



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2015-0073110  
(43) 공개일자 2015년06월30일

(51) 국제특허분류(Int. C1.)  
*G06Q 10/06* (2012.01) *G06Q 10/10* (2012.01)  
(21) 출원번호 10-2014-0183557  
(22) 출원일자 2014년12월18일  
심사청구일자 없음  
(30) 우선권주장  
14/137,473 2013년12월20일 미국(US)

(71) 출원인  
다쏘 시스템즈 아메리카스 코포레이션  
미국 매사추세츠 02451, 월섬, 와이만 스트리트  
175  
(72) 발명자  
메이어 폴 알  
미국 55119 미네소타주 세인트 폴 보니 레인 2196  
그라밧 커티스 에이  
미국 54665 위스콘신주 비로파 사우스 워싱턴 애  
비뉴 636  
콘란드 스티븐  
미국 55122 미네소타주 이건 플라자 드라이브  
1915 스위트 100  
(74) 대리인  
특허법인코리아나

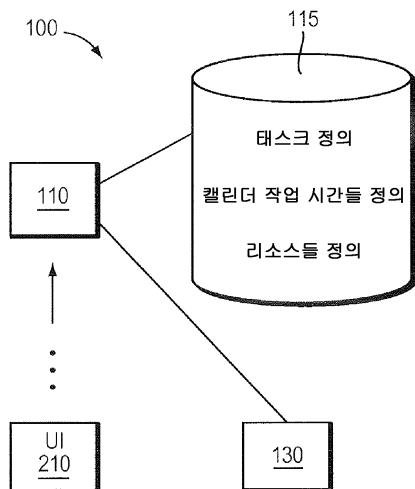
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 자동화된 스케줄링을 위한 컴퓨터 방법 및 장치

### (57) 요약

전자 캘린더를 위한 컴퓨터 기반의 스케줄링 및 재스케줄링 방법, 장치 및 시스템. 전자 캘린더는 스케줄링 된 태스크들의 시간 진행을 시간의 수평으로 또는 수직으로 지향된 뷰로 예시한다. 전자 캘린더 및 그의 그 래피 사용자 인터페이스 (GUI) 는 사용자를 위한 뷰잉 패턴을 제공한다. 뷰잉 패턴에서의 방향은 스케줄링된 태스크들의 시간 진행에 있어서 일자들-시간들을 증가시키는 것/ 감소시키는것과 아주 밀접하다. 스케줄링-재스케줄링 엔진은 캘린더로 사용자 요청된 태스크들을 스케줄링하고 영향을 받은 이전에 스케줄링된 태스크들을 자동적으로 재스케줄링한다. 스케줄링-재스케줄링 엔진은 또한 사용자 선택된 리소스들 (예컨대, 사람, 로봇들, 툴들, 등) 을 스케줄링된 태스크 각각에 대하여 스케줄링한다.

### 대표도 - 도1



## 명세서

### 청구범위

#### 청구항 1

디지털 프로세서를 이용하는 단계;

스케줄링된 태스크들을 시간 진행 및 각각의 태스크 시작 일자/시간들의 시간 순서로 디스플레이하는 전자 캘린더를 제공하는 단계; 및

상기 디지털 프로세서에 의해 디스플레이된 상기 전자 캘린더와의 사용자 상호작용에 응답하여, 상기 캘린더에서 사용자 요청되고 미스케줄링된 태스크를 자동적으로 스케줄링하고 임의의 영향받은 이전에 스케줄링된 태스크들을 재스케줄링하여, 상기 캘린더가 상기 사용자 요청되고 미스케줄링된 태스크 및 상기 이전에 스케줄링된 태스크들을 포함하는 태스크들의 논리적으로 올바른 스케줄을 유지하도록 하는 단계를 포함하는, 스케줄링 방법.

#### 청구항 2

제 1 항에 있어서,

상기 전자 캘린더는 시간 진행의 일자 및 시간을 증가시키는 것이 상기 전자 캘린더의 뷰잉 패턴의 일자 및 시간을 증가시키는 것과 동일한 방향으로 보여지도록 각각의 시작 일자/시간들의 시간 순서로 스케줄링된 태스크들을 디스플레이하는, 스케줄링 방법.

#### 청구항 3

제 2 항에 있어서,

상기 일자 및 시간을 증가시키는 방향은 좌측에서 우측으로인, 스케줄링 방법.

#### 청구항 4

제 1 항에 있어서,

스케줄링된 태스크들 및 연관되는 일자들 또는 시간들을 현재 디스플레이된 일자들 및 시간들에 대해 상대적으로 미래로 디스플레이하고, 반대의 경우도 마찬가지로 스케줄링된 태스크들 및 연관되는 일자들 및 시간들을 시간의 역방향으로 디스플레이하기 위하여, 상기 캘린더의 뷰들을 디스플레이 스크린의 하나의 사이드에서 반대 사이드로 진행시킴으로써 수평으로 지향된 뷰잉 패턴을 구현하는 상기 전자 캘린더에 그래픽 사용자 인터페이스를 제공하는 단계를 더 포함하는, 스케줄링 방법.

#### 청구항 5

제 4 항에 있어서,

상기 그래픽 사용자 인터페이스는 드래그 앤드 드롭 기술을 이용하는, 스케줄링 방법.

#### 청구항 6

제 1 항에 있어서,

상기 캘린더에 디스플레이된 사용자 선택된 스케줄링된 태스크와의 사용자 상호작용에 응답하여, 상기 캘린더에서 상기 사용자 선택된 태스크를 사용자 요청된 일자 및 시간으로 자동적으로 재스케줄링하고, 임의의 영향받은 다른 스케줄링된 태스크들을 자동적으로 재스케줄링하는 단계를 더 포함하는, 스케줄링 방법.

#### 청구항 7

제 6 항에 있어서,

상기 임의의 영향받은 다른 스케줄링된 태스크들의 자동적 재스케줄링은 선형 재스케줄링을 이용하는, 스케줄링

방법.

### 청구항 8

제 1 항에 있어서,

상기 임의의 영향받은 이전에 스케줄링된 태스크들의 재스케줄링은 각각의 스케줄링된 태스크의 지속기간 및 태스크들에 대한 유효한 작업 일자-시간들의 이용가능성에 기초한 선형 재스케줄링인, 스케줄링 방법.

### 청구항 9

제 1 항에 있어서,

상기 스케줄링된 태스크들은 상점 주문 동작들 및 제조 생산 중 적어도 하나의 작업 태스크들인, 스케줄링 방법.

### 청구항 10

제 1 항에 있어서,

상기 전자 캘린더와의 사용자 상호작용에 응답하여, 사용자 선택된 리소스를 상기 캘린더에서 사용자 선택된 스케줄링된 태스크로 스케줄링하는 단계를 더 포함하는, 스케줄링 방법.

### 청구항 11

시간 갑각과 밀접한 뷰잉 패턴을 가지도록 지향된 시간 진행으로 스케줄링된 태스크들을 디스플레이하는 적어도 하나의 전자 캘린더를 제공하도록 구성된 디지털 프로세서로서, 상기 전자 캘린더는 스케줄링된 태스크들을 각각의 시작 일자/시간들의 시간 순서로 디스플레이하는, 상기 디지털 프로세서; 및

상기 디지털 프로세서에 의해 실행되는 스케줄링 엔진으로서, 상기 스케줄링 엔진은, 상기 디지털 프로세서에 의해 디스플레이된 상기 캘린더와의 사용자 상호작용에 응답하여, 상기 캘린더에서 사용자 요청되고 미스케줄링된 태스크를 자동적으로 스케줄링하고 임의의 영향받은 이전에 스케줄링된 태스크들을 재스케줄링하여, 상기 캘린더가 상기 사용자 요청되고 미스케줄링된 태스크 및 상기 이전에 스케줄링된 태스크들을 포함하는 태스크들의 논리적으로 올바른 스케줄을 유지하도록 구성된, 상기 스케줄링 엔진; 및

상기 디지털 프로세서에 통신가능하게 커플링되며 상기 전자 캘린더의 뷰들을 디스플레이하는 디스플레이 모니터를 포함하는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 12

제 11 항에 있어서,

상기 시간 진행의 일자 및 시간을 증가시키는 것은 상기 뷰잉 패턴의 일자 및 시간을 증가시키는 것과 동일한 방향으로 보여지는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 13

제 12 항에 있어서,

상기 일자 및 시간을 증가시키는 방향은 좌측에서 우측으로인, 컴퓨터 기반의 스케줄링 장치.

### 청구항 14

제 11 항에 있어서,

상기 전자 캘린더에 커플링되고, 스케줄링된 태스크들을 현재 디스플레이된 일자들 및 시간들에 대해 상대적으로 미래의 일자들 및 시간들에 디스플레이하고, 반대의 경우도 마찬가지로 스케줄링된 태스크들을 시간의 역방향으로 일자들 및 시간들에 디스플레이하기 위하여, 상기 캘린더의 뷰들을 상기 디스플레이 모니터의 하나의 사이드에서 반대 사이드로 진행시킴으로써 수평으로 지향된 뷰잉 패턴을 구현하는 그래픽 사용자 인터페이스를 더 포함하는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 15

제 14 항에 있어서,

상기 그래픽 사용자 인터페이스는 드래그 앤드 드롭 기법들을 이용하는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 16

제 11 항에 있어서,

상기 스케줄링 엔진은, 상기 캘린더에 디스플레이된 사용자 선택된 스케줄링된 태스크와의 사용자 상호작용에 응답하여, 상기 캘린더에서 상기 사용자 선택된 태스크를 사용자 요청된 일자 및 시간으로 자동적으로 재스케줄링하고, 임의의 영향받은 다른 스케줄링된 태스크들을 자동적으로 재스케줄링하도록 더 구성되는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 17

제 11 항에 있어서,

상기 스케줄링 엔진은 선형 재스케줄링을 이용하는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 18

제 11 항에 있어서,

상기 스케줄링된 태스크들은 상점 주문 동작들, 및 제조 생산 중 적어도 하나의 작업 태스크들인, 컴퓨터 기반의 스케줄링 장치.

### 청구항 19

제 11 항에 있어서,

상기 스케줄링 엔진은, 상기 전자 캘린더와의 사용자 상호작용에 응답하여, 사용자 선택된 리소스를 상기 캘린더에서 사용자 선택된 스케줄링된 태스크로 스케줄링하도록 구성되는, 컴퓨터 기반의 스케줄링 장치.

### 청구항 20

컴퓨터 프로그램 제품으로서,

하나 이상의 전자 캘린더들을 제공하고 상기 캘린더들 중 임의의 캘린더의 사용자 조작들을 지원하는 비일시적 컴퓨터 관독가능 매체; 및

상기 컴퓨터 관독가능 매체로 구현된, 컴퓨터 프로그램 코드를 포함하고,

상기 컴퓨터 프로그램 코드는, 프로세서에 의해 실행될 때, 주어진 캘린더에 대해,

상기 전자 캘린더에서의 스케줄링된 태스크들을 디스플레이하고;

상기 전자 캘린더와의 사용자 상호작용에 응답하여, 사용자 요청된 태스크를 자동적으로 스케줄링하고 상기 전자 캘린더에서 임의의 영향받은 이전에 스케줄링된 태스크들을 재스케줄링하여, 상기 전자 캘린더가 사용자 선택된 태스크 및 상기 이전에 스케줄링된 태스크들을 포함하는 태스크들의 논리적으로 올바른 스케줄을 유지하는 것을 제공하며,

상기 디스플레이는 수평으로 지향된 뷰잉 패턴을 따른 시간 진행에 있는, 컴퓨터 프로그램 제품.

