



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0039564  
(43) 공개일자 2019년04월12일

- (51) 국제특허분류(Int. Cl.)  
G06F 3/0488 (2013.01) G06F 1/16 (2006.01)  
G06F 3/023 (2006.01) G06F 9/445 (2018.01)  
G06F 9/451 (2018.01)
- (52) CPC특허분류  
G06F 3/04886 (2013.01)  
G06F 1/1643 (2013.01)
- (21) 출원번호 10-2019-7007192
- (22) 출원일자(국제) 2017년09월22일  
심사청구일자 2019년03월12일
- (85) 번역문제출일자 2019년03월12일
- (86) 국제출원번호 PCT/US2017/053007
- (87) 국제공개번호 WO 2018/057925  
국제공개일자 2018년03월29일
- (30) 우선권주장  
62/398,984 2016년09월23일 미국(US)  
15/711,848 2017년09월21일 미국(US)

- (71) 출원인  
애플 인크.  
미국 캘리포니아 (우편번호 95014) 쿠파티노 원  
애플 파크 웨이
- (72) 발명자  
드리센, 크리스토퍼 에이.  
미국 캘리포니아주 쿠파티노 인피니트 루프 1  
테트마이어, 존 알.  
미국 캘리포니아주 쿠파티노 인피니트 루프 1  
켈리, 테일러 에이.  
미국 캘리포니아주 쿠파티노 인피니트 루프 1
- (74) 대리인  
장덕순, 백만기

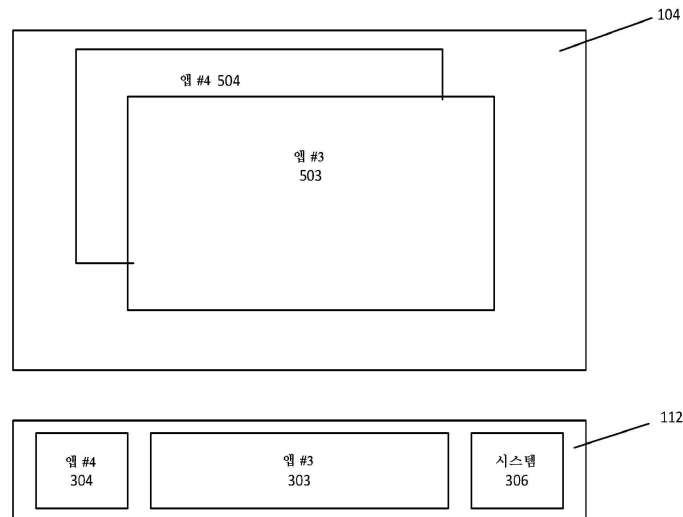
전체 청구항 수 : 총 30 항

(54) 발명의 명칭 동적 기능 열 맞춤화

(57) 요약

디바이스는 터치스크린을 포함하는 동적 기능 열(dynamic function row, DFR)을 포함할 수 있다. 디바이스 프로세서는 DFR을 제어하는 DFR 에이전트를 동작시킬 수 있다. DFR 에이전트는 규칙들 및/또는 컨텍스트들에 기초하여 DFR 상에 디스플레이하기 위한 이미지들을 생성 및/또는 선택할 수 있고, DFR은 이미지들을 디스플레이할 수 있다.

대표도 - 도5d



(52) CPC특허분류

*G06F 1/165* (2013.01)

*G06F 1/1673* (2013.01)

*G06F 3/0238* (2013.01)

*G06F 3/0486* (2013.01)

*G06F 9/44505* (2013.01)

*G06F 9/451* (2018.02)

---

## 명세서

### 청구범위

#### 청구항 1

방법으로서,

디바이스의 메인 디스플레이에 의해, 상기 디바이스의 동적 기능 열(dynamic function row, DFR) 상에 디스플레이하기 위한 제2 사용자 인터페이스(user interface, UI)를 보여주는 제1 UI를 디스플레이하는 단계 - 상기 DFR은 터치스크린을 포함함 -;

상기 디바이스의 입력 디바이스에 의해, 상기 제1 UI를 통해 이루어진 상기 제2 UI에 대한 맞춤화를 수신하는 단계;

상기 디바이스의 DFR 에이전트에 의해, 상기 맞춤화를 통합하는 맞춤화된 제2 UI의 이미지를 생성하는 단계; 및  
상기 DFR에 의해, 상기 이미지를 디스플레이하는 단계를 포함하는, 방법.

#### 청구항 2

제1항에 있어서, 상기 제2 UI는 복수의 등록된 요소들 중 적어도 하나를 포함하고, 각각의 등록된 요소는 상기 디바이스 상에서 실행되는 복수의 애플리케이션들 중 적어도 하나로부터의 것인, 방법.

#### 청구항 3

제2항에 있어서, 상기 맞춤화는, 상기 입력 디바이스에 의해, 적어도 하나의 등록된 요소를 상기 제2 UI의 적어도 일부분을 표현하는 제1 UI 영역으로 드래그하는 것을 포함하는, 방법.

#### 청구항 4

제2항에 있어서, 상기 맞춤화는 하나 이상의 등록된 요소들의 선택, 하나 이상의 등록된 요소들의 배열, 하나 이상의 등록된 요소들의 삭제, 또는 이들의 조합을 포함하는, 방법.

#### 청구항 5

제2항에 있어서,

상기 제1 UI는 등록된 요소들의 디폴트 세트를 포함하고;

상기 맞춤화는 상기 디폴트 세트의 선택을 포함하는, 방법.

#### 청구항 6

제5항에 있어서, 상기 맞춤화는, 상기 입력 디바이스에 의해, 단일 선택으로서 상기 디폴트 세트를 상기 제2 UI의 적어도 일부분을 표현하는 제1 UI 영역으로 드래그하는 것을 포함하는, 방법.

#### 청구항 7

제2항에 있어서, 상기 맞춤화는 적어도 2개의 등록된 요소들 사이에 공간을 추가하는 것을 포함하는, 방법.

#### 청구항 8

제1항에 있어서, 상기 맞춤화는 적어도 하나의 규칙에 의해 제약되는, 방법.

#### 청구항 9

제8항에 있어서, 상기 DFR 에이전트에 의해, 상기 맞춤화가 적어도 하나의 규칙을 위반할 때 상기 맞춤화를 거부하는 단계를 추가로 포함하는, 방법.

**청구항 10**

제1항에 있어서, 상기 맞춤형 제2 UI를 디스플레이하는 단계는, 상기 맞춤형 제2 UI를 상기 제1 UI로부터 상기 DFR로 드래그하라는, 상기 디바이스의 상기 입력 디바이스로부터의 커맨드를 처리하는 단계를 포함하는, 방법.

**청구항 11**

제10항에 있어서, 상기 맞춤형 제2 UI는 복수의 등록된 요소들 중 적어도 하나를 포함하고, 각각의 등록된 요소는 상기 디바이스 상에서 실행되는 복수의 애플리케이션들 중 적어도 하나로부터의 것인, 방법.

**청구항 12**

제11항에 있어서, 상기 커맨드는 하나 이상의 등록된 요소들의 선택, 하나 이상의 등록된 요소들의 배열, 하나 이상의 등록된 요소들의 삭제, 또는 이들의 조합을 포함하는, 방법.

**청구항 13**

제11항에 있어서,

상기 제1 UI는 등록된 요소들의 디폴트 세트를 포함하고;

상기 커맨드는 상기 디폴트 세트의 선택을 포함하는, 방법.

**청구항 14**

제11항에 있어서, 상기 커맨드는 적어도 2개의 등록된 요소들 사이에 공간을 추가하는 것을 포함하는, 방법.

**청구항 15**

제1항에 있어서, 상기 입력 디바이스는 상기 DFR의 상기 터치스크린을 포함하는, 방법.

**청구항 16**

디바이스로서,

제2 사용자 인터페이스(UI)를 보여주는 제1 UI를 디스플레이하도록 구성된 메인 디스플레이;

터치스크린을 포함하고 상기 제2 UI를 디스플레이하도록 구성된 동적 기능 열(DFR);

상기 제1 UI를 통해 이루어진 상기 제2 UI에 대한 맞춤화를 수신하도록 구성된 입력 디바이스; 및

DFR 에이전트에 의해, 상기 맞춤화를 통합하는 맞춤형 제2 UI의 이미지를 생성하도록 구성된 적어도 하나의 프로세서를 포함하며,

상기 DFR은 상기 이미지를 디스플레이하도록 구성되는, 디바이스.

**청구항 17**

제16항에 있어서, 상기 제2 UI는 복수의 등록된 요소들 중 적어도 하나를 포함하고, 각각의 등록된 요소는 상기 디바이스 상에서 실행되는 복수의 애플리케이션들 중 적어도 하나로부터의 것인, 디바이스.

**청구항 18**

제17항에 있어서, 상기 맞춤화는, 상기 입력 디바이스에 의해, 적어도 하나의 등록된 요소를 상기 제2 UI의 적어도 일부분을 표현하는 제1 UI 영역으로 드래그하는 것을 포함하는, 디바이스.

**청구항 19**

제17항에 있어서, 상기 맞춤화는 하나 이상의 등록된 요소들의 선택, 하나 이상의 등록된 요소들의 배열, 하나 이상의 등록된 요소들의 삭제, 또는 이들의 조합을 포함하는, 디바이스.

**청구항 20**

제17항에 있어서,

상기 제1 UI는 등록된 요소들의 디폴트 세트를 포함하고;

상기 맞춤화는 상기 디폴트 세트의 선택을 포함하는, 디바이스.

**청구항 21**

제20항에 있어서, 상기 맞춤화는, 상기 입력 디바이스에 의해, 단일 선택으로서 상기 디폴트 세트를 상기 제2 UI의 적어도 일부분을 표현하는 제1 UI 영역으로 드래그하는 것을 포함하는, 디바이스.

**청구항 22**

제17항에 있어서, 상기 맞춤화는 적어도 2개의 등록된 요소들 사이에 공간을 추가하는 것을 포함하는, 디바이스.

**청구항 23**

제16항에 있어서, 상기 맞춤화는 적어도 하나의 규칙에 의해 제약되는, 디바이스.

**청구항 24**

제23항에 있어서, 상기 적어도 하나의 프로세서는, 상기 DFR 에이전트에 의해, 상기 맞춤화가 적어도 하나의 규칙을 위반할 때 상기 맞춤화를 거부하도록 추가로 구성되는, 디바이스.

**청구항 25**

제16항에 있어서, 상기 맞춤화된 제2 UI를 디스플레이하는 것은, 상기 맞춤화된 제2 UI를 상기 제1 UI로부터 상기 DFR로 드래그하라는, 상기 디바이스의 상기 입력 디바이스로부터의 커맨드를 처리하는 것을 포함하는, 디바이스.

**청구항 26**

제16항에 있어서, 상기 맞춤화된 제2 UI는 복수의 등록된 요소들 중 적어도 하나를 포함하고, 각각의 등록된 요소는 상기 디바이스 상에서 실행되는 복수의 애플리케이션들 중 적어도 하나로부터의 것인, 디바이스.

**청구항 27**

제17항에 있어서, 상기 커맨드는 하나 이상의 등록된 요소들의 선택, 하나 이상의 등록된 요소들의 배열, 하나 이상의 등록된 요소들의 삭제, 또는 이들의 조합을 포함하는, 디바이스.

**청구항 28**

제17항에 있어서,

상기 제1 UI는 등록된 요소들의 디폴트 세트를 포함하고;

상기 커맨드는 상기 디폴트 세트의 선택을 포함하는, 디바이스.

**청구항 29**

제17항에 있어서, 상기 커맨드는 적어도 2개의 등록된 요소들 사이에 공간을 추가하는 것을 포함하는, 디바이스.

**청구항 30**

제16항에 있어서, 상기 입력 디바이스는 상기 DFR의 상기 터치스크린을 포함하는, 디바이스.

**발명의 설명**

**기술 분야**

[0001] 이전 출원에 대한 이익 주장

[0002] 본 출원은 2016년 9월 23일에 출원된 미국 가출원 제62/398,984호의 이익을 주장하며, 이 출원은 이에 의해 전체적으로 참조로서 포함된다.

[0003] 본 개시내용은 일반적으로 동적 기능 열(dynamic function row)에 대한 디스플레이 데이터 및 커맨드들을 관리하는 것에 관한 것이다.

**배경 기술**

[0004] 데스크톱 및 랩톱 컴퓨터와 같은 많은 컴퓨팅 디바이스들은 키보드들을 포함한다. 많은 키보드들은 키보드 상의 키들의 단일 열 내에 배열될 수 있는 기능 키들(예컨대, F1 내지 F12)을 포함한다. 일부 키보드들은 동일한 열 내에 이스케이프 키 및/또는 다른 키들(스크린 인쇄, 스크롤 잠금, 일시정지 등)을 포함한다. 키들의 이러한 열은 "기능 열"로서 지칭될 수 있다.

**발명의 내용**

[0005] 일부 구현예들에서, 컴퓨팅 디바이스는 컴퓨팅 디바이스의 컨텍스트 특정 기능들에 대한 사용자 액세스를 제공하기 위한 동적 기능 열(DFR)을 포함할 수 있다. 예를 들어, DFR 또는 "터치 바"는 컨텍스트 특정 기능들을 표현하는 그래픽 요소들을 제시하는 동적으로 생성된 그래픽 사용자 인터페이스(GUI)를 디스플레이할 수 있는 조합된 디스플레이 및 입력 디바이스(예컨대, 터치스크린)일 수 있다. DFR은 디스플레이된 그래픽 요소들을 선택하는 사용자 입력을 수신할 수 있고, 컴퓨팅 디바이스 상에서 대응하는 기능들을 호출할 수 있다. DFR은 상이한 상황들에서 상이한 인터페이스 요소들을 제공할 수 있다. 예를 들어, DFR은 사용자가 현재 사용하고 있는 애플리케이션 또는 애플리케이션들에 관련된 제어부들을 제공할 수 있거나, 검출된 이벤트들(예컨대, 수신된 착신 메시지, 또는 기능 열 키들을 디스플레이하라는 사용자 커맨드 등)에 응답하여 특정 제어부들을 제공할 수 있다. 본 명세서에 기술된 시스템들 및 방법들은 이들 및 다른 DFR 특징부들을 가능하게 하고 관리할 수 있다.

[0006] 특정 구현예들은 적어도 하나의 이점들을 제공한다. DFR은 사용자의 컨텍스트 또는 컴퓨팅 디바이스의 사용에 관련된 GUI 기능들을 제공할 수 있다. DFR은 디바이스 컨텍스트에 기초하여, 예를 들어 어떤 애플리케이션들이 사용 중인지 또는 임의의 통지들이 활성화인지 여부에 기초하여, 동적으로 업데이트될 수 있다. 사용자들 및 애플리케이션들은 DFR에 의해 제공되는 GUI들을 맞춤화할 수 있다. GUI는 전통적인 기능 키들과는 대조적으로, DFR을 통해 호출될 기능들에 대한 시각적 큐들을 제공할 수 있다(예컨대, F1은 상이한 애플리케이션들에 대한 상이한 기능들을 가질 수 있지만, 사용자는 키 자체를 보는 것에 의해서는 기능이 무엇인지 알지 못한다). DFR을 관리하는 DFR 에이전트는 GUI 요소들을 자동으로 선택, 배열, 및 변경할 수 있다. DFR 에이전트는 개별 애플리케이션들이 DFR과 직접 상호작용할 필요가 없도록 GUI를 생성하고 입력들을 처리할 수 있다.

[0007] 하나 이상의 구현예들의 상세사항들이 첨부 도면 및 이하의 설명에서 기재된다. 다른 특징들, 양태들, 및 잠재적 이점들이 설명 및 도면으로부터 그리고 청구범위로부터 명백해질 것이다.

**도면의 간단한 설명**

- [0008] 도 1은 DFR을 갖는 예시적인 디바이스이다.
- 도 2는 예시적인 DFR 제어 하드웨어의 블록도이다.
- 도 3은 예시적인 DFR 및 키보드 레이아웃이다.
- 도 4는 예시적인 DFR 에이전트의 블록도이다.
- 도 5a는 동일한 앱에 관련된 데이터가 양쪽 디스플레이 상에 보여지는, 예시적인 메인 디스플레이 및 DFR이다.
- 도 5b는 동일한 앱에 관련된 데이터가 양쪽 디스플레이 상에 보여지는, 예시적인 메인 디스플레이 및 DFR이다.
- 도 5c는 시스템 우선순위 데이터가 DFR 상에 보여지는, 예시적인 메인 디스플레이 및 DFR이다.
- 도 5d는 다수의 소스들로부터의 요소들이 DFR 상에 보여지는, 예시적인 메인 디스플레이 및 DFR이다.
- 도 6a 내지 도 6c는 예시적인 항목 트리들을 도시한다.
- 도 7a 및 도 7b는 복수의 예시적인 DFR 레이아웃들을 도시한다.

도 8은 예시적인 DFR 레이아웃 편집기 인터페이스를 도시한다.

도 9는 가장 앞쪽(front most) 애플리케이션에 기초한 예시적인 DFR 이미지 생성 프로세스의 흐름도이다.

도 10은 시스템 우선순위 이벤트에 기초한 예시적인 DFR 이미지 생성 프로세스의 흐름도이다.

도 11은 DFR 입력들을 검출 및 라우팅하기 위한 예시적인 프로세스의 흐름도이다.

도 12는 DFR 요소들을 선택하기 위한 예시적인 프로세스의 흐름도이다.

도 13은 하나 이상의 제약에 부합하는 DFR 요소들을 선택하기 위한 예시적인 프로세스의 흐름도이다.

도 14는 도 1 내지 도 14의 특징들 및 프로세스들을 구현하는 예시적인 시스템 아키텍처의 블록도이다.

다양한 도면들에서의 유사한 도면 부호들은 유사한 요소들을 가리킨다.

### 발명을 실시하기 위한 구체적인 내용

[0009]

DFR-장착형 디바이스

[0010]

일부 구현예들에서, 컴퓨팅 디바이스는 동적 기능 열(DFR)을 포함할 수 있다. 예를 들어, DFR은, 다른 것들 중에서도, 전통적인 키보드 기능 열의 기능을 제공할 수 있는 터치 감응형 디스플레이 디바이스일 수 있다. DFR을 포함하는 컴퓨팅 디바이스는 DFR에 의해 디스플레이된 정보 및/또는 DFR에 의해 수신된 사용자 입력을 관리하도록 구성된 하드웨어, 소프트웨어, 및/또는 펌웨어를 포함할 수 있다. 예를 들어, 디바이스는 DFR 에이전트를 인스턴스화할 수 있고, 이는 DFR을 관리할 수 있다. DFR 에이전트는 디바이스의 현재 상태에 기초하여 DFR 상에 디스플레이할 이미지들을 생성할 수 있다. DFR 에이전트는 하나 이상의 디바이스 애플리케이션에 의해 등록된, DFR 상에 디스플레이하기 위한 요소들을 포함하는 이미지들을 생성할 수 있다. DFR 에이전트는, 응답자 체인 순서에 의해 정의된 애플리케이션들의 우선순위에 기초하여, 요소들이 어떻게, 어디에, 그리고/또는 언제 디스플레이될 수 있는지를 정의하는 제약들의 세트에 기초하여, 사용자 맞춤화들, 및/또는 다른 인자들에 기초하여, 디스플레이하기 위한 등록된 요소들을 선택할 수 있다. DFR은 이미지를 디스플레이할 수 있다. 추가적으로, 사용자는 DFR을 터치하여, DFR 디스플레이 상에 제시된 그래픽 요소들에 의해 표현되는 기능들 및/또는 커맨드들을 선택하는 입력을 제공할 수 있다. DFR 에이전트는 커맨드들을 파싱하고 이들을 적절한 애플리케이션에 전송할 수 있다.

[0011]

도 1은 예시적인 디바이스(100)이다. 대표적인 디바이스(100)는 랩톱 컴퓨터로서 도시되어 있지만, 디바이스(100)는 예를 들어, 랩톱 컴퓨터, 데스크톱 컴퓨터, 컴퓨터 단말기, 텔레비전 시스템, 태블릿 컴퓨터, 전자책 리더기, 스마트폰, 스마트 워치, 및 웨어러블 컴퓨터를 포함하지만, 이에 한정되지 않는 다양한 전자 디바이스들 중 하나일 수 있다. 디바이스(100)는 케이스(102), 메인 디스플레이(104), 카메라(106), 터치패드(108), 키보드(110), 및/또는 DFR(112)을 포함할 수 있다. DFR(112)은 터치 감응형 디스플레이, 디스플레이 제어기, 터치 입력 제어기, 및/또는 본 명세서에서 논의되는 GUI 및 입력 특징부들을 제공하도록 구성된 다른 하드웨어를 포함할 수 있다. 예시적인 디바이스(100)에서, DFR(112)은 물리적 키들을 포함하는 전통적인 기능 열 대신에 제공되고, 기능 열이 전형적으로 표준 키보드 레이아웃 상에서 발견될 수 있는 곳에 위치된다. 예를 들어, 일부 키보드들은 이스케이프 키, 기능 키들(F1 내지 F12), 및/또는 추가의 키들을 포함하는 물리적 키들의 열을, 물리적 숫자 키들 및 문자 키 위에서 갖는다. 일부 실시예들에서, DFR(112)은, 이스케이프 키 및 키들(F1 내지 F12)을 포함하는 물리적 키 열 대신에, 물리적 숫자 키들 및 문자 키들 위에 위치될 수 있다. 다른 실시예들에서, DFR(112)은 물리적 키들을 포함하는 기능 열에 더하여 제공될 수 있고/있거나, 디바이스(100) 상의 다른 곳에 위치될 수 있고/있거나, 열과는 상이한 구성들 및/또는 배향들을 가질 수 있다. 예를 들어, DFR(112)은 숫자-패드 형상의 직사각형 또는 수직 컬럼과 유사한 형상을 가질 수 있다. 또한, DFR(112)은 물리적으로 분리된 다수의 디스플레이들로 구성될 수 있다(예컨대, 물리적으로 분리된 2개의 별개의 디스플레이 영역들을 포함하는 DFR(112)).

[0012]

도 2는 디바이스(100)의 예시적인 DFR 제어 하드웨어(200)의 블록도이다. 디바이스(100)는 메인 프로세서(202)(예컨대, X86 프로세서 또는 다른 적합한 프로세서)를 포함할 수 있다. 예를 들어, 메인 프로세서(202)는 디바이스(100)의 일차 운영 체제 및/또는 디바이스(100) 상에서 실행되는 임의의 시스템 또는 사용자 애플리케이션들을 실행하도록 구성되는 애플리케이션 프로세서일 수 있다. 메인 프로세서(202)는 버스(212)(예를 들어, 프론트-사이드 버스, 하이퍼 트랜스포트, 빠른 경로 상호연결부, 다이렉트 미디어 인터페이스, 또는 다른 버스)를 통해 플랫폼 제어기 허브(214)에, 그리고 플랫폼 제어기 허브(214)를 통해, 디바이스(100)의 다른 컴포넌트



트들(예를 들어, 비디오 카드, 오디오 카드, 네트워크 카드, 메모리, 하드 드라이브, 입력 디바이스(들) 등)에 결합될 수 있다. 메인 프로세서(202)는, 예를 들어 운영 체제(예컨대, iOS, 윈도우즈, 리눅스 등) 및 애플리케이션들을 실행하는 일반적인 디바이스(100) 기능을 제어할 수 있다. 메인 프로세서(202)는 열 조절, 전력 사용, 배터리 충전, 비디오 모드 스위칭, 슬립, 대기, 및 다른 기능들을 관리하도록 구성된 시스템 관리 제어(system management control, SMC)(220) 펌웨어를 포함할 수 있다. SMC(220)는 디바이스(100)가 전력 온되는 동안 항상 활성일 수 있어서, 예를 들어, 메인 프로세서(202)를 슬립 또는 대기 모드들로부터 웨이크시킬 수 있다.

[0013] 일부 구현예들에서, 디바이스(100)는 이차 프로세서(204)를 포함할 수 있다. 예를 들어, 이차 프로세서(204)는 시스템 온 칩(system on chip, SoC), 코프로세서, ARM 프로세서 등일 수 있다. 이차 프로세서(204)는 메인 프로세서(202) 상에서 동작하는 운영 체제와 상이한 운영 체제를 실행할 수 있다. 예를 들어, 이차 프로세서(204)는 iOS, watchOS, 실시간 운영 체제, 임베디드 시스템용 운영 체제, 또는 리눅스(Linux) 변형과 같은 운영 체제를 실행할 수 있다. 이차 프로세서(204)는 카메라(106), DFR(112), 및/또는 다른 디바이스(들)(216)(예컨대, 터치 식별 센서, 주변 광 센서 등)를 동작시킬 수 있다. 이차 프로세서(204)는 열 조절, 전력 사용, 하드웨어 전력 기능, 슬립, 대기, 및 다른 기능들을 관리하도록 구성된 전력 관리 유닛(PMU)(218) 펌웨어를 포함할 수 있다. PMU(218)는 디바이스(100)가 전력 온되는 동안 항상 활성일 수 있어서, 그것은 이차 프로세서(204)를 완전 동작 모드로 복원할 수 있고/있거나, 예를 들어 메인 프로세서(202)가 저전력 상태에 있는 동안 이차 프로세서(204)가 메인 프로세서(202)와 통신하게 할 수 있다.

