



등록특허 10-2693229



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2024년08월07일

(11) 등록번호 10-2693229

(24) 등록일자 2024년08월05일

(51) 국제특허분류(Int. Cl.)

G06F 3/04842 (2022.01) G06F 3/0481 (2022.01)

G06Q 10/06 (2012.01)

(52) CPC특허분류

G06F 3/04842 (2022.01)

G06F 3/0481 (2013.01)

(21) 출원번호 10-2023-7004145(분할)

(22) 출원일자(국제) 2015년12월07일

심사청구일자 2023년02월03일

(85) 번역문제출일자 2023년02월03일

(65) 공개번호 10-2023-0022272

(43) 공개일자 2023년02월14일

(62) 원출원 특허 10-2017-7016482

원출원일자(국제) 2015년12월07일

심사청구일자 2020년11월10일

(86) 국제출원번호 PCT/US2015/064163

(87) 국제공개번호 WO 2016/099965

국제공개일자 2016년06월23일

(30) 우선권주장

14/572,745 2014년12월16일 미국(US)

(56) 선행기술조사문헌

JP11096024 A*

(뒷면에 계속)

전체 청구항 수 : 총 20 항

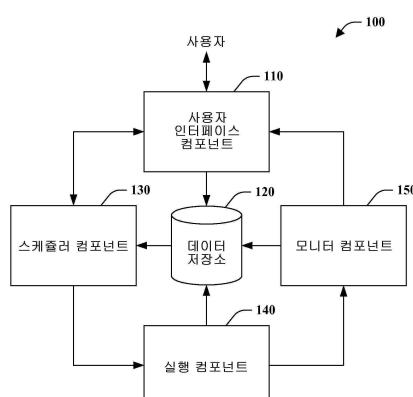
심사관 : 이후락

(54) 발명의 명칭 작업 스케줄링 및 모니터링 기법

(57) 요약

데이터 변환 작업의 뷰는 사용자 인터페이스에 의해 제시될 수 있다. 후속하여 작업 사이의 의존성에 기초하여 작업이 선택된 이후에 관련 작업이 자동으로 식별될 수 있다. 실행 상태가 또한 결정되고 제시되어서 예를 들어, 작업의 성공적인 실행 및 실패한 실행이 구분될 수 있다. 또한, 성공적으로 실행하는데 실패한 작업 실행의 선택은 실패한 또는 성공적으로 실행하는데 실패할 것이라고 예측되는 관련 작업 실행의 식별을 트리거할 수 있다.

대표도 - 도1



(52) CPC특허분류

G06Q 10/06311 (2013.01)

(72) 발명자

스톰 크리스티나

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

네츠 아미르

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

체옹 치우 잉

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

플라스코 마이클 제이

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

그리얼리쉬 케빈

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

엘라-리베라 지오바니 엠

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

칼슨 소니아 피

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

해닝거 마크 더블유

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

바흐 폴라 엠

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

네틀레톤 데이비드 제이

미국 위싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 어텐션: 페이던트 그룹 도켓팅 (빌딩 8/1000) 마이크로소프트 테크놀로지 라이센싱, 엘 엘씨

(56) 선행기술조사문현

WO2010001555 A1*

JP2006243996 A*

JP2009230584 A

WO2013122842 A1

US20130290974 A1

US20110264711 A1

KR1020060087999 A

KR1020140113310 A

*는 심사관에 의하여 인용된 문현

명세서

청구범위

청구항 1

며신 실행가능한 명령어를 저장하는 메모리에 결합된 적어도 하나의 프로세서를 포함하는 시스템에 의해 수행되는 방법으로서,

상기 며신 실행가능한 명령어는, 상기 프로세서에 의해 실행될 때, 상기 시스템이 단계들을 수행하게 제어하되, 상기 단계들은,

디스플레이 디바이스 상에 표시하기 위해, 데이터 세트를 포함하는 연속체(continuum)의 데이터의 조각들을 포함하는 하나 이상의 데이터 슬라이스에 관한 데이터 변환 작업 스케줄의 타임라인 뷰를 전달하는 단계 - 각각의 데이터 슬라이스는 데이터 변환 작업에 의해 생성 혹은 소비되고, 길이와 시작 시간 및 종료 시간을 포함함 - 와,