## 발명의 설명

### 배경기술

[0001] 현재, 항공우주 산업 제조업자들은 종이와 짧은 수기 노트들 (예컨대, 스티커식 노트들)의 조합을 이용하여 태스크들을 보고, 스케줄링하고, 관리하도록 강요받는다. 이 스케줄 디스플레이의 방법은 바 차트로 알려져 있다. 일련의 태스크들이 수평 "바" 상에 펼쳐지며 사람의 이름들이 바 또는 개개의 태스크들로 할당된다.

[0002] 작업 현장 (shop floor) 태스크들을 보고, 스케줄링하고 관리하는 이 종이-기반의 방법은 다음의 한계들을 가진다.

- [0003] 외부 시스템이 태스크 스케줄들의 변화들 및 리소스들의 할당의 변화들을 개시하기 위하여 사용될 수도 있다. 종이-기반의 방법에서, 이들 변화들은 태스크 관리 정보가 수동으로 업데이트되기 전까지는 반영되지 않는다.
- [0004] 종이-기반의 방법에서, 데이터 디스플레이의 수동 구성은 에러에 취약하다. 에러는 수동 표기에서 있을 수도 있고/있거나 에러는 디스플레이된 데이터가 종종 짧고 축약된 리더에 의한 해석에서 있을 수도 있다.
- [0005] 종이-기반의 방법에서, 데이터 디스플레이의 수동 구성은 시간 소모적이며 비효율적이다.
- [0006] 종이-기반의 방법에서, 임계 데이터는 시기적절한 방법으로 디스플레이되지 않으며 임계 데이터는 단일 로케이션에서만 이용가능하다.
- [0007] 종이-기반의 방법에서 데이터 디스플레이의 수동 구성은 과거에서의 일자/시간 또는 상점 스케줄 때문에 이용할 수 없는 시간에 대한 태스크에 대한 스케줄링과 같은 논리적 에러들을 방지하지 못한다. 유사한 방법으로, 이용불가능한 리소스들이 태스크에 할당될 수도 있다.
- [0008] 더욱이, 종이-기반의 방법에서, 새로운 태스크를 또 다른 태스크에 의해 현재 사용되는 시간 슬롯에 삽입하는 것은 영향을 받은 태스크들을 자동적으로 재스케줄링하지 않는다. 수동 재스케줄링은 (위와 유사하게) 논리적 에러들에 취약하며, (앞에서 언급한 바와 같이) 시간 소모적이고 비효율적이다.

## 발명의 내용

### 해결하려는 과제

[0009] 본 발명은 그 기술 분야에서 전술한 문제들을 해결한다.

### 과제의 해결 수단

- [0010] 본 발명에서의 실시형태들은 다음의 이점들 및 특징들을 제공한다.
- [0011] 스케줄링된 태스크들은, 태스크들이 생성될 때 또는 스케줄이 외부 시스템에 의해 업데이트될 때, 수평 위치에 자동적으로 컴퓨터 디스플레이된다.
- [0012] 태스크들은 데스크탑 또는 태블릿 디바이스들을 통하여 임의의 영역에 보여지거나 관리된다.
- [0013] 태스크를 원하는 디스플레이되는 시간 슬롯으로 GUI 드래그함으로써 태스크들은 용이하게 재스케줄링된다.
- [0014] 태스크가 디스플레이된 시간 슬롯으로 드롭되면, 재스케줄러 (즉, 선형 재스케줄링 알고리즘을 이용한 스케줄링 엔진)은 작업 센터 캘린더에 설정된 작업 시간 및 각각의 태스크에 설정된 태스크 지속기간에 기초하여 태스크들을 수평으로 스택한다.
- [0015] 태스크들은 과거의 시간 슬롯 또는 작업 스케줄 내에 맞지 않는 시간 슬롯에 잘못하여 스케줄링될 수 없다.
- [0016] 또 다른 태스크의 스케줄링에 의해 영향을 받은 스케줄링된 태스크들은 자동적으로 재스케줄링된다. 태스크의 준성들 및 현재의 스케줄이 또 다른 스케줄링된 태스크로 하여금 지연되어 시작되도록 초래할 때, 이 제약이 해결할 사용자에게 보여진다.
- [0017] 이용불가능한 리소스들은 태스크 시간 슬롯에 할당되지 않을 수도 있다.
- [0018] 따라서, 실시형태들은 다음을 제공한다:
- [0019] (i) 데이터를 관리하고 보는 시간 감소,
  - [0020] (ii) 현저하게 향상된 데이터 무결성 및 이용가능성, 및
  - [0021] (iii) 현저하게 향상된 프로젝트 조정 및 관리.
- [0022] 실시형태들은 디스플레이된 태스크를 원하는 시간 슬롯으로 드래그함으로써 태스크들이 용이하게 재스케줄링되는 그래픽 사용자 인터페이스를 제공한다. 이에 응답하여, 필요시 재스케줄링 엔진은 현재-다운스트림 (now-downstream) 태스크들을 자동적으로 재스케줄링한다.
- [0023] 가용 리소스들 (예컨대, 사람, 기계류, 장비, 툴들, 등)은 사람의 표시 또는 표현을 디스플레이된 태스크로 드래그함으로써 스케줄링된 태스크에 할당될 수도 있다. 이용불가능한 리소스들은 태스크 시간 슬롯에 할당되

지 않을 수도 있다.

[0024] 본 발명을 구현하는 스케줄링 방법은 스케줄링된 태스크들을 시간 진행 및 각각의 태스크 시작 일자/시간들의 시간 순서로 디스플레이하는 하나 이상의 전자 캘린더들을 제공하는 디지털 프로세서를 포함한다. 디스플레이된 시간 진행은 수평으로 지향된 뷰잉 패턴을 따르거나 또는 수직으로 지향된 뷰잉 패턴을 따를 수도 있다. 뷰잉 패턴에서의 방향은 시간 진행에서의 일자/시간의 증가/감소와 같은 것을 의미한다.

[0025] 디스플레이된 전자 캘린더와의 사용자 상호작용에 응답하여, 컴퓨터-구현 방법은 캘린더에서 사용자 요청되고 미스케줄링된 태스크를 자동적으로 스케줄링하고 임의의 영향받은 이전에 스케줄링된 태스크들을 재스케줄링한다. 그 결과, 캘린더는 사용자 요청된 태스크 및 이전에 스케줄링된 태스크들을 포함하는 태스크들의 논리적으로 올바른 스케줄을 유지한다.

[0026] 전자 캘린더는 시간 진행의 일자 및 시간을 증가시키는 것이 전자 캘린더의 뷰잉 패턴의 일자 및 시간을 증가시키는 것과 동일한 방향으로 보여지도록 각각의 태스크 시작 일자들/시간들의 시간 순서로 스케줄링된 태스크들을 디스플레이한다.

### 도면의 간단한 설명

[0027] 전술한 것은 첨부 도면들에 예시된 바와 같은, 본 발명의 예시적인 실시형태들의 다음의 더 구체적인 설명으로부터 명백히 알 수 있을 것이며, 도면 중, 유사한 참조 부호들은 상이한 뷰들 전반에 걸쳐서 동일한 부품들을 지칭한다. 도면들은 반드시 축척대로 도시하지는 않으며, 오히려 본 발명의 실시형태들을 예시하는 것을 강조한다.

도 1 은 일 실시형태의 구성도이다.

도 2a 내지 도 2d 는 실시형태들에서 태스크들을 스케줄링하는 그래픽 사용자 인터페이스의 개략 예시도들이다.

도 3a 내지 도 3c 는 실시형태들에서 태스크들을 재스케줄링하는 그래픽 사용자 인터페이스의 개략 예시도들이다.

도 4a 내지 도 4c 는 사람 또는 리소스를 스케줄링된 태스크에 할당하는 그래픽 사용자 인터페이스의 개략 예시도들이다.

도 5 는 컴퓨터 시스템의 블록도이다.

도 6 은 도 5 의 컴퓨터 시스템으로 형성된 컴퓨터 네트워크의 개략도이다.

### 발명을 실시하기 위한 구체적인 내용

[0028] 본 발명의 예시적인 실시형태들의 설명이 뒤따른다.

[0029] 도 1 내지 도 6 에 예시된 것은 본 발명의 원리들을 구현하는 자동적인 시각적 스케줄링 툴 (100) 이다. 툴 (100) 은 작업 태스크들의 수평으로 지향된 시간 라인 뷰를 가진 작업 센터 캘린더 (110) (즉, 전자 캘린더) 를 제공한다. 작업 태스크들은 예를 들어, 제조 작업 현장상에서의 상점 주문 동작들 및 생산 관련된 태스크들, 또는 다른 필드들에서, 사람, 로봇들, 장비, 기계류, 툴들 또는 다른 리소스들 등에 스케줄링되고 할당된 태스크들일 수도 있다. 이하에서 명백해지는 바와 같이, 작업 센터 캘린더 (110) 의 시간 라인은 좌측에서 우측으로 미래로 진행하는 일자들 및 시간들을 예시한다. 작업 센터 캘린더 (110) 은 시간 라인을 따라 각각의 시작 시간에 의해 배열된 스케줄링된 작업 태스크들을 나타낸다. 작업 센터 캘린더 (110) 에서 작업 태스크들의 수평 디스플레이하는 차기 작업 및/또는 작업의 시간 순서의 용이한 지각을 가능하게 한다.

[0030] 툴 (100) 은 드래그 앤드 드롭 그래픽 사용자 인터페이스 (210) 및 스케줄링 엔진 (130) 을 이용한다. 작업 센터 캘린더 (110) 은 사용자 요청들에 즉각 대응하며 다수의 작업 태스크들의 중앙집중화된 마스터 스케줄 (하나 이상의 스케줄일 수도 있다) 을 생성하고 유지한다. (재스케줄러 구성요소 포함한) 스케줄링 엔진 (130) 은 스케줄링되고 디스플레이된 작업 태스크들을 사용자 입력 및 태스크 지속기간에 기초하여 재조정한다. 데이터베이스 (115) 는 작업 센터 캘린더 (110) 의 동작들 및 기능을 지원한다.