[0014] 일부 구현예들에서, 메인 프로세서(202) 및 이차 프로세서(204)는 링크(210)를 통해 서로 통신할 수 있다. 예를 들어, 링크(210)는 USB2 링크 또는 유사한 데이터 링크일 수 있다. 예를 들어, 메인 프로세서(202)는 DFR(112) 상에 디스플레이하기 위한 이미지들을 생성하고 이들을 링크(210)를 통해 이차 프로세서(204)에 전달하여, 이차 프로세서(204)가 DFR(112) 상에 이미지들을 디스플레이하게 할 수 있다. 일부 구현예들에서, 이차 프로세서(204)는 DFR(112)에 대해 행해진 터치 입력들을 수신하고 링크(210)를 통해 터치 입력 데이터를 메인 프로세서(202)에 전달하여, 메인 프로세서(202)가 입력들을 처리하게 할 수 있다.

[0015] 일부 구현예들에서, 메인 프로세서(202) 및 이차 프로세서(204)는 인터-집적 회로(I<sup>2</sup>C) 버스(209)를 통해 서로 통신할 수 있다. 메인 프로세서(202)는 이차 프로세서(204)의 메모리 레지스터(206)("메일박스")에 데이터를 배치하기 위해 I<sup>2</sup>C 버스(209)를 사용할 수 있다. 메일박스 레지스터(206)는, PMU(218)에 대한 커맨드들이 메인 프로세서(202)에 의해 기록되는, PMU 스크래치패드로서의 역할을 할 수 있다. 예를 들어, 메인 프로세서(202)는 링크(210)가 비활성일 때 이차 프로세서(204)와 통신하기 위해 메모리 레지스터(206)에 데이터를 배치할 수 있다.

[0016] 일부 구현예들에서, 메인 프로세서(202) 및 이차 프로세서(204)는 범용 입/출력(GPIO) 경로들(208)을 통해 서로 결합될 수 있다. 각각의 GPIO 경로(208)는 메인 프로세서(202)에서의 GPIO 핀 및 이차 프로세서(204)에서의 GPIO 핀을 포함할 수 있으며, 이들은 서로 결합될 수 있다. 각각의 프로세서는 그의 각자의 GPIO 핀들을 입력들 또는 출력들로서 설정하도록 구성될 수 있다. 프로세서의 핀이 출력으로서 설정될 때, 프로세서는 핀을 로우(로직 0 또는 로우 전압) 또는 하이(로직 1 또는 하이 전압)로 구동하고, 그렇게 함으로써 데이터 비트를 다른 프로세서에 전송할 수 있다. 프로세서의 핀이 입력으로서 설정되는 경우, 프로세서는 핀 상의 전압이 변화할 때를 검출하고 이에 응답하여 처리를 수행할 수 있다. 예를 들어, 메인 프로세서(202)는 GPIO 경로들(208)을 사용하여, 이차 프로세서(204)가 메일박스 레지스터(206) 내의 데이터를 판독하게 하는 인터럽트를 트리거하는 것과 같은, 이차 프로세서(204)에 의한 동작을 트리거할 수 있다. 이차 프로세서(204)는 GPIO 경로들(208)을 사용하여 메인 프로세서(202)에 확인응답을 전송할 수 있다.

[0017] 도 3은 랩톱 컴퓨터와 같은 디바이스(100)를 위한 예시적인 DFR(112) 및 키보드(110) 레이아웃이다. DFR(112)은 기능 열이 다른 방식으로 예상될 수 있는 위치에서 키보드(110) 위에 배치될 수 있다. 도 3이 랩톱 컴퓨터를 위한 예시적인 레이아웃을 제공하지만, 일부 실시예들에서, 디바이스(100)는 데스크톱 컴퓨터일 수 있고, DFR(112)은 디바이스(100)의 키보드 주변기기 상에, 별개의 전용 I/O 주변기기 상에, 데스크톱 컴퓨터의 본체 상에 등에 배치될 수 있다.

[0018] 일부 구현예들에서, DFR(112)은 하나 이상의 DFR 그래픽 요소(300)를 제시할 수 있다(이 예에서는, 5개의 DFR 그래픽 요소(300A 내지 300E)를 제시할 수 있지만, 임의의 수의 DFR UI 요소들(300)이 가능할 수 있다). DFR UI 요소들(300)은 텍스트, 심볼, 이미지, 및/또는 다른 시각적 정보를 포함할 수 있다. 시각적 정보는 후술되는 바와 같이 프로세서(202)에 의해 동적으로 생성될 수 있거나, 또는 사전-생성된 시각적 정보 중에서 선택될



수 있다. DFR UI 요소들(300A 내지 300E)은 컴퓨팅 디바이스(100) 또는 컴퓨팅 디바이스(100) 상에서 실행되는 다양한 애플리케이션들 중 임의의 것에 의해 제공되는 기능들을 표현할 수 있다. 예를 들어, DFR UI 요소들(300A 내지 300E)은 별개의 터치 UI 영역들에 제시된 전통적인 기능 "키들"의 가상 표현을 제공하여, 별개의 요소들(300A 내지 300E)이 별개의 커맨드들에 대응하도록 할 수 있다. 예를 들어, 요소(300D)는 오디오 볼륨을 낮추기 위한 UI 요소일 수 있고, 요소(300E)는 오디오 볼륨을 상승시키기 위한 UI 요소일 수 있다. DFR UI 요소들(300D 및 300E)은 오디오 볼륨 커맨드들에 대응하는 심볼들 또는 이미지들을 디스플레이할 수 있다. DFR UI 요소(300D) 상의 사용자의 터치 입력(예컨대, 그것의 선택)은 디바이스(100)의 오디오 출력 볼륨을 낮추기 위한 커맨드를 등록할 수 있고, 요소(300E) 상의 터치 입력은 후술되는 바와 같이 디바이스(100)의 오디오 출력 볼륨을 상승시키기 위한 커맨드를 등록할 수 있다.

[0019] DFR(112)은 다양한 기능 제어부들에 대응하는 다양한 DFR UI 요소들(300)을 디스플레이할 수 있다. 일부 DFR UI 요소들(300)은 시스템-레벨 기능을 제공할 수 있고/있거나 다양한 애플리케이션들에 적용가능할 수 있다. 그러한 DFR UI 요소들(300)은 볼륨, 스피커들의 무음화, 스크린 밝기, 재생, 무선 네트워크 수신기의 인에이블/디스에이블, 멀티-스크린, 도움말 기능, 시스템 검색, 최근 및/또는 빈번하게 사용되는 애플리케이션 시작, 메시지 액세스, 문서 액세스, 파일 시스템 액세스, 캘린더, 및/또는 다른 것들에 대한 제어부들을 포함할 수 있다.

[0020] 일부 구현예들에서, DFR UI 요소들(300)은 특정 애플리케이션들에 맞추어진 기능을 제공할 수 있다. 예를 들어, 기능은 애플리케이션 컨텍스트에 기초하여 맞춤형되거나 동적으로 생성될 수 있다. 기능은 유사한 기능들을 갖는 복수의 애플리케이션들에 기초하여 맞추어지거나 동적으로 생성될 수 있다. 기능은 애플리케이션들 내의 특정 특징들에 기초하여 맞춤형되거나 동적으로 생성될 수 있다. 그러한 DFR UI 요소들(300)은 텍스트 편집에 관련된 애플리케이션들 또는 특징들에 대한 폰트 선택, 폰트 크기, 폰트 색상, 폰트 효과, 하이라이팅, 목록 기능, 텍스트 자동 완성 제안, 특수 문자, 절단, 복사, 붙여넣기, 및/또는 다른 것들; 웹 브라우징에 관련된 애플리케이션들 또는 특징들에 대한 주소 표시줄, 리로드, 북마크, 앞으로, 뒤로, 새 탭, 이력, 다운로드, 및/또는 다른 것들; 그래픽 편집에 관련된 애플리케이션들 또는 특징들에 대한 브러시 크기, 브러시 형상, 색상, 형상 효과, 텍스트 상자 편집, 자르기(cropping), 회전, 필터, 절단, 복사, 붙여넣기, 및/또는 다른 것들, 및 많은 다른 것들에 대한 제어부들을 포함할 수 있다. 이러한 제어부들의 목록은 총망라하고자 하는 것이 아니라, 일부 실시예들에서 가능한 DFR UI 요소(300) 특징들 및 기능들을 예시하기 위해서만 의도된다.

[0021] 일부 경우에, 활성 DFR UI 요소들(300A 내지 300E)은 빈 공간에 의해 서로 분리될 수 있다. 예를 들어, DFR UI 요소들(300A, 300C)은 이미지들을 디스플레이하고 입력들을 수용하는 활성 요소들일 수 있는 반면, DFR UI 요소(300B)는 이미지를 디스플레이하지 않고 입력을 수용하지 않는 빈 공간일 수 있다. 일부 구현예들에서, DFR UI 요소(300B)는 DFR UI 요소들(300A, 300C)을 분리하고자 하는 이미지일 수 있다. 예를 들어, DFR UI 요소(300B)는 DFR UI 요소들(300A, 300C)을 분리하는 수직 바 또는 라인일 수 있다.

[0022] *DFR 에이전트*

[0023] 도 4는 예시적인 DFR 에이전트(400)의 블록도이다. 일부 구현예들에서, 컴퓨팅 디바이스(100)는 DFR 에이전트(400)를 포함할 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 단일 하드웨어, 소프트웨어, 또는 펌웨어 요소일 수 있다. 다른 구현예들에서, DFR 에이전트(400)는 후술되는 DFR 에이전트(400) 기능들을 수행하는 다수의 하드웨어, 소프트웨어, 또는 펌웨어 요소들을 포함할 수 있다. DFR 에이전트(400)는 DFR(112) 상에 디스플레이된 이미지들을 관리하고, DFR(112)을 통해 수신된 커맨드들을, 어느 이미지가 사용자에게 의해 선택되는 것에 기초하여 커맨드들이 의도된 애플리케이션에 전송할 수 있다. 예를 들어, DFR 에이전트(400)는 디바이스(100) 애플리케이션들과 통신하여 정보를 수신하여, 정보를 포함하는 DFR(112)에 대한 GUI들을 디스플레이 및 생성할 수 있다. 사용자가 DFR(112)을 터치함으로써 커맨드를 입력할 때, DFR 에이전트(400)는 어느 DFR UI 요소(300)가 눌러졌는지를 결정하고, 눌러진 DFR UI 요소(300)와 연관된 커맨드를 DFR UI 요소(300)를 등록된 애플리케이션에 전송할 수 있다.

[0024] 일부 구현예들에서, DFR 에이전트(400)는 등록 모듈(410)을 포함할 수 있다. 예를 들어, 등록 모듈(410)은 애플리케이션들로부터 DFR UI 요소(300)의 등록들을 수신하고 처리할 수 있다. 아래에서 더 상세히 설명되는 바와 같이, 등록 모듈(410)은 하나 이상의 애플리케이션으로부터 DFR UI 요소 데이터를 수신할 수 있다. DFR UI 요소 데이터는 DFR UI 요소(300)에 의해 디스플레이될 이미지들 및/또는 텍스트를 포함할 수 있다. 예를 들어, 이미지들 및/또는 텍스트는 DFR UI 요소(300)에 의해 제시될 수 있고, DFR UI 요소(300)가 사용자에게 의해 선택될 때 어떤 기능이 수행될 것인지를 사용자에게 나타낼 수 있다.

- [0025] 일부 구현예들에서, DFR 에이전트(400)는 맞춤화 모듈(420)을 포함할 수 있다. 예를 들어, 맞춤화 모듈(420)은 DFR UI 요소들(300) 및/또는 DFR(112) 상에 디스플레이된 DFR UI 요소들(300)의 조합들에 대한 사용자 맞춤화들을 수신하고 처리할 수 있다. 예를 들어, 아래에서 더 상세히 설명되는 바와 같이, 맞춤화 모듈(420)은 DFR UI 요소들(300)을 선택하고 DFR(112) 자체 또는 시뮬레이션된 DFR 디스플레이로 그것들을 드래그하기 위한 사용자 인터페이스를 제공할 수 있다. 사용자는 다양한 이벤트들 및/또는 상태들에 대해 디스플레이될 DFR UI 요소들(300)을 선택하여서, 시스템(100)이 주어진 이벤트를 경험하거나 주어진 상태에 들어갈 때, DFR 에이전트(400)가 그 이벤트 또는 상태에 대해 사용자에게 의해 선택된 DFR UI 요소들(300)을 디스플레이할 수 있게 한다.
- [0026] 일부 구현예들에서, DFR 에이전트(400)는 시스템 상태 모듈(430)을 포함할 수 있다. 예를 들어, 시스템 상태 모듈(430)은 디스플레이하기 위한 DFR UI 요소들(300)을 선택하는 데 사용하기 위해 시스템(100) 이벤트들을 식별하고/하거나 시스템(100) 상태를 결정할 수 있다. 예를 들어, 아래에서 더 상세히 설명되는 바와 같이, 시스템 상태 모듈(430)은 시스템(100) 상태를 모니터링하여, 착신 통화와 같은 이벤트가 발생하는 때, 및/또는 시스템 상태가 변화하는 때, 예컨대 사용자가 상이한 애플리케이션 또는 애플리케이션의 상이한 부분에서 작업하기 시작하는 때를 검출할 수 있다. DFR 에이전트(400)는 시스템 상태 모듈(430)에 의해 결정된 바와 같이 시스템(100) 상태를 사용하여, 디스플레이하기 위한, 상태에 대응하는 DFR UI 요소들(300)을 선택할 수 있다.
- [0027] 일부 구현예들에서, DFR 에이전트(400)는 규칙 모듈(440)을 포함할 수 있다. 예를 들어, 규칙 모듈(440)은 하나 이상의 규칙을 사용하여, 등록된 DFR UI 요소들(300) 중에서 디스플레이되어야 하는 것을 선택할 수 있다. 규칙들은 DFR 에이전트(400)가, 그 시간에 디바이스(100) 컨텍스트에 맞는 DFR UI 요소들(300)을 발견하게 하도록 구성될 수 있다. 규칙 모듈(440) 및 그의 예시적인 기능들의 추가 논의는 하기의 "요소 선택" 및 "응답자 체인" 섹션들에 제공된다.
- [0028] 일부 구현예들에서, DFR 에이전트(400)는 컨텍스트 모듈(450)을 포함할 수 있다. 예를 들어, 컨텍스트 모듈(450)은 DFR UI 요소(300) 등록에서 제출된 DFR UI 요소들(300)의 설명들에 기초하여 주어진 컨텍스트에 관련된 DFR UI 요소들(300)을 선택할 수 있다. 예를 들어, 컨텍스트-기반 선택에서, DFR UI 요소들(300)은 응답자들(즉, 응답자 체인의 순서화된 계층구조 내의 애플리케이션들)보다는 특정 컨텍스트들과 연관될 수 있고, 컨텍스트 모듈(450)은 디바이스(100) 컨텍스트에 기초하여 DFR UI 요소들(300)을 선택할 수 있다. 예를 들어, 텍스트 다이얼로그 박스와 상호작용하는 사용자가 컨텍스트일 수 있고, 애플리케이션 상태가 컨텍스트일 수 있고, 내부 디바이스 상태가 컨텍스트일 수 있고, 착신 메시지 또는 다른 이벤트가 컨텍스트일 수 있는 등이다. DFR 에이전트(400)는 DFR UI 요소(300) 등록에서 제출된 요소들(300)의 설명들에 기초하여 주어진 컨텍스트와 관련된 DFR UI 요소들(300)을 선택할 수 있다.
- [0029] 일부 구현예들에서, DFR 에이전트(400)는 제약 모듈(460)을 포함할 수 있다. 예를 들어, 제약 모듈(460)은 DFR UI 요소들(300)의 순서화되지 않은(unordered) 세트를 선택하고, DFR UI 요소들(300) 및 DFR UI 요소들(300)의 순서화된(ordered) 그룹들에 제약들의 세트를 적용할 수 있다. 제약들을 적용함으로써, 제약 모듈(460)은 어느 DFR UI 요소들(300)이 어떤 배열로 디스플레이될지를 선택할 수 있다. 제약들은 어느 DFR UI 요소들(300)이 가시적인지 그리고 가시적인 DFR UI 요소들(300)이 DFR(112) 상에 어떻게 배열되는지를 확립할 수 있다. 예를 들어, 제약들은 "이들 3개의 DFR UI 요소(300)가 좌측에서 우측으로, 또는 우측에서 좌측으로 소정 순서로 나타나야 한다" 또는 "이들 3개의 DFR UI 요소(300)는 그들 사이에 적어도 특정된 양의 공간을 갖고서 나타나야 한다" 또는 "이 DFR UI 요소(300)는 좌측에 나타나야 한다" 또는 "이 DFR UI 요소(300)는 우측에 나타나야 한다" 등과 같은 관계들을 정의할 수 있다. 제약들은 DFR UI 요소들(300) 사이의 고정된 공간 또는 유연한(예컨대, 크기조정가능한) 공간의 양을 정의할 수 있다.
- [0030] 일부 구현예들에서, DFR 에이전트(400)는 디스플레이 조립 모듈(470)을 포함할 수 있다. 예를 들어, 디스플레이 조립 모듈(470)은 DFR(112)에 의한 제시를 위해 선택된 DFR UI 요소들(300)을 통합하는 디스플레이 데이터를 생성할 수 있다. 일단 규칙 모듈(440), 컨텍스트 모듈(450), 및/또는 제약 모듈(460)이 디스플레이하기 위한 DFR UI 요소들(300)을 선택 및 배열했으면, 디스플레이 조립 모듈(470)은, 예를 들어 미리규정된 순서로 배열된 각각의 DFR UI 요소(300)를 포함하는 비트맵 또는 다른 이미지를 생성함으로써, 디스플레이될 데이터를 생성할 수 있다. 디스플레이 조립 모듈(470)은 생성된 이미지를 디스플레이를 위해 DFR(112)에 전송할 수 있다.
- [0031] 일부 구현예들에서, DFR 에이전트(400)는 입력 모듈(480)을 포함할 수 있다. 입력 모듈(480)은 DFR(112)에 대한 사용자 입력들을 수신할 수 있고, DFR(112) 상에 디스플레이된 DFR UI 요소들(300)의 배열에 관한 데이터를 사용하여 사용자에게 의해 선택된 커맨드를 결정할 수 있다. 예를 들어, 데이터는 선택된 DFR UI 요소들(300)이 DFR(112) 상에서 어떻게 배열되는지를 정의하는 디스플레이 조립 모듈(470)로부터의 데이터를 포함할 수 있다.

데이터는 사용자가 DFR(112)을 누른 곳을 정의하는 DFR(112)로부터의 데이터를 포함할 수 있다. 입력 모듈(480)은 디스플레이 조립 모듈(470)로부터 요소들(300)의 배열에 관한 데이터를 수신할 수 있으므로, 입력 모듈(480)은 어느 요소들(300)이 디스플레이되고 있는지 그리고 그들이 DFR(112) 상에서 어디에서 제시되는지를 결정할 수 있다. 입력 모듈(480)은 DFR(112)로부터 사용자가 DFR(112)를 터치한 곳을 설명하는 데이터를 수신할 수 있다. 입력 모듈(480)은 어느 요소(300)가 선택되었는지를 결정하기 위해 디스플레이 조립 모듈(470) 및 DFR(112)로부터의 데이터를 상관시킬 수 있다. 입력 모듈(480)은 선택된 요소(300)와 연관된 애플리케이션에게 통지할 수 있다. 예를 들어, DFR 에이전트(400)는 선택된 요소에 의해 표현되는 커맨드 또는 기능에 대응하는 애플리케이션 API를 호출할 수 있다.

[0032] DFR 에이전트(400)는 애플리케이션들과 DFR(112) 사이의 중개자로서 기능할 수 있다. 예를 들어, 애플리케이션들은 DFR(112) 상에 디스플레이하기 위한 데이터를 등록 모듈(410)에 제출한다. 디스플레이 조립 모듈(470)은 디스플레이하기 위한 이미지를 생성하고, 이미지를 DFR(112)에 전송한다. 따라서, 애플리케이션들은 DFR(112) 상에 데이터를 직접 디스플레이하지 않을 수 있다. 입력 모듈(480)은 DFR(112)에 대해 이루어진 입력들을 수신하고 커맨드 데이터를 애플리케이션들에 전송한다. 따라서, 애플리케이션들은 DFR(112)에 대해 이루어진 입력들을 직접 수신하지 않을 수 있다.

[0033] 요소 등록

[0034] 일부 구현예들에서, DFR 에이전트(400)는 하나 이상의 DFR UI 요소(300)를 각각 등록하는 하나 이상의 애플리케이션으로부터의 등록들을 수용할 수 있다. 각각의 DFR UI 요소(300)는 DFR(112)의 섹션으로서 디스플레이될 그래픽 요소를 포함할 수 있다. 섹션은 총 DFR(112) 디스플레이 영역의 100%만큼 클 수 있지만, 많은 DFR UI 요소들(300)은 더 작을 수 있다. 각각의 DFR UI 요소(300)는 그 자신의 그래픽, 텍스트, 및/또는 기능을 가질 수 있다. 일부 애플리케이션들은 단일 DFR UI 요소(300)를 등록할 수 있는 반면, 다른 애플리케이션들은 다수의 DFR UI 요소들(300)을 등록할 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 DFR(112)의 총 크기 및/또는 DFR(112)의 총 크기보다 작은 하나 이상의 허용가능한 DFR UI 요소(300) 크기들을 정의할 수 있다.

[0035] DFR UI 요소(300)를 등록하기 위해, 애플리케이션은 요소 콘텐츠를, DFR 에이전트(400)에 의해 정의된 올바른 크기인 비트맵(또는 상이한 유형의 다른 이미지)으로 렌더링할 수 있다. 예를 들어, DFR 에이전트(400)는 DFR(112)의 전체 디스플레이 영역에 걸쳐 있는 DFR UI 요소(300)에 대한 크기를 정의할 수 있다. DFR 에이전트(400)는 DFR(112)의 디스플레이 영역의 일부분을 커버하는 DFR UI 요소(300)에 대한 크기를 정의할 수 있다(예컨대, 10%, 또는 100개의 픽셀들). 일부 구현예들에서, DFR 에이전트(400)는 DFR UI 요소들(300)을 DFR(112)에 의해 디스플레이하기 위한 합성 이미지로 조립할 수 있다. 예를 들어, 애플리케이션 또는 다수의 애플리케이션들은 도 3의 DFR UI 요소들(300A 내지 300E)을 등록할 수 있다. DFR 에이전트(400)는 DFR UI 요소들(300)을, 5개의 별개의 이미지를 제시하기보다는, DFR(112)의 디스플레이 상에 제시하기 위한 단일 합성 이미지로 조립할 수 있다.

[0036] 일부 구현예들에서, 애플리케이션에 의해 제출된 DFR UI 요소(300)에 대한 등록 데이터는 DFR(112) 비트맵에 통합될 이미지를 정의하는 이미지 데이터 및/또는 DFR UI 요소(300)의 기능을 설명하는 메타데이터를 포함할 수 있다. 등록 모듈(410)은 제출된 등록 데이터를 수신하고 DFR UI 요소(300)를 등록할 수 있다. 등록 모듈(410)은 후술되는 바와 같이 DFR(112)에 대한 디스플레이들을 구축하는 데 있어서 향후 사용을 위해, 등록된 DFR UI 요소(300)를 디바이스(100) 메모리에 저장할 수 있다.

[0037] 이미지 데이터는 비트맵 또는 다른 이미지 파일을 포함할 수 있다. 일부 실시예들에서, 애플리케이션으로부터 비트맵 또는 다른 이미지 파일을 수신하는 대신에, 등록 모듈(410)은 언어 기반 이미지 렌더링을 수행할 수 있다. 이러한 접근법에서, 등록 모듈(410)은 애플리케이션에 의해 제공되는 파라미터들에 기초하여 디스플레이하기 위한 이미지를 생성 또는 선택할 수 있다. 예를 들어, 애플리케이션은 그것이 특정된 제목, 기능, 및 크기를 갖는 요소(300)를 등록하고 있음을 나타내는 정보를 제공할 수 있다. 등록 모듈(410)은 특정된 파라미터들에 따라 요소(300)를 구축할 수 있다. 일부 구현예들에서, 등록 모듈(410)은 등록 모듈(410)에 이용가능한 이미지들의 라이브러리로부터 이미지를 선택함으로써 요소(300)를 구축할 수 있다. 라이브러리는 그들이 표현하는 커맨드들에 따라 식별되는 이미지들을 포함할 수 있다. 예를 들어, 하나의 이미지는 "미디어 재생 일시정지" 이미지로서 태깅될 수 있고, 다른 이미지는 "폰트 크기 증가" 이미지로서 태깅될 수 있고, 다른 이미지는 "착신 메시지 수신" 이미지로서 태깅될 수 있다. 애플리케이션이 DFR UI 요소(300)의 기능을 "미디어 재생 일시정지"로서 특정하는 경우, 등록 모듈(410)은 "미디어 재생 일시정지" 이미지로서 태깅된 이미지를 선택할 수 있다.



