



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년12월04일

(11) 등록번호 10-1574811

(24) 등록일자 2015년11월30일

- (51) 국제특허분류(Int. Cl.)  
*G06F 15/16* (2006.01) *G06F 9/44* (2006.01)
- (21) 출원번호 10-2010-7006157
- (22) 출원일자(국제) 2008년09월12일  
 심사청구일자 2013년08월13일
- (85) 번역문제출일자 2010년03월19일
- (65) 공개번호 10-2010-0074149
- (43) 공개일자 2010년07월01일
- (86) 국제출원번호 PCT/US2008/076136
- (87) 국제공개번호 WO 2009/042421  
 국제공개일자 2009년04월02일
- (30) 우선권주장  
 11/861,877 2007년09월26일 미국(US)
- (56) 선행기술조사문헌  
 US20020133814 A1  
 US20030182656 A1

- (73) 특허권자  
 마이크로소프트 테크놀로지 라이센싱, 엘엘씨  
 미국 위싱턴주 (우편번호 : 98052) 레드몬드 원  
 마이크로소프트 웨이
- (72) 발명자  
 센, 알버트 씨.에스.  
 미국 98052-6399 위싱턴주 레드몬드 원 마이크로  
 소프트 웨이 마이크로소프트 코포레이션 국제 특  
 허부 내  
 베이터, 크리스토퍼 제이.  
 미국 98052-6399 위싱턴주 레드몬드 원 마이크로  
 소프트 웨이 마이크로소프트 코포레이션 국제 특  
 허부 내  
 (뒷면에 계속)
- (74) 대리인  
 제일특허법인

전체 청구항 수 : 총 20 항

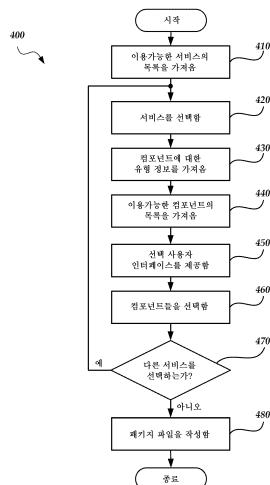
심사관 : 권현수

(54) 발명의 명칭 분산형 확장가능 애플리케이션의 생성 및 배포

### (57) 요 약

분산형 애플리케이션을 생성하는 것은 원격 서버 클러스터 상에서 이용가능한 컴포넌트들의 목록 중에서 일군의 컴포넌트들을 선택하는 것을 포함한다. 선택된 컴포넌트들을 설치하는 데 필요한 데이터가 원격 서버 클러스터로부터 수신된다. 수신된 데이터에 응답하여 일련의 명령어들이 생성된다. 이어서, 일련의 명령어들이 저장된다. 선택된 컴포넌트들을 설치하는 데 필요한 데이터를 추출하기 위해 일련의 명령어들이 처리된다. 원격 서버 클러스터에 컴포넌트들을 설치하는 것을 가능하게 하기 위해 선택된 컴포넌트들을 설치하는 데 필요한 데이터가 원격 서버 클러스터로 전송된다.

### 대 표 도 - 도4



(72) 발명자

**톰, 리차드 더블유.**

미국 98052-6399 위성던주 레드몬드 원 마이크로소프트 웨이 마이크로소프트 코포레이션 국제 특허부내

**고피나스, 라비쿠마르 비.**

미국 98052-6399 위성던주 레드몬드 원 마이크로소프트 웨이 마이크로소프트 코포레이션 국제 특허부내

**블롬퀴스트, 브라이언 씨.**

미국 98052-6399 위성던주 레드몬드 원 마이크로소프트 웨이 마이크로소프트 코포레이션 국제 특허부내

---

**카니간티, 마드하빌라타**

미국 98052-6399 위성던주 레드몬드 원 마이크로소프트 웨이 마이크로소프트 코포레이션 국제 특허부내

**치우, 테이비드**

미국 98052-6399 위성던주 레드몬드 원 마이크로소프트 웨이 마이크로소프트 코포레이션 국제 특허부내

## 명세서

### 청구범위

#### 청구항 1

분산형 애플리케이션을 생성하는 방법으로서, 상기 방법은 프로세서에 의해 실행되고,

원격 서버 클러스터 상에서 이용가능한 서비스들의 목록에서 서비스를 선택하는 단계 - 각 서비스는 상이한 기능을 제공함 - 와,

선택된 서비스 각각에 대하여, 상기 원격 서버 클러스터 상의 상기 선택된 서비스로부터 이용 가능한 컴포넌트들의 목록으로부터 컴포넌트의 그룹을 선택하는 단계와,

상기 원격 서버 클러스터의 상이한 서비스들로부터 상기 선택된 컴포넌트를 설치하기 위해 요구되는 데이터를 수신하는 단계와,

상기 분산형 애플리케이션을 생성하기 위해 사용되는 상기 상이한 서비스들로부터의 상기 데이터의 적어도 일부를 포함하는 명령어들의 목록을 생성하는 단계와,

상기 명령어들의 목록을 저장하는 단계와,

상기 컴포넌트들에 대한 메타 정보(meta-information)을 기술하는 매니페스트(manifest)를 포함하는 패키지 파일을 생성하는 단계 - 상기 매니페스트는 잠재적인 충돌을 검출하고 컴포넌트 이용가능성을 판정하기 위해 상기 서비스들에 의해 이용 가능하며, 상기 메타 정보는 팩클릿(packlet) 없이 상기 충돌을 검출하는데 이용 가능하고, 상기 팩클릿은 상기 선택된 서비스에 특정적인 바이너리 데이터를 포함함 -

를 포함하는 방법.

#### 청구항 2

제1항에 있어서, 상기 명령어들의 목록을 저장하는 단계는,

수신된 데이터를 상기 명령어들의 목록에 인코딩하는 단계와,

상기 명령어들의 목록에 워터마크를 인코딩하는 단계

를 더 포함하는 방법.

#### 청구항 3

제1항에 있어서, 이용 가능한 상기 서비스들의 목록을 판정하기 위해 상기 원격 서버 클러스터에 대한 호출을 수행하는 단계

를 더 포함하는 방법.

#### 청구항 4

제1항에 있어서, 상기 선택된 서비스 상에서 이용 가능한 컴포넌트를 선택하는 단계는, 상기 선택된 서비스 상에서 이용 가능한 컴포넌트의 유형에 특정적인 선택 인터페이스 데이터를 수신하는 단계를 더 포함하는 방법.

#### 청구항 5

제1항에 있어서,

상기 선택된 컴포넌트들을 설치하는 데 요구되는 데이터를 추출하기 위해 상기 명령어들의 목록을 처리하는 단계와,

상기 원격 서버 클러스터에 상기 선택된 컴포넌트들을 설치하는 것을 가능하게 하기 위해 상기 선택된 컴포넌트들을 설치하는 데 요구되는 상기 데이터를 상기 원격 서버 클러스터로 전송하는 단계

를 더 포함하는 방법.

#### 청구항 6

제5항에 있어서, 상기 명령어들의 목록을 처리하는 단계는,

상기 선택된 컴포넌트들에 대한 기술을 상기 선택된 컴포넌트들과 연관된 서비스들로 전송하는 단계와,

상기 서비스들로부터 충돌 정보를 수신하는 단계와,

상기 선택된 컴포넌트들 내에 충돌이 존재하는지 여부를 판정하기 위해 상기 충돌 정보를 중앙집중식으로 처리하는 단계와,

충돌이 존재하지 않을 때, 상기 충돌 정보를 중앙집중식으로 처리한 것에 응답하여, 상기 원격 서버 클러스터에 상기 컴포넌트들을 설치하는 것을 가능하게 하기 위해 상기 선택된 컴포넌트들을 설치하는 데 요구되는 데이터를 상기 원격 서버 클러스터로 전송하는 단계

를 더 포함하는 방법.

#### 청구항 7

제6항에 있어서, 상기 명령어들의 목록을 처리하는 단계는,

충돌이 존재하는 경우, 상기 충돌이 자동으로 처리될 수 있는지 여부를 판정하는 단계와,

상기 충돌이 자동으로 처리될 수 있는지 여부의 상기 판정에 응답하여 충돌을 자동으로 처리하는 단계

를 더 포함하는 방법.