[0031] 일 실시형태에서, 데이터베이스 (115) 는 각각의 작업 태스크에 대하여 다음을 저장한다: (i) 표현, 및 (ii) 시간 지속기간 정보. 표현은 태스크의 설명, 이름 또는 타이틀, 및/또는 아이콘 (심볼, 썬네일 이미지 또는 다른 시각적 표시자) 로 형성될 수도 있다. 데이터베이스 (115) 는 캘린더 (110) 에 대해 스케줄링된 작업

태스크 당 각각의 작업 시간을 저장한다. 데이터베이스 (115) 는 또한, 요일들 및 유효한 작업 교대들 (작업 시간들) 인 하루 당 시간들, 따라서 작업 태스크들 및/또는 리소스들에 할당되거나 또는 작업 태스크들 및/또는 리소스들에 대하여 스케줄링될 가능한 후보들 뿐만 아니라, (예를 들어, 공휴일, 재고 정리, 등으로 인해) 유효하지 않은 년 (year) 중 주들 및 월 (month) 중 일들 (days) 을 캘린더 (110) 에 대해 저장한다. 이러한 방법으로, 데이터베이스 (115) 는 작업 태스크 정의들 및 캘린더 작업 시간 정의들, 및 작업 태스크들 및 리소스들의 스케줄 할당들을 저장한다.

[0032] 특히, 실시형태들에서, 스케줄링된 태스크들은 수평 일자-시간 기반의 포맷으로 디스플레이된다. 바람직하게는, 스케줄링된 태스크들은 태스크들이 생성되거나 또는 스케줄이 외부 시스템에 의해 업데이트될 때 수평 위치로 자동적으로 디스플레이된다. 태스크들은 데스크탑 또는 태블릿 디바이스들을 이용하여 임의의 디스플레이 영역 (예컨대, 원도우들, 네트워크 텱, 등) 에서 보여지거나 관리될 수도 있다.

[0033] 사용자는 대상 태스크의 표현을 유효한 일자-시간 위치로 드래그함으로써 스케줄링된 태스크를 재스케줄링하거나 비-스케줄링된 태스크를 스케줄링할 수 있다. 비-유효한 일자-시간 위치들은 과거의 임의의 일자-시간 또는 미리 결정된 작업 캘린더 일자-시간 범위(들) 내에 맞지 않는 임의의 일자-시간을 포함한다.

[0034] 사용자가 대상 태스크를 시간 슬롯으로 드롭하면, 스케줄링 엔진 (130) 은 작업 센터 캘린더에 설정된 작업 시간 및 각각의 태스크에 설정된 태스크 지속기간에 기초하여 태스크들을 수평으로 스택하기 위하여 선형 재스케줄링 알고리즘을 이용한다.

[0035] 따라서, 태스크의 스케줄링에 의해 영향을 받은 모든 태스크들은, 각각의 태스크의 지속기간 및 미리 결정된 작업 캘린더 일자-시간 범위(들) 내에 맞는 일자-시간의 이용가능성에 기초하여 (스케줄링 엔진 (130) 에 의해) 자동적으로 선형적으로 재스케줄링된다.

[0036] 사용자는 태스크를 과거의 시간 슬롯 또는 작업 스케줄내에 맞지 않는 시간 슬롯에 잘못하여 스케줄링할 수 없다.

[0037] 사용자는 과거의 일자-시간에 대해 스케줄링되지 않은 태스크의 실행을 돋기 위하여 유효한 리소스들을 할당할 수 있다. 리소스는 그 리소스가 목표 태스크의 스케줄내에 맞는 일자-시간 범위 동안 또 다른 태스크를 돋기 위하여 이미 스케줄링되면 유효한 리소스가 아니다.

[0038] 도 2a 내지 도 4c 에 예시된 것은 작업 센터 캘린더 (110) 및 대응하는 그래픽 사용자 인터페이스 (210) (GUI) 의 비한정적인 예시적인 실시형태이다. 도 2a 에 도시된 것은 작업 센터 캘린더 (110), 미-스케줄링된 태스크들 (118) 의 표현들을 나타내는 사용자 스크롤 가능한 영역 (202), 및 여러 가용 리소스들 (예컨대, 사람, 로봇들, 툴들, 다른 장비, 기계류, 등) (120a, …120n) (일반적으로, (120)) 의 표시들을 가진 사용자 탐색 가능한 영역 (204) 로 형성되는 메인 뷰 (200) (GUI (210) 의 부분) 이다. 일반적으로, 작업 센터 캘린더 (110) 는 년, 월, 주, 일, 및 시간 단위들의 시간 계층에 의해 배열된다. 사용자 인터페이스 (210) 는 사용자로 하여금 줌 인/아웃 기능들에 의해 시간 그레뉼래티의 여러 레벨들 (예컨대, 하루 중의 시간들 대 요일들) 에서, 그리고 순방향/역방향 네비게이션 및 탐색 기능들에 의해 상이한 시간 기간들 (예를 들어, 어떤 월 및 년 중의 어떤 주에서의 날들의 범위들) 에서 캘린더 (110) 를 볼 수 있도록 한다. 이들 기능들의 공통 그래픽 사용자 인터페이스 기술 동작 및 구현예가 이용된다.

[0039] 예시된 예에서, 툴 (100) 은 미래로 진행하는 스크린 뷰 (200) 의 좌측에서 우측 사이드들로 배열된 일들의 범위를 가지는 캘린더 (110) 를 디스플레이한다. 디스플레이된 일들에 바로 앞서는 (선행하는) 일들은 GUI (210) 의 터치 스크린 스크롤링 또는 기타 등등에 의해 스크린 뷰 (200) 의 좌측 사이드로의 연속체 상에서 이용 가능하다. 이와 유사하게, 디스플레이된 일들에 바로 이어서 계속되는 (미래의) 일들은 GUI (210) 의 터치 스크린 스크롤링에 의해 스크린 뷰 (200) 의 우측 사이드로의 연속체 상에서 액세스 가능하다. 이러한 수평 디스플레이 레이아웃 및 뷰잉 패턴 (예컨대, 시간의 진행과 아주 밀접하게 좌측에서 우측으로 지향된) 은 사용자가 한 주에서 하루 당, 또는 하루에서 시간 당 또는 다른 시간 척도로 스케줄링된 작업 태스크들 (112) 의 순서를 시각화할 수 있도록 한다. 캘린더 (110) 데이터 및 스케줄링된 태스크들 (112) 의 이 방위 및 프리젠테이션은 한눈에 시간 순서의 용이한 지각 및 스케줄링되고 미스케줄링된 시간의 감지를 제공한다.

[0040] 일부 실시형태들에서, 툴 (110) 이 로딩되면, 디폴트 시간 범위가 디스플레이된다. 툴 (100) 은 사용자가 디스플레이된 시간 범위를 옵션적으로 변경할 수 있도록 한다. 일단 사용자가 원하는 시간 범위에서 전자 캘린더 (110) 뷰를 설정하면, 도 2a-2d 에 도시한 바와 같이 사용자는 미스케줄링된 작업 태스크 (118) 를 스케줄링한다. 사용자는 커서 제어 디바이스 하이라이트 선택에 의해서 또는 터치 및 선택된 작업 태스크 (118)

상의 콘택 압력을 유지함으로써 미스캐줄링된 태스크들의 영역 (202) 으로부터 원하는 작업 태스크 (118) (그의 표현) 를 선택한다. 콘택 압력 (또는, 커서 제어 디바이스 선택/상대적인 동작) 을 유지하면서, 도 2a 및 도 2b 에 예시된 바와 같이 사용자는 선택된 작업 태스크 표현 (118) 을 미스캐줄링된 태스크들 영역 (202) 으로부터 캘린더 (110) 로 이동 (드래그) 한다. 특히, 도 2c 에 예시된 바와 같이 사용자는 전자 캘린더 (110) 에서의 스캐줄링된 태스크들 (112a…n) (일반적으로, 112) 중에서 원하는 일자 및 시간의 선택된 작업 태스크 표현 (118) 을 드래그 및 드롭한다. 일반 드래그 앤드 드롭 기술이 이용된다.

[0041] 이에 응답하여, 스캐줄링 엔진 (130) (도 1 에 도시) 은 사용자 선택된 일자/시간에 사용자 선택된 작업 태스크 (118) 를 시간 슬롯에 스캐줄하고 (이전에 스캐줄링된) 현재 다운-스트림 (now down-stream) 작업 태스크들 (112) 을 자동적으로 재스캐줄링한다. 이를 달성하기 위해, 스캐줄링 엔진 (130) 은 (i) 사용자 선택된 일자/시간의 유효성을 결정하고 (즉, 사용자 선택된 일자/시간이 과거가 아니며 도 1 에 도시된 데이터베이스 (115) 에 정의된 작업 시간들 내에 있다는 것을 입증하고), 그리고 (ii) 선형 재스캐줄링 알고리즘을 적용한다.

그 결과, 툴 (100) 은 사용자 선택된 일자/시간에 시작하고 작업 태스크 (118) 에 의해 정의된 지속기간을 가진 캘린더 스캐줄에 사용자 선택된 작업 태스크 표현 (118) 을 디스플레이한다. 도 2d 는 예시적이다.

툴 (100) 은 또한 따라서 조정되거나 또는 재스캐줄링된 시작 시간들 (일자/시간들) 에 현재 다운스트림 작업 태스크들 (112) (그의 표현들) 을 작업 센터 캘린더 (110) 에 디스플레이한다. 스캐줄링 엔진 (130) 은 따라서 업데이트된 캘린더 스캐줄 (작업 태스크들 (112) 할당들) 을 데이터베이스 (115) 에 저장한다.

[0042] 유사한 방법으로, 사용자가 선택된 작업 태스크 (118) 를 전자 캘린더 (110) 에서의 스캐줄링된 작업 태스크 (112) 상에 (즉, 스캐줄링된 작업 태스크 112에 이전에 할당된 이미 점유된 시간 슬롯상에) 드래그 및 드롭하면, 스캐줄링 엔진 (130) 은 대응하는 시간 슬롯이 점유되거나 선택되었다는 것을 인식한다. 스캐줄링 엔진 (130) 은 (i) 충돌되는 시간 슬롯으로부터 대상 스캐줄링된 작업 태스크 (112) 를 재스캐줄링하고, (ii) 대상 시간 슬롯에 사용자 선택된 작업 태스크 (118) 를 스캐줄링하고, 그리고 (iii) 다운 스트림 작업 태스크들 (112) 를 재스캐줄링한다. 스캐줄링 엔진 (130) 은 따라서 업데이트된 캘린더 스캐줄 (작업 태스크들 (112a…n) 로의 일자/시간 슬롯 할당들) 을 데이터베이스 (115) 에 저장한다.

[0043] 도 2a-2d 에서와 같은 캘린더 (110) 에서의 하나 이상의 태스크들 (118) 의 스캐줄링에 후속하여, 도 3a-3c 에 예시된 바와 같이 사용자는 스캐줄링된 작업 태스크 (112) 를 재스캐줄링할 수도 있다. 터치 투 셀렉트 앤 드 드래그 (touch to select and drag) 기술을 이용하여, 사용자는 전자 캘린더 (110) 에서 현재 (할당된) 하나의 시간 슬롯 (232) 에서 원하는 스캐줄링된 작업 태스크 (112) 를 선택하고 (도 3c 에 도시) 상기 원하는 스캐줄링된 작업 태스크 (112) 를 원하는 시간 슬롯 (230) 으로 이동 (드래그) 한다 (도 3b 에 도시). 이에 응답하여, 스캐줄링 엔진 (130) 은 데이터베이스 (115) 에서의 캘린더 (110) 정의들에서 설정된 작업 시간들 및 데이터베이스 (115) 에서 정의되고 저장된 작업 태스크들의 각각의 시간 지속기간들에 기초하여 캘린더 (110) 에서 영향을 받은 스캐줄링된 작업 태스크들 (112) 을 재스캐줄링한다. 바람직하게는 스캐줄링 엔진 (130) 은 아래에 상세히 설명된 선형 재스캐줄링 알고리즘을 적용하는 재스캐줄러를 포함한다. 스캐줄링 엔진 (130) 은 업데이트된 캘린더 스캐줄을 데이터베이스 (115) 에 저장한다. 툴 (100) 은 도 3c 에 예시된 바와 같이 GUI (210) 에서 최종 캘린더 (110) 스캐줄을 디스플레이한다.