- [0038] 일부 구현예들에서, DFR 에이전트(400)는 DFR UI 요소(300)에 대한 메타데이터를 수신할 수 있다. 메타데이터는 DFR UI 요소(300)와 연관된 커맨드를 정의하는 정보를 포함할 수 있다. 예를 들어, DFR UI 요소 데이터는 DFR UI 요소(300)가 사용자에게 의해 선택될 때 DFR 에이전트(400)가 호출해야 하는 애플리케이션 프로그래밍 인터페이스(API)를 식별할 수 있다. DFR UI 요소 데이터는 DFR UI 요소(300)를 애플리케이션에 링크시키는 정보(예컨대, 애플리케이션 식별자)를 포함할 수 있다. 등록 모듈(410)은 수신된 DFR UI 요소 데이터를 등록하여서, 다른 DFR UI 요소들(300)이 등록된 데이터를 사용하여, 디스플레이하기 위한 DFR UI 요소들(300)을 선택하고 디스플레이되고 있는 DFR UI 요소들(300)과 연관된 커맨드들을 처리할 수 있도록 할 수 있다.
- [0039] DFR 에이전트(400)는 메타데이터 제출들에 대한 표준화된 포맷 또는 언어를 특정할 수 있다. 예를 들어, 메타데이터 제출은 애플리케이션 식별자, 기능 식별자(예컨대, API 식별자), 및 일부 경우들에서, 디스플레이 파라미터들을 포함할 수 있다. 애플리케이션 식별자는 어떤 애플리케이션이 DFR UI 요소(300)와 연관되는지를 DFR 에이전트(400)에게 알릴 수 있다. 기능 식별자는 어떤 애플리케이션 커맨드 또는 API가 DFR UI 요소(300)와 연관되는지를 DFR 에이전트(400)에게 알릴 수 있다. 디스플레이 파라미터들은, 아래에서 더 상세히 기술되는 바와 같이, DFR UI 요소(300)를 디스플레이하기 위한 규칙들 및/또는 제약들을 포함할 수 있다. 예를 들어, 미디어 재생 제어 DFR UI 요소(300)에 대한 등록 메타데이터는 하기의 정보를 포함할 수 있다: "애플리케이션 - 미디어 재생기; 기능 - 일시정지; 파라미터 - 좌측 예지 > 정지 요소의 좌측 예지". 이 예에서, 애플리케이션 식별자는 DFR 에이전트(400)에게, 사용자가 DFR UI 요소(300)를 선택할 때, 커맨드가 미디어 재생기 애플리케이션에 대해 의도된다는 것을 알릴 수 있다. 기능 식별자는 DFR 에이전트(400)에게, 사용자가 DFR UI 요소(300)를 선택할 때, 커맨드가 미디어 재생 일시정지 커맨드임을 알릴 수 있다. 디스플레이 파라미터는, 아래에서 더 상세히 기술되는 바와 같이, DFR 에이전트(400)에게, DFR UI 요소(300)가 다른 DFR UI 요소들(300)에 관련하여 DFR(112) 상에서 배치되어야 하는 곳을 알릴 수 있다.
- [0040] DFR 에이전트(400)는 메타데이터를 사용하여, DFR UI 요소(300)의 사용자 선택을, 애플리케이션에 의해 처리되는 커맨드로 변환할 수 있다. 예를 들어, 애플리케이션이 DFR UI 요소(300)를 등록할 때, DFR 에이전트(400)는 DFR UI 요소(300)의 목적을 정의하는 DFR UI 요소(300)에 대한 메타데이터를 수신할 수 있다. 메타데이터는 후술되는 바와 같이, DFR 에이전트(400)가 DFR UI 요소(300)의 사용자 선택을, 대응하는 애플리케이션 커맨드와 연관시키도록 할 수 있다. 예를 들어, 애플리케이션은 밀줄친 문자를 보여주는 비트맵 및 텍스트에 밀줄을 치기 위한 제어부로서 DFR UI 요소(300)를 정의하는 메타데이터를 포함하는 DFR UI 요소(300)를 등록할 수 있다. 이러한 DFR UI 요소(300)의 사용자 선택은 DFR UI 요소(300)를 등록한 애플리케이션에서 텍스트 밀줄치기를 토글링할 수 있다. 예를 들어, 밀줄 DFR UI 요소(300)의 사용자 선택을 수신하는 것에 응답하여, DFR 에이전트(400)는 대응하는 API를 호출할 수 있으며, 이는 애플리케이션이 텍스트의 일부 선택된 부분에 밀줄치게 할 수 있다.
- [0041] 일부 구현예들에서, 애플리케이션은 애플리케이션에 대한 복수의 제어부들을 포함하는 단일 DFR UI 요소(300)를 등록할 수 있다. 예를 들어, DFR UI 요소(300)에 대한 등록 메타데이터는 DFR UI 요소(300) 내에서 복수의 별개의 영역들을 특정할 수 있으며, 여기서 각각의 상이한 영역과의 사용자 상호작용들은 상이한 결과들을 생성한다. 예를 들어, 미디어 재생 제어 DFR UI 요소(300)는 하기 정보를 포함할 수 있다: "애플리케이션 - 미디어 재생기; 좌측 50% 기능 - 일시정지; 우측 50% 기능 - 정지". 이 예에서, 애플리케이션 식별자는 DFR 에이전트(400)에게, 사용자가 DFR UI 요소(300)를 선택할 때, 커맨드가 미디어 재생기 애플리케이션에 대해 의도된다는 것을 알릴 수 있다. 기능 식별자들은 DFR 에이전트(400)에게, 사용자가 DFR UI 요소(300)의 좌측 절반을 선택할 때, 커맨드는 미디어 재생 일시정지 커맨드이고, 사용자가 DFR UI 요소(300)의 우측 절반을 선택할 때, 커맨드는 미디어 재생 정지 커맨드임을 알릴 수 있다.
- [0042] 일부 구현예들에서, 애플리케이션은 하나 이상의 개별 DFR UI 요소(300)를 등록할 수 있으며, 각각은 애플리케이션에 대한 단일 제어부 또는 몇몇 제어부들을 포함한다. 일부 실시예들에서, 별개의 애플리케이션들은 동일하거나 유사한 기능을 갖는 DFR UI 요소들(300)을 등록할 수 있다(일부 경우에, 동일한 그래픽 및 기능을 갖는, 동일한 DFR UI 요소들(300)). 애플리케이션들은 등록 동안 DFR UI 요소(300) 기능을 표현할 수 있어서, DFR 에이전트(400)는 공통성을 결정할 수 있다. 이것은, 예를 들어, 앱 A 및 앱 B 둘 모두가 동일한 등록된 제어부를 갖는 경우에, 유용할 수 있다. 예를 들어, 제1 DFR UI 요소 메타데이터는 "애플리케이션 - 앱 A; 기능 - 폰트 선택; 파라미터 - 없음"을 포함할 수 있고, 제2 DFR UI 요소 메타데이터는 "애플리케이션 - 앱 B; 기능 - 폰트 선택; 파라미터 - 없음"을 포함할 수 있다. DFR 에이전트(400)는 상이한 애플리케이션들에 관련된 DFR UI 요소들(300)을 디스플레이하는 것으로부터 스위칭할 때 공통 DFR UI 요소들(300)에 대한 연속성을 보존할 수 있다. 예를 들어, DFR 에이전트(400)가 앱 A DFR UI 요소들(300)을 보여주는 것으로부터 앱 B DFR UI 요소들(300)로

스위칭할 때, 폰트 선택 DFR UI 요소(300)는 DFR(112) 상의 동일한 장소에서 디스플레이될 수 있다. 이는 또한 단일 애플리케이션의 상이한 기능 레벨들이 동일하거나 유사한 DFR UI 요소들(300)을 등록했을 때 유용할 수 있다. 애플리케이션은, 아래에서 더 상세히 기술되는 바와 같이, 각각이 그 자신의 기능을 갖는 여러 기능 레벨들(예컨대, 서브섹션들)을 가질 수 있다.

[0043] 일부 구현예들에서, DFR UI 요소(300)는 요소들의 그룹을 포함할 수 있다. 예를 들어, 그룹 요소는 다수의 요소들을 포함할 수 있다(분리되지 않아야 하는 것들의 개념적인 그룹화를 형성함). 예를 들어, 그룹 요소는 분리되지 않아야 하는 복수의 미디어 제어 DFR UI 요소들(300)(예컨대, 재생 제어부들)을 포함할 수 있다. 그룹 요소 등록 데이터는 어느 DFR UI 요소들(300)이 그룹 내에 나타나는지를 특정할 수 있고, DFR UI 요소들(300)에 대한 특정 순서를 정의할 수 있다. 따라서, 그룹 요소가 DFR(112) 상에 디스플레이될 때, 그룹 내의 모든 DFR UI 요소들(300)은 매년 동일한 고정된 순서로 나타날 수 있다.

[0044] DFR UI 요소(300)는 윈도우 또는 다이얼로그와 유사하지만 DFR 에이전트(400)에 의해 완전히 관리되는 것으로 간주될 수 있다. 다시 말하면, DFR 에이전트(400)는 디바이스(100)의 사용자가 윈도우들의 크기 또는 위치에 대한 간접적인 제어만을 갖는 윈도우 시스템을 제공하는 것으로 간주될 수 있다. 사용자의 행동들은 어느 DFR UI 요소들(300)이 디스플레이되는지에 영향을 미칠 수 있다(예컨대, 사용자가 애플리케이션을 열고 DFR 에이전트(400)가 그 애플리케이션에 의해 등록된 요소들(300)을 디스플레이하거나, 또는 사용자는, 주어진 디바이스(100) 상태에서 디스플레이된 그들이 보고 싶어하는 DFR UI 요소들(300)을 선택한다). 그러나, DFR 에이전트(400)는, 윈도우화된 GUI에서 윈도우를 드래그하고/하거나 크기조정하는 것과는 대조적으로, DFR UI 요소들(300)의 실제 선택 및 배열을 수행한다. DFR 에이전트(400)는, DFR UI 요소들(300) 내의 변화들 및/또는 디스플레이되고 있는 DFR UI 요소들(300) 사이의 전이들에 대한 중앙 조정된 애니메이션을 제공할 수 있다.

[0045] 추가적으로, 애플리케이션은 DFR UI 요소들(300)을 등록하기만 하고 DFR 에이전트(400)는 디스플레이하기 위한 DFR UI 요소들(300)을 선택하기 때문에, 애플리케이션은 DFR(112) 하드웨어에 액세스하지 못할 수 있다. 실제로, 애플리케이션은 자신의 DFR UI 요소(300)가 DFR(112) 상에 디스플레이되는지 또는 메인 디스플레이(104) 상에서 가상으로 실행되고 있는지(예를 들어, 디버깅 또는 사용자 맞춤화를 위해) 결정할 수 없다. 애플리케이션에 비가시적인 방식으로 가상으로 DFR UI 요소(300)를 실행하는 것은 정확한 시뮬레이션 및/또는 디버깅을 허용할 수 있는데, 이는 애플리케이션이 시뮬레이션 또는 디버그 모드에 배치되는 것으로 인해 상이하게 거동하지 않을 것이기 때문이다.

[0046] 요소 선택

[0047] 일부 구현예들에서, DFR 에이전트(400)는 DFR(112)의 디스플레이 상에 제시하기 위해 등록된 DFR UI 요소들(300)을 선택할 수 있다. 예를 들어, 등록된 DFR UI 요소들(300)은 DFR(112) 상에 반드시 보여지지 않는다. 시스템 상태 모듈(430)은 시스템(100) 이벤트들(예컨대, 착신 통화 또는 메시지, 특정 DFR UI 요소들(300)을 디스플레이하기 위한 사용자 커맨드, 경보 등) 및/또는 시스템(100) 상태들(예컨대, 사용자가 상호작용하고 있는 특정 애플리케이션, 사용자가 상호작용하고 있는 애플리케이션의 특정 부분, 현재 실행중인 다른 애플리케이션들)을 검출할 수 있다. 맞춤화 모듈(420)은 다양한 상황들에서 어느 DFR UI 요소들(300)이 디스플레이될 것인지에 관한 사용자의 선호도를 설명하는 사용자 선호도 정보를 수신할 수 있다. DFR 에이전트(400)는, 디스플레이하기 위한 DFR UI 요소들(300)을 선택하기 위해 규칙 모듈(440), 컨텍스트 모듈(450), 및 제약 모듈(460) 중 하나 이상을 사용하여 시스템 상태 모듈(430) 및/또는 맞춤화 모듈(420)에 의해 제공되는 데이터를 평가할 수 있다. DFR UI 요소들(300)이 DFR(112)의 디스플레이 상에 제시되도록 선택되었으면, 디스플레이 조립 모듈(470)은 DFR(112) 상에 제시하기 위한 선택된 DFR UI 요소들(300)을 포함하는 이미지를 생성할 수 있다.

[0048] 규칙 모듈(440)은 하나 이상의 규칙을 사용하여, 등록된 DFR UI 요소들(300) 중에서 디스플레이되어야 하는 것을 선택할 수 있다. 규칙들은 DFR 에이전트(400)가, 그 시간에 디바이스(100) 컨텍스트에 맞는 DFR UI 요소들(300)을 발견하게 하도록 구성될 수 있다. 예를 들어, 규칙 모듈(440)은, 디바이스(100)가 착신 통화를 수신하고 있을 때, DFR(112)은 통화를 핸들링하기 위한 하나 이상의 DFR UI 요소(300)를 디스플레이해야 한다는 규칙을 규정할 수 있다(예컨대, 통화에 응답하고, 통화를 해제하고, 음성메일에 보내고, 통화 정보를 제공하는 것들을 위한 제어부들을 제공하는 것). 규칙들은 또한 선택된 DFR UI 요소들(300)이 어떻게 배열되고 크기설정되는지를 정의할 수 있다. 예를 들어, 착신 통화 시나리오에서, 규칙 모듈(440)은, 통화 DFR UI 요소들(300)이 전체 DFR(112)을 채워야 한다는 규칙, 또는 통화 DFR UI 요소들(300)이 DFR(112)의 좌측 절반 상에 나타나야 한다는 규칙을 규정할 수 있다. 규칙들을 사용하여, DFR 에이전트(400)는 시스템(100) 상태에 기초하여 DFR UI 요소들(300) 사이에서 스위칭할 수 있다.

- [0049] 도 5a는 예시적인 메인 디스플레이(104) 및 DFR(112)이다. 이 예는 동일한 앱에 관련된 데이터가 메인 디스플레이(104) 및 DFR(112) 둘 모두 상에 보여지는 시나리오를 예시한다. 하나의 예시적인 규칙에 따르면, DFR 에이전트(400)는 어느 애플리케이션이 가장 앞쪽에 있는지를 추적할 수 있고, 그 애플리케이션에 의해 등록된 하나 이상의 DFR UI 요소(300)를 디스플레이할 수 있다. 예를 들어, 도 4에 도시된 바와 같이, 앱 #1 윈도우(501)는 메인 디스플레이(104) 상에서 현재 활성화되며, 이는 사용자가 현재 앱 #1과 관여되어 있음을 암시한다. 다른 애플리케이션 윈도우가 열려 있을 수도 있지만(예컨대, 앱 #2 윈도우(502)), 백그라운드에 있을 수도 있거나 그렇지 않으면 사용자에게 의해 현재 사용되고 있지 않을 수 있다. 따라서, 앱 #1은 가장 앞쪽 애플리케이션으로서 간주될 수 있고, DFR 에이전트(400)는 가장 앞쪽 애플리케이션이 앱 #1에 의해 등록된 DFR UI 요소(301)를 선택하는 우선순위를 갖는다는 것을 나타내는 규칙을 따를 수 있다. DFR UI 요소(301)는 단일 기능을 제공하는 별개의 요소일 수 있거나, 또는 다수의 기능을 제공하는 그룹 요소일 수 있다(즉, 앱 #1에 의해 등록된 복수의 DFR UI 요소(300)로부터 조립된 단일 DFR UI 요소(301)).
- [0050] 앱 #1이 다수의 요소를 등록한 경우, DFR UI 요소(301)는 사용자가 현재 관여하고 있는 앱 #1의 특정 양태에 기초하여 추가로 선택될 수 있다. 예를 들어, 앱 #1이 수신자 다이얼로그 및 메시지 상자 다이얼로그에 관련된 별개의 등록된 DFR UI 요소들(300)을 갖는 이메일 애플리케이션인 경우, DFR 에이전트(400)는, 사용자가 수신자 다이얼로그 내에서 타이핑하고 있을 때 수신자 다이얼로그 요소들(예컨대, 주소록 요소, 자동완성 요소, cc/bcc 요소 등)을 디스플레이하고, 사용자가 메시지 상자 내에서 타이핑하고 있을 때 메시지 상자 요소들(예컨대, 폰트 선택 요소, 자동완성 요소, 서명 요소, 하이퍼링크 요소, 객체 삽입 요소 등)을 디스플레이할 수 있다.
- [0051] 도 5b는 예시적인 메인 디스플레이(104) 및 DFR(112)이다. 이 예는, 사용자가 도 5a의 예와는 상이한 앱에서 작업을 시작한 후에 동일한 앱에 관련된 데이터가 메인 디스플레이(104) 및 DFR(112) 둘 모두 상에 보여지는 시나리오를 예시한다. 도시된 바와 같이, 사용자가 앱 #1에서 작업하는 것으로부터 앱 #2에서 작업하는 것으로 스위칭할 때, 앱 #2 윈도우(502)는 메인 디스플레이(104) 상에서 활성화로 될 수 있다. 따라서, 앱 #2는 가장 앞쪽 애플리케이션으로서 간주될 수 있고, DFR 에이전트(400)는 가장 앞쪽 애플리케이션이 앱 #2에 의해 등록된 DFR UI 요소(302)를 선택하는 우선순위를 갖는다는 것을 나타내는 규칙을 따를 수 있다. DFR UI 요소(302)는 단일 기능을 제공하는 별개의 요소일 수 있거나, 또는 다수의 기능을 제공하는 그룹 요소일 수 있다(즉, 앱 #2에 의해 등록된 복수의 DFR UI 요소(300)로부터 조립된 단일 DFR UI 요소(302)).
- [0052] 도 5c는 시스템 우선순위 데이터가 DFR(112) 상에 보여지는, 예시적인 메인 디스플레이(104) 및 DFR(112)이다. 다른 예시적인 규칙에 따르면, 일부 DFR UI 요소들(300)은 시스템 우선순위 요소들일 수 있다. 예를 들어, 시스템 우선순위 요소들은 시스템 우선순위 애플리케이션들에 의해 등록될 수 있고/있거나, 소정 디바이스(100) 우선순위 상태들 또는 시스템 모달 이벤트들(예를 들어, 사용자가 애플리케이션 또는 디바이스(100)를 계속해서 사용하기 전에 이벤트를 다루어야 하는 경우)에 관련될 수 있다. 규칙은, 시스템 우선순위 이벤트들에 응답하여, 소정의 DFR UI 요소들(300)은 다른 무엇이 일어나고 있는지에 관계없이(예컨대, 메인 디스플레이(104) 상에서 어느 윈도우가 활성화인지에 관계없이) 제시될 수 있음을 요구할 수 있다. 도 5c에 도시된 바와 같이, 예를 들어, 앱 #2 윈도우(502)가 디스플레이(104) 상에서 가장 앞쪽에 있음에도 불구하고 시스템 우선순위 DFR UI 요소(305)가 DFR(112) 상에 보여진다.
- [0053] 이러한 규칙을 트리거하는 시스템 우선순위 이벤트는 규칙에 의해 특정된 임의의 이벤트일 수 있다. 일부 예들은 다음을 검출하는 것을 포함할 수 있다: 착신 메시지(예컨대, 오디오 또는 비디오 통화 또는 채팅 메시지) - 여기서, DFR 에이전트(400)는 메시지 애플리케이션 기능을 포함하는 시스템 우선순위 DFR UI 요소(305)를 선택할 수 있음 -; 활성 스크린 기록 세션 - 여기서, DFR 에이전트(400)는 스크린 기록 기능을 포함하는 시스템 우선순위 DFR UI 요소(305)를 선택할 수 있음 -; 특정 사용자 커맨드(예컨대, 사용자가, 기능 열을 디스플레이하기 위한 커맨드를 입력하는 것) - 여기서, DFR 에이전트(400)는 커맨드와 관련된 시스템 우선순위 DFR UI 요소(305)를 선택할 수 있음(예컨대, 기능 키들의 세트) -; 통지 또는 팝업 메시지를 수신하는 것 - 여기서, DFR 에이전트(400)는 통지 또는 메시지를 다루기 위한 제어부들을 포함하는 시스템 우선순위 DFR UI 요소(305)를 선택할 수 있음 -; 및/또는 기타 이벤트들. 관련된 예시적인 규칙에 따르면, 이벤트가 종료될 때, DFR 에이전트(400)는 이벤트 전에 보여졌던 DFR UI 요소(300)를 디스플레이하는 것으로 복귀할 수 있다. 도 5c의 예에서, 이것은 도 5b의 디스플레이로의 복귀일 수 있다. 다른 관련된 예시적인 규칙에 따르면, DFR 에이전트(400)는 시스템 우선순위 DFR UI 요소(305)를 해제하라는 사용자 커맨드(예컨대, 메시지 무시 커맨드 또는 기능 열 숨김 커맨드)에 응답하여 이벤트 전에 보여졌던 DFR UI 요소(300)를 디스플레이하는 것으로 복귀할 수 있다.
- [0054] 규칙들은 미리정의된 것일 수 있고/있거나 동적으로 설정될 수 있다. 일부 구현예들에서, 미리정의된 규칙들은 시스템(100) 설정들로 인해 DFR 에이전트(400)가 적용되는 규칙들일 수 있다. 예를 들어, 키보드(110)는 "디스



플레이 F-키" 버튼을 포함할 수 있다. DFR 에이전트(400)는 사용자가 디스플레이 F-키 버튼을 누를 때(또는 누르고 유지할 때) DFR(112) 상의 F-키 요소들(300)(예컨대, 기능 키들(F1 내지 F12)을 위한 요소들(300), 이스케이프 키 요소(300), 및/또는 다른 것들)의 디스플레이를 요구하는 미리정의된 규칙을 적용할 수 있다. 따라서, 사용자가 디스플레이 F-키 버튼을 누를 때(또는 누르고 유지할 때), DFR(112)은 이미지들/심볼들을 디스플레이하는 것으로부터 F-키 요소들(300)을 디스플레이하는 것으로 스위칭할 수 있다.

[0055] 동적으로 확립된 규칙들은 애플리케이션들에 의해 등록될 수 있다. 다른 예에서, 착신 메시지에 응답하여 메시지 DFR UI 요소(300)의 디스플레이를 요구하는 규칙은 동적으로 확립된 규칙일 수 있다. 예를 들어, 거동은 메시지에 책임이 있는 애플리케이션에 의해 등록 메타데이터에 정의될 수 있다. 애플리케이션은 DFR UI 요소(300)를 등록하고, DFR UI 요소(300)가 소정 조건들 하에서 규칙에 의해 디스플레이되게 하는 데 관심을 표명할 수 있다. DFR 에이전트(400)는 시스템 모달 애플리케이션들에 대한 우선순위들의 계층구조를 포함할 수 있는 전체 규칙들을 우선순위화하고 확립하기 위해, 다른 애플리케이션 관심들에 대조하여 이러한 관심을 평가할 수 있다.

[0056] 도 5d는 예시적인 메인 디스플레이(104) 및 DFR(112)이다. 이 예에서, 다수의 소스로부터의 요소들이 DFR(112) 상에 보여진다. 일부 경우에, 앱 #3에 의해 가장 앞쪽 윈도우(503)에 등록된 DFR UI 요소(303)는 전체 DFR(112)을 채우지 않을 수 있다. DFR 에이전트(400)는 이들 경우에 디스플레이하기 위한 하나 이상의 추가적인 DFR UI 요소들(예컨대, 앱 #4 DFR UI 요소(304), 시스템 DFR UI 요소(306))을 선택할 수 있다. DFR 에이전트(400)는 예를 들어 후술되는 바와 같이 응답자 트리를 순회(traverse)함으로써 추가적인 애플리케이션 DFR UI 요소들(예컨대, DFR UI 요소(304))을 선택할 수 있다.