#### 청구항 8

분산형 애플리케이션을 생성하는 컴퓨터-실행가능 명령어들을 포함하는 컴퓨터-관독가능 매체로서,

상기 명령어들은,

이용 가능한 서비스들의 목록을 판정하기 위해 원격 서버 클러스터에 호출을 수행하는 동작과,

분산 컴퓨터 시스템 상에서 이용 가능한 서비스들의 목록 중에서 서비스를 선택하는 동작과,

상기 선택된 서비스 상에서 이용 가능한 컴포넌트를 선택하는 동작과,

상기 선택된 컴포넌트에 관련된 설치 데이터를 상기 분산 컴퓨터 시스템으로부터 수신하는 동작과,

상기 컴포넌트에 대한 메타 정보를 기술하는 매니페스트를 포함하는 패키지 파일을 생성하는 동작 - 상기 매니페스트는 상기 수신된 데이터에 응답하여 잠재적인 충돌을 검출하고 컴포넌트 이용가능성을 판정하기 위해 상기 서비스들에 의해 이용 가능하며, 상기 패키지 파일은 상기 분산형 애플리케이션을 생성하는데 사용되는 설치 데이터를 포함하고, 상기 메타 정보는 패클릿 없이도 상기 충돌을 검출하는데 이용 가능하고, 상기 패클릿은 상기 선택된 서비스에 특정적인 바이너리 데이터를 포함함 - 과,

상기 패키지 파일을 저장하는 동작

을 수행하는, 컴퓨터-관독가능 매체.

#### 청구항 9

제8항에 있어서, 상기 패키지 파일을 저장하는 동작은,

상기 패키지 파일에 상기 선택된 컴포넌트를 기술하는 상기 매니페스트를 인코딩하는 동작과,

상기 패키지 파일에 상기 설치 데이터를 패이로드로서 인코딩하는 동작과,

상기 패키지 파일에 위터마크를 인코딩하는 동작

을 더 포함하는, 컴퓨터-관독가능 매체.

#### 청구항 10

제9항에 있어서, 상기 패키지 파일에 상기 워터마크를 인코딩하는 동작은, 상기 패키지 파일에 포함된 데이터의 암호화된 해쉬(hash)를 생성하는 동작을 더 포함하는, 컴퓨터-관독가능 매체.

### 청구항 11

제8항에 있어서, 상기 선택된 서비스 상에서 이용 가능한 컴포넌트를 선택하는 동작은,  
상기 서비스 상에서 이용 가능한 컴포넌트의 유형에 특정적인 선택 인터페이스 데이터를 수신하는 동작과,  
상기 선택 인터페이스 데이터에 응답하여 선택 인터페이스를 발생하기 위해 상기 선택 인터페이스 데이터를 배포 인터페이스(deployment interface)로 전달하는 동작  
을 더 포함하는, 컴퓨터-관독가능 매체.

### 청구항 12

제8항에 있어서,  
상기 명령어들은,  
상기 설치 데이터를 추출하기 위해 상기 패키지 파일을 처리하는 동작과,  
상기 분산형 애플리케이션의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 분산 컴퓨터 시스템으로 전송하는 동작  
을 더 수행하는, 컴퓨터-관독가능 매체.

### 청구항 13

제9항에 있어서,  
상기 매니페스트를 상기 선택된 컴포넌트와 연관된 상기 선택된 서비스로 전송하는 동작과,  
상기 선택된 서비스로부터 충돌 정보를 수신하는 동작과,  
충돌이 존재하는지 여부를 판정하기 위해 상기 충돌 정보를 중앙집중식으로 처리하는 동작과,  
상기 충돌 정보를 중앙집중식으로 처리하는 것에 응답하여 상기 분산형 애플리케이션의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 선택된 서비스로 전송하는 동작  
을 더 포함하는, 컴퓨터-관독가능 매체.

### 청구항 14

제12항에 있어서, 상기 패키지 파일을 처리하는 동작은,  
상기 선택된 서비스가 이용 가능한지 여부를 판정하는 동작과,  
상기 선택된 서비스가 이용 가능할 때 상기 분산형 애플리케이션의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 분산 컴퓨터 시스템으로 전송하는 동작  
을 더 포함하는, 컴퓨터-관독가능 매체.

### 청구항 15

제12항에 있어서, 상기 설치 데이터를 추출하기 위해 상기 패키지 파일을 처리하는 동작은,  
사용자 퍼미션(permission)에 기초하여 상기 선택된 컴포넌트가 이용 가능한지 여부를 판정하는 동작과,  
상기 선택된 컴포넌트가 이용 가능할 때 상기 분산형 애플리케이션의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 분산 컴퓨터 시스템으로 전송하는 동작  
을 더 포함하는, 컴퓨터-관독가능 매체.

### 청구항 16

분산형 컴퓨팅 애플리케이션의 제어를 중앙집중화하는 시스템으로서,  
프로세서 및 컴퓨터-판독가능 매체와,  
상기 컴퓨터-판독가능 매체 상에 저장되고 상기 프로세서 상에서 실행되는 운영 환경과,  
상기 컴퓨터-판독가능 매체 상에 저장되고 상기 프로세서 상에서 실행되는 솔루션 프레임워크  
를 포함하고,  
상기 솔루션 프레임워크는,  
서비스 클러스터 상에서 이용 가능한 서비스들의 목록 중에서 서비스를 선택하고,  
상기 선택된 서비스 상에서 이용 가능한 컴포넌트를 선택하며,  
상기 선택된 컴포넌트에 관련된 설치 데이터를 상기 선택된 서비스로부터 수신하고,  
상기 컴포넌트에 대한 메타 정보를 기술하는 매니페스트를 포함하는 패키지 파일을 생성하되, 상기 매니페스트  
는 상기 수신된 데이터에 응답하여 잠재적인 충돌을 검출하고 컴포넌트 이용가능성을 판정하기 위해 상기 서비  
스들에 의해 이용 가능하며, 상기 패키지 파일은 상기 분산형 애플리케이션을 생성하는데 사용되는 설치 데이터  
를 포함하고, 상기 메타 정보는 패클릿 없이도 상기 충돌을 검출하는데 이용 가능하고, 상기 패클릿은 상기 선  
택된 서비스에 특정적인 바이너리 데이터를 포함하며,  
상기 패키지 파일을 상기 컴퓨터-판독가능 매체 상에 저장하도록  
구성되어 있는 시스템.

#### 청구항 17

제16항에 있어서, 상기 솔루션 프레임워크는 또한,  
상기 패키지 파일에 상기 선택된 컴포넌트를 기술하는 상기 매니페스트를 저장하는 동작과,  
상기 패키지 파일에 상기 설치 데이터를 저장하는 동작과,  
상기 패키지 파일에 워터마크를 저장하는 동작  
에 의해 상기 컴퓨터-판독가능 매체 상에 상기 패키지 파일을 저장하도록 구성되어 있는 시스템.

#### 청구항 18

제16항에 있어서, 상기 솔루션 프레임워크는 또한,  
상기 서비스 상에서 이용 가능한 컴포넌트의 유형에 특정적인 커스텀(custom) 선택 인터페이스를 수신하고,  
상기 커스텀 선택 인터페이스를 배포 인터페이스로 전달하도록  
구성되어 있는 시스템.

#### 청구항 19

제17항에 있어서, 상기 솔루션 프레임워크는 또한,  
상기 설치 데이터를 추출하기 위해 상기 패키지 파일을 처리하고;  
상기 분산형 컴퓨팅 애플리케이션의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 분산 컴퓨터 시스템으  
로 전송하도록  
구성되어 있는 시스템.

#### 청구항 20

제19항에 있어서, 상기 솔루션 프레임워크는 또한,  
상기 매니페스트를 상기 선택된 컴포넌트와 연관된 상기 선택된 서비스로 전송하는 동작과,

상기 서비스로부터 충돌 정보를 수신하는 동작과,  
 충돌이 존재하는지 여부를 판정하기 위해 상기 충돌 정보를 중앙집중식으로 처리하는 동작과,  
 상기 충돌 정보를 중앙집중식으로 처리한 것에 응답하여 상기 컴포넌트의 설치를 가능하게 하기 위해 상기 설치 데이터를 상기 선택된 서비스로 전송하는 동작과,  
 배포후 호출(post-deployment call)을 상기 서비스 클러스터로 전송하는 동작  
 와 의해 상기 설치 데이터를 추출하기 위해 상기 패키지 파일을 처리하도록 구성되는 시스템.