[0044] 스캐줄링 엔진 (130) 은 작업 태스크가 과거의 시간 슬롯에서, 또는 데이터베이스 (115) 에서 캘린더 (110) 에 대해 정의된 작업 시간들내에 맞지 않는 시간 슬롯에서, 스캐줄링되거나 재스캐줄링되지 않도록 한다.

[0045] 도 4a 내지 도 4c 는, GUI (210) 를 통하여, 도 4a 내지 도 4c 에 도시된 바와 같이 어떻게 사용자가 전자 캘린더 (110) 에서 가용 리소스들 (예컨대, 사람, 로봇들, 툴들, 장비, 등) (120a…n) 을 스캐줄링된 작업 태스크들 (112a…n) 에 할당할 수도 있는지를 예시한다. 드래그 앤드 드롭 GUI 기법들을 이용하여, GUI (210) 는 사용자가 리소스들의 영역 (204) 으로부터 원하는 리소스 (120) 를 선택하고 사용자 선택된 리소스 (120) 의 표현의 복사본을 영역 (204) 에서 캘린더 (110) 영역으로 원하는 스캐줄링된 작업 태스크 (112) 로 이동 (드래그) 하도록 한다.

[0046] 스캐줄링 엔진 (130) 은 사용자 선택된 리소스 (120) 의 이용가능성을 데이터베이스 (115) 로부터 결정함으로써 즉각 응답한다. 리소스들 (120a…n) 의 할당된 시간들 및 유효한 가용 작업 시간들의 각각의 스캐줄들은 작업 태스크들 (112a…n 및 118a…n) 의 스캐줄들과 유사하게 데이터베이스 (115) 에 저장될 수도 있다. 따라서, 스캐줄링 엔진 (130) 은 사용자 선택된 리소스 (120) 가 목표 스캐줄링된 작업 태스크 (112) 의 대응하는 시간 슬롯들 (시작 일자/시간 및 지속기간) 에 대하여 가용 시간을 가지고 있다고, 뿐만 아니라 대응하는 시간 슬롯이 사용자 선택된 리소스 (120) 의 유효한 작업 시간들 내에 있다고 결정한다. 이러한 방법으로, 툴

(100) 은 이용불가능한 리소스들 (120) 이 작업 태스크들 (112) 및 대응하는 시간 슬롯들에 할당되는 것을 방지 한다. 스케줄링 엔진 (130)이 사용자 선택된 리소스 (120)의 이용가능성을 확정적으로 결정한 후에, 스케줄링 엔진 (130)은 선택된 작업 태스크 (112) 및 대응하는 시간 슬롯을 리소스 (120)에 할당한다. 결국, 스케줄링 엔진 (130)은 데이터베이스 (115)에서 캘린더 스케줄 및 리소스 작업 태스크/시간 슬롯 할당들을 업데이트한다. 도 4c에 나타낸 바와 같이 툴 (100)/GUI (210)는 최종 작업 센터 캘린더 (110)를 디스플레이 한다.

[0047] 이제 스케줄링 /재스케줄링 엔진 (130)으로 돌아가서, 하나의-실시형태에서 구현에는 다음과 같다.

[0048] 사용자가 특정의 태스크 (아래에서, 'currentTask'로 지칭됨)를 드래그한 후에, 스케줄링 엔진 (130)은 다음 알고리즘을 실행한다. 본원에서 시작 시간 및 종료 시간을 언급할 때, 시간/일자 조합이 의도된다 - 하루의 일자 및 시각 양쪽이 고려된다.

[0049] 1) currentTask의 드래그 위치에 기초하여 시작 시간을 재계산한다. 스케줄링 엔진 (130)은 태스크를 구성 가능한 증가분 (즉, 가장 가까운 시간, 가장 가까운 하루, 가장 가까운 15 분 기간, 등)으로 '스냅 (snap)'하도록 구성된다.

[0050] 2) 새로운 시작 시간이 기존 태스크의 중간에 있는지를 결정하고, 만약 그렇다면, 시작 시간을 그 태스크의 끝으로 이동시킨다

[0051] a. 모든 태스크들의 부하 시작 시간 및 종료 시간 및 순서를 연대순으로 로딩한다.

[0052] b. 각각의 태스크를 분류한다. currentTask.startTime > existingTask.startTime 이고 currentTask.startTime < existingTask.endTime 이면, currentTask.startTime=existingTask.endTime로 설정한다

[0053] 3) 새로운 시작 시간 및 태스크 지속기간에 기초하여, 종료 시간을 재계산한다. 작업 일 (day) 길이를 조정 한다.

[0054] 4) 다음 태스크들을 이동시킨다 - 그의 종료 일자가 또 다른 태스크에 교차하도록 태스크가 드래그되면, 제 2 태스크의 시작 일자는 조정되어야 한다. 이 기능은 회귀적으로 호출된다는 점에 주의한다 - 제 2 태스크가 이동되면, 그것은 제 3 태스크가 이동 등이 되도록 강요할 수도 있다.

[0055] a. 모든 태스크들의 시작 시간 및 종료 시간 및 순서를 연대순으로 로딩한다. 시작 시간이 currentTask.startTime 보다 더 긴 제 1 태스크를 로케이트한다.

[0056] b. 리스트에 남아있는 모든 태스크들을 루프한다. 이 리스트 내 각각의 existingTask에 대해, existingTask.startTime < currentTask.endTime 및 existingTask.startTime > currentTask.startTime이면,

[0057] i. existingTask의 지속기간을 계산한다

[0058] ii. existingTask.startTime = currentTask.endTime를 설정한다

[0059] iii. 지속기간에 기초하여 existingTask.endTime를 설정한다

[0060] iv. 작업 일 길이를 조정한다

[0061] v. existingTask를 수평 디스플레이에 위치시킨다.

[0062] vi. (회귀적 콜에서 이제 currentTask이 된) existingTask에 대하여 moveFollowingTasks를 회귀적으로 콜한다.

[0063] 5) 덮어 씌어진 태스크들을 이동시킨다 - 또 다른 태스크를 완전히 커버하도록 태스크가 드래그되면, 커버하는 태스크는 currentTask를 뒤따르도록 이동된다

[0064] a. 모든 태스크들의 시작 시간 및 종료 시간 및 순서를 연대순으로 로딩한다. 모든 태스크들 (existingTask)을 루프한다.

[0065] b. currentTask.startTime < existingTask.startTime 및 currentTask.endTime > existingTask.endTime이면,

[0066] i. existingTask의 지속기간을 계산한다

[0067] ii. existingTask.startTime = currentTask.endTime을 설정한다

- [0068] iii. 지속기간에 기초하여 existingTask.endTime 를 설정한다
- [0069] iv. 작업 일 길이를 조정한다
- [0070] v. existingTask 를 수평 디스플레이에 위치시킨다
- [0071] 6) 태스크들을 재정리한다 - 일단 스케줄 엔진 (130) 이 태스크들의 모두를 이동하였으면, 엔진은 어떻게 태스크 시작/종료 시간들이 변경되었는지에 기초하여 데이터를 재정리해야 한다
- [0072] a. 가장 이른 시작 시간을 가진 태스크를 찾는다. 리스트의 시작에 위치시킨다.
- [0073] b. reorderTasks 를 회귀적으로 콜하고, 리스트의 시작에 배치된 태스크에 들어간다. 이것은 제 2 태스크를 설정하고 그런 다음 제 3 태스크, 등을 찾기 위하여 회귀적으로 콜할 것이다.
- [0074] 작업 센터 캘린더 (110) 및/또는 스케줄링 엔진 (130) 은 다음을 고려한다:
- [0075] 수평 스케줄 디스플레이에서, 실시형태들은 줌 기능을 구현한다. 이것은 사용자가 더 길거나 더 짧은 시간 기간들을 보기 위하여 전자 캘린더 (110) 의 디스플레이를 '줌' 하도록 허용한다. 툴 (100) 알고리즘들은 디스플레이의 줌 레벨에 기초하여 태스크 엘리먼트들 (스케줄링된 작업 태스크들 (112)) 의 위치 및 사이징을 재계산한다.
- [0076] 스케줄 디스플레이는 또한 상이한 작업 스케줄들을 고려한다. 캘린더 (110) 스케줄들은 예를 들어, 8 시간, 16 시간, 또는 24 시간 작업 일들로 구성될 수 있다. 스케줄링된 작업 태스크들 (112) 을 이동할 때, 만약 태스크 (112) 가 작업 일의 끝을 지났다면, 종료 시간은 대상 태스크 (112) 가 다음 유효한 작업 일에 완료되도록 조정된다. 예를 들어, 만약 상점이 오전 8:00 에 열어서 오후 4:00 에 닫는다면, 그리고 6 시간 태스크 (112) 가 오후 3:00 에 시작하도록 스케줄링되면, 스케줄링 엔진 (130) 은 태스크 (112) 의 종료 시간을 다음 유효한 작업 일의 오후 12:00 에 자동적으로 설정한다.
- [0077] 또 다른 실시형태에서, 룸 마다 작업 센터 캘린더 (110) 가 있을 수도 있다. 예를 들어, 어떤 공장 룸은 각각의 작업 센터 캘린더 (110) 를 가진다. 태스크들 (112, 118) 및 리소스들 (120) 은 위에서 설명된 기법들 및 특징들을 이용하여 이 캘린더상에 스케줄링된다. 게다가, 태스크들 (112, 118) 은 하나의 공장 룸의 캘린더 (110) 으로부터 상이한 공장 룸의 각각의 캘린더 (110) 로 전송될 수도 있다. 스케줄링 엔진 (130) 은 (대상 캘린더 (110) 이내에서 뿐만 아니라) 복수의 작업 센터 캘린더들 (110) 에 걸쳐서 태스크들 (112, 118) 및 영향을 받은 사전 스케줄링된 태스크들 (112) 을 유효하게 스케줄링/ 재스케줄링하기 위하여 위에서 설명된 알고리즘들 및 기법들을 이용한다.
- [0078] 도 5 는 본원에서 컴퓨터 시스템으로서 또한 지칭될 수도 있는 예시적인 컴퓨터-지원 설계 스테이션 (300) 의 개략 블록도이다. 본원에서 사용될 때, 용어들 "컴퓨터-지원 설계 스테이션" 및 "컴퓨터 시스템" 은 일반적으로 위에서 설명한 프로세스들 및/또는 위에서 설명한 것들에 관련될 수도 있는 임의의 추가적인 프로세스들을 수행하는데 이용될 수도 있는 임의의 적합한 컴퓨팅 디바이스를 지칭한다.
- [0079] 예시적인 실시형태에서, 컴퓨터-지원 설계 스테이션 (300) 은 위에서 설명한 프로세스들 및/또는 위에서 설명한 것들과 관련될 수도 있는 임의의 추가적인 프로세스들을 수행하는 하나 이상의 프로세서들 (302) (CPU) 를 포함한다. 용어 "프로세서" 는 일반적으로, 시스템들 및 마이크로제어기들을 포함하는 임의의 프로그래밍 가능 시스템, RISC (reduced instruction set circuits), 주문형 집적 회로들 (ASIC), 프로그래밍 가능 로직 회로들, 및/또는 본원에서 설명한 기능들을 수행할 수 있는 임의의 다른 회로 또는 프로세서를 지칭하는 것으로 이해되어야 한다. 상기 예들은 단지 예시적이며, 따라서, 어떤 방법으로든 용어 "프로세서" 의 정의 및/또는 의미를 한정하려고 의도되지 않는다.
- [0080] 위에서 설명한 것들과 관련될 수도 있는 위에서 설명한 프로세스들 및/또는 임의의 추가적인 프로세스들의 단계들은 예를 들어, 시스템 버스 (306) 에 의해 프로세서 (302) 에 동작가능하게 및/또는 통신가능하게 커플링된 메모리 영역 (304) 과 같은, 비일시성 컴퓨터-판독가능 매체에, 컴퓨터-실행가능한 명령들로서 저장될 수도 있다. 본원에서 사용할 때의 "메모리 영역" 은, 일반적으로 전자 캘린더 툴 (100) 을 이용하여 자동적으로 스케줄링 및 재스케줄링하는 것을 지원하기 위하여 하나 이상의 프로세서들에 의해 실행가능한 프로그램 코드 및 명령들의 임의의 수단을 지칭한다. 메모리 영역 (304) 은 하나, 또는 하나 보다 많은, 메모리의 유형들을 포함할 수도 있다. 예를 들어, 메모리 영역 (304) 은 비-휘발성 RAM, 자기 RAM, 강유전체 RAM, 및/또는 다른 형태들의 RAM 을 포함할 수 있는 랜덤-액세스 메모리 (RAM) (308) 를 포함할 수도 있다. 메모리 영역 (304) 은 또한 판독 전용 메모리 (ROM) (310) 및/또는 플래시 메모리 및/또는 전기적으로-소거가능한 프로그래