사용자로부터 상기 뷰 내에서 상기 하나 이상의 데이터 슬라이스 중에서 데이터 슬라이스를 선택하는 제1 신호를 수신하는 단계와,

데이터 의존성에 기초하여 상기 사용자가 선택한 데이터 슬라이스에 대한 하나 이상의 관련된 데이터 슬라이스를 자동으로 결정하는 단계 - 상기 하나 이상의 관련된 데이터 슬라이스는 상기 사용자가 선택한 데이터 슬라이스에 의존하는 하나 이상의 데이터 슬라이스 또는 상기 사용자가 선택한 데이터 슬라이스가 의존하는 하나 이상의 데이터 슬라이스 중 적어도 하나를 포함함 - 와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 타임라인 뷰 내에서 하나 이상의 다른 데이터 슬라이스와 시각적으로 구별되는 상기 사용자가 선택한 데이터 슬라이스 및 상기 하나 이상의 관련된 데이터 슬라이스의 시각화를 전달하는 단계

를 포함하는,

방법.

청구항 2

제 1 항에 있어서,

상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 작업에 의해 생성하기로 스케줄링된 하나 이상의 데이터 슬라이스를 전달하는 단계를 더 포함하는,

방법.

청구항 3

제 1 항에 있어서,

상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 작업에 의해 이전에 생성된 하나 이상의 데이터 슬라이스를 전달하는 단계를 더 포함하는,

방법.

청구항 4

제 1 항에 있어서,

상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 작업에 의해 생성하기로 스케줄링된 데이터 슬라이스와 시각적으로 구별되는 이전에 생성된 데이터 슬라이스를 전달하는 단계를 더 포함하는,
방법.

청구항 5

제 1 항에 있어서,
상기 단계들은,
하나 이상의 데이터 슬라이스를 생성하는 동안 계산 부하를 결정하는 단계와,
상기 디스플레이 디바이스 상에 표시하기 위해, 상기 하나 이상의 데이터 슬라이스에 맞춰진 상기 계산
부하를 전달하는 단계
를 더 포함하는,
방법.

청구항 6

제 1 항에 있어서,
상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 적어도 하나의 데이터 슬라이스가 성공적으로
생성되는 데에 실패했다는 표시를 전달하는 단계를 더 포함하는,
방법.

청구항 7

제 6 항에 있어서,
상기 단계들은,
성공적으로 생성하는 데 실패한 적어도 하나의 데이터 슬라이스 중 하나를 선택하는 제2 신호를 수신하
는 단계와,
선택한 데이터 슬라이스와 관련된 하나 이상의 관련된 데이터 슬라이스를 자동적으로 결정하는 단계 -
상기 관련된 데이터 슬라이스는, 성공적으로 생성하는데 실패한 상기 선택한 데이터 슬라이스가 의존하는 하나
이상의 데이터 슬라이스 또는 상기 선택한 데이터 슬라이스에 대한 의존성에 기초하여 성공적으로 생성하는데
실패할 것으로 예측되는 하나 이상의 데이터 슬라이스 중 적어도 하나를 포함함 - 와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 선택한 데이터 슬라이스 및 상기 하나 이상의 관
련된 데이터 슬라이스의 강조된 버전을 전달하는 단계
를 더 포함하는,
방법.

청구항 8

제 6 항에 있어서,
상기 단계들은,
성공적으로 생성하는 데 실패한 상기 적어도 하나의 데이터 슬라이스 중 하나를 선택하는 제2 신호를

수신하는 단계와,

상기 제2 신호에 응답하여 상기 적어도 하나의 데이터 슬라이스의 생성을 재스케줄링하는 단계를 더 포함하는

방법.