[0057] 시스템 DFR UI 요소(306)는 애플리케이션-레벨 제어보다는 시스템-레벨 제어를 제공하는 요소일 수 있다. 예를 들어, 시스템 DFR UI 요소들(306)은 밝기 제어부들, 윈도우 배열 제어부들(예컨대, 앱 노출, 모든 앱 보기, 데스크톱 보기), 미디어 제어부들, 디스크 꺼내기 제어부들 등을 제공할 수 있다. 일부 경우에, 시스템 DFR UI 요소들(306)은 항상 존재할 수 있거나, 또는 시스템 우선순위 DFR UI 요소(305)가 DFR(112)을 인계받지 않는 한 항상 존재할 수 있다. 예를 들어, DFR 에이전트(400)는 시스템 DFR UI 요소(306)가 항상 DFR(112)의 우측에 나타나야 한다고 진술하는 규칙을 시행할 수 있다. 이러한 예에서, DFR 에이전트(400)는 DFR(112)의 나머지를 가장 앞쪽 애플리케이션의 DFR UI 요소들(300)로 채우지만 시스템 DFR UI 요소(306)를 DFR(112)의 우측에 남겨둘 수 있다. 시스템 DFR UI 요소들(306)의 일부 예들은 밝기, 볼륨, 무선 네트워킹 토글, 및/또는 다른 것들과 같은 일반적으로 유용한 제어부들을 제공할 수 있다.

[0058] 일부 실시예들에서, DFR 에이전트(400)는 전체 DFR(112)을 인계받도록 허용된 애플리케이션들의 화이트리스트를 유지할 수 있다. 예시적인 규칙에 따르면, 가장 앞쪽 애플리케이션이 화이트리스트 상에 있는 경우, 그것은 전체 DFR(112)을 인계받는 능력을 가질 수 있다. 예를 들어, DFR 에이전트(400)는 앱 #1 및 앱 #2를 포함하는 화이트리스트를 가질 수 있다. 따라서, 도 5a 및 도 5b에 도시된 바와 같이, 앱 #1 DFR UI 요소(301) 및 앱 #2 DFR UI 요소(302)는 DFR(112)을 채울 수 있다. 가장 앞쪽 애플리케이션이 화이트리스트 상에 있지 않은 경우, DFR 에이전트(400)는 전체 DFR(112)을 채우지 않는 애플리케이션에 의해 등록된 DFR UI 요소들을 선택할 수 있고, DFR(112) 디스플레이 공간의 나머지를 다른 DFR UI 요소들(300)로 채울 수 있다. 예를 들어, 앱 #3은 화이트리스트 상에 있지 않을 수 있으므로, DFR 에이전트(400)는, 도 5c에 도시된 바와 같이, 앱 #3 DFR UI 요소(303) 및 추가적인 DFR UI 요소들(304, 306)을 선택하여 DFR(112)을 채울 수 있다.

[0059] *응답자 체인*

[0060] 위에서 논의된 바와 같이, 다수의 애플리케이션들은 DFR UI 요소들(300)을 등록할 수 있고, DFR 에이전트(400)는 규칙들에 기초하여 어느 DFR UI 요소들(300)을 디스플레이할지를 결정할 수 있다. 이러한 결정을 용이하게 하기 위해, 컨텍스트 모듈(450)은 시스템 상태 모듈(430)에 의해 제공되는 시스템(100) 상태 데이터에 기초할 수 있는 애플리케이션들에 대한 응답자 체인들을 생성할 수 있다. 응답자 체인은 현재 시행되고 있는 임의의 규칙들에 대한 각각의 애플리케이션의 관계를 정의할 수 있다(예를 들어, 어느 애플리케이션이 가장 앞쪽에 있는지, 시스템 모달 이벤트가 진행 중인지 여부 등에 기초하여). 규칙 모듈(440)은 디스플레이하기 위한 DFR UI 요소들(300)을 선택하기 위해 응답자 체인을 순회할 수 있다.

[0061] 일부 구현예들에서, 응답자 체인은 애플리케이션의 하나 이상의 기능 레벨을 정의할 수 있다. 예를 들어, 애플리케이션은 애플리케이션 전체에 걸쳐 일반적으로 적용가능할 수 있는 기능들을 포함하는 일반적인 기능 레벨을 가질 수 있다. 애플리케이션은 또한 특정 애플리케이션 상태들에만 적용가능한 기능들을 포함하는 하나 이상의 특정 기능 레벨을 가질 수 있다. 애플리케이션은 특정 기능 레벨들에 관련된 DFR UI 요소들(300)을 등록할 수



있다. 예를 들어, 이메일 애플리케이션은 새로운 이메일 작성 기능, 이메일 검색 기능, 및/또는 연락처 관리 기능을 포함하는 일반적인 기능 레벨을 가질 수 있다. 이메일 애플리케이션은 "새로운 이메일 작성" DFR UI 요소(300), "이메일 검색" DFR UI 요소(300), 및/또는 "연락처 검색" DFR UI 요소(300)와 같은, 일반적인 기능 레벨에 특히 관련된 DFR UI 요소들(300)을 등록할 수 있다. 이메일 애플리케이션은, 수신자 추가 기능, 이메일 전송 기능, 및/또는 이메일 저장 기능을 포함하는, 사용자가 이메일 편집 윈도우에서 작업하고 있을 때 특히 적용가능한 기능을 포함하는 기능 레벨을 가질 수 있다. 이메일 애플리케이션은 "수신자 추가" DFR UI 요소(300), "이메일 전송" DFR UI 요소(300), 및/또는 "이메일 저장" DFR UI 요소(300)와 같은, 이메일 편집 윈도우 기능 레벨과 특히 관련된 DFR UI 요소들(300)을 등록할 수 있다. 이메일 애플리케이션은 폰트 선택 기능, 폰트 크기 선택 기능, 및/또는 폰트 효과 선택 기능을 포함하는, 사용자가 다이얼로그 박스에서 텍스트를 편집하고 있을 때 특히 적용가능한 기능을 포함하는 기능 레벨을 가질 수 있다. 이메일 애플리케이션은 "폰트 선택" DFR UI 요소(300), "폰트 크기 감소" DFR UI 요소(300), 및/또는 "밑줄" DFR UI 요소(300)와 같은, 텍스트 편집 기능 레벨과 특히 관련된 DFR UI 요소들(300)을 등록할 수 있다.

[0062] DFR 에이전트(400)는 디바이스(100) 상태에 대한 그들의 관련성에 기초하여 DFR UI 요소들(300)을 배열하기 위해 응답자 체인을 순회할 수 있으며, 그 결과들은 항목 트리로서 표현될 수 있다. 도 6a는 예시적인 항목 트리(600)이다. 도 6a에서, "W → X → Y → Z"는 응답자 체인이고, A는 응답자 체인을 지배하는 현재 상태를 표현한다(예컨대, 디바이스(100)에서 일어나는 것에 기초하여 적용되는 규칙들). 각각의 번호가 매겨진 블록 또는 "그룹" 블록(예컨대, 그룹 Ψ, 그룹 Ω)은 DFR UI 요소(300)를 표현한다. 일부 예들에서, DFR UI 요소들(300)은 애플리케이션 제어부들 및/또는 제어부들 사이의 배치를 위한 미리정의된 고정된 또는 유연한 공간들을 포함할 수 있다. 각각의 열은 애플리케이션의 단일 기능 레벨에 대응하는 DFR UI 요소들(300)의 그룹을 표현한다.

[0063] 각각의 열 내에서, DFR UI 요소들(300)은 순서화된 디폴트 세트에 배열될 수 있다. 일부 경우에, DFR UI 요소들(300)은 아래에서 더 상세히 논의되는 바와 같이 맞춤화가능할 수 있다. 만약 그렇다면, 애플리케이션의 등록은 제거될 수 없는 DFR UI 요소들(300)의 목록, 디폴트로 존재하지 않지만 추가될 수 있는 DFR UI 요소들(300)의 목록, 및 DFR UI 요소들(300)의 순서화된 디폴트 세트를 제공할 수 있다. 이러한 항목 트리(600)는 단지 예로서 제시되는데, 그 이유는 응답자 체인들이 기능 레벨들 및 DFR UI 요소들(300)의 임의의 배열을 가질 수 있어서, 순회될 때 다양한 상이한 항목 트리들(600)을 생성하게 된다. 또한, 본 명세서에 기술된 예들은 전체 DFR(112)에 대한 단일 응답자 체인을 가정하지만, 일부 실시예들은 복수의 응답자 체인들(예컨대, 좌측 응답자 체인, 우측 응답자 체인, 및 중앙 응답자 체인)을 사용하여 DFR(112)의 서브섹션들을 채울 수 있으며, 그 각각은 순회될 때 별개의 항목 트리(600)를 생성할 수 있다.

[0064] 일부 구현예들에서, DFR 에이전트(400)는, 항목 트리(600)로부터의 어느 DFR UI 요소(300)가 DFR(112)의 디스플레이 상에 제시될지를 결정할 때, 억제(suppression) 규칙들을 적용할 수 있다. 억제 규칙은 DFR 에이전트(400)가 응답자 체인을 순회하기 위해 적용할 수 있는 규칙의 하나의 유형일 수 있다. 디바이스(100) 상태는 사용자가 디바이스(100)와 상호작용함에 따라 변할 수 있다. 예를 들어, 디바이스(100)는 부팅할 수 있고, 열린 윈도우들을 갖는 어떠한 활성 애플리케이션들도 갖지 않을 수 있다. 사용자는 웹 브라우저 윈도우를 열어, 디바이스(100) 상태가 변하게 할 수 있다. 사용자는 웹 브라우저 주소 표시줄을 클릭하고 그 내에 타이핑할 수 있으며, 이는 디바이스(100) 상태가 다시 변하게 할 수 있다. 이어서, 사용자는 디스플레이된 웹사이트를 클릭하고 웹사이트를 읽기 위해 스크롤링하며, 이는 디바이스(100) 상태가 또 다시 변하게 할 수 있다. 응답자 체인에서의 각각의 기능 레벨은 현재 디바이스(100) 상태에 따라 더 포커싱되거나 덜 포커싱될 수 있다. 더 포커싱된 기능 레벨은 덜 포커싱된 기능 레벨보다 현재 디바이스(100) 상태에 더 관련될 수 있다. 하나의 예시적인 억제 규칙 하에서, DFR 에이전트(400)는 디스플레이하기 위한 DFR UI 요소들(300)을 선택할 때 응답자 체인 내의 더 포커싱된 기능 레벨들로 DFR UI 요소들(300)을 우선순위화할 수 있다. 다른 예시적인 억제 규칙 하에서, DFR 에이전트(400)는 디스플레이하기 위한 요소들을 선택할 때 덜 포커싱된 기능 레벨들로 DFR UI 요소들(300)을 우선순위화할 수 있다.

[0065] 예를 들어, 사용자가 웹 브라우저 윈도우를 열 때, 웹 브라우저에 의해 등록된 DFR UI 요소들(300)은 디폴트 시스템 DFR UI 요소들(300)보다 더 포커싱될 수 있다. 사용자가 주소 표시줄과 상호작용할 때, 주소 표시줄에 특히 관련되는 웹 브라우저에 의해 등록된 DFR UI 요소들(300)(예컨대, 북마크, 최근에 방문한 URL 등)은 웹 브라우저에 의해 등록된 다른 DFR UI 요소들(300)보다 더 포커싱될 수 있다. 사용자가 웹사이트와 상호작용하는 것으로 스위칭할 때, 주소 표시줄에 특정된 DFR UI 요소들(300)은 덜 포커싱될 수 있는 반면, 웹사이트 디스플레이에 특정된 DFR UI 요소들(300)(예컨대, 줌, 뒤로, 앞으로 등)은 더 포커싱될 수 있다.

[0066] 다른 예에서, 디바이스(100)는 메인 디스플레이(104) 상의 윈도우가 수신자 다이얼로그 박스 및 메시지 다이얼

로그 박스를 포함하는 이메일 애플리케이션을 실행하고 있을 수 있다. 사용자가 수신자 다이얼로그 박스에 타이핑하고 있을 때, 애플리케이션의 그 양태에 관련된 DFR UI 요소들(300)(예컨대, 주소록 검색)은 메시지 다이얼로그 박스에 관련된 DFR UI 요소들(300)(예컨대, HTML 메시지 포매팅 옵션들)보다 더 포커싱될 수 있으며, 이는 이어서 이메일 애플리케이션에 관련되지 않은 DFR UI 요소들(300)(예컨대, 시스템 볼륨 제어부들)보다 더 포커싱될 수 있다. 사용자가 메시지 다이얼로그 박스에 타이핑하고 있을 때, 메시지 다이얼로그 박스에 관련된 DFR UI 요소들(300)은 수신자 다이얼로그 박스에 관련된 DFR UI 요소들(300)보다 더 포커싱될 수 있다.

[0067] DFR 에이전트(400)는 응답자 체인 내의 어느 DFR UI 요소들(300)이 디스플레이되어야 하는지를 결정하기 위해 억제제를 사용할 수 있다. 예를 들어, 임의의 주어진 DFR UI 요소(300)에 대한 억제 규칙은 다음의 4개의 상이한 형태들 중 하나를 가질 수 있다: 더 포커싱된 항목들에 의해 억제되는, 덜 포커싱된 항목들에 의해 억제되는, 그리고 더 포커싱된 항목들을 억제하는, 또는 덜 포커싱된 항목들을 억제하는. 억제된 DFR UI 요소들(300)은 억제되지 않은 DFR UI 요소들(300)보다 더 낮은 우선순위를 가질 수 있고, 그에 따라 DFR 에이전트(400)는 디스플레이하기 위한 억제되지 않은 DFR UI 요소들(300)을 선택할 수 있으며, 이는 단지, 모든 억제되지 않은 DFR UI 요소들(300)이 선택된 후 남은 공간이 있을 경우 억제된 DFR UI 요소들(300)을 선택한다. 예를 들어, DFR 에이전트(400)는 볼륨 제어 DFR UI 요소(300)가 보다 포커싱된 항목들에 의해 억제된다는 것을 규정하는 억제 규칙을 시행할 수 있다. 사용자가 이메일 애플리케이션을 사용하고 있는 경우, 이메일 애플리케이션에 의해 등록된 연락처 목록 DFR UI 요소(300)는 이메일 애플리케이션에 관련되지 않는 볼륨 제어 DFR UI 요소(300)보다 더 포커싱될 수 있다. 따라서, DFR 에이전트(400)는 볼륨 제어 DFR UI 요소(300) 대신에 디스플레이하기 위해 연락처 목록 DFR UI 요소(300)를 선택할 수 있다. 다른 예에서, 사용자가 이메일 애플리케이션에서 메시지 텍스트를 편집하고 있을 때, 응답자 체인의 텍스트 편집 기능 레벨에서의 텍스트 밑줄 DFR UI 요소(300)는 응답자 체인의 덜 포커싱된 메시지 윈도우 기능 레벨에서 이메일 전송 DFR UI 요소(300)보다 더 포커싱될 수 있다. DFR 에이전트(400)는, 더 포커싱된 항목들이 우선순위를 갖고 이메일 전송 DFR UI 요소(300) 대신에 디스플레이를 위해 밑줄 DFR UI 요소(300)를 선택한다고 규정하는 억제 규칙을 시행할 수 있다.

[0068] 검출된 착신 메시지 또는 애플리케이션과의 사용자 상호작용과 같은 디바이스(100) 이벤트가 발생할 때, DFR 에이전트(400)는 디스플레이하기 위한 DFR UI 요소들(300)을 선택하기 위해 응답자 체인을 순회할 수 있다. DFR 에이전트(400)는 응답자 체인에서 이벤트 A에 가장 가까운 기능 레벨 내의 DFR UI 요소들(300)로 시작할 수 있다. 예를 들어, DFR 에이전트(400)는 더 포커싱된 DFR UI 요소들(300)이 덜 포커싱된 DFR UI 요소들(300)을 억제하는 규칙을 시행할 수 있다. 도 6a의 항목 트리(600)에서, 1 및 2로 라벨링된 DFR UI 요소들(300)은 이벤트 A에 가장 가깝다(예컨대, W는 응답자 체인의 가장 포커싱된 레벨일 수 있다). DFR 에이전트(400)는 가장 가까운 기능 레벨 내의 DFR UI 요소들(300) 중 임의의 것이 이벤트에 관련되는지 여부를 결정할 수 있다. 만약 그렇다면, 이들은 선택될 수 있다. 그렇지 않은 경우, DFR 에이전트(400)는 가장 가까운 기능 레벨을 건너뛰고 다음의 가장 가까운 것(예컨대, 도 6a에서의 제2 레벨, 여기서 X는 응답자 체인의 다음 가장 포커싱된 레벨임)으로 이동하고 항목 트리(600)를 따라 동일한 분석 및 선택 등을 수행할 수 있다. 디스플레이 조립 모듈(470)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 생성하고, 디스플레이를 위해 이미지를 DFR(112)에 전송할 수 있다.

[0069] 다른 예에서, DFR 에이전트(400)는 덜 포커싱된 DFR UI 요소들(300)이 더 포커싱된 DFR UI 요소들(300)을 억제하는 규칙을 시행할 수 있다. 도 6a의 항목 트리(600)에서, 1 및 2로 라벨링된 DFR UI 요소들(300)은 이벤트 A에 가장 가깝다(예컨대, W는 응답자 체인의 가장 덜 포커싱된 레벨일 수 있다). DFR 에이전트(400)는 가장 가까운 기능 레벨 내의 DFR UI 요소들(300) 중 임의의 것이 이벤트에 관련되는지 여부를 결정할 수 있다. 만약 그렇다면, 이들은 선택될 수 있다. 그렇지 않은 경우, DFR 에이전트(400)는 가장 가까운 기능 레벨을 건너뛰고 다음 가장 가까운 것(예컨대, 도 6a에서의 제2 레벨, 여기서 X는 응답자 체인의 다음으로 가장 적게 포커싱된 레벨임)으로 이동하고 항목 트리(600)를 따라 동일한 분석 및 선택 등을 수행할 수 있다. 디스플레이 조립 모듈(470)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 생성하고, 디스플레이를 위해 이미지를 DFR(112)에 전송할 수 있다.

[0070] 유사하게, 기능 레벨로부터의 DFR UI 요소들(300)이 선택되었고, DFR(112) 상에 더 많은 DFR UI 요소들(300)을 위한 더 많은 공간이 여전히 있을 때, DFR 에이전트(400)는 추가 DFR UI 요소들(300)을 선택하기 위해 항목 트리(600)를 따라 계속할 수 있다. 디스플레이 조립 모듈(470)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 생성하고, 디스플레이를 위해 이미지를 DFR(112)에 전송할 수 있다.

[0071] 일부 구현예들에서, 항목 트리(600)는 응답자 체인 내의 다른 항목들에 대한 플레이스홀더들을 포함할 수 있다. 예를 들어, 애플리케이션은 또한 도 6a에서 "기타" 블록들로서 도시된, "기타 항목 프록시"로 지칭되는 항목을

등록할 수 있다. 기타 항목 프록시 등록은, DFR UI 요소들(300)이 DFR(112) 상에 배치되고 공간이 남아있는 경우, 다른 기능 레벨들 상에 있고/있거나 다른 애플리케이션들에 의해 등록된 다른 DFR UI 요소들(300)이 추가될 수 있다는 것을, DFR 에이전트(400)에 알릴 수 있다. 기타 항목 프록시는, 예를 들어, 다른 DFR UI 요소들(300)이 이미-추가된 DFR UI 요소들(300)에 대해 관련하여 배치될 수 있는 곳에 대한 애플리케이션의 선호도를 정의할 수 있다.

[0072] 기능 레벨은, 기타 항목 프록시를 포함함으로써 다른 기능 레벨들로부터의 DFR UI 요소들(300)이 그 자신의 DFR UI 요소들(300)과 함께 배치되도록 할 수 있다. 레벨이 기타 항목 프록시를 포함하지 않고, DFR(112) 상에 더 많은 DFR UI 요소들(300)에 대한 공간이 있는 경우, DFR 에이전트(400)는 그 레벨의 DFR UI 요소들(300)을 완전히 건너뛰고 항목 트리(600) 아래로 이동할 수 있다. 따라서, 기타 항목 프록시를 등록하는 것에 대한 실패는, 레벨의 요소들(300)이, DFR 에이전트(400)가 그 레벨에 대한 요소들(300)만이 관련되어 있다고 결정하는 상황들에서만 디스플레이되어야 한다는 것을 나타내는 표시일 수 있다.

[0073] 일부 경우에, DFR 에이전트(400)는 주요(예컨대, 중앙에 위치한(centered)) 요소(300)로서 디스플레이되도록 선택된 DFR UI 요소(300)를 지정할 수 있다. 애플리케이션들은 DFR UI 요소들(300)을 DFR(112) 상에서 두드러지게 특징화되어야 하는 주요 요소들로서 등록할 수 있다. DFR 에이전트(400)는, 항목 트리(600)로부터, 주요한 것으로서 등록된 어떤 선택된 DFR UI 요소(300)가 항목 트리(600) 내의 이벤트에 가장 가까운지를 결정할 수 있고, 그 DFR UI 요소(300)를 주요 요소(300)로서 지정할 수 있다. 디스플레이 조립 모듈(470)이 디스플레이하기 위한 이미지를 생성할 때, 주요 DFR UI 요소(300)는 이미지 내에서 중앙에 위치될 수 있다.

[0074] 다른 실시예들에서, 주요 DFR UI 요소들(300)은 반드시 중앙에 위치되어야 하는 것은 아닐 수 있다. 예를 들어, DFR 에이전트(400)는, 주요 DFR UI 요소(300)가 DFR(112)의 좌측에 나타나도록 요구되는 규칙을 시행할 수 있다. 따라서, 디스플레이 조립 모듈(470)이 디스플레이하기 위한 이미지를 생성할 때, 주요 DFR UI 요소(300)는 이미지의 좌측에 배치될 수 있다.

[0075] 도 7a는 도 6a의 항목 트리(600)로부터 형성되었을 수 있는 복수의 예시적인 DFR(112) 레이아웃들을 도시한다. 레이아웃(112A)은, DFR 에이전트(400)가 응답자 체인을 순회하고 그리고 항목 트리(600)의 제2 레벨을 건너뛰면서 항목 트리(600)의 제1, 제3, 및 제4 레벨들로부터 모든 이용가능한 DFR UI 요소(300)를 선택하는 것으로부터 기인할 수 있다. 예를 들어, 제2 레벨은 이벤트 트리거링 레이아웃(112A)에 관련되지 않을 수 있고 억제될 수 있다. 레이아웃(112B)은, DFR 에이전트(400)가 응답자 체인을 순회하고 그리고 항목 트리(600)의 제1 및 제2 레벨들로부터의 모든 이용가능한 DFR UI 요소(300) 및 항목 트리(600)의 제3 레벨로부터의 단일 DFR UI 요소(300)를 선택하는 것으로부터 기인할 수 있다. 예를 들어, DFR(112)은 제1 및 제2 레벨 DFR UI 요소들(300)이 선택된 후 단지 하나 이상의 DFR UI 요소(300)를 위한 공간을 가졌을 수 있고, DFR 에이전트(400)는 항목 트리(600)의 제3 레벨로부터 가장 관련되거나 가장 중요한 DFR UI 요소(300)를 선택했을 수 있다. 레이아웃(112C)은, DFR 에이전트(400)가 응답자 체인을 순회하고 그리고 항목 트리(600)의 제1 및 제2 레벨들로부터 모든 이용가능한 DFR UI 요소(300)를 선택하는 것으로부터 기인할 수 있다. 이들 레이아웃들(112A 내지 112C)은 단지 예로서 제시된다. 다른 응답자 체인들은 상이한 레이아웃들을 생성할 수 있고, 심지어 도 6a의 항목 트리(600)는 상이한 규칙들이 적용되는 경우 상이한 레이아웃들을 생성할 수 있다.

[0076] 일부 구현예들에서, 항목 트리들은 분할되고 재조직화될 수 있다. 예를 들어, 디바이스(100)는 특정 사용자 언어를 사용하여 기능하도록 구성될 수 있다. 사용자는 영어, 스페인어, 아랍어, 또는 다른 언어들로 입력들을 허용하하고 데이터를 디스플레이하도록 디바이스(100)를 설정할 수 있다. 일부 언어들은 좌측에서 우측으로 읽을 수 있고, 다른 언어들은 우측에서 좌측으로 읽을 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 디폴트로 DFR(112) 레이아웃들을 좌측에서 우측으로의 방식으로 배열할 수 있다. 디바이스(100)가 우측에서 좌측으로의 언어를 사용하는 것으로 기능하도록 구성될 때, DFR 에이전트(400)는 DFR(112) 레이아웃들을 우측에서 좌측으로 읽도록 재배열할 수 있다.