밍가능 판독 전용 메모리 (EEPROM) 를 포함할 수도 있다. 다른 형태들의 메모리 단독 또는 이와 조합하여, 하드-디스크 드라이브 (HDD) (312) 와 같은, 임의의 다른 적합한 자기, 광학, 및/또는 반도체 메모리가, 메모리 영역 (304) 에 포함될 수도 있다. HDD (312) 는 또한 메시지들을 프로세서 (302) 로 송신하고 그로부터 수신하는데 사용을 위해 디스크 제어기 (314) 에 커플링될 수도 있다. 더욱이, 메모리 영역 (304) 은 또한, 적합한 카트리지 디스크, CD-ROM, DVD, 또는 USB 메모리와 같은, 분리식 또는 착탈식 메모리 (316) 이거나 또는 이를 포함할 수도 있다. 상기 예들은 단지 예시적이며, 따라서, 어떤 방법으로든 용어 "메모리 영역" 의 정의 및/또는 의미를 한정하려고 의도되지 않는다.

[0081] 컴퓨터-지원 설계 스테이션 (300) 은 또한 디스플레이 제어기 (320) 에 커플링된, 예컨대 동작가능하게 커플링된 디스플레이 디바이스 (318) 를 포함한다. 디스플레이 제어기 (320) 는 디스플레이 디바이스 (318) 에 의한 디스플레이를 위해 시스템 버스 (306) 를 통해서 데이터를 수신한다. 디스플레이 디바이스 (318) 는, 한정 없이, 모니터, 텔레비전 디스플레이, 플라즈마 디스플레이, 액정 디스플레이 (LCD), 발광 다이오드들에 기초한 디스플레이 (LED), 유기 LED들에 기초한 디스플레이 (OLED), 폴리머 LED들에 기초한 디스플레이, 표면-전도전자 에미터들에 기초한 디스플레이, 투영된 및/또는 반사된 이미지를 포함하는 디스플레이, 또는 임의의 다른 적합한 전자 디바이스 또는 디스플레이 메카니즘일 수도 있다. 더욱이, 디스플레이 디바이스 (318) 는 연관된 터치스크린 제어기를 가진 터치스크린을 포함할 수도 있다. 상기 예들은 단지 예시적이며, 따라서, 어떤 방법으로든 용어 "디스플레이 디바이스" 의 정의 및/또는 의미를 한정하려고 의도되지 않는다.

[0082] 게다가, 컴퓨터-지원 설계 스테이션 (300) 은 네트워크 (도 5 에 미도시) 와 통신할 때에 사용을 위한 네트워크 인터페이스 (322) 를 포함한다. 더욱이, 컴퓨터-지원 설계 스테이션 (300) 은 키보드 (324) 와 같은 하나 이상의 입력 디바이스들, 및/또는 롤러 볼, 마우스, 터치패드 등과 같은 포인팅 디바이스 (326) 를 포함한다. 입력 디바이스들은 시스템 버스 (306) 에 추가로 커플링되는 입력/출력 (I/O) 인터페이스 (328) 에 커플링되어 그에 의해 제어된다.

[0083] 디스플레이 디바이스 (318), 키보드 (324), 포인팅 디바이스 (326) 뿐만 아니라, 디스플레이 제어기 (320), 디스크 제어기 (314), 네트워크 인터페이스 (322), 및 I/O 인터페이스 (328) 의 일반적인 특징들 및 기능의 설명은 이들 특징들이 알려져 있으므로, 본원에서 간결성을 위해 생략된다.

[0084] 사용자 인터페이스 (210) 에 대한 드래그 앤드 드롭 기법들은 터치 스크린, 커서 제어 디바이스 또는 그 기술분야에서 공통인 다른 I/O 기술에 의해 구현될 수도 있다.

[0085] 도 6 은 위에서 설명한 프로세스들 및/또는 위에서 설명한 것들과 관련될 수도 있는 추가적인 프로세스들과 같은, 모델링된 오브젝트의 컴퓨터-지원 설계에서의 사용을 위한 예시적인 시스템 (400) 의 개략 블록도이다. 예시적인 실시형태에서, 메모리 영역 (402) 는 스캐줄링된 태스크들 (112) 데이터, 스캐줄링된 리소스들 (120) 데이터, 태스크들 정의들, 리소스들 정의들 및 캘린더 작업 시간들 정의들 및/또는 미스캐줄링된 태스크들 (118) 데이터와 같은, 작업 센터 캘린더 (110) 데이터를 저장할 때에 사용을 위한 하나 이상의 저장 디바이스들 (404) 를 포함한다. 일부 실시형태들에서, 메모리 영역 (402) 는 서버 (406) 에 커플링되며, 그 서버는 결과적으로 네트워크 (412) 를 통해서 관리자 시스템 (408) 및/또는 사용자 시스템 (410) 에 커플링된다. 저장 디바이스들 (404) 는 하나 이상의 데이터베이스들로서 구현될 수도 있거나, 단일 또는 다수의 지리적 사이트들에 로케이트될 수도 있거나, 또는 서버 (406) 와 통합될 수도 있다.

[0086] 주지하는 바와 같이, 네트워크 (412) 는 인터넷과 같은 공중 네트워크, 또는 LAN 또는 WAN 네트워크와 같은 사설 네트워크, 또는 이들의 임의의 조합일 수 있으며, 또한 PSTN 또는 ISDN 서브-네트워크들을 포함할 수 있다.

네트워크 (412) 는 또한 이더넷 네트워크와 같은 유선일 수 있거나, 또는 EDGE, 3G 및 4G 무선 셀룰러 시스템들을 포함한 셀룰러 네트워크와 같은 무선일 수도 있다. 무선 네트워크는 또한 WiFi, Bluetooth, 또는 기타의 임의의 다른 무선 유형의 통신일 수 있다. 따라서, 네트워크 (412) 는 단지 예시적이며 본 향상들의 범위를 조금도 한정하지 않는다.

[0087] 당업자는 인식하고 있는 바와 같이, 관리자 시스템 (408) 및/또는 사용자 시스템 (410) 은 도 5 를 참조하여 위에서 설명된 바와 같은 컴퓨터-지원 설계 스테이션, 또는 기지의 임의의 다른 컴퓨팅 시스템일 수 있다. 더욱이, 관리자 시스템 (408) 및/또는 사용자 시스템 (410) 은 위에서 설명한 프로세스들 및/또는 위에서 설명한 것들에 관련될 수도 있는 임의의 추가적인 프로세스들을 수행하도록 구성되는 것으로 이해되어야 한다.

[0088] 서버 (406) 는 위에서 설명한 프로세스들을 실행하는 컴퓨터-판독가능 명령들을 저장하고 이를 명령들을 네트워크 (412) 를 통해서 관리자 시스템 (408) 및/또는 사용자 시스템 (410) 에 제공한다. 더욱이, 서버 (406)

는 또한 필요에 따라 메모리 영역 (402) 으로부터의 데이터를 관리자 시스템 (408) 및 사용자 시스템 (410) 에 제공할 수 있다. 이와 같이, 도 6 은 클라우드 컴퓨팅, 분산 컴퓨팅 및 기타 등등을 통한 시스템 (400) 의 구현예들을 포함한다.

[0089] 모델링된 오브젝트의 컴퓨터-지원 설계에 사용하기 위한 시스템들, 방법들, 장치, 컴퓨터 프로그램 제품들, 및 컴퓨터-판독가능 저장 매체들의 예시적인 실시형태들이 위에서 자세히 설명되어 있다. 이 시스템들, 방법들, 장치, 컴퓨터 프로그램 제품들, 및 컴퓨터-판독가능 저장 매체들은 본원에서 설명하는 특정의 실시형태들에 한정되기 보다는 오히려, 방법들, 프로그램 제품들 및/또는 저장 매체들의 동작들뿐만 아니라, 시스템 및/또는 장치의 구성요소들은, 본원에서 설명되는 다른 동작들 및/또는 구성요소들과는 독립적으로 그리고 개별적으로 이용될 수도 있다. 또, 설명되는 동작들 및/또는 구성요소들은 또한 다른 시스템들, 방법들, 장치, 프로그램 제품들 및/또는 저장 매체들에 정의되거나, 또는 그와 함께 사용될 수도 있으며, 오직 본원에서 설명하는 바와 같은, 시스템들, 방법들, 장치, 프로그램 제품들 및 저장 매체들로 실시하는 것에 한정되지 않는다.