청구항 9

제 1 항에 있어서,

상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 작업 및 데이터 세트의 표현 뿐만 아니라 상기 작업과 데이터 세트 사이의 연결을 포함하는 작업 저작 다이어그램(job authoring diagram)에서 상기 선택한 데이터 슬라이스 및 상기 하나 이상의 관련된 데이터 슬라이스를, 전달하는 단계를 더 포함하는

방법.

청구항 10

컴퓨터 시스템으로서,

메모리에 결합된 프로세서를 포함하고,

상기 프로세서는 상기 메모리에 저장된 머신 실행가능한 명령어를 실행하도록 구성되며,

상기 머신 실행가능한 명령어는 실행될 때 단계들을 수행하되, 상기 단계들은,

디스플레이 디바이스 상에 표시하기 위해, 데이터 세트를 포함하는 연속체의 데이터의 조각을 포함하는 하나 이상의 데이터 슬라이스에 관한 데이터 변환 작업 스케줄의 타임라인 뷰를 전달하는 단계 - 각각의 데이터 슬라이스는 데이터 변환 작업에 의해 생성 혹은 소비되고, 길이와 시작 시간 및 종료 시간을 포함함 - 와,

사용자로부터 상기 뷰 내에서 상기 하나 이상의 데이터 슬라이스 중에서 데이터 슬라이스를 선택하는 제1 신호를 수신하는 단계와,

데이터 의존성에 기초하여 상기 사용자가 선택한 데이터 슬라이스에 대한 하나 이상의 관련된 데이터 슬라이스를 자동으로 결정하는 단계 - 상기 하나 이상의 관련된 데이터 슬라이스는 상기 사용자가 선택한 데이터 슬라이스에 의존하는 하나 이상의 데이터 슬라이스 또는 상기 사용자가 선택한 데이터 슬라이스가 의존하는 하나 이상의 데이터 슬라이스 중 적어도 하나를 포함함 - 와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 뷰 내에서 하나 이상의 다른 데이터 슬라이스와 시각적으로 구별되는 상기 사용자가 선택한 데이터 슬라이스 및 상기 하나 이상의 관련된 데이터 슬라이스의 시각화를 전달하는 단계

를 포함하는,

컴퓨터 시스템.

청구항 11

제 10 항에 있어서,

상기 단계들은,

데이터 슬라이스 생성과 함께 컴퓨터 리소스 활용을 모니터링하는 단계와,

상기 디스플레이 디바이스 상에 표시하기 위해, 하나 이상의 개별적인 데이터 슬라이스에 맞춰진 상기 리소스 활용의 시각화를 전달하는 단계

를 더 포함하는,

컴퓨터 시스템.

청구항 12

제 10 항에 있어서,
상기 단계들은,
데이터 슬라이스 생성의 성공 또는 실패를 판정하는 단계와,
성공적인 데이터 슬라이스 생성과는 상이한 실패한 데이터 슬라이스 생성을 제시하는 단계
를 더 포함하는,
컴퓨터 시스템.

청구항 13

제 10 항에 있어서,
상기 단계들은, 실패한 데이터 슬라이스 생성과 관련해서 수신한 신호에 기초해서 데이터 슬라이스의 생성을 재
스케줄링하는 단계를 더 포함하는,
컴퓨터 시스템.

청구항 14

제 10 항에 있어서,
상기 단계들은,
생성하는 데 실패한 적어도 하나의 데이터 슬라이스 중 하나를 선택하는 제2 신호를 검출하는 단계와,
선택한 데이터 슬라이스와 관련되고, 상기 선택한 데이터 슬라이스에 대한 의존성에 기초하여 실패할
것으로 예측되는, 하나 이상의 관련된 데이터 슬라이스를 자동적으로 결정하는 단계와,
상기 디스플레이 디바이스 상에 표시하기 위해, 상기 선택한 데이터 슬라이스 및 상기 하나 이상의 관
련된 데이터 슬라이스의 강조된 버전을 전달하는 단계
를 더 포함하는,
컴퓨터 시스템.