### 발명의 설명

#### 배경기술

[0001] 분산 컴퓨팅 네트워크를 통해 설치된 소프트웨어는 사용자가 개별적인 서버 클러스터 상에서 호스팅되는 다수의 서비스를 이용하는 엔드-투-엔드(end-to-end) 애플리케이션 및 솔루션을 생성할 수 있게 한다. 이들 엔드-투-엔드 애플리케이션은 각각이 선택에 따라 서로 다른 서버 클러스터 상에 분산되어 있는, 비즈니스 데이터(business data), 페이지 설계, 페이지 레이아웃, 및 비즈니스 로직(business logic)과 같은 컴포넌트들을 포함할 수 있다. 이들 애플리케이션의 분산형 성질로 인해, 이들 애플리케이션의 배포는 조정(coordination), 인증, 컨텐츠 충실도(content fidelity), 충돌 관리 및 확장성(scalability) 등의 영역들에서 다양한 어려움을 제기한다. 예를 들어, 이용 가능한 서비스의 수가 급격히 증가함에 따라, 장래의 서비스들이 이전에 생성된 애플리케이션들과 완전한 후방 호환성을 유지하면서 애플리케이션에 쉽고 유연하게 포함될 수 있도록 이들 애플리케이션이 확장 가능해야만 한다.

#### 발명의 내용

[0002] 이 요약은 이하에서 상세한 설명에 더 기술되는 일련의 개념들을 간략화된 형태로 소개하기 위해 제공된 것이다. 이 요약은 청구된 발명 대상의 중요한 특징들 또는 필수적인 특징들을 확인하기 위한 것이 아니며, 청구된 발명 대상의 범위를 정하는 데 보조 수단으로 사용되기 위한 것도 아니다.

[0003] 분산형 애플리케이션을 생성하는 방법은 원격 서버 클러스터(remote server cluster) 상에서 이용 가능한 컴포넌트들의 목록 중에서 일군의 컴포넌트들을 선택하는 단계를 포함한다. 선택된 컴포넌트들을 설치하는 데 필요한 데이터가 원격 서버 클러스터로부터 수신된다. 수신된 데이터에 응답하여 명령어들의 목록이 생성되고, 이 명령어들의 목록이 저장된다.

[0004] 유형의 컴퓨터-판독 가능 매체는 분산형 애플리케이션을 생성하는 컴퓨터-실행 가능 명령어들을 갖는다. 이 명령어들은 분산 컴퓨터 시스템 상에서 이용 가능한 서비스들의 목록 중에서 서비스를 선택하는 것을 포함한다. 선택된 서비스에서 이용 가능한 컴포넌트가 선택된다. 선택된 컴포넌트에 관련된 설치 데이터가 분산 컴퓨터 시스템으로부터 수신된다. 수신된 데이터에 응답하여 패키지 파일이 생성된다. 이 패키지 파일이 저장된다.

[0005] 분산형 컴퓨팅 애플리케이션의 제어를 중앙집중화하는 시스템은 프로세서 및 컴퓨터-판독 가능 매체를 포함한다. 이 시스템은 또한 컴퓨터-판독 가능 매체 상에 저장되고 프로세서 상에서 실행되는 운영 환경을 포함한다. 또한, 컴퓨터-판독 가능 매체 상에 저장되고 프로세서 상에서 실행되는 솔루션 프레임워크(solution framework)가 포함되어 있다. 이 솔루션 프레임워크는 서비스 클러스터 상에서 이용 가능한 서비스들의 목록 중에서 서비스를 선택하도록 구성되어 있다. 선택된 서비스에서 이용 가능한 컴포넌트가 선택된다. 선택된 서비스로부터, 선택된 컴포넌트(들)에 관련된 설치 데이터가 수신된다. 수신된 데이터에 응답하여 패키지 파일이 생성된다. 이 패키지 파일이 이어서 컴퓨터-판독 가능 매체 상에 저장된다.

[0006] 이들 및 다른 특징들 및 이점들이 이하의 상세한 설명을 읽어보고 관련 도면들을 살펴보면 명백하게 될 것이다. 이상의 개괄적인 설명 및 이하의 상세한 설명 둘다가 단지 설명을 위한 것이며 제한하는 것이 아니라는 것을 잘 알 것이다. 그 중에서도 특히, 본 명세서에 기술된 다양한 실시예들이 방법, 장치, 또는 이들의 조합으로서 구현될 수 있다. 마찬가지로, 다양한 실시예들이 하드웨어로만 된 실시예, 소프트웨어로만 된 실시예, 또는 소프트웨어 양태 및 하드웨어 양태를 겸비한 실시예의 형태를 취할 수 있다. 따라서, 본 명세서의 개시 내용은 제한하는 의미로 보아서는 안 된다.

#### 도면의 간단한 설명

[0007]

도면에서, 유사한 참조 번호는 유사한 구성요소를 나타낸다.

도 1은 본 명세서에 기술된 컴퓨터-구현 방법의 구현예들에 대한 운영 환경의 블록도이다.

도 2는 솔루션 프레임워크의 구현예에 대한 운영 환경을 나타낸 블록도이다.

도 3은 패키지 파일의 구현예를 나타낸 흐름도이다.

도 4는 일군의 컴포넌트를 선택하는 동작을 나타낸 흐름도이다.

도 5는 패키지 파일을 생성하는 동작을 나타낸 흐름도이다.

도 6은 분산 컴퓨팅 환경에서 패키지 파일을 배포하는 동작을 나타낸 흐름도이다.

### 발명을 실시하기 위한 구체적인 내용

[0008]

이제부터, 유사한 참조 번호가 유사한 구성요소를 나타내고 있는 도면들을 참조하여, 다양한 실시예들에 대해 기술할 것이다. 상세하게는, 도 1 및 대응하는 설명은 실시예들이 구현될 수 있는 적당한 컴퓨팅 환경에 대한 간략하고 전반적인 설명을 제공하기 위한 것이다.

[0009]

일반적으로, 최종 사용자와 서비스 클러스터들 사이에서 인터페이스하는 일을 맡고 있는 솔루션 프레임워크가 제공된다. 솔루션 프레임워크는 분산형 애플리케이션의 생성 및 배포 둘다를 중앙집중화할 수 있다. 분산형 애플리케이션의 생성 및 배포 둘다에 많은 이점들이 존재한다. 예를 들어, 분산형 애플리케이션은 많은 개별적인 서비스의 기능을 하나의 애플리케이션으로 결합시키기 위해 다수의 서로 다른 원격 서버 상에 위치하는 다수의 서비스 상의 컴포넌트들을 이용할 수 있다.

[0010]

솔루션 프레임워크는 생성 사용자 인터페이스와 서비스 클러스터 간의 매개체로서 역할함으로써 분산형 애플리케이션의 생성을 중앙집중화한다. 사용자는 특정의 서비스를 통합시키는 패키지 파일을 생성할 수 있게 하기 전에, 사용자는 그 서비스에서 인증을 받을 필요가 있다. 솔루션 프레임워크는 또한 사용자 자격증명 정보를 각각의 서비스 클러스터로 전달함으로써 이 인증 프로세스를 중앙집중화할 수 있다. 이어서, 솔루션 프레임워크는 인증된 사용자가 서비스 클러스터에서 이용가능한 다양한 컴포넌트들을 선택함으로써 분산형 애플리케이션을 정의할 수 있게 할 수 있다. 애플리케이션을 정의한 것에 응답하여, 패키지 파일이 생성된다. 파일의 컨텐츠에 기초하여 파일 내에 포함되어 있는 워터마크 메타데이터(watermark metadata)를 사용하여 패키지 파일의 컨텐츠 충실도가 유지된다.

[0011]

애플리케이션이 설치되기 전에, 패키지 파일이 애플리케이션을 정의하는 사용자로부터 다른 사용자로 전송되거나, 단순히 애플리케이션을 정의하는 사용자에 의해 설치될 수 있다. 설치 동안에, 충돌 해결도 역시 솔루션 프레임워크에 의해 중앙집중식으로 제어될 수 있다. 충돌이 해결된 경우, 솔루션 프레임워크는 모든 관련 서비스 클러스터와 통신을 할 수 있고 정의된 패키지 파일에 기초하여 컴포넌트들을 올바른 위치에 설치할 수 있다.

[0012]

이제부터 도 1을 참조하여, 다양한 실시예들에서 이용되는 컴퓨터(100)에 대한 컴퓨터 아키텍처에 대해 기술할 것이다. 도 1에 도시된 컴퓨터 아키텍처는 테스크톱 컴퓨터 또는 모바일 컴퓨터(mobile computer)로서 구성될 수 있고, 중앙 처리 장치("CPU")(5), 랜덤 액세스 메모리("RAM")(9) 및 판독 전용 메모리("ROM")(10)를 포함하는 시스템 메모리(7), 및 메모리를 CPU(5)에 결합시키는 시스템 버스(12)를 포함하고 있다.