[0090] 본원에서 설명되는 것들과 같은, 컴퓨터 또는 컴퓨터 시스템은 적어도 하나의 프로세서 또는 프로세싱 유닛 및 시스템 메모리를 포함한다. 컴퓨터 또는 컴퓨터 시스템은 적어도 일부 유형의 컴퓨터 판독가능 매체들을 일반적으로 갖는다. 비한정적인 일 예로서, 컴퓨터 판독가능 매체들은 컴퓨터 저장 매체들 및 통신 매체들을 포함한다. 컴퓨터 저장 매체들은 컴퓨터 판독가능 명령들, 데이터 구조들, 프로그램 모듈들, 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 착탈식 및 비-착탈식 매체들을 포함한다. 통신 매체들은 일반적으로 컴퓨터 판독가능 명령들, 데이터 구조들, 프로그램 모듈들, 또는 다른 데이터를 반송과 또는 다른 전송 메커니즘과 같은 변조된 데이터 신호로 수록하며, 임의의 정보 전달 매체들을 포함한다. 당업자들은 변조된 데이터 신호를 알고 있으며, 그 변조된 데이터 신호는 그의 특성들 세트 중 하나 이상을 갖거나 또는 그 신호에 정보를 인코딩하는 방법으로 변경된다. 상술한 것 중 임의의 것들의 조합들이 또한 컴퓨터 판독가능 매체들의 범위 내에 포함된다.

[0091] 위에서 설명한 프로세스들에서 사용하기 위한 예시적인 비일시성 컴퓨터-실행가능한 구성요소들은 프로세서 (302) (도 5 에 도시) 로 하여금 디스플레이 디바이스 (318) (도 5 에 도시) 상에 작업 센터 캘린더 (110) 을 제시하도록 하는 디스플레이 구성요소를 포함하지만 단지 포함하는 것에 제한되지 않는다. 구성요소들은 또한 프로세서 (302) 로 하여금, 예를 들어, I/O 인터페이스 (328) (도 5 에 도시) 를 통해서 사용자 입력들을 수신하게 하는 인터페이스 구성요소를 포함한다. 위에서 설명한 바와 같이, 사용자 입력들은 작업 태스크들 (112, 118) 및 리소스들 (120) (도 2a 내지 도 4c에 도시) 를 유효하게 스케줄링/제스케줄링하기 위하여, 그리고/또는 수평으로 지향된 뷰잉 패턴을 따라서 시간 진행으로 캘린더 (110) 스케줄을 보기 위하여 툴 (100) 의 사용에 관련될 수도 있다.

[0092] 더욱이, 구성요소들은 프로세서 (302) 로 하여금 작업 태스크들 (112) 를 자동적으로 제스케줄링하고 캘린더 (110) 에 대하여 논리적으로 올바른 스케줄을 유지하도록 하는 스케줄링 엔진 (130) 구성요소를 포함한다. 일부 실시형태들에서, 스케줄링 엔진 (130) 구성요소는 프로세서 (302) 로 하여금 위에서 설명한 프로세스에서 와 같은, 태스크들 지속기간 및 유효한 작업 일자들/시간들에 기초하여 작업 태스크들 (112) 의 선형 제스케줄링을 실행하도록 한다.

[0093] 본 발명은 예시적인 컴퓨터 시스템 환경과 관련하여 설명되지만, 본 발명의 실시형태들은 매우 많은 다른 범용 또는 특수 목적 컴퓨터 시스템 환경들 또는 구성들과 사용 가능하다. 컴퓨터 시스템 환경은 본 발명의 임의의 양태의 사용 범위 또는 기능에 관한 임의의 제한을 암시하려고 의도된 것이 아니다. 더욱이, 컴퓨터 시스템 환경은 예시적인 동작 환경에 예시된 구성요소들 중 임의의 하나 또는 조합에 관한 임의의 의존성 또는 요구사항을 갖는 것으로 해석되어서는 안된다. 본 발명의 양태들과 함께 사용하기에 적합할 수도 있는 널리 공지된 컴퓨터 시스템들, 환경들, 및/또는 구성들의 예들은, 개인용 컴퓨터들, 서버 컴퓨터들, 휴대용 또는 랩탑 디바이스들, 멀티프로세서 시스템들, 마이크로프로세서-기반의 시스템들, 셋톱 박스들, 프로그래밍가능 가전 제품, 모바일 전화기들, 네트워크 PCs, 미니 컴퓨터들, 메인프레임 컴퓨터들, 상기 시스템들 또는 디바이스들 중 임의의 것을 포함하는 분산 컴퓨팅 환경들 등을 포함하지만, 이에 한정되지 않는다.

[0094] 본 발명의 실시형태들은 프로그램 구성요소들 또는 모듈들과 같은, 하나 이상의 컴퓨터들 또는 다른 디바이스들에 의해 실행되는 컴퓨터-실행가능한 명령들의 일반적인 컨텍스트 (일반적인 컨텍스트) 로 설명될 수도 있다. 본 발명의 양태들은 구성요소들 또는 모듈들의 임의의 개수 및 구성으로 구현될 수도 있다. 예를 들어, 본 발명의 양태들은 도면들 및 설명되는 본원에서 예시된 특정의 컴퓨터-실행가능한 명령들 또는 특정의 구성요소들 또는 모듈들에 한정되지 않는다. 본 발명의 대안적인 실시형태들은 본원에서 예시된 및 설명되는 정보

다 더 많거나 더 적은 기능을 갖는 상이한 컴퓨터-실행가능한 명령들 또는 구성요소들을 포함할 수도 있다.

[0095] 본원에서 예시되고 설명되는 본 발명의 실시형태들에서 동작들의 실행 또는 수행의 순서는 달리 규정하지 않는 한, 필수적은 아니다. 즉, 동작들은 달리 규정하지 않는 한, 임의의 순서로 수행될 수도 있으며, 본 발명의 실시형태들은 본원에서 개시된 동작들 보다 추가적인 또는 더 적은 동작들을 포함할 수도 있다. 예를 들어, 또 다른 동작 이전, 동시에, 또는 이후에 특정의 동작을 실행하거나 또는 수행하는 것은 본 발명의 양태들의 범위 내인 것으로 고려된다.

[0096] 본 발명 또는 그의 실시형태들의 양태들의 엘리먼트들을 도입할 때, 관사들 "하나의 (a)", "한 (an)", "그 (the)", 및 "상기 (said)" 는 엘리먼트들 중 하나 이상인 것을 의미하도록 의도된다. 용어들 "포함하는 (comprising)", 포함한 (including)", 및 "갖는 (having)" 은 포괄적인 것으로 의도되며, 리스트된 엘리먼트들과는 다른 추가적인 엘리먼트들일 수도 있다는 것을 의미하도록 의도된다.

[0097] 이 서면 설명은 예들을 이용하여 최상의 모드를 포함한 본 발명을 개시하며, 또한 임의의 당업자로 하여금, 임의의 디바이스들 또는 시스템들을 제조하고 이용하는 것 및 임의의 포함된 방법들을 수행하는 것을 포함하여, 본 발명을 실시가능하게 한다. 본 발명의 특허의 범위는 청구항들에 의해 정의되며, 당업자들에게 떠오르는 다른 예들을 포함할 수도 있다. 이러한 다른 예들은, 청구항들의 문자적 전문용어 (문자적 전문용어)와는 상이하지 않은 구조적 엘리먼트들을 가지면, 또는 청구항들의 문자적 전문용어와는 크지 않은 차이들을 가진 동등한 구조적 엘리먼트들을 포함하면, 청구항들의 범위 내에서 있는 것으로 의도된다.

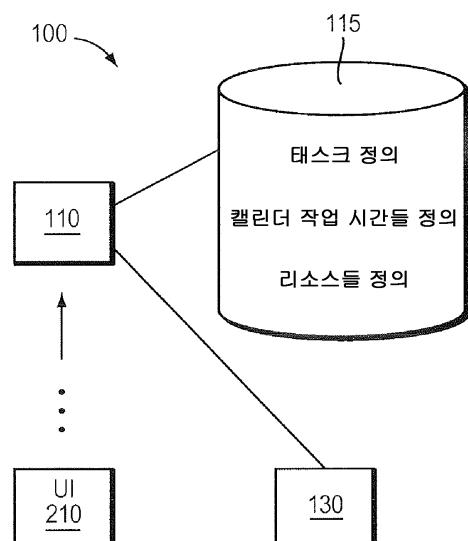
[0098] 모든 특허들의 교시들, 공개된 출원 및 본원에서 인용한 참고문헌들은 그들 전체로 참고로 포함된다.

[0099] 본 발명은 특히 그의 예시적인 실시형태들을 참조하여 도시하고 설명되었지만, 형상 및 세부 사항들에서의 여러 변화들이 첨부된 청구항들에 의해 포괄되는 본 발명의 범위로부터 일탈함이 없이, 이루어질 수도 있는 것으로 이해될 것이다.

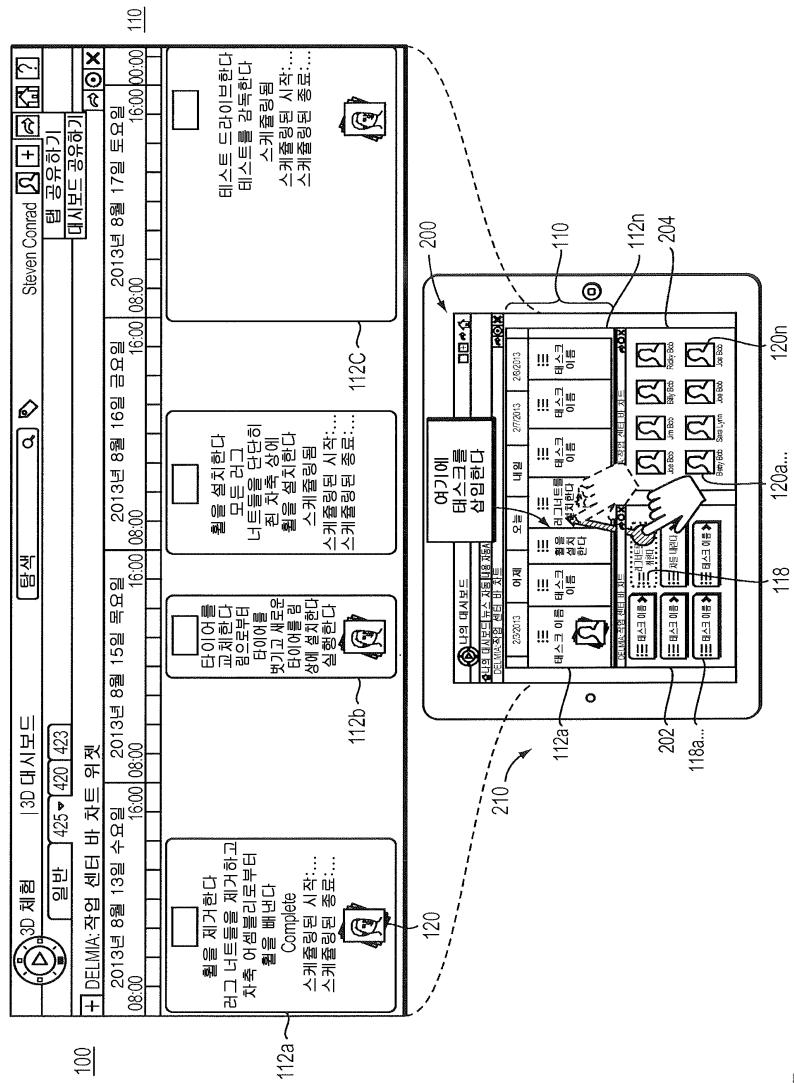
[0100] 예를 들어, 상술한 것은 작업 센터 캘린더 (110) 의 수평 방위를 가진 뷰잉 패턴을 예시한다. 작업 센터 캘린더의 수직 방위 및 대응하는 수직 뷰잉 감각 (뷰잉 패턴) 이 다른 실시형태들에 적합한 것으로 이해된다.