[0077] 도 6b는 예시적인 항목 트리(610)이다. 이러한 예에서, 항목 트리(610)는 2개의 레벨을 갖는다. 제1 레벨은 이벤트 A에 가장 가까운 1, 2, 및 3으로 라벨링된 DFR UI 요소들(300)을 포함한다. 제2 레벨은 4 및 5로 라벨링된 DFR UI 요소들(300)을 포함한다. 3으로 라벨링된 DFR UI 요소(300)는 주요 DFR UI 요소(300)일 수 있다. 도 7b는 응답자 체인을 순회하고 디스플레이를 위해 항목 트리(610)에서 각각의 DFR UI 요소(300)를 선택하는 것으로부터 기인할 수 있는 예시적인 DFR 레이아웃(112D)을 도시한다.

[0078] 주요 DFR UI 요소(300)로서 3으로 라벨링된 DFR UI 요소(300)를 유지하면서 DFR 레이아웃(112D)을 우측에서 좌측 레이아웃으로 재배열하기 위해, DFR 에이전트(400)는 응답자 체인을 순회하고 이어서 항목 트리(610)를 분할



할 수 있다. 도 6c는 항목 트리(610)가 3개의 트리들, 항목 트리(610A), 항목 트리(610B), 및 항목 트리(610C)로 분할된 예를 도시한다. 항목 트리(610B)는 DFR(112) 상에서 중앙에 위치되어 유지될 수 있는 주요 DFR UI 요소(300)를 포함하는 트리이다. 항목 트리(610A)는 응답자 체인의 순회 동안 주요 DFR UI 요소(300)의 좌측에 디스플레이하도록 선택된 DFR UI 요소들(300)을 포함한다. 항목 트리(610C)는 응답자 체인의 순회 동안 주요 DFR UI 요소(300)의 우측에 디스플레이하도록 선택된 DFR UI 요소들(300)을 포함한다. DFR 에이전트(400)는 각각의 항목 트리(610A 내지 610C)를 별도로 처리할 수 있고, 좌측 항목 트리(610A) 및 우측 항목 트리(610C) 내의 DFR UI 요소들(300)의 순서를 반전시킬 수 있다. DFR 에이전트(400)는 항목 트리(610B)로부터의 DFR UI 요소들(300)의 양측 상의 항목 트리들(610A, 610C)로부터의 반전된 DFR UI 요소들(300)을 조합함으로써 DFR 레이아웃을 구축할 수 있다. 예를 들어, 도 7b는 항목 트리들(610A 내지 610C)로부터 기인할 수 있는 DFR 레이아웃(112E)을 도시한다. 항목 트리(610A) 내의 DFR UI 요소들(300)은 DFR 레이아웃(112D)과 비교하여 반전되지만, 여전히 주요 DFR UI 요소(300)의 좌측으로 나타난다. 항목 트리(610C) 내의 DFR UI 요소들(300)은 DFR 레이아웃(112D)과 비교하여 반전되지만, 여전히 주요 DFR UI 요소(300)의 우측으로 나타난다.

[0079] 컨텍스트

[0080] 응답자 체인을 사용하는 것 대신에, 또는 응답자 체인을 사용하는 것에 더하여, 컨텍스트 모듈(450)은 컨텍스트에 기초하여 DFR UI 요소들(300)을 선택할 수 있다. 컨텍스트는 디바이스(100) 상태 또는 활동을 기술할 수 있다. 예를 들어, 텍스트 다이얼로그 박스와 상호작용하는 사용자가 컨텍스트일 수 있고, 애플리케이션 상태가 컨텍스트일 수 있고, 내부 디바이스 상태가 컨텍스트일 수 있고, 착신 메시지 또는 다른 이벤트가 컨텍스트일 수 있는 등이다. 컨텍스트-기반 선택에서, 요소들(300)은 응답자들(즉, 응답자 체인의 순서화된 계층구조 내의 애플리케이션들)보다는 특정 컨텍스트들과 연관될 수 있다. DFR 에이전트(400)는 DFR UI 요소(300) 등록에서 제출된 DFR UI 요소들(300)의 설명들에 기초하여 주어진 컨텍스트와 관련된 DFR UI 요소들(300)을 선택할 수 있다.

[0081] 일부 경우들에서, 여러 컨텍스트들이 디바이스(100)에 동시에 적용가능할 수 있다. 애플리케이션들은 DFR UI 요소들(300) 또는 DFR UI 요소들(300)의 그룹들을 컨텍스트들의 세트들에 대해 등록할 수 있으며, 이는 주어진 세트의 컨텍스트들이 등록 메타데이터 내에 동시에 존재할 때 DFR UI 요소들(300)의 선택 및 배열을 특정한다. 추가적으로, 컨텍스트들은 우선순위 레벨들을 가질 수 있다. DFR 에이전트(400)는 일부 컨텍스트들을 다른 컨텍스트들에 비해 우선순위화하고 가장 높은 우선순위 컨텍스트들에 관련된 DFR UI 요소들(300)을 먼저 선택할 수 있다. 예를 들어, 사용자는 착신 메시지가 디바이스(100)에서 수신될 때 문서에 텍스트를 타이핑하기 위해 텍스트 편집 애플리케이션을 사용하고 있을 수 있다. 일반적인 텍스트 편집 애플리케이션 컨텍스트, 특정 타이핑 기능 컨텍스트, 및 메시지 컨텍스트를 포함하는, 여러 컨텍스트들이 적용될 수 있다. DFR 에이전트(400)는 착신 메시지에 가장 높은 우선순위를 제공하고, 메시지에 관련된 DFR UI 요소들(300)을 디스플레이할 수 있다. 추가적인 DFR UI 요소들(300)을 위한 공간이 있는 경우, DFR 에이전트(400)는 다음으로 가장 높은 우선순위 컨텍스트(예컨대, 타이핑)로부터 그것들을 선택하고, 우선순위 레벨의 순서로 컨텍스트들을 통해 선택할 수 있는 등이다. 디스플레이 조립 모듈(470)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 생성하고, 디스플레이를 위해 이미지를 DFR(112)에 전송할 수 있다.

[0082] DFR 에이전트(400)가 DFR UI 요소들(300)을 선택하기 위해 응답자 체인 및 컨텍스트 둘 모두를 사용하는 실시예들에서, 응답자 체인 내의 각각의 기능 레벨의 위치, 및 각각의 애플리케이션의 억제 규칙들은, 컨텍스트들로서 간주될 수 있고 그에 따라 컨텍스트 모듈(450)에 의해 평가될 수 있다. 예를 들어, 하나의 컨텍스트는 체인 내의 제1 응답자로서 특정 기능 레벨의 상태일 수 있고, 다른 컨텍스트는 다른 체인 응답자에 의해 억제되는 체인 응답자의 상태일 수 있다. 도 6 및 도 7을 참조하면, DFR 디스플레이(112A)는 예를 들어, 가장 높은 우선순위 컨텍스트를 갖는 제1 레벨의 제1 응답자 위치, 및 제3 레벨이 제2 레벨을 억제하게 하는 억제 규칙으로부터 기인할 수 있다.

[0083] 제약

[0084] 일부 구현예들에서, DFR 에이전트(400)는 디스플레이하기 위한 DFR UI 요소들(300)을 선택 및/또는 배열하기 위해 제약들을 사용할 수 있다. 제약들은 DFR UI 요소들(300) 사이의 관계들을 표현할 수 있다. 제약들을 적용하는 것은 어느 요소들(300)이 가시적인지 그리고 가시적인 요소들(300)이 DFR(112) 상에 어떻게 배열되는지를 확립할 수 있다. 예를 들어, 제약들은 "이들 3개의 요소(300)가 좌측에서 우측으로, 또는 우측에서 좌측으로 소정 순서로 나타나야 한다" 또는 "이들 3개의 요소(300)는 그들 사이에 적어도 특정된 양의 공간을 갖고서 나타나야 한다" 또는 "이 요소(300)는 좌측에 나타나야 한다" 또는 "이 요소(300)는 우측에 나타나야 한다" 등과

같은 관계들을 정의할 수 있다. 제약들은 또한 요소들(300) 사이의 고정된 공간 또는 유연한(예컨대, 크기조정 가능한) 공간의 양을 정의할 수 있다. 복수의 서브요소들을 포함하는 그룹 요소들(300)은 그룹 요소들(300) 내의 서브-요소들에 대한 유사한 관계들을 특징하는 서브-제약들을 가질 수 있다. 그룹 요소(300) 자체는 또한 개별 요소들(300)과 동일한 방식으로 일반적인 제약들에 영향을 받을 수 있다.

[0085] 제약 모듈(460)은 DFR UI 요소들(300)의 순서화되지 않은 세트를 선택하고 DFR UI 요소들(300) 및 DFR UI 요소들(300)의 순서화된 그룹들에 제약들의 세트를 적용할 수 있다. 제약들을 적용함으로써, 제약 모듈(460)은 어느 DFR UI 요소들(300)이 어떤 배열로 디스플레이될지를 선택할 수 있다. 디스플레이 조립 모듈(470)은 선택된 DFR UI 요소들(300)을 갖는 이미지를, 제약들에 의해 정의된 배열로 생성할 수 있다.

[0086] 디스플레이하기 위한 DFR UI 요소들(300)을 선택 및 배열하기 위해, DFR 에이전트(400)는 현재 디바이스(100) 컨텍스트(들) 또는 상태에 적용가능한 DFR UI 요소들(300)에 어떤 제약들이 적용되는지를 결정함으로써, 제약 시스템을 확립할 수 있다. DFR 에이전트(400)는 제약 시스템에서의 제약들을 해결하여, 응답자 체인 또는 컨텍스트에 의해 규정된 순서로 제약들을 만족시키려고 시도할 수 있다. 제약들을 해결하기 위해, DFR 에이전트(400)는 규정된 순서로 제1 제약을 적용하여, 제1 제약에 따라 DFR UI 요소들(300)을 선택 및 배열할 수 있다. 추가적인 DFR UI 요소들(300)을 위한 공간이 있다고 가정하면, 공간이 채워질 때까지, DFR 에이전트(400)는 제2 제약을 적용하고, 추가적인 DFR UI 요소들(300)을 추가 및 배열하려고 시도하는 등을 수행할 수 있다.

[0087] 각각의 제약은 적어도 3개의 카테고리들 - 요구되는, 선택적인, 또는 맞춤의 - 중 하나에 속할 수 있다. DFR UI 요소(300)가 요구되는 제약에 의해 제약되는 경우, DFR 에이전트(400)는 제약을 만족시킬 수 있어야 하거나 그렇지 아니면 DFR UI 요소(300)는 디스플레이되지 않을 것이다. DFR UI 요소(300)가 선택적인 제약에 의해 제약되는 경우, DFR 에이전트(400)는 제약을 만족시키려고 시도할 수 있지만, 제약을 만족시키는 것이 다른 제약(예컨대, 요구되는 제약)과 충돌될 경우 제약을 위반하여 DFR UI 요소(300)를 여전히 디스플레이할 수 있다. DFR 에이전트(400)는 선택적인 제약과 유사하게 맞춤 제약을 처리할 수 있지만, 맞춤 제약은 충돌시 선택적인 제약보다 우선할 수 있는데, 그 이유는 맞춤 제약이 사용자-특정될 수 있고 따라서 그것이 사용자의 선호도에 대응하므로 더 중요한 것으로 간주되기 때문이다.

[0088] 제약들은 로직 및 선형 표현들의 혼합으로서 표현될 수 있다. 예를 들어, DFR UI 요소 X(300)가 DFR(112)의 좌측 에지 상에 나타나도록 요구하는 요구된 제약은 다음과 같이 표현될 수 있다:  $X$  좌측 에지 > 모든 다른 좌측 에지들 또는  $X$ 는 비가시적임. DFR 에이전트(400)가, 이러한 제약을 포함하는 제약 시스템에서의 제약들을 해결할 때, 제약이 만족될 수 없는 경우, 연관된 DFR UI 요소 X(300)는 디스플레이를 위해 선택되지 않을 수 있다. 다른 예에서, DFR UI 요소 Y(300)가 DFR(112)의 좌측 에지 상에 나타나도록 요구하는 선택적인 제약은 다음과 같이 표현될 수 있다:  $Y$  좌측 에지 > 모든 다른 좌측 에지들, 또는  $Y$  좌측 에지와 다른 DFR UI 요소의 우측 에지 사이의 차이가 가능한 한 작음. DFR 에이전트(400)가 이러한 제약을 포함하는 제약 시스템을 해결할 때, 제약이 만족될 수 없는 경우, 연관된 DFR UI 요소 Y(300)는 다른 제약들에 의해 지시된 바와 같이 하나 이상의 다른 DFR UI 요소들(300)의 우측에 있을 수 있지만, 다른 제약들을 고려하여 가능한 한 가장 좌측에 있을 수 있다. 선택적인 제약들은 또한 우선순위 값을 포함할 수 있어서, 선택적인 제약들이 각자의 우선순위 값들에 기초하여 제약 시스템에서 우선순위화될 수 있다. 예를 들어, 2개의 DFR UI 요소(300)가, 그들이 DFR(112) 상에서 가장 좌측에 있어야 한다고 진술하는 선택적인 제약들을 갖는 경우, 더 높은 값을 갖는 요소(300)는 더 낮은 값을 갖는 DFR UI 요소(300)의 좌측에 배치될 수 있다.

[0089] 도 6a 및 도 7a를 참조하면, DFR 디스플레이(112C)는 제약 시스템을 해결하는 것으로부터 기인할 수 있으며, 여기서 예를 들어 DFR UI 요소 2는 제약 - 2 우측 에지 > 모든 다른 우측 에지들 또는 2가 비가시적임 - 을 갖고, 그룹 요소 X는 제약 - X 우측 에지 > 모든 다른 우측 에지들, 또는 X 우측 에지와 다른 요소의 좌측 에지 사이의 차이가 가능한 한 작음 - 을 갖고, DFR UI 요소 1은 제약 - 1 좌측 에지 > 모든 다른 좌측 에지들 또는 1은 비가시적임 - 을 갖는다.

[0090] 제약-기반 DFR UI 요소(300) 선택 프로세스들은 위에서 논의된 우선순위-기반 요소(300) 선택 방법들과는 상이하게 기능할 수 있다. 우선순위-기반 선택에서, 각각의 DFR UI 요소(300)는 우선순위를 가질 수 있다. 너무 많은 DFR UI 요소들(300)이 이용가능할 때, DFR 에이전트(400)는 가장 높은 우선순위의 DFR UI 요소들(300)로 시작할 수 있고, DFR(112)에 맞는 DFR UI 요소들(300)의 세트가 생성될 때까지 우선순위 목록에 따라 내려갈 수 있다. 제약 모델을 사용하면, 고유 우선순위가 공급될 필요가 없다. 대신에, 제약은 DFR UI 요소(300)가 디스플레이되어야 하는 조건들을 정의하고, 조건들이 선택적인지 또는 요구되는지 여부를 정의할 수 있다. DFR 에이전트(400)는 DFR(112)에 맞는 DFR UI 요소들(300)의 세트를 생성하기 위해 가능한 한 많은 제약들을 만족시킬

수 있다.

[0091] *커맨드 처리*

[0092] DFR 에이전트(400)는 DFR(112) 터치 인터페이스를 사용하여 입력되는 커맨드를 처리하기 위해 DFR(112) 상의 DFR UI 요소들(300)의 배열을 사용할 수 있다. 사용자가 DFR(112)을 터치할 때, 입력 모듈(480)은 DFR(112)로부터 특정 터치 위치를 나타내는 데이터를 수신할 수 있다. 디스플레이 조립 모듈(470)은 어느 DFR UI 요소(300)가 터치된 위치에 있는지를 나타내는 데이터를 입력 모듈(480)에 공급할 수 있다. 따라서, 입력 모듈(480)은 터치 위치를 관련된 DFR UI 요소(300)와 상관시킬 수 있다. 관련된 DFR UI 요소(300)를 결정한 후에, 입력 모듈(480)은 터치 이벤트에 대해 어느 애플리케이션이 알아야 하는지를 결정할 수 있다. 예를 들어, 입력 모듈(480)은 어느 애플리케이션이 DFR UI 요소(300)를 등록했는지를 결정하고 등록 애플리케이션의 적절한 API를 결정할 수 있다. 입력 모듈(480)은 적절한 API를 호출하여 커맨드를 애플리케이션에 전달할 수 있다. 다수의 애플리케이션들이 디스플레이 상에 DFR UI 요소들(300)을 동시에(at once) 가질 수 있기 때문에, DFR 에이전트(400)는 이벤트 통지들을 각각의 애플리케이션에 개별적으로 전달할 수 있다.

[0093] DFR 에이전트(400)는 개별 DFR UI 요소들(300)을 DFR(112)의 개별 영역들 내로 배치하여, 어느 DFR UI 요소(300)가 사용자에게 의해 눌러지는지의 추적을 허용할 수 있다. 예를 들어, DFR 에이전트(400)가 디스플레이하기 위한 DFR UI 요소들(300)을 선택했다면, DFR 에이전트(400)는 DFR(112)을 영역들로 분할하고 각각의 DFR UI 요소(300)를 별개의 영역 내에 놓을 수 있다. 각각의 영역의 형상은, 들어오는 터치를 어떻게 해석할지를 나타내는 메타데이터를 제공할 수 있다. 영역들은 이미지로 합성되고 DFR(112) 상에 렌더링될 수 있다. 렌더링된 이미지는 DFR(112) 또는 다른 곳에 전송될 수 있다(예컨대, 시뮬레이션용 윈도우 온 스크린(window on screen), 녹화를 위한 영화 캡처, 스크린 캡처로서 파일로 전송된 하나의 프레임 등). 영역들은 또한 디버깅을 가능하게 할 수 있다. 예를 들어, 디버그 모드에서, DFR 에이전트(400)는 DFR(112)로 하여금, 어느 DFR UI 요소(300)가 어느 애플리케이션과 연관되어 사용자가 어느 애플리케이션들이 DFR(112)의 어느 영역들에 참여하고 있는지를 한눈에 볼 수 있도록 하는지를 말하는 텍스트를 오버레이하게 할 수 있다. 디버깅은, DFR(112)의 기능에 영향을 미치지 않으면서 그리고 애플리케이션이 디버그 모드를 인식하지 않고서, 수행될 수 있다.

[0094] *맞춤화*

[0095] 도 8은 예시적인 DFR 레이아웃 편집기 인터페이스(800)를 도시한다. 디바이스(100)는 DFR(112) 디스플레이들을 맞춤화하기 위한 애플리케이션을 포함할 수 있다. 예를 들어, 사용자는 DFR 맞춤화 모드에 진입할 수 있으며, 여기서 사용자는 DFR(112) 내에서 DFR UI 요소들(300)을 배치하고 배열할 수 있다. DFR 맞춤화 모드에서, 디바이스(100)는 메인 디스플레이(104) 상에 팔레트(802)를 디스플레이할 수 있다. 팔레트(802)는 DFR(112) 내에 놓일 수 있는 DFR UI 요소들(300)을 포함할 수 있다. 마우스 또는 터치 패드와 같은 입력 디바이스를 사용하여, 사용자는 일부 실시예들에서, 팔레트(802) 내의 하나 이상의 DFR UI 요소(300)를 선택하고 이들을 메인 디스플레이(104)로부터 DFR(112)로 드래그할 수 있다. 디바이스(100)가 DFR 맞춤화 모드에 있는 동안, 사용자는 또한 DFR(112) 상에 디스플레이되고 있는 하나 이상의 DFR UI 요소(300)를 선택하고, 이들을 DFR(112) 상의 다른 위치들로 드래그할 수 있다. 사용자는 또한 하나 이상의 DFR UI 요소(300)를 DFR(112)로부터 멀리 드래그하여, 이들을 완전히 제거하거나 이들을 메인 디스플레이(104)로 드래그할 수 있다. 다른 실시예들에서, 메인 디스플레이(104)는 또한 사용자가 DFR UI 요소들(300)을 그 내부로 드래그할 수 있는 가상 DFR을 디스플레이할 수 있다. 도 8의 예에서, 사용자는 DFR UI 요소들(300-1, 300-2, 300-3) 및 그룹 DFR UI 요소(300-X)를 드래그하여 이들을 DFR(112) 또는 가상 DFR 상에 원하는 대로 위치시켰다. 팔레트(802)는 DFR UI 요소들(300)의 하나 이상의 디폴트 세트(806)를 포함할 수 있다. DFR(112)을 디폴트 디스플레이로 복원하기 위해, 사용자는 전체 디폴트 세트(806)를 DFR(112)로 드래그할 수 있다.

[0096] 팔레트(802)의 내용은 디바이스(100)의 현재 상태에 의해 좌우될 수 있다. 예를 들어, 사용자가 이메일 애플리케이션을 사용하고 있고 맞춤화 모드에 진입하는 경우, 팔레트(802) 내의 DFR UI 요소들(300)은 이메일 애플리케이션에 의해 등록된 DFR UI 요소들(300) 및 일반적으로 적용가능한 시스템 DFR UI 요소들(예컨대, 오디오 볼륨, 스피커 요소들 등과 같은 시스템-범위의 제어들을 위한 DFR UI 요소들(300))을 포함할 수 있다. 일부 실시예들에서, 사용자는 동일한 인터페이스(800)를 사용하여, 또한 시스템 디폴트 상태에 대해 DFR(112)을 맞춤화할 수 있다. 맞춤화 모듈(420)은 상태에 대한 사용자의 편집들을 저장할 수 있고, 그 상태가 다시 마주치게 될 때, DFR 에이전트(400)는 사용자의 편집들을 포함하는 DFR(112)에 대한 디스플레이 이미지를 생성할 수 있다. 예를 들어, 규칙 모듈(440)은 사용자 편집들이 디폴트보다 우선하는 규칙을 적용할 수 있거나, 또는 제약 모듈(460)은 사용자의 편집들에 대응하는 제약들을 우선순위화할 수 있다.