[0013]

시동 중과 같은 때에 컴퓨터 내의 구성요소들 사이의 정보 전송을 돋는 기본 루틴을 포함하는 기본 입/출력 시스템은 ROM(10)에 저장되어 있다. 컴퓨터(100)는 또한 이하에서 보다 상세히 기술하게 될, 운영 체제(16), 애플리케이션 프로그램(24), 및 솔루션 프레임워크(26)를 저장하는 대용량 저장 장치(14)도 포함하고 있다.

[0014]

대용량 저장 장치(14)는 버스(12)에 연결된 대용량 저장장치 제어기(도시 생략)를 통해 CPU(5)에 연결되어 있다. 대용량 저장 장치(14) 및 그와 연관된 컴퓨터-판독가능 매체는 컴퓨터(100)에 대한 비휘발성 저장을 제공한다. 본 명세서에 포함된 컴퓨터-판독가능 매체에 대한 설명이 하드 디스크 또는 CD-ROM 드라이브 등의 대용량 저장 장치를 언급하고 있지만, 컴퓨터(100)에 의해 액세스 가능한 이용가능한 매체는 그 어떤 것이든지 컴퓨터-판독가능 매체가 될 수 있다.

[0015]

예로서, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체 및 통신 매체를 포함할 수 있지만 이에 제한되는 것은 아니다. 컴퓨터 저장 매체는 컴퓨터-판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보를 저장하는 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 이동식 및 비이동식 매체를 포함한다. 컴퓨터 저장 매체는 RAM, ROM, EPROM, EEPROM, 플래쉬 메모리 또는 기타 고상 메모리(solid state memory) 기술,

CD-ROM, DVD(digital versatile disk) 또는 기타 광 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 기타 자기 저장 장치, 또는 컴퓨터(100)에 의해 액세스될 수 있고 원하는 정보를 저장하는 데 사용될 수 있는 임의의 기타 매체를 포함하지만 이에 제한되는 것은 아니다.

다양한 실시예들에 따르면, 컴퓨터(100)는 인터넷과 같은 네트워크(18)를 통한 원격 컴퓨터로의 논리적 접속을 사용하여 네트워크화된 환경에서 동작할 수 있다. 컴퓨터(100)는 버스(12)에 연결된 네트워크 인터페이스 유닛(20)을 통해 네트워크(18)에 연결할 수 있다. 이 네트워크 연결은 무선 및/또는 유선일 수 있다. 네트워크 인터페이스 유닛(20)은 또한 기타 유형의 네트워크 및 원격 컴퓨터 시스템에 연결하는 데도 이용될 수 있다. 컴퓨터(100)는 또한 키보드, 마우스, 또는 전자 스타일러스(도 1에 도시되지 않음)를 비롯한 다수의 다른 장치들로부터 입력을 수신하여 처리하는 입/출력 제어기(22)도 포함할 수 있다. 유사하게, 입/출력 제어기(22)는 디스플레이 화면, 프린터, 또는 기타 유형의 출력 장치에 출력력을 제공할 수 있다.

앞서 간단히 언급한 바와 같이, 미국 워싱턴주 레드몬드 소재의 MICROSOFT CORPORATION의 WINDOWS VISTA 운영 체제 등의, 네트워크화된 퍼스널 컴퓨터의 동작을 제어하는 데 적합한 운영 체제(16)를 비롯한 다수의 프로그램 모듈 및 데이터 파일이 컴퓨터(100)의 대용량 저장 장치(14) 및 RAM(9)에 저장될 수 있다. 대용량 저장 장치(14) 및 RAM(9)은 또한 하나 이상의 프로그램 모듈도 저장할 수 있다. 상세하게는, 대용량 저장 장치(14) 및 RAM(9)은 또한 하나 이상의 애플리케이션 프로그램(24)을 저장할 수 있다. 예를 들어, 대용량 저장 장치(14)는 솔루션 프레임워크(26)를 저장할 수 있다. 솔루션 프레임워크(26)는 분산형 애플리케이션의 개발 및 설치를 중앙집중화한다.

도 2는 솔루션 프레임워크(26)가 동작하는 환경의 구현예를 나타낸 것이다. 솔루션 프레임워크(26)는 서비스 클러스터(230)에 결합될 수 있다. 서비스 클러스터(230)는 인터넷 또는 엑스트라넷(extranet) 등의 네트워크를 통해 솔루션 프레임워크(26)에 결합될 수 있다. 서비스 클러스터(230)는 N개의 개별 서비스[서비스(232, 234, 236)]를 포함할 수 있다. 개별 서비스는 서로 다른 장소에 저장될 수 있다. 예를 들어, 서비스(230)는 제1 장소에 있는 제1 서버에 저장될 수 있고, 서비스(240)는 제2 장소에 있는 제2 서버에 저장될 수 있다. 다른 구현 예들에서, 다수의 서비스가 하나의 장소에 저장될 수 있다. 그에 따라, 솔루션 프레임워크는 개별 서비스의 장소와 무관하게 동작할 수 있다.

각각의 서비스는 다양한 형태의 기능을 제공할 수 있다. 예를 들어, 서비스는 비즈니스 데이터, 페이지 설계 또는 레이아웃을 포함할 수 있다. 다른 예들에서, 서비스는 비즈니스 로직을 포함할 수 있다. 또 다른 예들에서, 서비스는 임의의 다른 형태의 기능을 포함할 수 있다.

솔루션 프레임워크(26)는 또한 생성 인터페이스(210) 및 배포 인터페이스(220)에도 결합될 수 있다. 생성 인터페이스(210)는 사용자가 일군의 선택된 서비스를 포함하는 패키지 파일을 정의할 수 있게 하는 인터페이스를 제공할 수 있다. 배포 인터페이스(220)는 사용자가 일군의 선택된 서비스를 설치하기 위해 패키지 파일을 배포할 수 있게 하는 인터페이스를 제공할 수 있다. 일부 구현 예들에서, 이들 인터페이스는 HTML(Hypertext Markup Language) 또는 XML(Extensible Markup Language)와 같은 마크업 언어로 코딩된 웹 인터페이스일 수 있다. 다른 구현 예들에서, 이들 인터페이스는 C# 또는 Java 등의 다른 언어들로 코딩될 수 있다.

생성 인터페이스(210)는 제1 장소에 있는 제1 컴퓨팅 환경에 위치할 수 있고 제1 사용자가 패키지 파일을 생성 할 수 있게 할 수 있는 반면, 배포 인터페이스(220)는 제2 장소에 있는 제2 컴퓨팅 환경에 위치할 수 있고 제2 사용자가 패키지 파일을 배포할 수 있게 할 수 있다. 이러한 구현예에서, 패키지 파일은 제1 컴퓨팅 환경으로부터 제2 컴퓨팅 환경으로 전송될 수 있다. 이러한 전송은 임의의 파일 전송 수단에 의해 달성될 수 있다. 예를 들어, 파일이 제2 컴퓨팅 환경으로 물리적으로 전달되는 디스크 등의 컴퓨터-판독가능 매체 상에 인코딩될 수 있다. 다른 예에서, 파일은 인터넷을 통해 전송되는 이메일 첨부 파일 등에 의해 2개의 컴퓨팅 환경 간의 네트워크 연결을 통해 저자적으로 전송될 수 있다.

다른 구현 예들에서, 생성 인터페이스(210) 및 배포 인터페이스(220)는 동일한 컴퓨팅 환경에 있을 수 있다. 이러한 구현 예에서, 동일한 사용자는 동일한 컴퓨팅 환경으로부터 패키지 파일의 생성 및 배포 둘다를 할 수 있다. 게다가 이러한 구현 예에서 이상에서 요구되는 패키지 파일의 전송이 회피될 수 있다.

그에 따라, 솔루션 프레임워크(26)는 패키지 파일의 생성 및 패키지 파일의 배포 둘다를 관리하고 중앙집중화한다.

도 3은 패키지 파일(300)의 한 구현예를 나타낸 것이다. 패키지 파일(300)은 768 비트(비트 0 내지 비트 767)를 포함할 수 있다. 패키지 파일(300)은 비트 0 내지 비트 31에 있는 헤더 부분을 포함할 수 있다. 패키지 파일(300)은 비트 32 내지 비트 767에 있는 데이터 부분을 포함할 수 있다.

