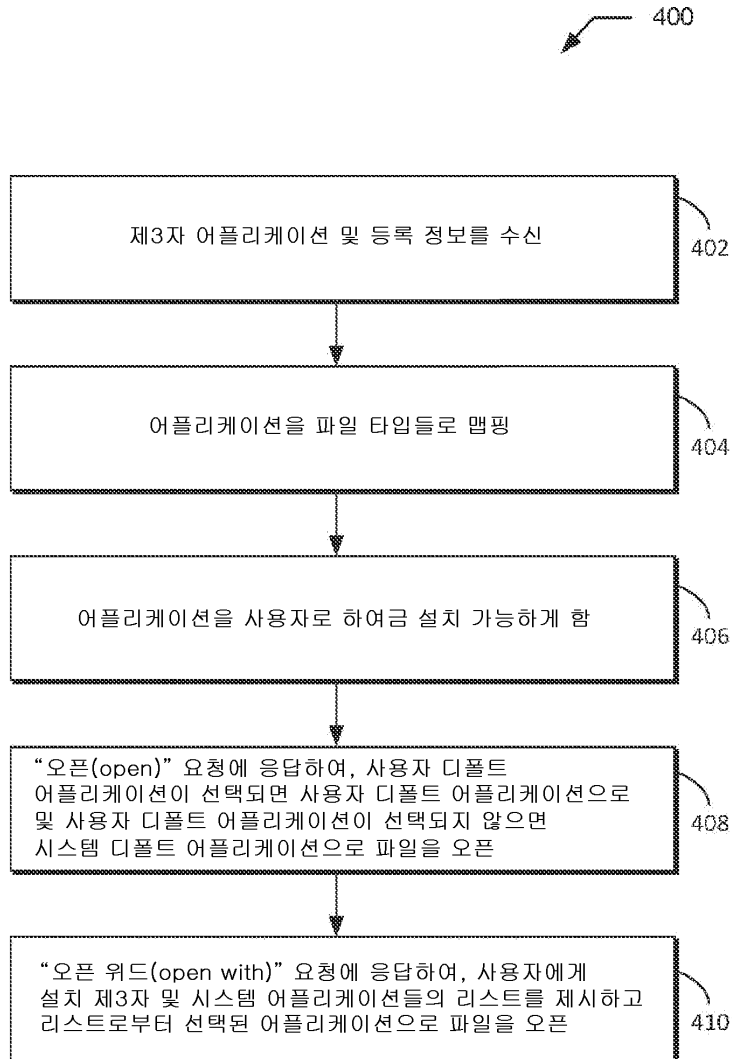
도면2



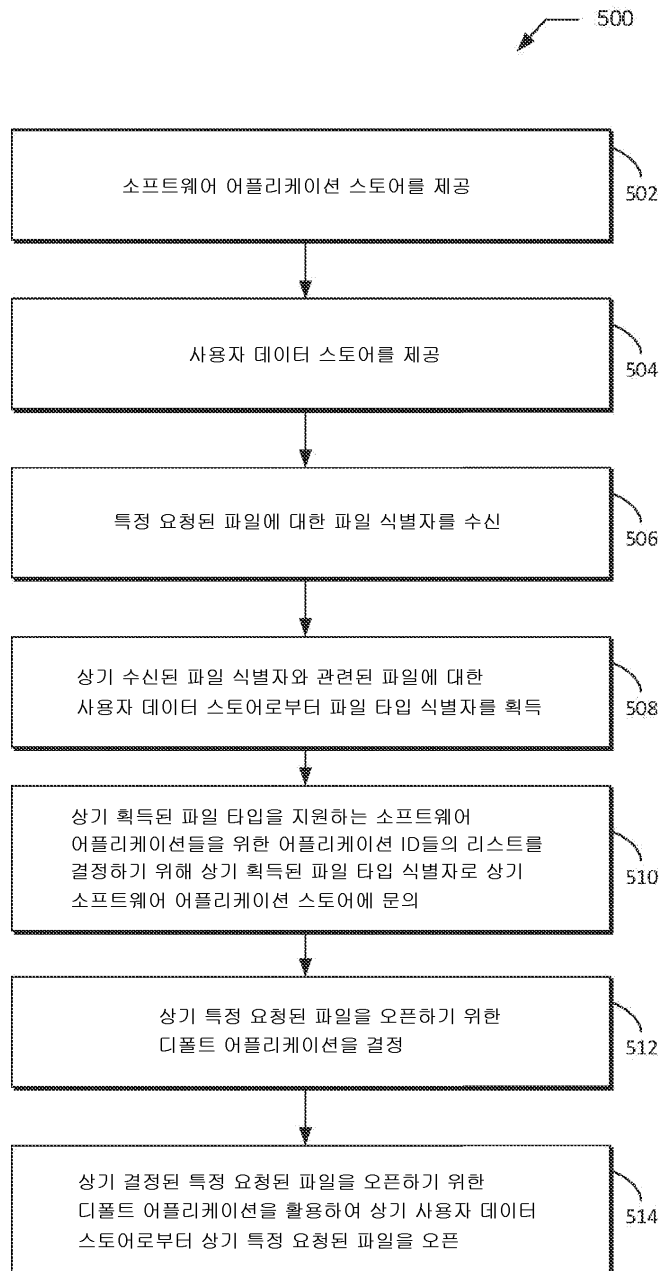
도면3



도면4



도면5



도면6