- [0097] 그룹 DFR UI 요소들(300)은 또한 인터페이스(800)를 통해 맞춤화가능할 수 있다. 사용자는 맞춤화를 위한 그룹을 선택하고 그룹 안팎으로 DFR UI 요소들(300)을 드래그하고 그룹 내에서 DFR UI 요소들(300)을 배열할 수 있다. 다수의 애플리케이션들 또는 이벤트들이 맞춤화된 그룹을 사용하는 경우, 맞춤화들은 그룹이 사용될 때마다 존재할 수 있다.
- [0098] 맞춤화는 사용자에게 유연성을 제공할 수 있지만, DFR(112) 상의 DFR UI 요소들(300)의 배치는 전술된 바와 같이 규칙들 및/또는 제약들에 의해 여전히 지배될 수 있다. 인터페이스(800)는 규칙들 또는 제약들을 위반하는 임의의 사용자 변경들을 거부할 수 있다. 예를 들어, DFR UI 요소 1이 DFR(112)의 좌측 상에 있도록 요구되는 경우, 사용자는 DFR UI 요소 3을 DFR UI 요소 1의 좌측으로 드래그할 수 없을 수 있다. 일부 그룹들 또는 DFR(112) 레이아웃들은 맞춤화가능한 것으로서 마킹되거나 등록될 수 있고/있거나 다른 것들은 그렇지 않을 수 있거나, 또는 그 반대일 수 있다. 전술된 바와 같이, 애플리케이션의 등록은 제거될 수 없는 DFR UI 요소들(300)의 목록을 제공할 수 있다. 이러한 경우에, 규칙은 사용자가 제거가능하지 않은 DFR UI 요소들(300)을 제거하는 것을 방지할 수 있다. 맞춤화할 때 DFR UI 요소들(300)을 분할하는 것 또는 그룹화된 DFR UI 요소들(300)을 분할하는 것을 방지하는 규칙들이 규정될 수 있다. 예를 들어, 사용자는 오디오 볼륨 업 DFR UI 요소(300)와 오디오 볼륨 다운 DFR UI 요소(300) 사이에 DFR UI 요소(300)를 배치하는 것이 제한될 수 있다. 따라서, 맞춤화되도록 허용되는 그룹들 또는 레이아웃들만이 인터페이스(800)에서 편집될 수 있다.
- [0099] *예시적인 프로세스*
- [0100] 도 9는 가장 앞쪽 애플리케이션에 기초한 예시적인 DFR 이미지 생성 프로세스(900)의 흐름도이다. 예를 들어, 프로세스(900)는 디바이스(100)의 상태의 변화에 응답하여 DFR(112) 상에 디스플레이하기 위한 이미지를 생성하기 위해 컴퓨팅 디바이스(100)에 의해 수행될 수 있다.
- [0101] 단계(902)에서, 컴퓨팅 디바이스(100)는 DFR(112) 상에 디폴트 그래픽 요소들을 제시할 수 있다. 예를 들어, 디바이스(100) 시동 시 또는 어떠한 애플리케이션들도 활성화하지 않을 때, DFR 에이전트(400)는 디폴트 디스플레이 데이터(예컨대, 오디오 볼륨 제어부들, 스크린 밝기 제어부들, 이스케이프 키 등과 같은 시스템 DFR UI 요소들(300)을 포함하는 이미지)를 DFR(112)에 전송할 수 있다. 구체적으로, 전술된 바와 같이, 시스템 상태 모듈(430)은 디바이스(100) 상태(즉, 활성화 애플리케이션이 없음)를 결정할 수 있다. DFR 에이전트(400)는 규칙들, 컨텍스트, 및/또는 제약들에 기초하여 디스플레이하기 위한 DFR UI 요소들(300)을 선택 및 배열하기 위해 상태를 사용할 수 있다. 어떠한 애플리케이션들도 활성화하지 않을 때, DFR UI 요소들(300)은 등록될 수 없으므로, DFR 에이전트(400)는 이용가능한 시스템 DFR UI 요소들(300)(예컨대, 스크린 밝기, 볼륨 무선 네트워크 토클 등) 중에서 DFR UI 요소들(300)을 선택할 수 있다. 디스플레이 조립 모듈(420)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 DFR(112)에 전송할 수 있다. 프로세스(900)는 이 예에서 디폴트 디스플레이로부터 시작하지만, 일부 경우에, DFR(112)은 이 단계에서 특정 디바이스(100) 애플리케이션 또는 상태에 관련된 이미지들을 디스플레이하고 있을 수 있다. 예를 들어, 하나 이상의 애플리케이션이 실행되고 있고 DFR UI 요소들(300)을 등록했을 때, DFR 에이전트(400)는 전술된 바와 같이 하나 이상의 등록된 DFR UI 요소(300) 및/또는 하나 이상의 시스템 DFR UI 요소(300)를 선택할 수 있다.
- [0102] 단계(904)에서, 컴퓨팅 디바이스(100)는 가장 앞쪽 애플리케이션이 변경되었음을 검출할 수 있다. 예를 들어, 사용자는 새로운 애플리케이션에서 작업을 시작하거나 디스플레이(104) 상의 열린 윈도우를 일차 윈도우로서 선택할 수 있다. 도 5a에서, 앱 #1이 가장 앞쪽 애플리케이션이다. 도 5b에서, 앱 #2가 가장 앞쪽 애플리케이션이다. DFR 에이전트(400)는 가장 앞쪽 애플리케이션(예컨대, 현재 사용자 입력 포커스를 갖는 애플리케이션)이 제1 애플리케이션으로부터 제2 애플리케이션으로 변경되었음을 나타내는 상태 데이터를 컴퓨팅 디바이스(100)의 운영 체제로부터 수신할 수 있다.
- [0103] 단계(906)에서, 컴퓨팅 디바이스(100)는 가장 앞쪽 애플리케이션으로부터 하나 이상의 DFR UI 요소(300)에 대한 등록 데이터를 수신할 수 있다. 예를 들어, 가장 앞쪽 애플리케이션은 요소들(300)을 DFR 에이전트(400)에 등록할 수 있다. 예를 들어, 사용자가 웹 브라우저를 활성화로 사용하고 있는 경우, 웹 브라우저는 주소 표시줄, 리로드, 북마크, 앞으로, 뒤로, 새 탭, 이력, 및/또는 다운로드를 위한 제어부들을 갖는 DFR UI 요소들(300)을 등록할 수 있다.
- [0104] 단계(908)에서, 컴퓨팅 디바이스(100)는 컴퓨팅 디바이스(100) 상에서 실행되는 다른 애플리케이션들로부터 하나 이상의 DFR UI 요소(300)에 대한 등록 데이터를 수신할 수 있다. 예를 들어, 다른 애플리케이션들은 요소들(300)을 DFR 에이전트(400)에 등록할 수 있다. 예를 들어, 사용자가 미디어 재생기를 시작했고 백그라운드에서 미디어 재생기(및 포그라운드에서 웹 브라우저)를 실행하고 있는 경우, 미디어 재생기는 미디어 재생 및 선택을



위한 제어부들을 갖는 DFR UI 요소들(300)을 등록할 수 있다.

- [0105] 단계(910)에서, 컴퓨팅 디바이스(100)는 요소 선택 규칙들에 기초하여, 등록된 DFR UI 요소들(300)을 평가할 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 요소 선택 섹션에서 전송된 바와 같은 규칙들에 기초하여, 등록된 DFR UI 요소들(300)을 평가할 수 있다. 예를 들어, 웹 브라우저는 전체 DFR(112)을 인계받도록 허용된 화이트리스트된 애플리케이션일 수 있다.
- [0106] 단계(912)에서, DFR 에이전트(400)는 규칙들에 기초하여 디스플레이하기 위한 DFR UI 요소들(300)을 선택할 수 있다. 웹 브라우저가 화이트리스트되는 예에서, DFR 에이전트(400)는 웹 브라우저에 의해 등록된 DFR UI 요소들(300)만을 선택하고 미디어 재생기에 의해 등록된 DFR UI 요소들(300)을 배제할 수 있다.
- [0107] 단계(914)에서, DFR 에이전트(400)는 선택된 DFR UI 요소들(300)을 포함하는 이미지(예컨대, 전송된 바와 같은 비트맵)를 생성하고 이미지를 DFR(112)에 전송할 수 있다. DFR(112)은 이미지를 디스플레이할 수 있다.
- [0108] 도 10은 시스템 우선순위 이벤트 발생에 기초한 예시적인 DFR 이미지 생성 프로세스(1000)의 흐름도이다. 예를 들어, 프로세스(1000)는 시스템 모달 이벤트에 응답하여 DFR(112) 상에 이미지를 디스플레이하고 시스템 모달 이벤트가 종료될 때 이전 이미지를 디스플레이하는 것으로 복귀하도록 컴퓨팅 디바이스(100)에 의해 수행될 수 있다.
- [0109] 단계(1002)에서, 컴퓨팅 디바이스(100)는 DFR(112) 상에 디폴트 그래픽 요소들을 제시할 수 있다. 예를 들어, 디바이스(100) 시동 시 또는 어떠한 애플리케이션들도 활성화하지 않을 때, DFR 에이전트(400)는 디폴트 디스플레이 데이터(예컨대, 시스템 DFR UI 요소들(300)을 포함하는 이미지)를 DFR(112)에 전송할 수 있다. 구체적으로, 전송된 바와 같이, 시스템 상태 모듈(430)은 디바이스(100) 상태(즉, 활성 애플리케이션이 없음)를 결정할 수 있다. DFR 에이전트(400)는 규칙들, 컨텍스트, 및/또는 제약들에 기초하여 디스플레이하기 위한 DFR UI 요소들(300)을 선택 및 배열하기 위해 상태를 사용할 수 있다. 어떠한 애플리케이션들도 활성화하지 않을 때, DFR UI 요소들(300)은 등록될 수 없으므로, DFR 에이전트(400)는 이용가능한 시스템 DFR UI 요소들(300)(예컨대, 스크린 밝기, 볼륨 무선 네트워크 토글 등) 중에서 DFR UI 요소들(300)을 선택할 수 있다. 디스플레이 조립 모듈(420)은 선택된 DFR UI 요소들(300)을 포함하는 이미지를 DFR(112)에 전송할 수 있다. 프로세스(1000)는 이 예에서 디폴트 디스플레이로부터 시작하지만, 일부 경우에, DFR(112)은 이 단계에서 특정 디바이스(100) 애플리케이션 또는 상태에 관련된 이미지들을 디스플레이하고 있을 수 있다. 예를 들어, 하나 이상의 애플리케이션이 실행되고 있고 DFR UI 요소들(300)을 등록했을 때, DFR 에이전트(400)는 전송된 바와 같이 하나 이상의 등록된 DFR UI 요소(300) 및/또는 하나 이상의 시스템 DFR UI 요소(300)를 선택할 수 있다.
- [0110] 단계(1004)에서, 시스템 우선순위 이벤트가 발생할 수 있다. 예를 들어, 디바이스(100)는 착신 메시지를 수신할 수 있고, 사용자는 디스플레이 F-키 버튼을 누를 수 있는 등이다. 도 5b 및 도 5c에 도시된 바와 같이, 시스템 우선순위 이벤트는 사용자가 작업하고 있는 애플리케이션과 관련되지 않을 수 있다. 예를 들어, 도 5b에서, 사용자는 앱 #2에서 작업하고 있고, DFR(112)은 앱 #2 DFR UI 요소(302)를 디스플레이하고 있다. 도 5c에서, 시스템 우선순위 이벤트의 발생 이후에, DFR(112)은, 사용자가 여전히 앱 #2에서 작업하고 있다는 사실에도 불구하고, 시스템 우선순위 DFR UI 요소(305)를 디스플레이하고 있다. 도 10의 예에서, 시스템 우선순위 이벤트는 사용자가 임의의 추가 동작들을 취하기 전에 응답하거나 해제해야 하는 착신 메시지 통지와 같은, 시스템 모달 이벤트이다.
- [0111] 단계(1006)에서, DFR 에이전트(400)는 시스템 모달 이벤트에 응답하여 디스플레이하도록 특정된 DFR UI 요소들(300)을 포함하는 이미지(예컨대, 전송된 바와 같은 비트맵)를 생성하고 이미지를 DFR(112)에 전송할 수 있다. DFR(112)은 이미지를 디스플레이할 수 있다.
- [0112] 단계(1008)에서, 시스템 우선순위 이벤트는 종료될 수 있다. 예를 들어, 착신 메시지가 응답될 수 있고, 사용자는 기능 열 키를 해제할 수 있고, 사용자는 메시지 통지를 해제할 수 있는 등이다.
- [0113] 단계(1010)에서, DFR 에이전트(400)는 시스템 모달 이벤트 이전에 활성화되었던 DFR(112) 디스플레이를 복원할 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 자동으로 복원을 수행할 수 있는데, 그 이유는 일단 시스템 우선순위 이벤트가 종료되면, 시스템(100)은 그것이 이벤트 이전에 있었던 상태로 복귀할 수 있기 때문이다. 따라서, DFR 에이전트(400)는 그것이 이전에 사용한 동일한 규칙들, 컨텍스트들, 및/또는 제약들에 따라 디스플레이 이미지를 생성할 수 있다. 이 예에서, 복원된 디스플레이는 디폴트 디스플레이이지만, 상태-특정 디스플레이가 DFR(112) 상에 있었다면(예컨대, 프로세스(900)에 따라 생성된 디스플레이), 그것은 동일한 방식으로 복원될 수 있다. 그러나, 디바이스(100)의 상태가 이벤트 동안 추가적인 변화들을 겪는 경우(예컨대, 사용자가 새

로운 애플리케이션에서 작업하기 시작하여, 새로운 애플리케이션이 가장 앞쪽이 되게 함), DFR 에이전트(400)는 이벤트 종료 시 새로운 상태에 따라 디스플레이 이미지를 생성할 수 있다. 예를 들어, DFR 에이전트(400)는, 사용자가 새로운 가장 앞쪽 애플리케이션에서 작업하기 전에 디스플레이되었던 DFR UI 요소들(300)보다는, 새로운 가장 앞쪽 애플리케이션에 의해 등록된 DFR UI 요소들(300)을 선택할 수 있다.

- [0114] 도 11은 DFR 입력들을 검출 및 라우팅하기 위한 예시적인 프로세스(1100)의 흐름도이다. 예를 들어, 컴퓨팅 시스템(100)은 DFR(112)이 이미지를 디스플레이하고 있고 사용자가 이미지를 터치함으로써 커맨드를 입력할 때 프로세스(1100)를 수행할 수 있다.
- [0115] 단계(1102)에서, DFR(112)은 DFR(112)을 터치하는 사용자에 의해 입력된 커맨드를 검출할 수 있다. DFR(112)은 터치의 위치를 나타내는 메타데이터를 DFR 에이전트(400)에 전송할 수 있다.
- [0116] 단계(1104)에서, DFR 에이전트(400)는 터치 위치를, 그 위치에 디스플레이된 DFR UI 요소(300)와 상관시킬 수 있다. 예를 들어, 도 5d를 참조하면, 위치 메타데이터가 터치가 DFR(112)의 좌측에서 등록되었음을 나타내는 경우, DFR 에이전트(400)는 터치를 DFR UI 요소(304)와 상관시킬 수 있다. 위치 메타데이터가 터치가 DFR(112)의 우측에서 등록되었음을 나타내는 경우, DFR 에이전트(400)는 터치를 DFR UI 요소(305)와 상관시킬 수 있다. 위치 메타데이터가 터치가 DFR(112)의 중심에서 등록되었음을 나타내는 경우, DFR 에이전트(400)는 터치를 DFR UI 요소(303)와 상관시킬 수 있다.
- [0117] 단계(1106)에서, DFR 에이전트(400)는 상관된 DFR UI 요소(300)의 선택의 의미를 결정할 수 있다. 예를 들어, 선택된 DFR UI 요소(300)가 "볼륨 업" 요소라고 DFR 에이전트(400)가 결정하는 경우, DFR 에이전트(400)는 시스템 볼륨 제어 애플리케이션이 볼륨 업 커맨드를 수신할 필요가 있을 수 있다고 결정할 수 있다. 선택된 DFR UI 요소(300)가 "통화 응답" 요소라고 DFR 에이전트(400)가 결정하는 경우, DFR 에이전트(400)는 메시징 애플리케이션이 통화 응답 커맨드를 수신할 필요가 있을 수 있다고 결정할 수 있다.
- [0118] 단계(1108)에서, DFR 에이전트(400)는 커맨드를 적절한 애플리케이션에 전송할 수 있다. 예를 들어, DFR 에이전트(400)는 전송된 바와 같이 커맨드에 대응하는 적절한 애플리케이션 API를 호출할 수 있다. 따라서, DFR 에이전트(400)는 DFR(112) 커맨드들의 초기 처리를 수행하여, DFR(112)과 디바이스(100) 애플리케이션들 사이의 추상화(abstraction) 계층을 제공할 수 있다.
- [0119] 도 12는 응답자 체인(600)을 사용하여 DFR UI 요소들을 선택하기 위한 예시적인 프로세스(1200)의 흐름도이다. 예를 들어, 컴퓨팅 시스템(100)은 응답자 체인(600)을 순회하고 이에 의해 디스플레이하기 위한 DFR UI 요소들(300)을 선택하는 프로세스(1200)를 수행할 수 있다.
- [0120] 단계(1202)에서, DFR 에이전트(400)는 디바이스(100) 상에서 실행되는 적어도 하나의 애플리케이션으로부터, 등록된 DFR UI 요소들(300)을 수신할 수 있다. 예를 들어, 사용자는 이메일 애플리케이션에서 작업하고 있을 수 있고, 이메일 애플리케이션은 복수의 DFR UI 요소(300)를 등록할 수 있다. 일부 구현예들에서, DFR 에이전트(400)는 초기에, 등록된 DFR UI 요소들(300)을 애플리케이션에 대한 디폴트 시동 순서로 디스플레이할 수 있다. 예를 들어, DFR 에이전트(400)는 초기 디스플레이를 위해 일반적인 기능 레벨 DFR UI 요소들(300)을 선택할 수 있다.
- [0121] 단계(1204)에서, DFR 에이전트(400)는 이벤트가 발생했거나 디바이스(100) 상태가 변했음을 나타내는 데이터를 수신할 수 있다. 예를 들어, 사용자는 애플리케이션의 특정 부분에서 작업을 시작할 수 있다. 사용자는 예를 들어 이메일 편집 윈도우를 열고/열거나 이메일 메시지 작성 상자에서 타이핑을 시작할 수 있다.
- [0122] 단계(1206)에서, DFR 에이전트(400)는 이벤트와 관련된 응답자 체인(600)을 결정할 수 있다. DFR 에이전트(400)는 이벤트로부터의 거리 순서로, 응답자 체인(600) 내의 기능 레벨들 및/또는 DFR UI 요소들(300)을 배열할 수 있다. 예를 들어, 사용자가 메시지 작성 박스에서 타이핑하고 있는 경우, 텍스트 편집 기능 레벨 DFR UI 요소들(300)은 가장 포커싱될 수 있고 따라서 응답자 체인(600) 내의 이벤트에 가장 가까울 수 있다. 이메일 윈도우 기능 레벨에 대한 DFR UI 요소들(300)은 다음으로 가장 포커싱될 수 있고, 응답자 체인(600)에서 다음에 올 수 있다. 이메일 애플리케이션에 대한 일반적인 기능 레벨 DFR UI 요소들(300)은 가장 적게 포커싱될 수 있고, 응답자 체인(600)에서 마지막으로 올 수 있다.
- [0123] 단계(1208)에서, DFR 에이전트(400)는 이벤트에 대한 제1 응답자를 발견하기 위해 응답자 체인(600)을 순회할 수 있다. 예를 들어, 도 6a에서, DFR UI 요소들 1 및 2를 포함하는 제1 레벨은 이벤트 A에 대한 제1 응답자일 수 있다.

- [0124] 단계(1210)에서, DFR 에이전트(400)는 전술된 바와 같이, 제1 응답자 DFR UI 요소들(300)을 DFR 이미지에 추가할 수 있다. 예를 들어, 도 7a의 DFR 디스플레이(112B)에서, DFR UI 요소들 1 및 2는 DFR(112)의 중심에 배치된다.
- [0125] 단계(1212)에서, 제1 응답자 DFR UI 요소들(300)이 DFR 이미지를 채우지 않는 경우, DFR 에이전트(400)는 디스플레이하기 위한 추가적인 DFR UI 요소들(300)을 발견하기 위해 추가로 응답자 체인(600)을 순회할 수 있다. 예를 들어, DFR 에이전트(400)는 도 6a의 응답자 체인(600)을 순회하고 DFR UI 요소들 3, 4 및 그룹 요소 X를 선택할 수 있다.
- [0126] 단계(1214)에서, DFR 에이전트(400)는 추가적인 DFR UI 요소들(300)을 이미지에 추가할 수 있다. 예를 들어, 도 7a의 DFR 디스플레이(112B)에서, DFR UI 요소들 3, 4 및 그룹 요소 X는 DFR(112) 내의 요소들 1 및 2의 양측 상에 배치된다. 전술된 바와 같이, DFR 에이전트(400)는 이미지를 생성하고 이를 디스플레이를 위해 DFR(112)에 전송할 수 있다.
- [0127] 도 13은 하나 이상의 제약에 부합하는 DFR 요소들을 선택하기 위한 예시적인 프로세스(1300)의 흐름도이다. 예를 들어, 컴퓨팅 시스템(100)은 프로세스(1300)를 수행하여, 하나 이상의 제약을 만족시키고 이에 의해 디스플레이하기 위한 DFR UI 요소들(300)을 선택할 수 있다.
- [0128] 단계(1302)에서, DFR 에이전트(400)는 디바이스(100) 상에서 실행되는 애플리케이션들로부터, 등록된 DFR UI 요소들(300)을 수신할 수 있다.
- [0129] 단계(1304)에서, DFR 에이전트(400)는 등록된 DFR UI 요소들(300)을 검사하여, 요소들(300)에 대해 등록된 제약들을 식별하고 전술된 바와 같이 제약들을 해결할 수 있다. 예를 들어, 도 7a를 참조하면, DFR 디스플레이(112A)는 제약 시스템을 해결하는 것으로부터 기인할 수 있으며, 여기서 DFR UI 요소 3은 제약 - 3 우측 에지 > 모든 다른 우측 에지들 또는 3이 비가시적임 - 을 갖고, 그룹 DFR UI 요소 Y는 제약 - Y 좌측 에지 > 모든 다른 좌측 에지들, 또는 Y 좌측 에지와 다른 DFR UI 요소의 우측 에지 사이의 차이가 가능한 한 작음 - 을 갖고, DFR UI 요소 7은 제약 - 7 우측 에지 > 모든 다른 우측 에지들, 또는 7이 비가시적임 - 을 갖는다.
- [0130] 단계(1306)에서, DFR 에이전트(400)는, 필요한 경우, 제약들로 인해 디스플레이될 수 없는 DFR UI 요소들(300)을 폐기할 수 있다. 예를 들어, 도 7a에서, DFR UI 요소 3에 대한 제약은 DFR UI 요소 7에 대한 제약과 충돌한다. DFR UI 요소 7 제약은 DFR UI 요소 3 제약보다 더 높은 우선순위를 가질 수 있어서, DFR UI 요소 3은 폐기될 수 있다.
- [0131] 단계(1308)에서, DFR 에이전트(400)는 제약들에 따라 나머지 DFR UI 요소들(300)을 배열할 수 있다. DFR 에이전트(400)는 결정된 배열로 DFR UI 요소들(300)을 포함하는 이미지를 생성할 수 있다.
- [0132] 단계(1310)에서, DFR 에이전트(400)는 디스플레이하기 위한 이미지를 DFR(112)에 전송할 수 있다.
- [0133] *그래픽 사용자 인터페이스*
- [0134] 상기 본 개시내용은 다양한 특징들, 프로세스들 또는 워크플로우들을 구현하기 위한 다양한 GUI들에 대해 설명한다. 이들 GUI들은 랩톱 컴퓨터, 데스크톱 컴퓨터, 컴퓨터 단말기, 텔레비전 시스템, 태블릿 컴퓨터, 이북리더기(e-book reader) 및 스마트폰들을 포함하지만, 이에 한정되지 않는 다양한 전자 디바이스들 상에 제시될 수 있다. 이들 전자 디바이스들 중 하나 이상은 터치 감응형 표면을 포함할 수 있다. 터치 감응형 표면은 다수의 동시적인 입력 점들을 처리할 수 있으며, 이는 입력 점 각각의 압력, 정도 또는 위치와 관련된 데이터의 처리를 포함할 수 있다. 이러한 처리는 핀칭(pinching) 및 스와이핑(swiping)을 포함하는, 다수의 손가락을 이용한 제스처들을 용이하게 할 수 있다.
- [0135] 본 개시내용에서 GUI 내의 사용자 인터페이스 요소들을 "선택한다(select)" 또는 "선택하는(selecting)"이라고 지칭할 경우, 이러한 표현들은 마우스 또는 다른 입력 디바이스를 가지고 사용자 인터페이스 요소 상부에서 클릭하거나 "호버링(hovering)"하는 것, 또는 하나 이상의 손가락 또는 스타일러스를 가지고 사용자 인터페이스 요소 상에 터치, 탭핑(tapping) 또는 제스처 입력(gesturing)하는 것을 포함하는 것으로 이해된다. 사용자 인터페이스 요소는 가상의 버튼, 메뉴, 선택기, 스위치, 슬라이더, 스크러버, 노브(knob), 썸네일, 링크, 아이콘, 라디오 버튼, 체크 박스 및 사용자로부터 입력을 수신하거나 사용자에게 피드백을 제공하기 위한 임의의 다른 메커니즘일 수 있다.
- [0136] *예시적인 시스템 아키텍처*