일(300)은 선택된 컴포넌트에 대해 기술하는 비트 32 내지 비트 63에 있는 매니페스트(manifest)도 포함할 수 있다. 매니페스트는 잠재적인 충돌 및 컴포넌트 이용가능성을 검출하기 위해 서비스들에 의해 사용될 수 있는 컴포넌트들에 관한 정보를 기술하는, 서비스 클러스터(230)로부터 수신된 정보를 포함하고 있을 수 있다. 예를 들어, 매니페스트는 패클릿(packlet)에 있는 컴포넌트들에 대해 기술하는, 패클릿 생성 시에 각각의 서비스 클러스터에 의해 반환되는 정보의 일부분을 포함할 수 있다. 물리적 패클릿이 없더라도 매니페스트 내의 메타-정보가 충돌을 검출하는 데 사용될 수 있도록 매니페스트가 설계된다. 헤더 및 매니페스트 다음에, 패키지 파일(300)은 비트 64 내지 비트 127에 있는 페이로드를 포함할 수 있다. 도 5를 참조하여 이하에서 더 상세히 기술되는 바와 같이, 패키지 파일(300)의 페이로드 부분은 선택된 서비스들로부터 수신된 정보를 포함하고 있다.

[0025] 보안을 위해, 패키지 파일(300)은 비트 128 내지 비트 639에 있는 512-비트 비밀키(Secret Key) 등의 공개키(public key)를 포함할 수 있다. 추가의 보안을 위해 그리고 파일 무결성의 검증을 가능하게 하기 위해, 도 6을 참조하여 이하에서 더 상세히 기술되는 바와 같이, 패키지 파일(300)은 비트 640 내지 비트 767에 있는 암호화된 128-비트 SHA-1(Secure Hash Algorithm 1) 해쉬 등의 워터마크를 포함할 수 있다. 다른 실시예들에서, 워터마크는 MD5(Message-Digest algorithm 5) 해쉬를 포함할 수 있다. 이 해쉬는 임의의 해싱 알고리즘을 패키지 파일 내의 페이로드 및 매니페스트에 적용함으로써 생성될 수 있다.

#### 솔루션 프레임워크 동작

[0027] 이제부터 도 4를 참조하여, 분산형 애플리케이션에 포함될 일련의 컴포넌트들을 정의하는 예시적인 프로세스(400)에 대해 기술할 것이다.

[0028] 본 명세서에 제공된 루틴들에 대한 설명을 보면, 다양한 실시예들의 논리 연산들이 (1) 컴퓨팅 시스템 상에서 실행되는 일련의 컴퓨터 구현 동작들 또는 프로그램 모듈들로서 및/또는 (2) 컴퓨팅 시스템 내의 상호연결된 기계 논리 회로들 또는 회로 모듈들로서 구현된다는 것을 잘 알 것이다. 이 구현은 본 발명을 구현하는 컴퓨팅 시스템의 성능 요건에 따른 선택의 문제이다. 그에 따라, 예시되어 있고 본 명세서에 기술된 실시예들을 이루고 있는 논리 연산들이 연산, 구조적 장치, 동작 또는 모듈 등 여러가지로 지칭될 수 있다. 이들 연산, 구조적 장치, 동작 및 모듈이 소프트웨어로, 펌웨어로, 특수 목적의 디지털 논리로, 또한 이들의 임의의 조합으로 구현될 수 있다.

[0029] 시작 동작 후에, 이 프로세스는 동작(410)으로 가서, 서비스들의 목록이 솔루션 프레임워크(26)에 수신된다. 이 프로세스는, 예를 들어, 사용자가 생성 인터페이스(210) 상에서 패키지 생성 페이지를 브라우징할 때 트리거될 수 있다. 이어서, 솔루션 프레임워크는 서비스 클러스터(230)를 호출하여 이용가능한 서비스들의 목록을 요청할 수 있다. 특정의 서비스가 이용가능한지 여부를 판정하기 위해, 솔루션 프레임워크(26)는 생성 인터페이스(210)와 서비스 클러스터(230) 내의 각각의 서비스 간의 퍼미션(permission)을 중앙집중식으로 중재할 수 있다. 예를 들어, 솔루션 프레임워크(26)는 사용자의 프로파일을 서비스 클러스터(230)로 전송하고, 그에 응답하여 사용자가 액세스할 수 있는 일련의 서비스를 수신할 수 있다. 그에 따라, 이용가능성은 솔루션 프레임워크(26)에 네트워크 연결되어 있는 서비스들은 물론 개개의 사용자 퍼미션에도 의존할 수 있다.

[0030] 이용가능한 서비스들의 목록이 솔루션 프레임워크(26)에 수신되어 마무리된 후에, 이 프로세스는 동작(420)으로 가서, 제1 서비스가 선택된다.

[0031] 서비스 내의 컴포넌트들을 보여달라는 사용자 요청을 나타내는, 생성 인터페이스(210)로부터 수신된 서비스를 선택하라는 명령에 응답하여 서비스가 선택될 수 있다. 서비스가 선택된 경우, 이 프로세스는 동작(430)으로 가서, 컴포넌트 유형들의 목록이 수집된다. 이 동작 동안에, 솔루션 프레임워크(26)는 선택된 서비스 상에서 이용가능한 개개의 컴포넌트 유형들의 목록을 전송하라는 요청을 선택된 서비스로 전달한다. 이 유형 정보는 이용가능한 컴포넌트의 유형을 기술하는 정보 및 개개의 컴포넌트를 선택하는 데 사용될 수 있는 사용자 인터페이스를 기술하는 정보를 포함할 수 있다. 일부 구현예들에서, 이 정보는 그 서비스에 특유한 컴포넌트 유형들 및 컴포넌트들을 선택하는 커스텀 사용자 인터페이스를 포함할 수 있다.

[0032] 이 프로세스는 이어서 동작(440)으로 가서, 이용가능한 컴포넌트들의 목록이 수집된다. 이상에 기술한 서비스들의 이용가능성과 유사하게, 컴포넌트들의 이용가능성은 컴포넌트들이 솔루션 프레임워크(26)에 연결되어 있는지 여부는 물론 퍼미션 검사(permission check)에도 의존할 수 있다.

[0033] 이 프로세스는 이어서 동작(450)으로 가서, 수신되는 선택된 컴포넌트들과 연관된 유형 정보가 처리되고 생성 인터페이스(210)로 전송되며, 이 생성 인터페이스(210)에서 선택 인터페이스가 사용자에게 제공된다. 특정 유형의 이용가능한 컴포넌트들을 선택하는 것을 용이하게 하도록 커스텀 선택 인터페이스가 디스플레이될 수 있도록

록, 이 선택 인터페이스가 선택된 서비스로부터 수신되는 유형 정보에 응답하여 생성된다. 이 커스텀 인터페이스는 사용자가 선택된 컴포넌트에 임의의 파라미터를 입력하는 메커니즘을 포함할 수 있다. 이를 파라미터는 선택된 컴포넌트가 페키징되어야 하는 방식을 솔루션 프레임워크 및 서비스 클러스터에 설명하는 부가의 메타-정보로서 기능할 수 있다.

[0034] 이 프로세스는 이어서 동작(460)으로 가서, 사용자가 도 2의 생성 인터페이스(210) 상에 제공된 선택 인터페이스를 사용하여 원하는 컴포넌트들을 선택한다. 일부 구현예들에서, 이 동작 동안에, 선택된 컴포넌트들의 목록이 서비스 클러스터(230)로 전송되어 서비스 클러스터(230)에 수신된다. 다른 구현예들에서, 동작(470) 이후에 각각의 선택된 서비스에 대해 이 프로세스가 완료될 때까지 선택된 컴포넌트들의 목록이 서비스 클러스터(230)로 전송되지 않는다.

[0035] 동작(470)으로 가서, 부가의 서비스가 선택되는지 여부에 관한 결정이 행해진다. 사용자가 부가의 서비스를 선택하는 경우, 이 프로세스는 동작(420)으로 되돌아가서, 그 다음의 선택된 서비스에 대해 인터페이스-발생 및 컴포넌트-선택 프로세스가 반복된다. 사용자가 부가의 서비스를 선택하지 않는 경우, 이 프로세스는 동작(480)으로 계속된다.

[0036] 동작(480)으로 계속하여, 도 5를 참조하여 이하에서 더 상세히 기술되는 바와 같이, 모든 패클릿들이 결합되어 하나의 페키지 파일로 포맷팅(format)된다. 이 프로세스는 이어서 종료 동작으로 가고 다른 동작들의 처리로 되돌아간다.

[0037] 이제부터 도 5를 참조하여, 동작(480)에서 생성되는 것과 같은 페키지 파일을 생성하는 예시적인 프로세스(500)에 대해 기술할 것이다.

[0038] 시작 동작 후에, 이 프로세스는 동작(510)으로 가서, 선택된 컴포넌트들의 목록이 솔루션 프레임워크(26)에 의해 처리된다. 선택된 컴포넌트들의 목록이 생성 인터페이스(210)로부터 솔루션 프레임워크(26)에 수신될 수 있다. 선택된 컴포넌트들의 목록이, 예를 들어, 도 4에 예시된 프로세스(400)에 따라 생성된 것일 수 있다.