## 도면

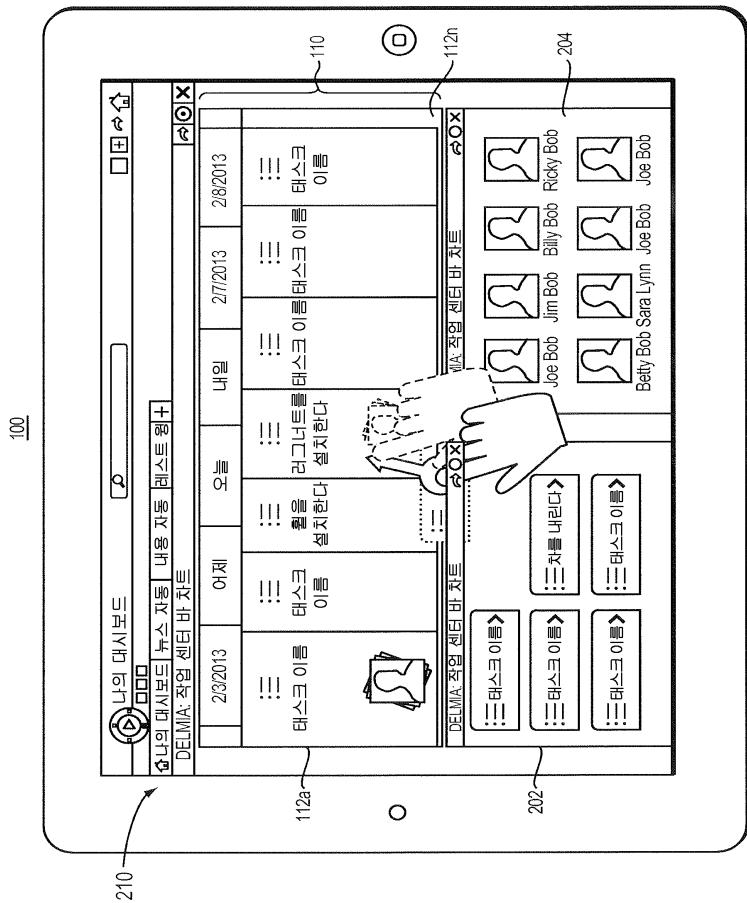
### 도면1



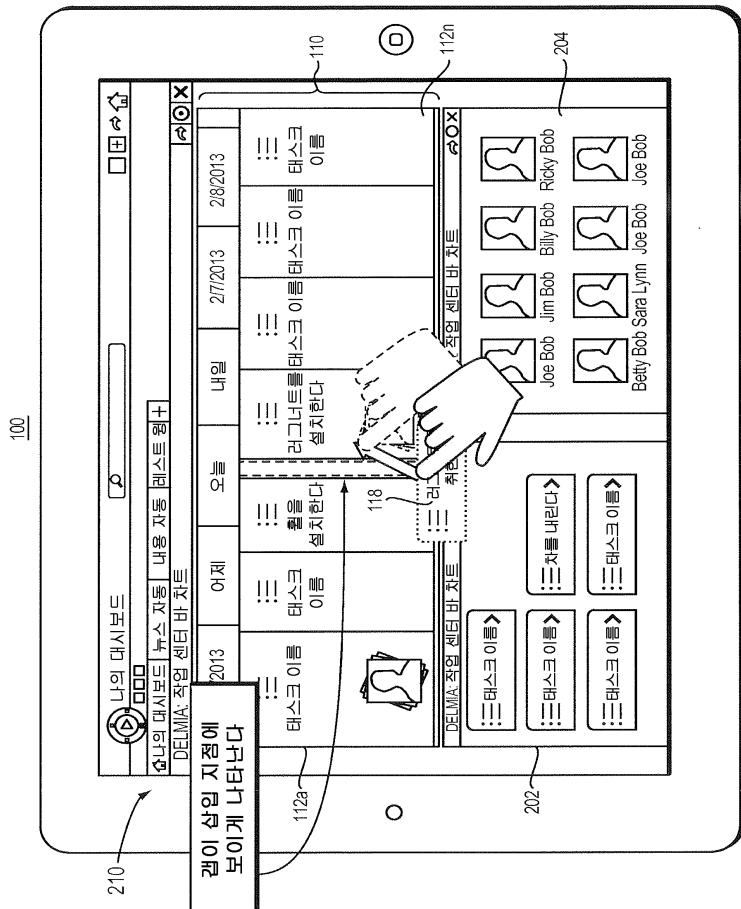
## 도면2a



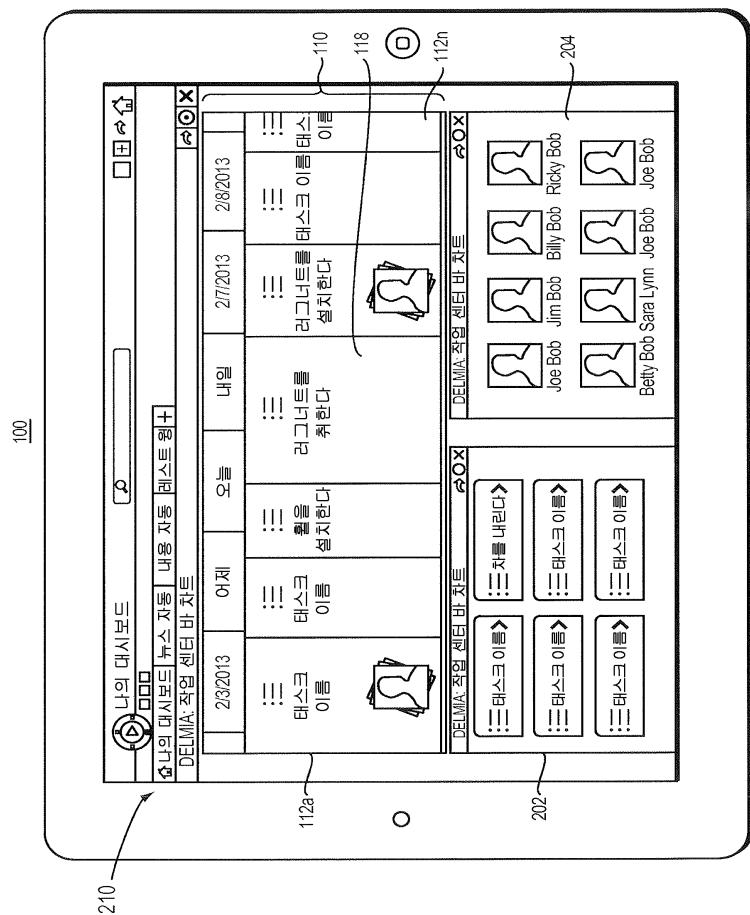
## 도면2b



## 도면2c

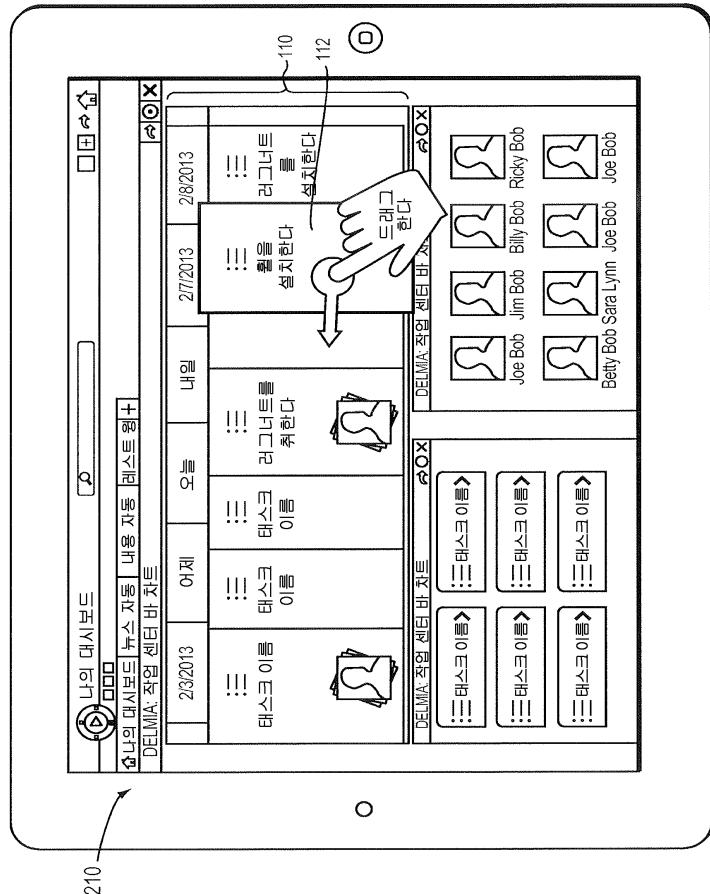


## 도면2d



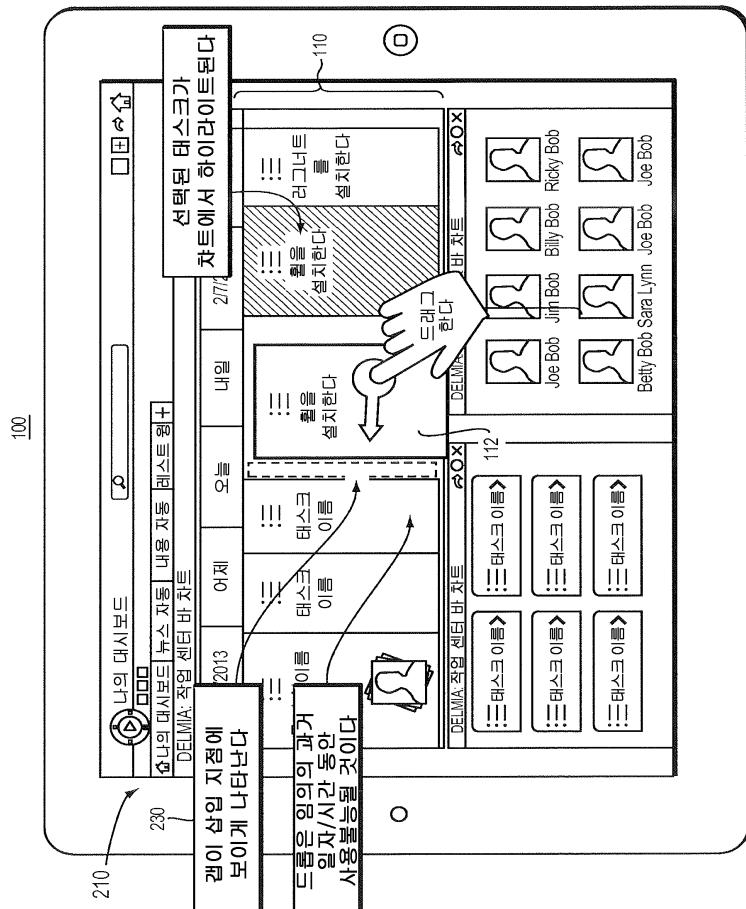
## 도면3a

100