- [0137] 도 14는 도 1 내지 도 13의 특징부들 및 프로세스들을 구현할 수 있는 예시적인 컴퓨팅 디바이스(1400)의 블록도이다. 컴퓨팅 디바이스(1400)는 메모리 인터페이스(1402), 하나 이상의 데이터 프로세서, 이미지 프로세서 및/또는 중앙 처리 장치(1404), 및 주변기기 인터페이스(1406)를 포함할 수 있다. 예를 들어, 하나 이상의 프로세서(1404)는 메인 프로세서(202) 및 이차 프로세서(204)를 포함할 수 있다. 메모리 인터페이스(1402), 하나 이상의 프로세서(1404) 및/또는 주변기기 인터페이스(1406)는 별도의 컴포넌트들일 수 있거나 하나 이상의 집적 회로에 통합될 수 있다. 컴퓨팅 디바이스(1400) 내의 다양한 컴포넌트들은 하나 이상의 통신 버스들 또는 신호 라인들에 의해 결합될 수 있다.
- [0138] 센서, 디바이스 및 서브시스템은 다수의 기능들을 용이하게 하기 위하여 주변기기 인터페이스(1406)에 결합될 수 있다. 예를 들어, 모션 센서(1410), 광 센서(1412) 및 근접 센서(1414)는 배향, 조명 및 근접 기능들을 용이하게 하기 위하여 주변기기 인터페이스(1406)에 결합될 수 있다. 다른 센서들(1416)이 또한 GNSS(global navigation satellite system)(예컨대, GPS 수신기), 온도 센서, 생체인식 센서, 자력계 또는 다른 감지 디바이스와 같은 주변기기 인터페이스(1406)에 연결되어, 관련 기능들을 용이하게 할 수 있다.
- [0139] 카메라 서브시스템(1420) 및 광학 센서(1422), 예컨대, CCD(charged coupled device) 또는 CMOS(complementary metal-oxide semiconductor) 광학 센서는 사진 및 비디오 클립의 녹화와 같은 카메라의 기능을 용이하게 하도록 이용될 수 있다. 카메라 서브시스템(1420) 및 광학 센서(1422)는, 예컨대 안면 인식 분석을 수행함으로써, 사용자의 인증 동안에 사용될 사용자의 이미지를 수집하는 데 사용될 수 있다.
- [0140] 통신 기능들은 무선 주파수 수신기와 송신기 및/또는 광학(예컨대, 적외선) 수신기와 송신기를 포함할 수 있는, 하나 이상의 무선 통신 서브시스템(1424)을 통하여 용이하게 될 수 있다. 통신 서브시스템(1424)의 특정 설계와 구현은 컴퓨팅 디바이스(1400)가 동작하도록 의도되는 통신 네트워크(들)에 의존할 수 있다. 예를 들어, 컴퓨팅 디바이스(500)는 GSM 네트워크, GPRS 네트워크, EDGE 네트워크, Wi-Fi 또는 WiMax 네트워크, 및 블루투스™ 네트워크를 통해 동작하도록 설계된 통신 서브시스템들(1424)을 포함할 수 있다. 특히, 무선 통신 서브시스템(1424)은 디바이스(1400)가 다른 무선 디바이스를 위한 기지국으로서 구성될 수 있도록 호스팅 프로토콜을 포함할 수 있다.
- [0141] 오디오 서브시스템(1426)은 화자 인식, 음성 복제, 디지털 녹음, 및 전화 통신 기능과 같은 음성-지원 기능들을 용이하게 하기 위하여 스피커(1428) 및 마이크로폰(51430)에 결합될 수 있다. 오디오 서브시스템(1426)은 예를 들어 음성 커맨드, 성문 감정(voiceprinting) 및 음성 인증의 처리를 용이하게 하도록 구성될 수 있다.
- [0142] I/O 서브시스템(1440)은 터치 표면 제어기(1442) 및/또는 기타 입력 제어기(들)(51444)를 포함할 수 있다. 터치 표면 제어기(1442)는 터치 표면(1446)에 결합될 수 있다. 터치 표면(1446)과 터치 표면 제어기(1442)는, 예를 들어, 용량성, 저항성, 적외선 및 표면 탄성파(surface acoustic wave) 기술들뿐만 아니라, 터치 표면(1446)과의 하나 이상의 접촉점을 결정하기 위한 다른 근접 센서 어레이들 또는 다른 요소들을 포함하지만 이들로 제한되지 않는 복수의 터치 감응성 기술들 중 임의의 기술을 사용하여 접촉 및 이동 또는 그의 중단을 검출할 수 있다.
- [0143] 기타 입력 제어기(들)(1444)는 하나 이상의 버튼들, 로커 스위치(rocker switch), 지동륜/thumb-wheel), 적외선 포트, USB 포트, 및/또는 포인터 디바이스, 예컨대 스타일러스와 같은 기타 입력/제어 디바이스들(1448)에 결합될 수 있다. 하나 이상의 버튼(도시되지 않음)은 스피커(1428) 및/또는 마이크로폰(1430)의 볼륨 제어를 위한 업/다운 버튼을 포함할 수 있다.
- [0144] 일 구현예에서, 제1 지속기간 동안의 버튼 누름은 터치 표면(1446)의 잠금을 결합해제할 수 있고; 제1 지속기간보다 긴 제2 지속기간 동안의 버튼 누름은 컴퓨팅 디바이스(1400)에 대한 전력을 켜거나 끌 수 있다. 제3 지속기간 동안의 버튼 누름은, 사용자가 마이크로폰(1430)에 커맨드를 말하여 디바이스가 음성 커맨드를 실행하게 할 수 있는, 음성 제어, 또는 음성 커맨드, 모드를 활성화시킬 수 있다. 사용자는 버튼들 중 하나 이상의 버튼의 기능을 맞춤화할 수 있다. 터치 표면(1446)은 예를 들어 가상 또는 소프트 버튼 및/또는 키보드를 구현하는 데에 또한 사용될 수 있다.
- [0145] 컴퓨팅 디바이스(1100)는 DFR(1180)을 포함할 수 있다. DFR(1180)은 터치 감응형 디스플레이, 디스플레이 제어기, 터치 입력 제어기, 및/또는 GUI를 디스플레이하고 GUI와의 사용자 상호작용으로부터 커맨드들을 수신하도록 구성된 다른 하드웨어를 포함할 수 있다.
- [0146] 일부 구현예들에서, 컴퓨팅 디바이스(1400)는 MP3, AAC 및 MPEG 파일과 같은, 레코딩된 오디오 및/또는 비디오 파일을 제시할 수 있다. 일부 구현예들에서, 컴퓨팅 디바이스(1400)는 아이팟(iPod)™과 같은 MP3 재생기의 기



능을 포함할 수 있다. 그러므로, 컴퓨팅 디바이스(1400)는 아이팟과 호환되는 36-핀 커넥터를 포함할 수 있다. 기타 입력/출력 및 제어 디바이스가 또한 사용될 수 있다.

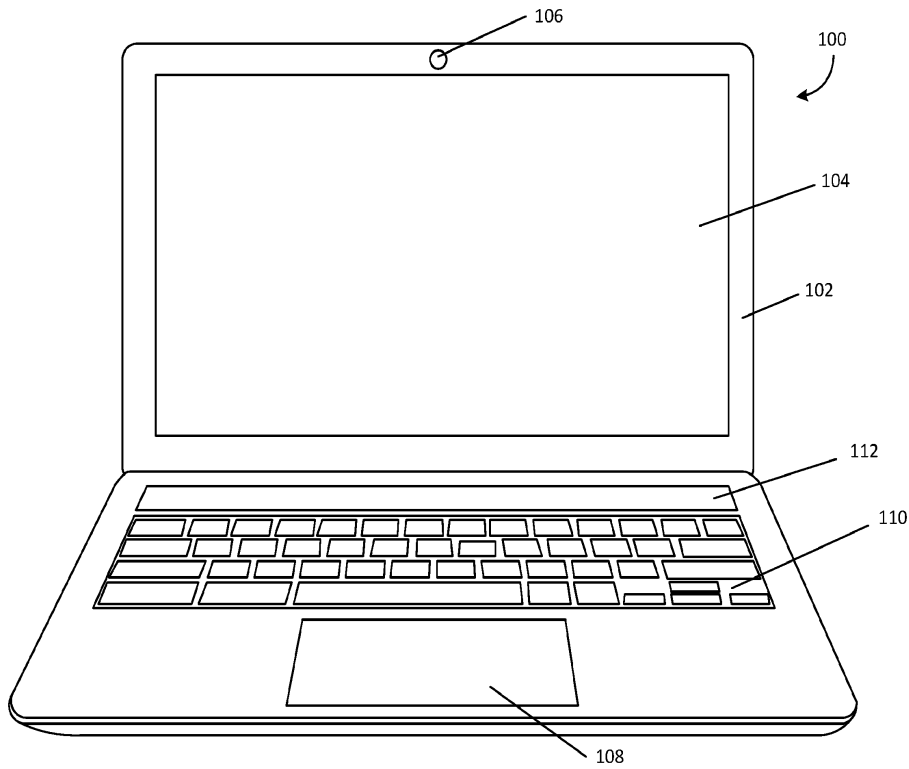
- [0147] 메모리 인터페이스(1402)는 메모리(1450)에 결합될 수 있다. 메모리(1450)는 고속 랜덤 액세스 메모리 및/또는 비휘발성 메모리, 예컨대 하나 이상의 자기 디스크 저장 디바이스, 하나 이상의 광학 저장 디바이스, 및/또는 플래시 메모리(예를 들어, NAND, NOR)를 포함할 수 있다. 메모리(1450)는 운영 체제(1452), 예컨대 다윈(Darwin), RTXC, 리눅스(LINUX), 유닉스(UNIX), OS X, 윈도우즈(WINDOWS), 또는 임베디드 운영 체제, 예컨대 VxWorks를 저장할 수 있다.
- [0148] 운영 체제(1452)는 기본 시스템 서비스들을 취급하고 하드웨어 의존형 태스크들을 수행하기 위한 명령어들을 포함할 수 있다. 일부 구현예들에서, 운영 체제(1452)는 커널(예를 들어, 유닉스 커널)일 수 있다. 일부 구현예들에서, 운영 체제(1452)는 음성 인증을 수행하기 위한 명령어들을 포함할 수 있다. 예를 들어, 운영 체제(1452)는 도 1 내지 도 13을 참조하여 설명한 바와 같이 DFR 특징부들을 구현할 수 있다.
- [0149] 메모리(1450)는 또한 하나 이상의 추가 디바이스, 하나 이상의 컴퓨터 및/또는 하나 이상의 서버와의 통신을 용이하게 하는 통신 명령어들(1454)을 저장할 수 있다. 메모리(1450)는 그래픽 사용자 인터페이스 처리를 용이하게 하는 그래픽 사용자 인터페이스 명령어들(1456); 센서 관련 처리 및 기능들을 용이하게 하는 센서 처리 명령어들(1458); 전화 관련 프로세스들 및 기능들을 용이하게 하는 전화 명령어들(1460); 전자 메시징 관련 프로세스들 및 기능들을 용이하게 하는 전자 메시징 명령어들(1462); 웹 브라우징 관련 프로세스들 및 기능들을 용이하게 하는 웹 브라우징 명령어들(1464); 미디어 처리 관련 프로세스들 및 기능들을 용이하게 하는 미디어 처리 명령어들(1466); GNSS 및 내비게이션 관련 프로세스들 및 명령어들을 용이하게 하는 GNSS/내비게이션 명령어들(1468); 및/또는 카메라 관련 프로세스들 및 기능들을 용이하게 하는 카메라 명령어들(1470)을 포함할 수 있다.
- [0150] 메모리(1450)는 도 1 내지 도 13을 참조하여 설명한 바와 같이 DFR 프로세스들 및 기능들과 같은 기타 프로세스들 및 기능들을 용이하게 하는 DFR 관리 명령어들(1472)을 저장할 수 있다.
- [0151] 메모리(1450)는 또한 웹 비디오 관련 프로세스들 및 기능들을 용이하게 하는 웹 비디오 명령어들; 및/또는 웹 쇼핑 관련 프로세스들 및 기능들을 용이하게 하는 웹 쇼핑 명령어들과 같은 기타 소프트웨어 명령어들(1474)을 저장할 수 있다. 일부 구현예들에서, 미디어 처리 명령어들(1466)은 오디오 처리 관련 프로세스들 및 기능들 및 비디오 처리 관련 프로세스들 및 기능들을 각각 용이하게 하는 오디오 처리 명령어들 및 비디오 처리 명령어들로 나뉜다.
- [0152] 위에 확인된 명령어들 및 애플리케이션들 각각은 상기 설명한 하나 이상의 기능을 수행하기 위한 명령어들의 세트에 대응할 수 있다. 이들 명령어는 별개의 소프트웨어 프로그램들, 절차들, 또는 모듈들로서 구현될 필요는 없다. 메모리(1450)는 추가적인 명령어 또는 더 적은 수의 명령어를 포함할 수 있다. 또한, 컴퓨팅 디바이스(1400)의 다양한 기능들은, 하나 이상의 신호 처리 및/또는 응용 주문형 집적 회로를 포함하는 하드웨어 및/또는 소프트웨어로 구현될 수 있다.
- [0153] 기술된 특징들은 데이터 저장 시스템, 적어도 하나의 입력 디바이스, 및 적어도 하나의 출력 디바이스로부터 데이터 및 명령어들을 수신하도록 그리고 그들에 데이터 및 명령어들을 송신하도록 결합된 적어도 하나의 프로그래밍가능 프로세서를 포함하는 프로그래밍가능 시스템 상에서 실행가능할 수 있는 하나 이상의 컴퓨터 프로그램으로 구현될 수 있다. 컴퓨터 프로그램은 소정의 활동을 수행하거나 소정의 결과를 초래하기 위해 컴퓨터에 직접 또는 간접적으로 사용될 수 있는 명령어들의 세트이다. 컴퓨터 프로그램은 컴파일러형 또는 해석형 언어를 비롯한 임의의 형태의 프로그래밍 언어(예컨대, 오브젝티브-C(Objective-C), 자바(Java))로 기록될 수 있으며, 그것은 독립형 프로그램으로서 또는 모듈, 컴포넌트, 서브루틴, 또는 컴퓨팅 환경에서 사용하기에 적합한 다른 유닛으로서를 비롯해 임의의 형태로 배치될 수 있다.
- [0154] 명령어들의 프로그램의 실행에 적합한 프로세서들은, 예로서, 임의의 종류의 컴퓨터의, 범용 마이크로프로세서와 특수 목적 마이크로프로세서 둘 모두, 및 단일 프로세서 또는 다수의 프로세서들 또는 코어들 중 하나를 포함할 수 있다. 일반적으로, 프로세서는 판독 전용 메모리 또는 랜덤 액세스 메모리 또는 둘 모두로부터 명령어들과 데이터를 수신할 수 있다. 컴퓨터의 본질적인 요소들은 명령어들을 실행하기 위한 프로세서, 및 명령어들과 데이터를 저장하기 위한 하나 이상의 메모리를 포함할 수 있다. 일반적으로, 컴퓨터는 또한 데이터 파일을 저장하기 위한 하나 이상의 대용량 저장 디바이스를 포함하거나 그와 통신하도록 동작가능하게 결합될 수 있으며; 그러한 디바이스는 자기 디스크, 예를 들어 내부 하드 디스크 및 착탈식 디스크; 자기-광학 디스크; 및 광 디스크를 포함할 수 있다. 컴퓨터 프로그램 명령어들과 데이터를 유형적으로 구체화하기에 적합한 저장 디

바이스는 모든 형태의 비휘발성 메모리를 포함할 수 있으며, 예로써, 반도체 메모리 디바이스, 예컨대 EPROM, EEPROM, 및 플래시 메모리 디바이스; 자기 디스크, 예컨대 내부 하드 디스크 및 착탈식 디스크; 자기-광학 디스크; 및 CD-ROM 및 DVD-ROM 디스크를 포함한다. 프로세서 및 메모리는 ASIC들에 의해 보완되거나 그들에 포함될 수 있다.

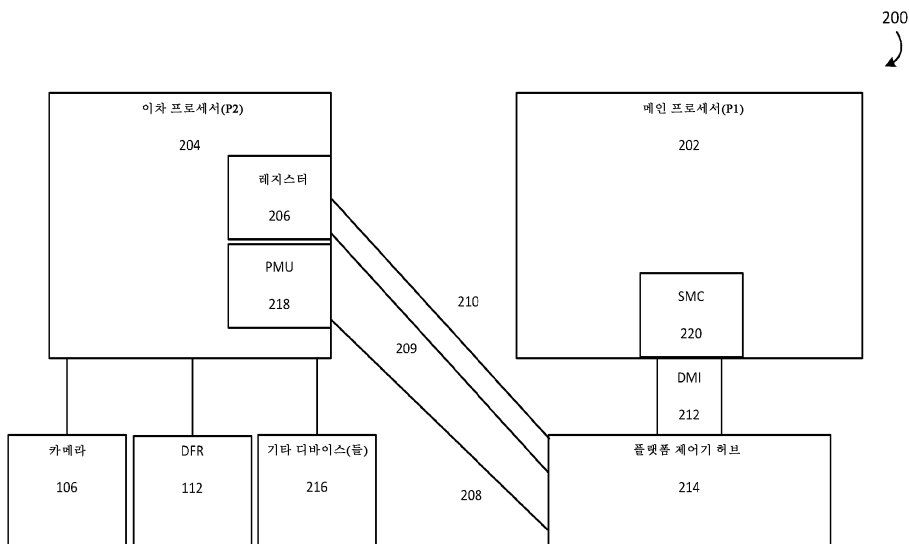
- [0155] 사용자와의 상호작용을 제공하기 위해, 특징들은 사용자에게 정보를 디스플레이하기 위한 CRT(음극선관) 또는 LCD(액정 디스플레이) 모니터와 같은 디스플레이 디바이스, 및 사용자가 컴퓨터에 입력을 제공할 수 있게 하는 마우스 또는 트랙볼과 같은 포인팅 디바이스 및 키보드를 갖는 컴퓨터 상에서 구현될 수 있다.
- [0156] 특징들은 데이터 서버와 같은 백-엔드 컴포넌트를 포함하는 컴퓨터 시스템, 또는 애플리케이션 서버 또는 인터넷 서버와 같은 미들웨어 컴포넌트를 포함하는 컴퓨터 시스템, 또는 그래픽 사용자 인터페이스 또는 인터넷 브라우저, 또는 이들의 임의의 조합을 갖는 클라이언트 컴퓨터와 같은 프론트-엔드 컴포넌트를 포함하는 컴퓨터 시스템에서 구현될 수 있다. 시스템의 컴포넌트들은 통신 네트워크와 같은, 임의의 형태 또는 매체의 디지털 데이터 통신에 의해 연결될 수 있다. 통신 네트워크의 예들은, 예컨대 LAN, WAN, 및 인터넷을 이루는 컴퓨터들과 네트워크들을 포함한다.
- [0157] 컴퓨터 시스템은 클라이언트들 및 서버들을 포함할 수 있다. 클라이언트와 서버는 대체로 서로 원격일 수 있으며, 전형적으로 네트워크를 통해 상호작용할 수 있다. 클라이언트와 서버의 관계는 각자의 컴퓨터 상에서 실행되고 서로 클라이언트-서버 관계를 갖는 컴퓨터 프로그램들에 의해 발생할 수 있다.
- [0158] 개시된 실시예들의 하나 이상의 특징들 또는 단계들은 API를 사용하여 구현될 수 있다. API는 호출하는 애플리케이션과, 서비스를 제공하거나 데이터를 제공하거나 동작 또는 계산을 수행하는 다른 소프트웨어 코드(예를 들어, 운영 체제, 라이브러리 루틴, 함수) 사이에 전달되는 하나 이상의 파라미터들을 정의할 수 있다.
- [0159] API는 API 사양서에 정의된 호출 규약에 기초하여 파라미터 목록 또는 다른 구조를 통해 하나 이상의 파라미터들을 송신 또는 수신하는 프로그램 코드 내의 하나 이상의 호출들로서 구현될 수 있다. 파라미터는 상수, 키, 데이터 구조, 객체, 객체 클래스, 변수, 데이터 타입, 포인터, 배열, 목록, 또는 다른 호출일 수 있다. API 호출들 및 파라미터들은 임의의 프로그래밍 언어로 구현될 수 있다. 프로그래밍 언어는 프로그래머가 API를 지원하는 기능들에 액세스하기 위해 이용할 어휘 및 호출 규약을 정의할 수 있다.
- [0160] 일부 구현예들에서, API 호출은 애플리케이션을 실행하는 디바이스의 능력들, 예컨대 입력 능력, 출력 능력, 처리 능력, 전력 능력, 통신 능력 등을 애플리케이션에게 보고할 수 있다.
- [0161] 다양한 실시예들이 전술되었지만, 이들은 예로써 제시되었을 뿐, 제한하는 것은 아니라는 점을 이해해야 한다. 형태 및 상세 사항의 다양한 변화가 사상 및 범주로부터 벗어남이 없이 그 내에서 이루어질 수 있음이 통상의 기술자에게 명백할 것이다. 실제로, 상기 설명을 읽은 후에, 대안적인 실시예들을 구현하는 방법이 통상의 기술자에게 명백할 것이다.
- [0162] 또한, 기능 및 이점을 강조하는 임의의 도면들이 예시 목적으로만 제시된다는 것을 이해하여야 한다. 개시된 방법 및 시스템은 각각 충분히 유동적이고 구성가능하여, 도시된 것과 다른 방식으로 이용될 수 있다.
- [0163] 용어 "적어도 하나" 가 종종 명세서, 청구범위 및 도면에 사용될 수 있지만, 용어 "a", "an", "the", "상기" 등은 또한 명세서, 청구범위 및 도면에서 "적어도 하나" 또는 "상기 적어도 하나"를 의미한다.
- [0164] 마지막으로, 명시적인 언어 "~ 위한 수단" 또는 "~ 위한 단계"를 포함하는 청구항만이 35 U.S.C. 112(f) 하에서 해석되어야 한다는 것이 출원인의 의도이다. 어구 "~ 위한 수단" 또는 "~ 위한 단계"를 명확하게 포함하지 않는 청구범위는 35 U.S.C. 112(f) 하에서 해석되지 않는다.