[0039] 이 프로세스는 이어서 동작(520)으로 가서, 선택된 컴포넌트가 위치하는 제1 서비스가 선택된다. 이 선택은 솔루션 프레임워크(26)가 선택된 컴포넌트들의 목록 상의 컴포넌트들이 포함되어 있는 모든 서비스들에 걸쳐 반복하는 자동적인 프로세스의 일부이다. 따라서, 생성 인터페이스(210)를 통한 사용자의 개입, 또는 사용자로부터의 입력 없이 솔루션 프레임워크(26)에 의해 직접 서비스가 선택될 수 있다. 솔루션 프레임워크(26)가 제1 서비스를 선택한 경우, 이 프로세스는 동작(530)으로 가서, 선택된 서비스와 연관된 컴포넌트들에 대한 설명이 솔루션 프레임워크로부터 제1 선택된 서비스로 전송된다. 즉, 솔루션 프레임워크(26)는 사용자가 선택한 서비스 내에 위치한 컴포넌트들의 목록을 그 서비스로 전송한다.

[0040] 그 서비스가 컴포넌트들의 목록을 수신한 경우, 이 프로세스는 동작(540)으로 가서, 그 서비스는 선택된 컴포넌트들과 연관된 정보를 포함하는 패클릿을 전송함으로써 그 요청에 응답한다. 패클릿은 요청된 데이터를 포함하는 이진 데이터 스트림은 물론 요청된 데이터에 관한 메타-정보를 기술하는 매니페스트도 포함할 수 있다.

[0041] 동작(550)에서, 다른 서비스가 선택되는지 여부에 관한 판정이 행해진다. 솔루션 프레임워크(26)가 선택된 컴포넌트들 전부를 처리하는 데 필요한 서비스들 전부를 순환하여 처리한 경우, 더 이상 서비스가 선택될 필요가 없으며, 이 프로세스는 동작(560)으로 계속된다. 솔루션 프레임워크(26)가 선택된 컴포넌트들에 대한 서비스들 전부를 순환하지 않은 경우, 이 프로세스는 동작(520)으로 되돌아가서, 그 다음 서비스가 선택된다. 다른 실시 예들에서, 이 프로세스가 비동기적으로 실행될 수 있다. 즉, 멀티-쓰래드 환경(multi-threaded environment)에서 솔루션 프레임워크(26)는 모든 서비스들을 동시에 호출하여 패클릿을 수신하고 이어서 패클릿들이 돌아올 때 이들을 조립한다. 따라서, 이 프로세스는 비동기적으로도 선형적/순차적으로도 실행될 수 있다.

[0042] 동작(560)에서, 수신된 패클릿들이 페키지 파일에 부가된다. 예를 들어, 패클릿들이 서로 연결되어 페키지 파일(300)의 페이로드 부분에 부가될 수 있다. 다른 구현예들에서, 패클릿들이 페키지 파일(300)의 페이로드 내에 포함되기 전에, 패클릿들에 대한 추가적인 처리가 실행될 수 있다.

[0043] 동작(570)에서, 워터마크가 페키지 파일(300)에 부가된다. 일부 구현예들에서, 워터마크는 단순히 패클릿들로부터 생성된 해쉬일 수 있다. 다른 구현예들에서, 워터마크는 패클릿들 및 페키지 파일에 포함된 기타 데이터(헤더 부분, 매니페스트 부분, 또는 비밀키와 같은 보안 부분 등)로부터 생성될 수 있다. 일부 예들에서, 워터마크는 SHA-1 해싱 알고리즘을 사용하여 생성될 수 있다. 그에 따라, 해쉬는 나중에 데이터의 무결성을 검증하는 데 사용될 수 있으며, 페키지 파일(300)이 변조되었는지 여부를 판정하기 위해 포함될 수 있다. 이 프로세

스는 이어서 종료 동작으로 가고 다른 동작들의 처리로 되돌아간다.

[0044] 이제부터 도 6을 참조하여, 패키지 파일(300)을 배포하는 예시적인 프로세스(600)에 대해 기술할 것이다.

[0045] 시작 동작 후에, 이 프로세스는 동작(610)으로 가서, 패키지 파일(300)이 배포 인터페이스(220)로부터 솔루션 프레임워크(26)에 수신된다. 일부 예들에서, 패키지 파일(300)이 패키지 파일(300)의 생성 동안에 패키지 파일(300)을 수신했던 시스템으로부터 바로 솔루션 프레임워크(26)로 전송될 수 있다. 이것은 패키지 파일(300)을 생성했던 사용자가 또한 이 패키지 파일(300)을 배포하는 상황들에서 일어날 수 있다. 다른 예들에서, 패키지 파일(300)이 다른 사용자로부터 솔루션 프레임워크(26)로 전송될 수 있다. 예를 들어, 패키지 파일(300)이 생성 인터페이스(210)에서 제1 사용자에 의해 수신되고 이어서 컴팩트 디스크(CD)를 통해 전달되거나 배포 인터페이스(220)에서 이메일을 통해 제2 사용자에게 전송될 수 있고, 이어서 솔루션 프레임워크(26)로 전송된다.

[0046] 패키지 파일(300)이 솔루션 프레임워크(26)에 수신된 경우, 이 프로세스는 동작(620)으로 간다. 동작(620)에서, 패키지 파일(300)이 유효한 것인지 여부의 판정이 행해진다. 이 판정은 워터마크를 참조하여 행해질 수 있다. 예를 들어, 패키지 파일(300)이 다양한 전송 동안에 오염되었거나 패키지 파일(300)이 의도적으로 변조된 경우, 워터마크가 패키지 파일(300) 내에 포함된 데이터와 더 이상 적절히 부합되지 않을 수 있다. 다른 예들에서, 패키지 파일이 유효한 것인지 여부를 판정하는 데 다른 기준들이 사용될 수 있다. 예를 들어, 파일 확장자가 변경되었는지 여부가 참조될 수 있다. 다른 예들에서, 파일 크기가 참조될 수 있다. 즉, 기본 최대 파일 크기가 할당될 수 있고 기본 파일 크기보다 큰 어떤 것이라도 유효하지 않은 것으로 플래깅(flag)될 수 있다. 따라서, 패키지 파일(300)이 유효한 것인지 여부의 판정이 부분적으로는 워터마크에 의존할 수 있고 부분적으로는 패키지 파일(300)의 다른 속성들에 의존할 수 있다. 그에 따라, 패키지 파일(300)이 유효한 경우, 이 프로세스는 동작(630)으로 가서, 처리가 계속된다.

[0047] 패키지 파일(300)이 더 이상 유효하지 않은 경우, 이 프로세스는 동작(680)으로 가서, 배포가 중단되고, 이 프로세스는 종료 동작으로 계속된다. 일부 예들에서, 중단 동작(680)은 배포 인터페이스에서 사용자에게 오류 메시지를 제공하는 것을 포함할 수 있다. 다른 예들에서, 중단 동작(680)은 또한 정정가능한 오류들이 솔루션 프레임워크(26)에 의해 자동으로 정정되어 그 프로세스가 계속될 수 있도록 자동 오류 정정도 포함할 수 있다.

[0048] 동작(630)에서, 패키지 파일(300)이 처리되고 페이로드가 추출된다. 어느 서비스들이 필요한지를 결정하기 위해 페이로드가 처리된다. 이어서, 솔루션 프레임워크(26)는 패키지 파일에서 필요로 하는 서비스들 모두가 이용가능한지 여부를 판정한다. 패키지 파일(300) 내에 포함되어 있는 컴포넌트가 특정의 서비스에 존재하는 경우, 그 서비스가 필요하게 된다. 예를 들어, 패키지 파일(300)이 생성되었을 때와 패키지 파일(300)이 배포되고 있을 때 사이에 시간이 경과한 경우, 서비스 클러스터(230)에서 이용가능한 서비스들 중 하나 이상이 더 이상 이용가능하지 않을 수 있다.

[0049] 이상에서 기술된 바와 같이, 예를 들어, 패키지 파일(300)을 배포하는 사용자가 특정의 서비스에 액세스할 퍼미션을 갖고 있지 않은 경우 또는 서비스가 단순히 네트워크로부터 분리된 경우에도, 서비스가 더 이상 이용가능하지 않을 수 있다. 필요한 서비스가 이용가능하지 않은 경우, 이 프로세스는 동작(680)으로 가서, 배포가 중단된다. 필요한 서비스들 모두가 이용가능한 경우, 이 프로세스는 동작(640)으로 간다.