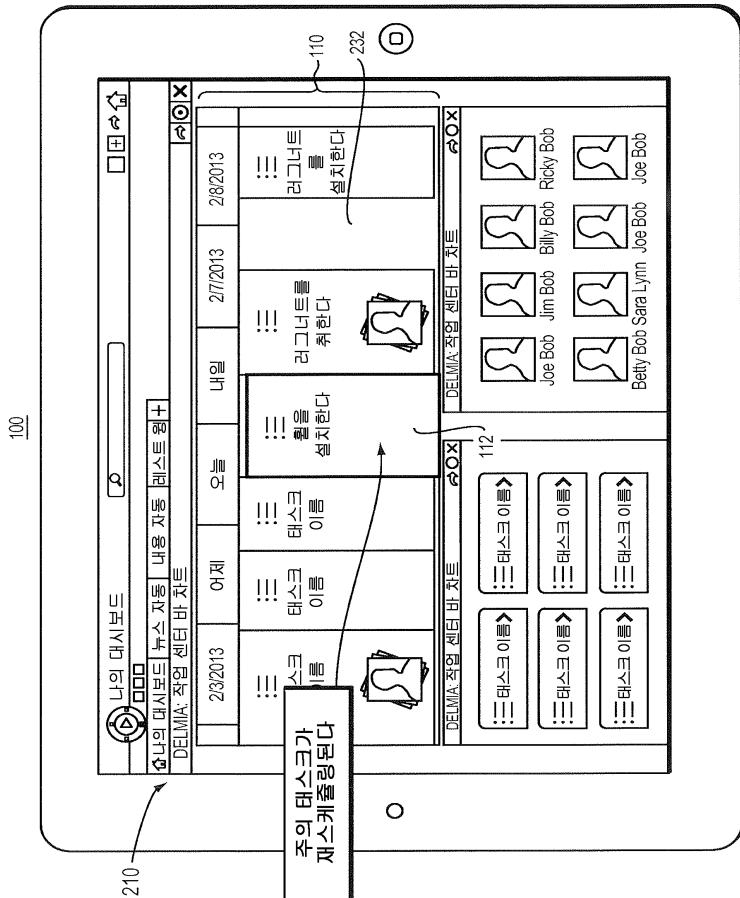


210

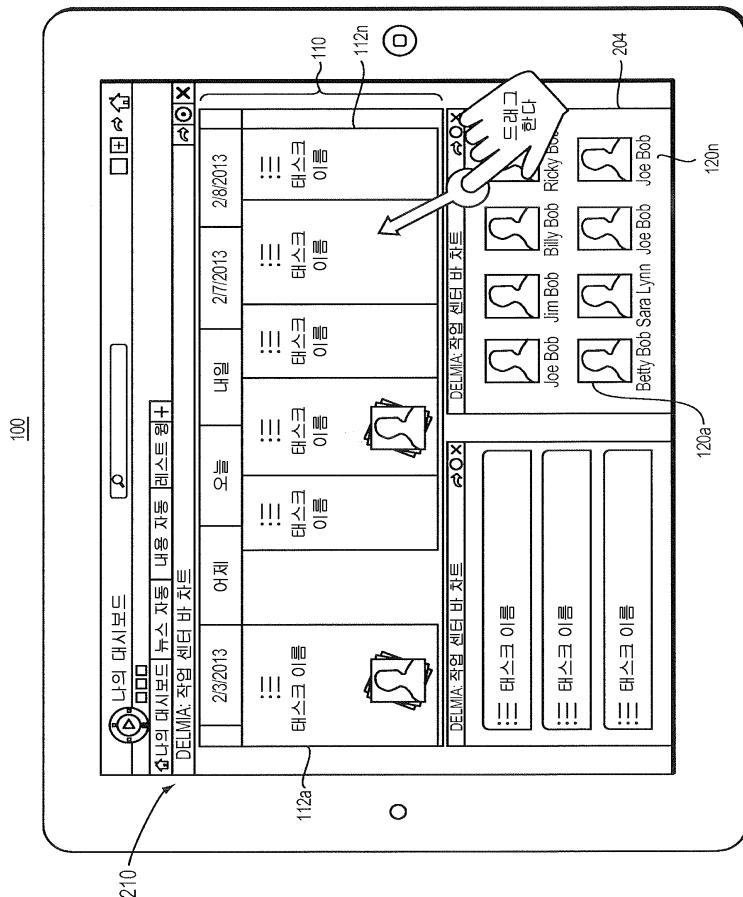
### 도면3b



## 도면3c

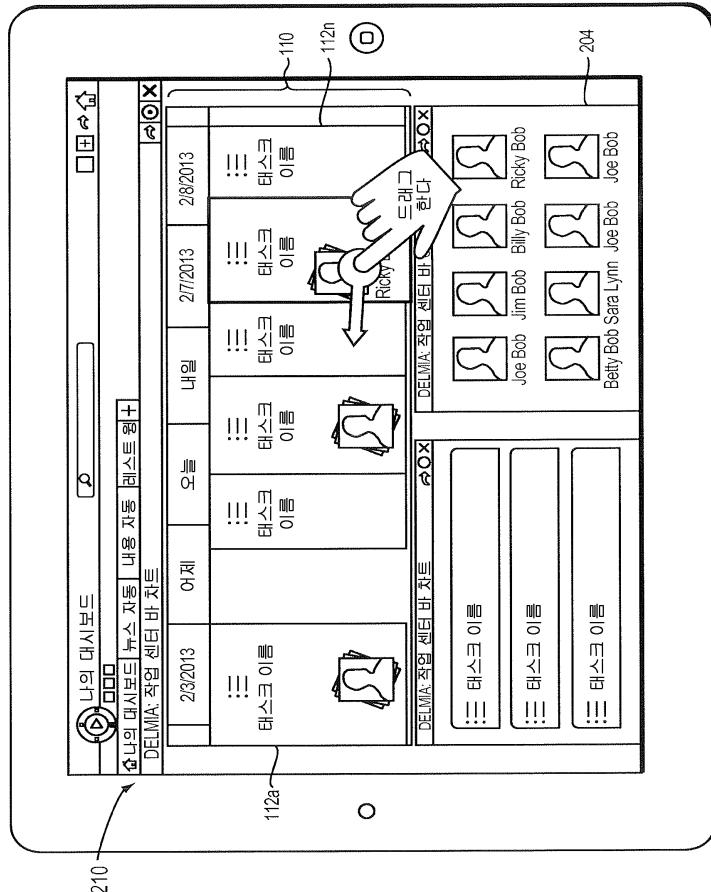


## 도면4a



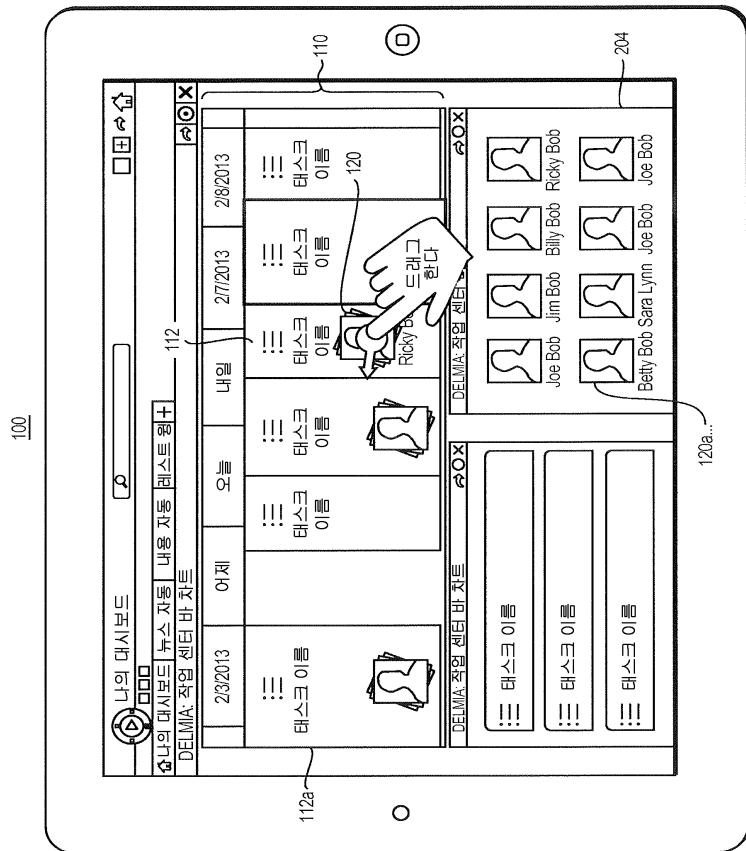
## 도면4b

100

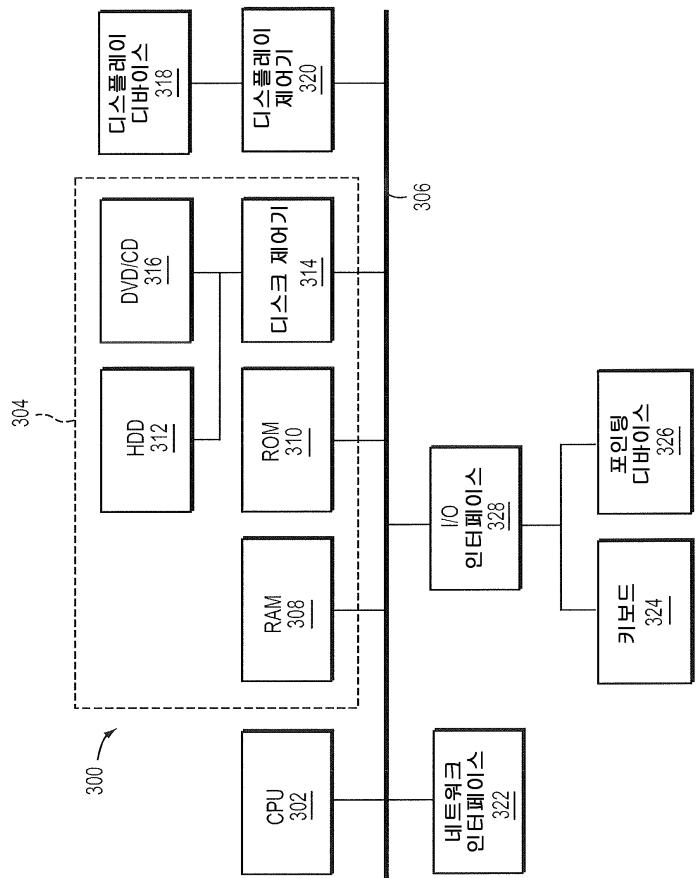


210

### 도면4c



도면5



도면6