도면

도면1

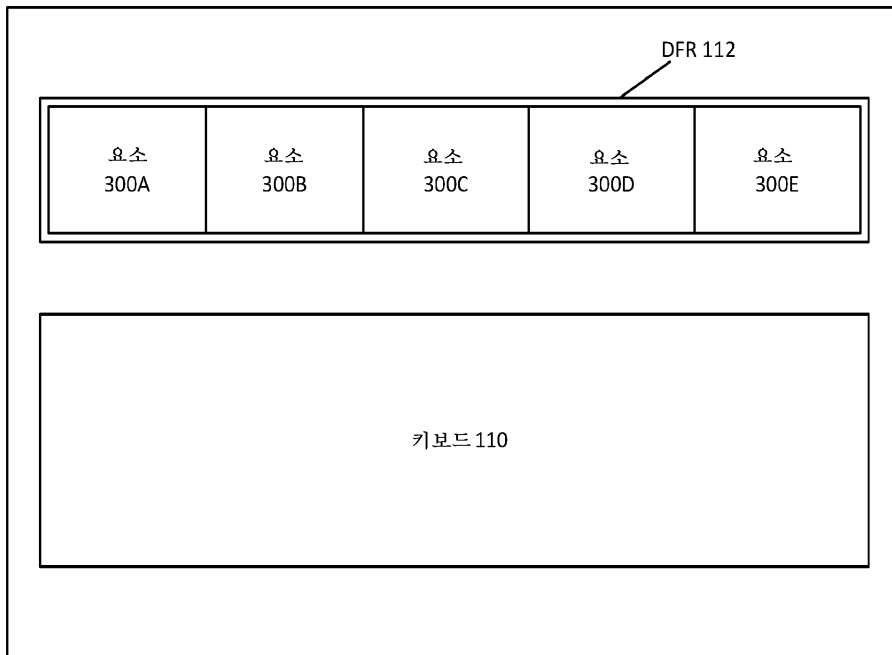


도면2

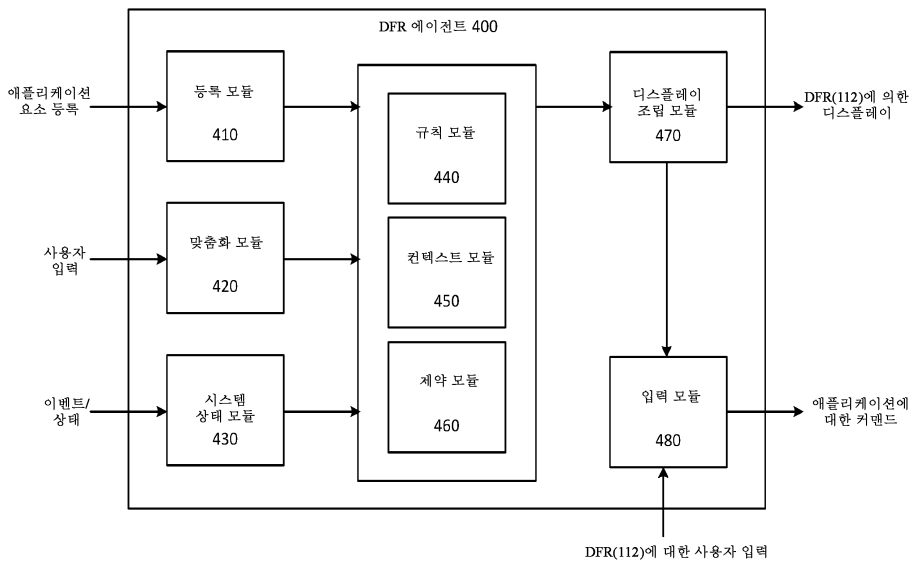




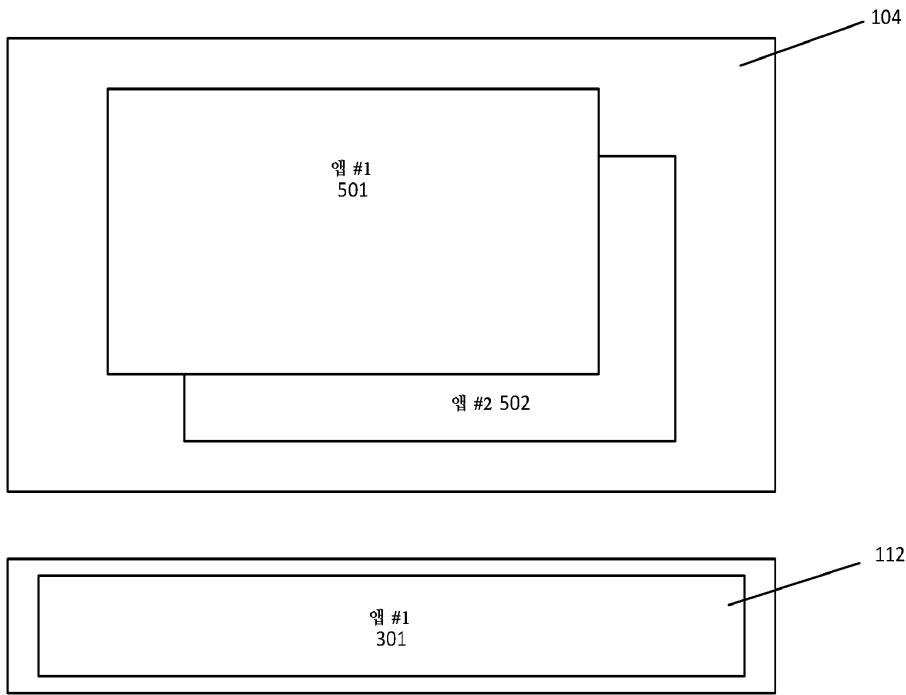
도면3



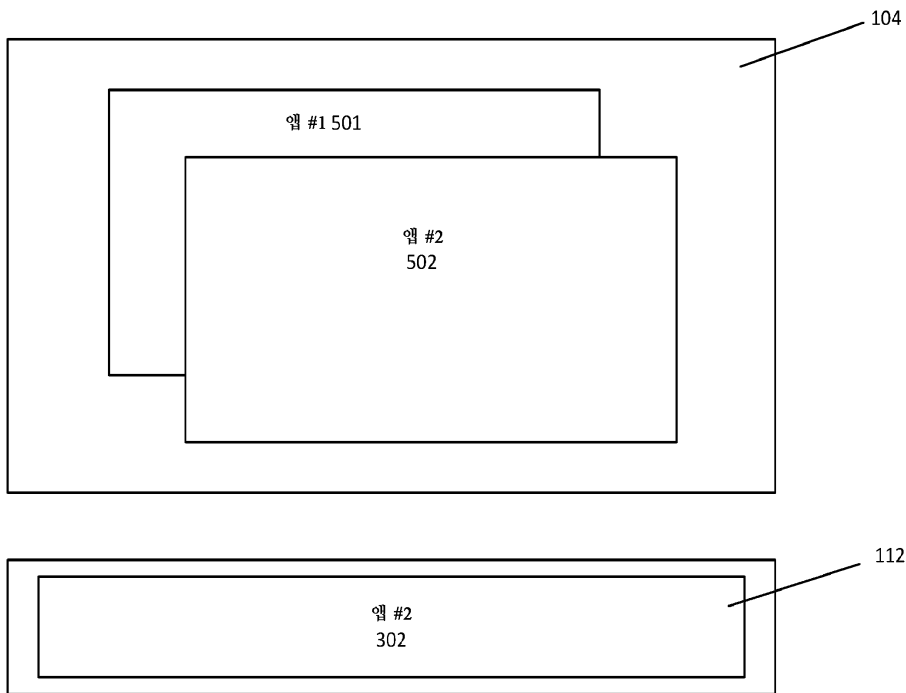
도면4



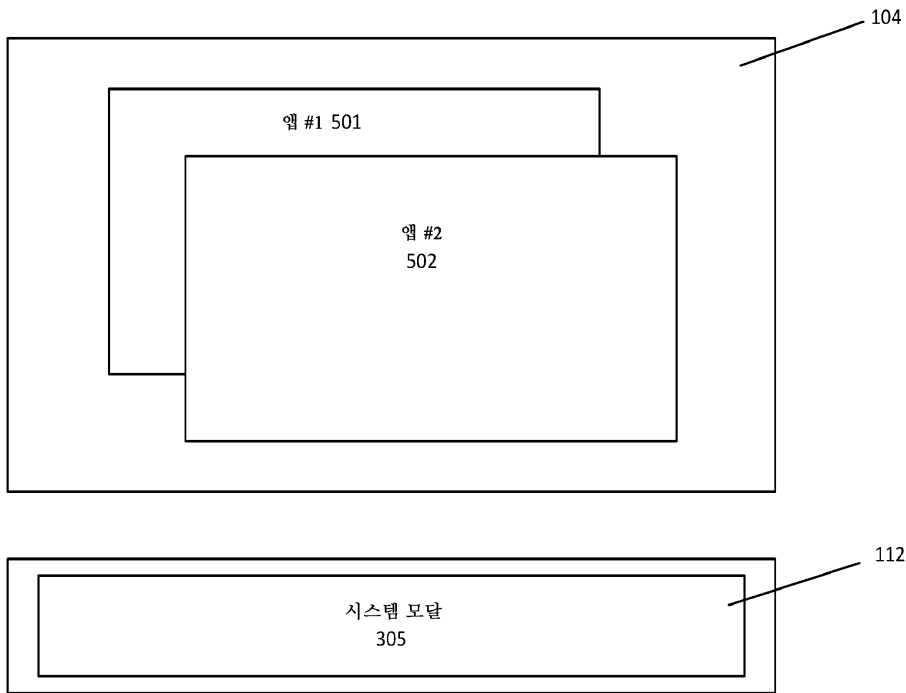
도면5a



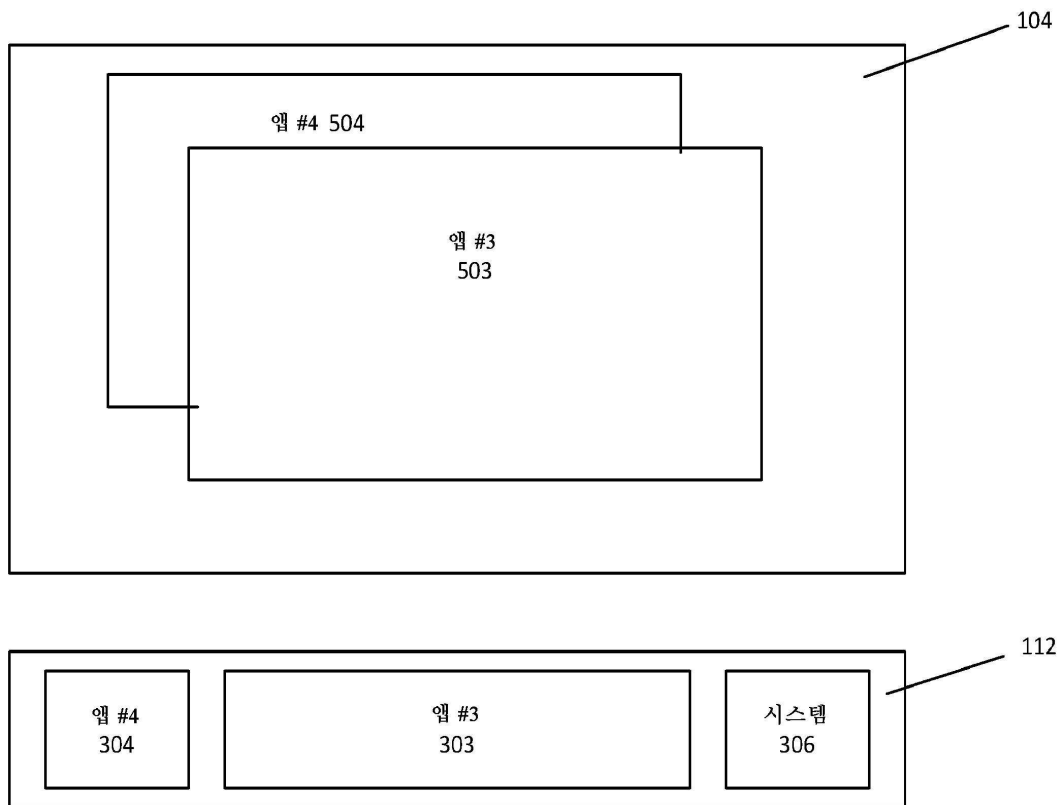
도면5b



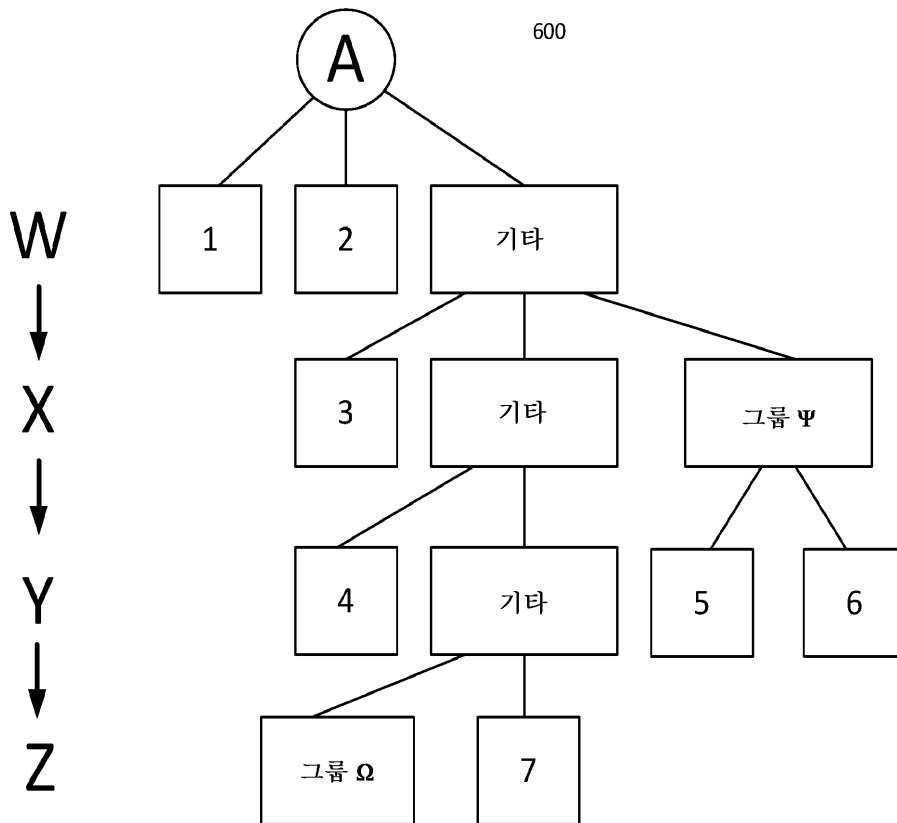
도면5c



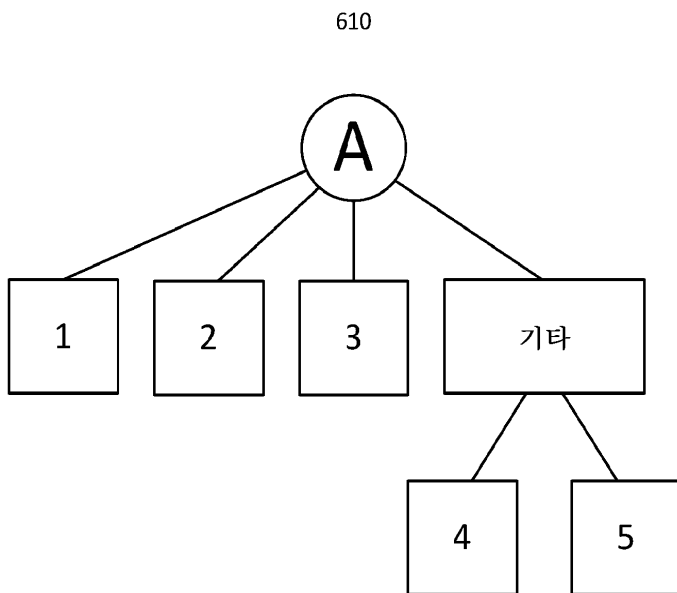
도면5d



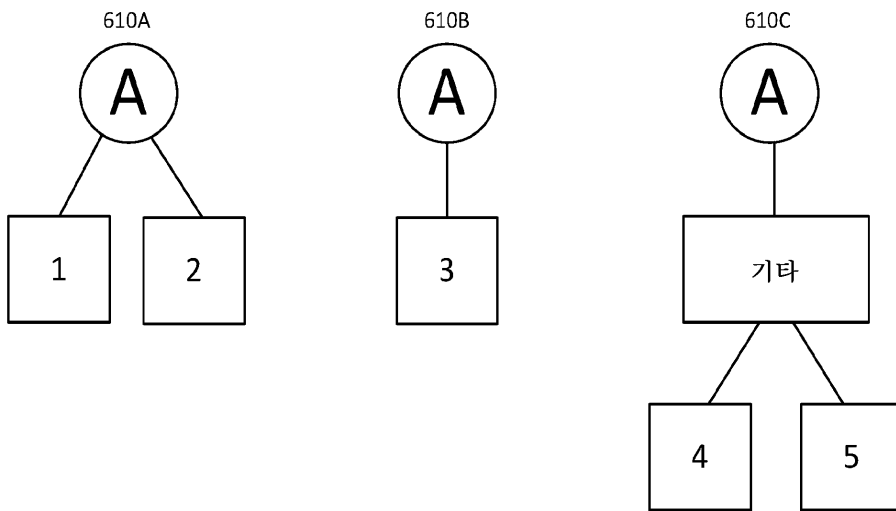
도면6a



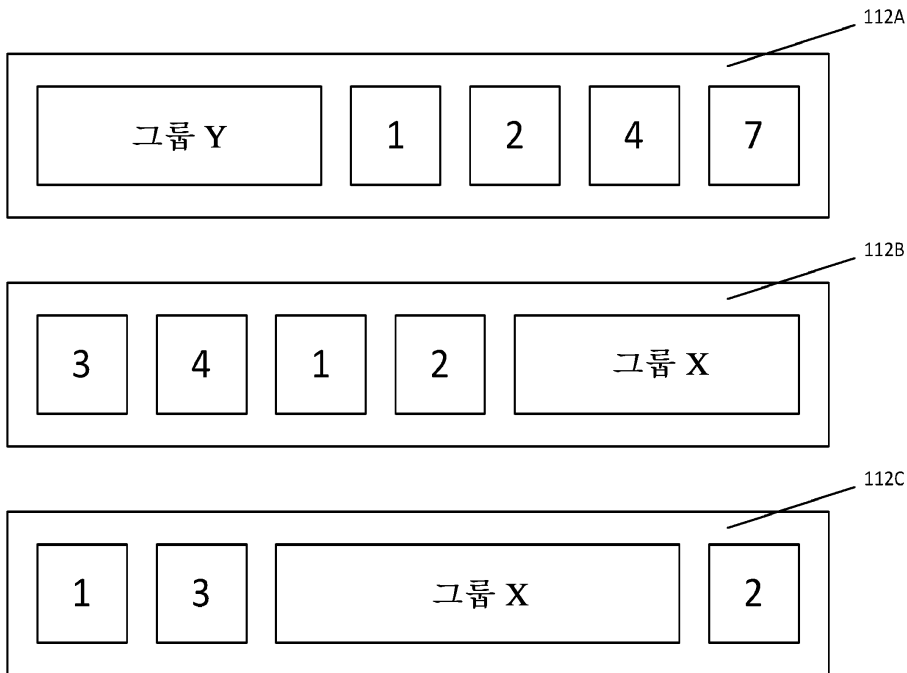
도면6b



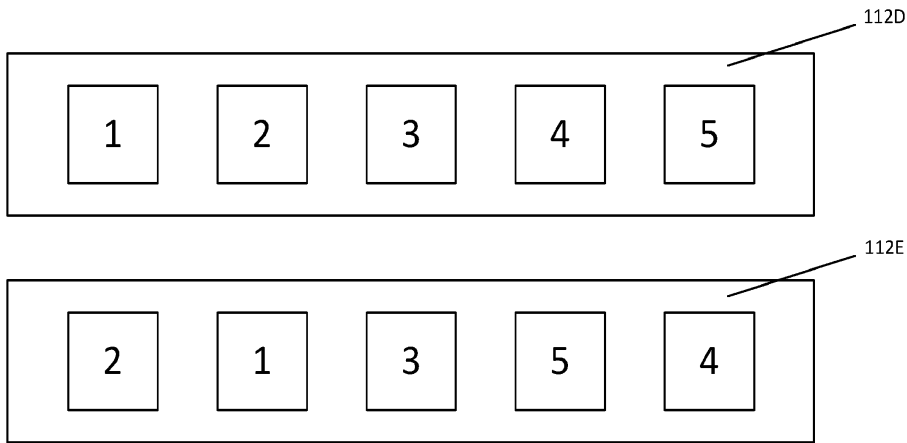
도면6c



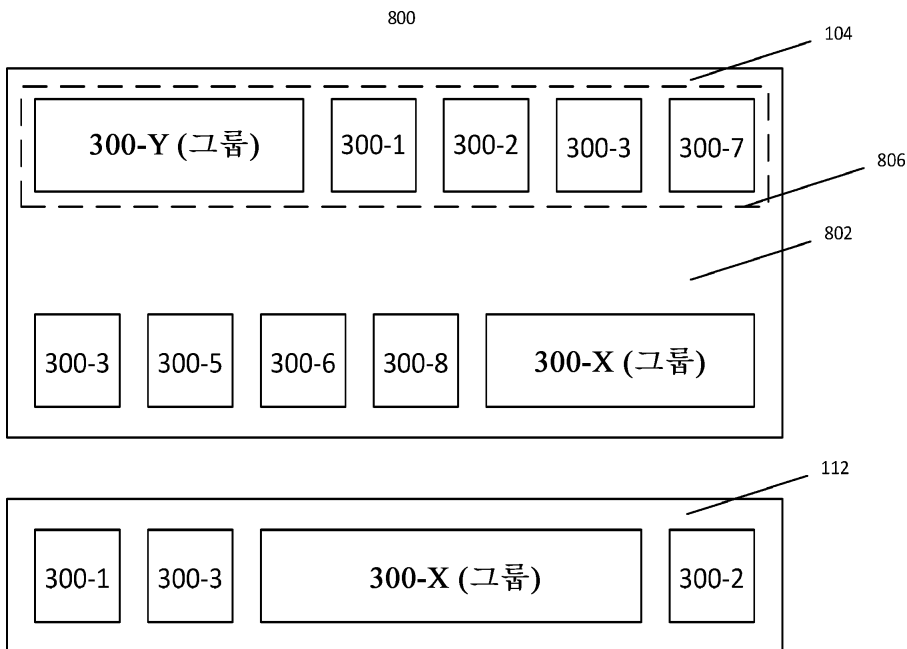
도면7a



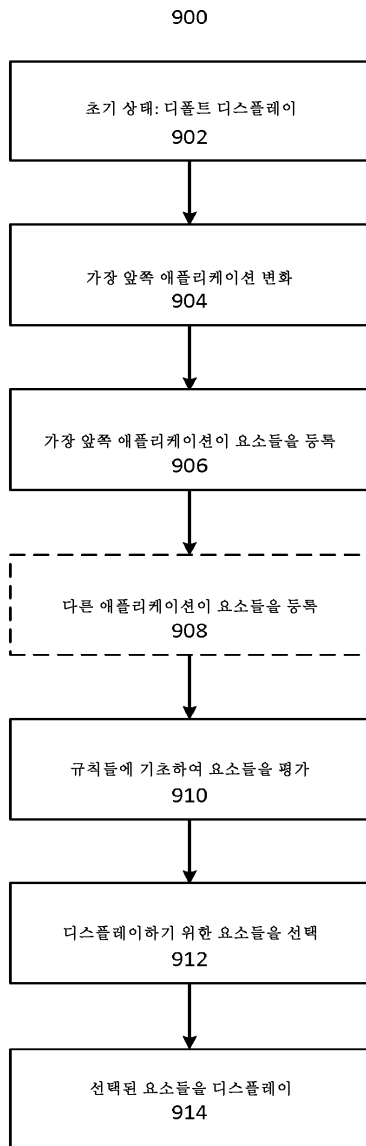
도면7b



도면8

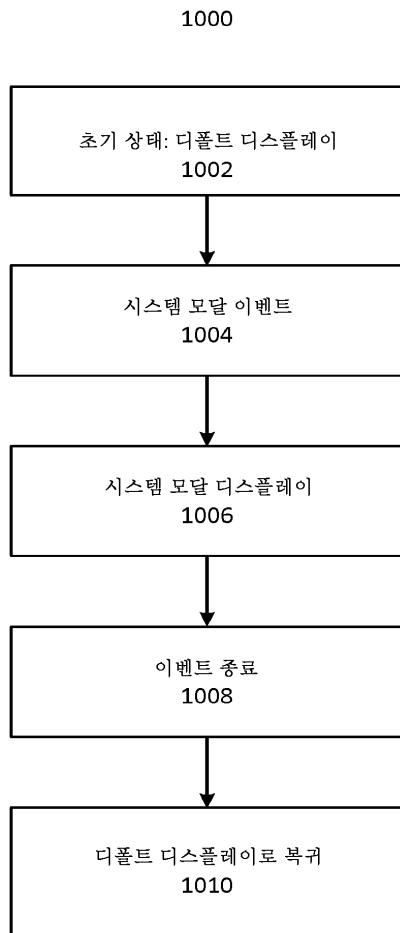


도면9

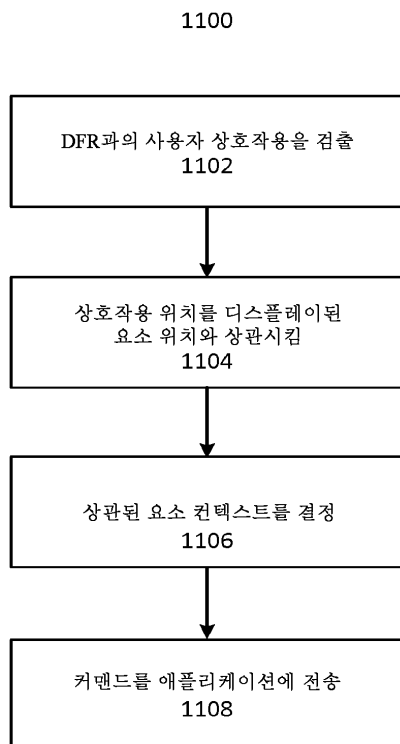




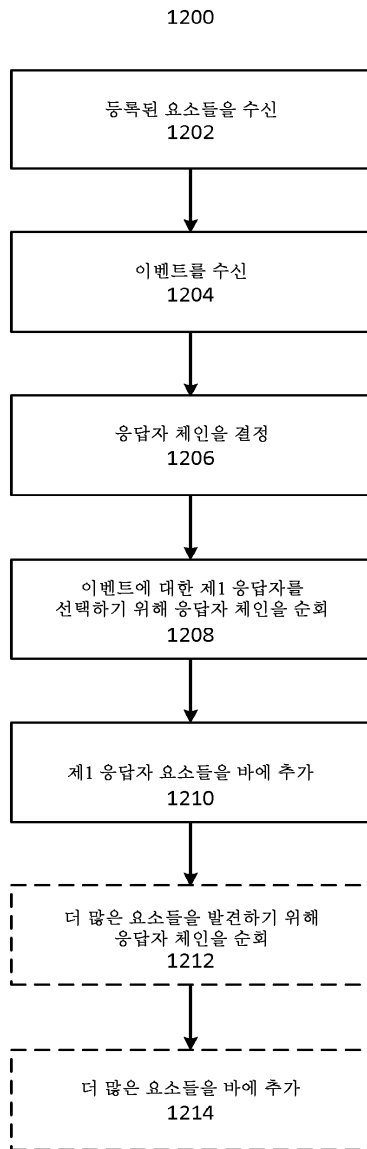
도면10



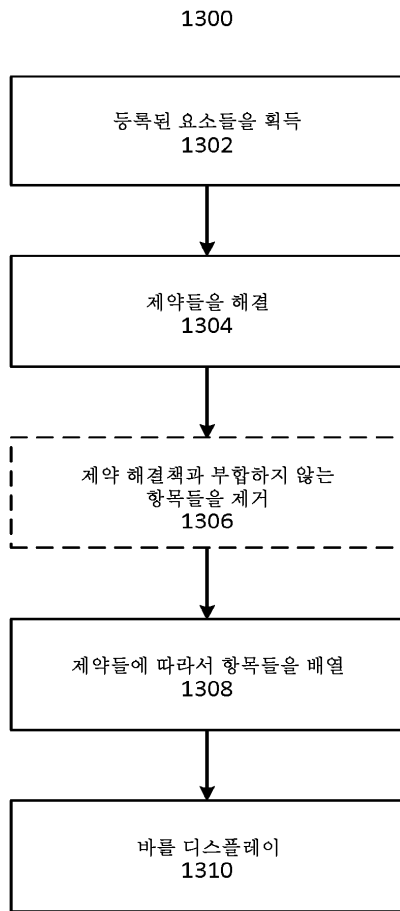
도면11



도면12



도면13



도면14