[0050] 동작(640)에서, 제1 서비스가 선택된다. 이 선택은 솔루션 프레임워크(26)가 패키지 파일(300)의 페이로드 내의 컴포넌트들에 의해 이용되는 서비스들 모두를 통해 반복되는 자동적인 프로세스의 일부이다. 따라서, 배포 인터페이스(220)를 통한 사용자의 개입, 또는 사용자로부터의 입력 없이 솔루션 프레임워크(26)에 의해 직접 서비스가 선택될 수 있다. 솔루션 프레임워크(26)가 제1 서비스를 선택하는 경우, 이 프로세스는 동작(650)으로 가서, 충돌 검사가 실행된다.

[0051] 동작(650)에서, 패키지 파일(300)의 컨텐츠가 서비스 클러스터(230)로 전송될 수 있다. 일부 구현 예들에서, 고비용의 전송 시간을 절감하기 위해 매니페스트만이 전송될 수 있다. 매니페스트는 패클릿에 있는 컴포넌트들에 대해 기술하는, 패클릿 생성 시에 각각의 서비스 클러스터에 의해 반환되는 정보의 일부분을 포함할 수 있다. 물리적 패클릿이 없더라도 매니페스트 내의 메타-정보가 충돌을 검출하는 데 사용될 수 있도록 매니페스트가 설계된다. 이어서, 각각의 서비스가 패키지 파일(300)(또는 다른 구현 예들에서는 매니페스트)의 컨텐츠를 조사하고, 컴포넌트 상세, 컴포넌트 충돌, 및 패키지 파일(300)의 배포 동안에 사용자가 부딪칠 수 있는 임의의 다른 배포 문제를 솔루션 프레임워크(26)에 보고한다. 이 정보를 다시 솔루션 프레임워크(26)에 보고함으로써, 충돌 검사 프로세스의 제어가 중앙집중화될 수 있다. 충돌이 존재하는 경우, 이 프로세스는 동작(670)으로 간다. 충돌이 존재하지 않는 경우, 이 프로세스는 동작(660)으로 가서, 배포 프로세스가 계속된다. 다른 실시 예들에

서, 충돌 검사 프로세스가 비동기적으로 실행될 수 있다. 즉, 멀티-쓰레드 프로세스에서 솔루션 프레임워크(26)는 동시에 충돌을 검출한다. 따라서, 이 프로세스는 비동기적으로도 선형적/순차적으로도 실행될 수 있다.

[0052] 동작(670)에서, 충돌이 솔루션 프레임워크에 의해 자동으로 정정될 수 있는지 여부에 관한 판정이 행해진다. 솔루션 프레임워크가 충돌을 자동으로 정정할 수 있는 경우, 솔루션 프레임워크는 충돌을 자동으로 정정할 수 있고, 프로세스가 동작(660)으로 계속될 수 있다. 일부 구현 예들에서는, 사용자에게 정정된 충돌을 알려주기 위해 배포 인터페이스(220)를 통해 경고 메시지가 사용자에게 제공될 수 있다. 솔루션 프레임워크(26)가 충돌을 자동으로 정정할 수 없는 경우, 치명적인 충돌을 나타내는 오류 메시지가 사용자에게 제공될 수 있고, 이 프로세스는 동작(680)으로 가서, 배포가 중단된다. 또 다른 구현 예들에서, 사용자는 충돌하는 컴포넌트들을 덮어쓰기(overwrite)하는 어떤 충돌이라도 자동으로 오버라이드 override(override)할 수 있는 기능을 제공받을 수 있다. 이러한 구현 예들에서, 사용자는 패키지 파일(300)을 배포하려고 시도하기 전에 이것을 할 필요가 있을 수 있고, 충돌이 일어나는 경우, 컴포넌트들이 패키지 파일(300)로부터의 새로운 컴포넌트들로 덮어쓰기될 수 있다.

[0053] 동작(660)에서, 선택된 컴포넌트들에 관련된 패클릿들을 서비스로 전송함으로써, 선택된 서비스의 컴포넌트들이 배포된다. 각각의 패클릿은 서비스에 특유한 일련의 이진 데이터를 포함할 수 있다. 선택된 서비스는 데이터 스트림을 배포를 위한 관련 데이터로 다시 역직렬화(de-serialize)하는 방법을 알고 있을 수 있다. 데이터가 서비스에 배포되어 있는 경우, 이 프로세스는 동작(690)으로 계속된다.

[0054] 동작(690)에서, 다른 서비스가 선택될 것인지 여부의 판정이 행해진다. 부가의 컴포넌트들이 다른 서비스에 배포되어야만 하는 경우, 이 프로세스는 동작(640)으로 되돌아가서, 그 다음 서비스가 선택된다. 컴포넌트들 모두가 배포된 경우, 이 프로세스는 종료 동작으로 간다.

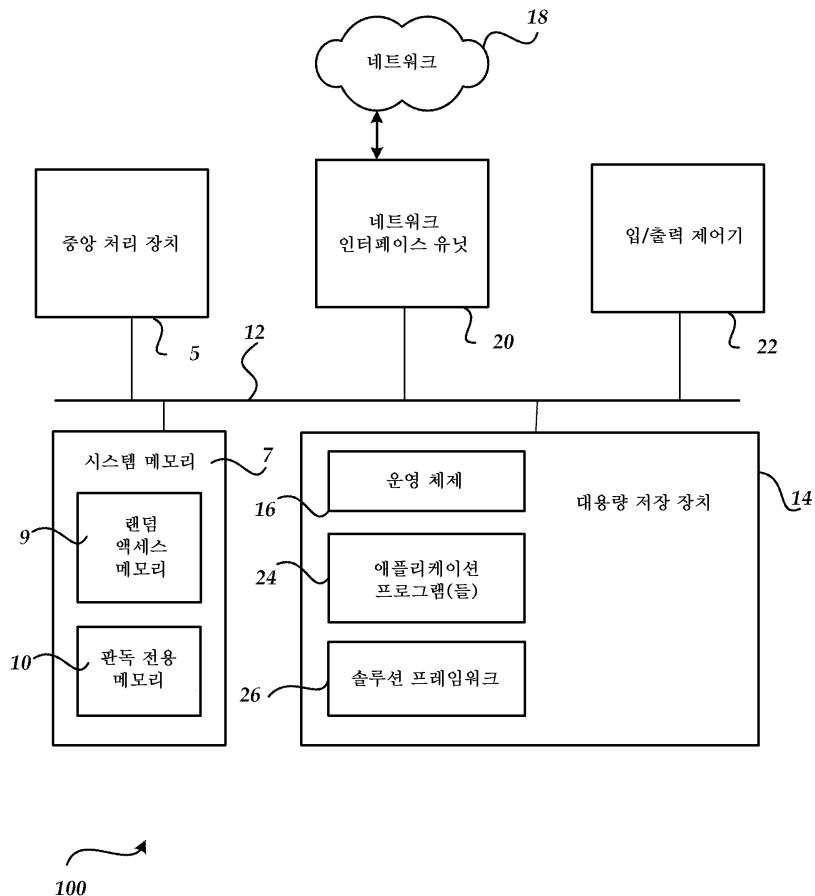
[0055] 다른 구현 예들에서, 어떤 패클릿들도 솔루션 프레임워크(26)로부터 서비스 클러스터(230)로 전송되기 전에 충돌이 있는지 검사하기 위해 선택된 컴포넌트들 모두가 처리될 수 있다. 예를 들어, 서비스들 모두가 순환될 수 있고, 충돌 정보가 솔루션 프레임워크에서 수집되어 중앙집중식으로 처리될 수 있고, 충돌이 제거되는 경우, 패클릿들이 서비스 클러스터(230)로 전송된다. 이와 같이, 모든 충돌이 중앙집중식으로 처리될 수 있다.

[0056] 게다가, 배포후 프로세스(post deployment process)가 실행될 수 있다. 예를 들어, 패클릿을 배포한 후에, 각각의 서비스 클러스터가 배포된 컴포넌트들에 관한 정보를 보고할 수 있다. 이 배포후 정보(post-deployment information)는 이어서 각각의 서비스 클러스터로 다시 전달될 수 있다. 따라서, 전체 패키지의 배포가 완료된 후에 서비스 클러스터들 전부가 다시 한번 전달된다. 이렇게 함으로써 서비스 클러스터가 전체 시스템에 걸쳐 배포된 모든 컴포넌트들에 관한 정보를 통보받을 수 있다. 이와 같이, 각각의 클러스터는 그 정보에 기초하여 배포후 동작들(post-deployment operations)을 실행할 수 있다. 예를 들어, 제1 서비스 클러스터 및 제2 서비스 클러스터 내의 컴포넌트들이 서로 아주 밀접하게 관련되어 있을 수 있고, 서로에 관련된 최종적인 배포 정보를 얇으로써, 서비스들은 이것이 많은 분리된 조각들보다는 엔드-투-엔드의 밀접하게 연계된 분산형 애플리케이션(end-to-end, closely-tied distributed application)의 배포라는 사실을 보강하는 비즈니스 로직을 실행할 수 있다.

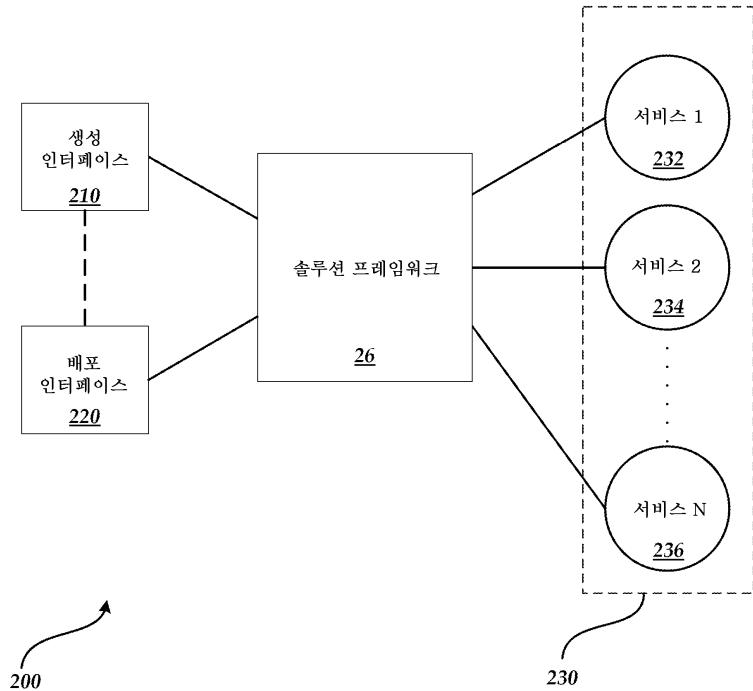
[0057] 이상의 명세서, 예 및 데이터는 본 발명의 구성의 제조 및 사용에 대한 완전한 설명을 제공한다. 본 발명의 많은 실시 예들이 본 발명의 사상 및 범위를 벗어나지 않고 이루어질 수 있기 때문에, 본 발명은 이후에 첨부된 청구항들 내에 속한다.

## 도면

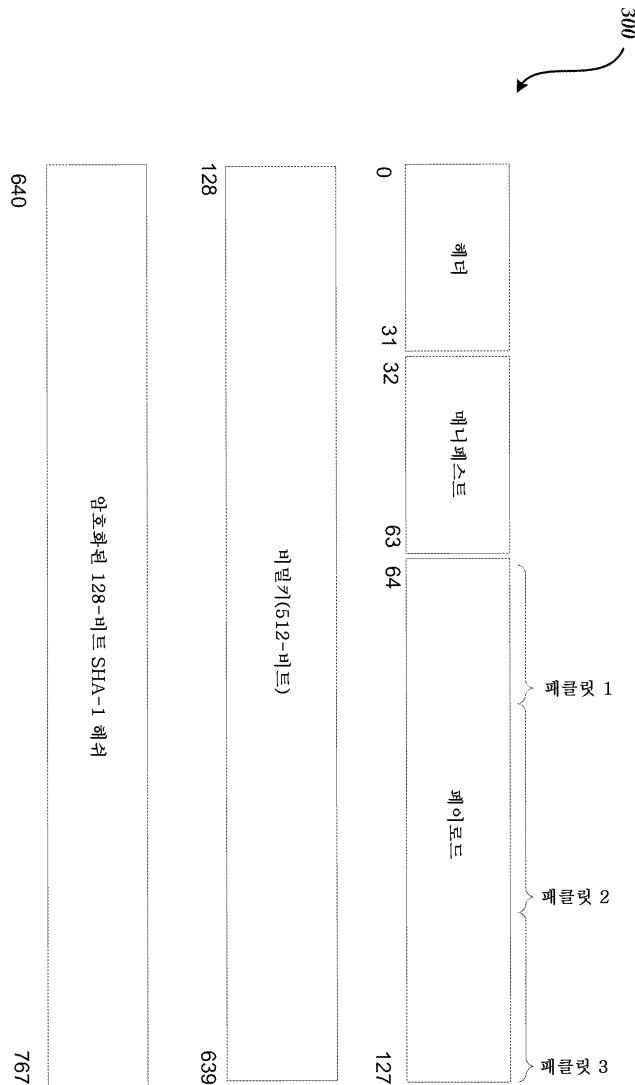
## 도면1



도면2



## 도면3



640

767

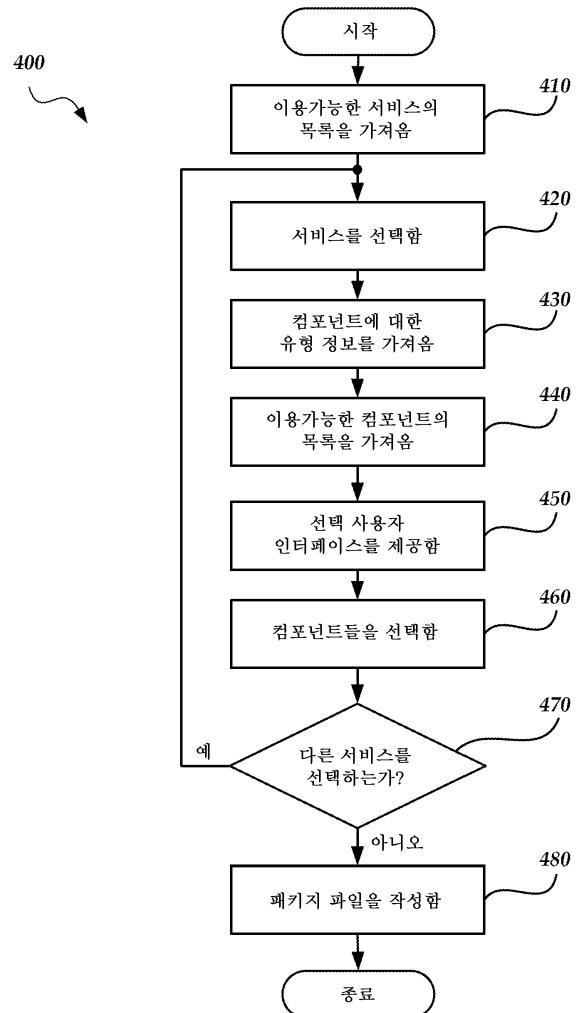
암호화된 128-비트 SHA-1 해시

128

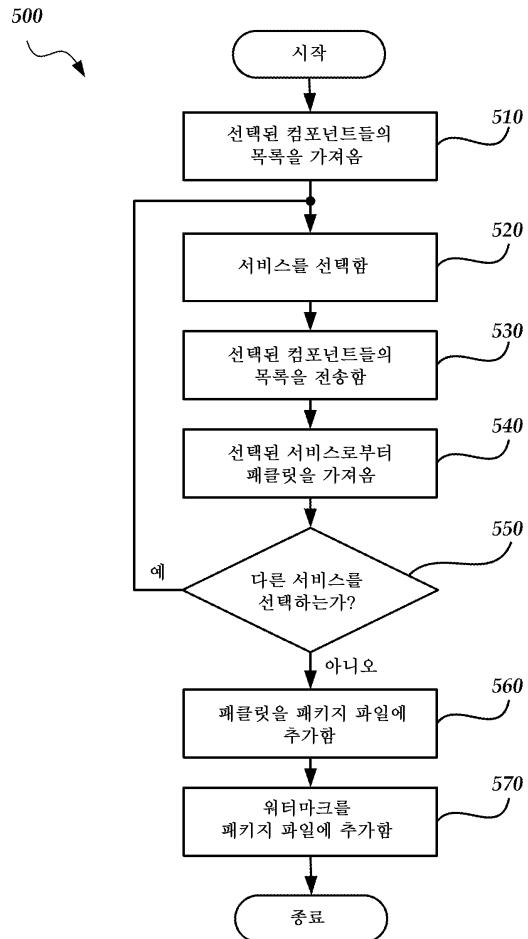
639

비밀키(512-비트)

## 도면4



## 도면5



## 도면6