청구항 15

제 10 항에 있어서,
상기 단계들은, 상기 디스플레이 디바이스 상에 표시하기 위해, 작업 및 데이터 세트의 표현 뿐만 아니라 상기
작업과 데이터 세트 사이의 연결을 포함하는 작업 저작 다이어그램에서 상기 선택한 데이터 슬라이스 및 상기
하나 이상의 관련된 데이터 슬라이스를, 전달하는 단계를 더 포함하는,
컴퓨터 시스템.

청구항 16

명령어가 저장된 컴퓨터 관독가능 비일시적 저장 매체로서,

상기 명령어는 실행될 때 적어도 하나의 프로세서로 하여금 방법을 수행하게 하고,

상기 방법은,

디스플레이 디바이스 상에 표시하기 위해, 데이터 세트를 포함하는 연속체의 데이터의 조각을 포함하는 하나 이상의 데이터 슬라이스에 관한 데이터 변환 작업 스케줄의 타임라인 뷰를 전달하는 단계 - 각각의 데이터 슬라이스는 데이터 변환 작업에 의해 생성 혹은 소비되고, 길이와 시작 시간 및 종료 시간을 포함함 - 와,

사용자로부터 상기 뷰 내에서 상기 하나 이상의 데이터 슬라이스 중에서 데이터 슬라이스를 선택하는 제1 신호를 수신하는 단계와,

데이터 의존성에 기초하여 상기 사용자가 선택한 데이터 슬라이스에 대한 하나 이상의 관련된 데이터 슬라이스를 자동으로 결정하는 단계 - 상기 하나 이상의 관련된 데이터 슬라이스는 상기 사용자가 선택한 데이터 슬라이스에 의존하는 하나 이상의 데이터 슬라이스 또는 상기 사용자가 선택한 데이터 슬라이스가 의존하는 하나 이상의 데이터 슬라이스 중 적어도 하나를 포함함 - 와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 뷰 내에서 하나 이상의 다른 데이터 슬라이스와 시각적으로 구별되는 상기 사용자가 선택한 데이터 슬라이스 및 상기 하나 이상의 관련된 데이터 슬라이스의 시각화를 전달하는 단계

를 포함하는,

컴퓨터 관독가능 비일시적 저장 매체.

청구항 17

제 16 항에 있어서,

상기 방법은, 상기 디스플레이 디바이스 상에 표시하기 위해, 생성하기로 스케줄링된 데이터 슬라이스와 시각적으로 구별되는 이전에 생성된 데이터 슬라이스를 전달하는 단계를 더 포함하는,

컴퓨터 관독가능 비일시적 저장 매체.

청구항 18

제 16 항에 있어서,

상기 방법은,

상기 하나 이상의 데이터 슬라이스를 생성하는 동안 계산 부하를 결정하는 단계와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 하나 이상의 데이터 슬라이스에 맞춰진 상기 계산 부하를 전달하는 단계

를 더 포함하는,

컴퓨터 관독가능 비일시적 저장 매체.

청구항 19

제 16 항에 있어서,

상기 방법은, 상기 디스플레이 디바이스 상에 표시하기 위해, 적어도 하나의 데이터 슬라이스가 성공적으로 생성하는 데에 실패했다는 표시를 전달하는 단계를 더 포함하는,

컴퓨터 관독가능 비일시적 저장 매체.

청구항 20

제 16 항에 있어서,

상기 방법은,

성공적으로 생성하는 데 실패한 적어도 하나의 데이터 슬라이스 중 하나를 선택하는 제2 신호를 검출하는 단계 와,

선택한 데이터 슬라이스와 관련된 하나 이상의 관련된 데이터 슬라이스를 자동적으로 결정하는 단계 - 상기 관련된 데이터 슬라이스는, 성공적으로 생성하는데 실패한 상기 선택한 데이터 슬라이스가 의존하는 하나 이상의 데이터 슬라이스 또는 상기 선택한 데이터 슬라이스에 대한 의존성에 기초하여 성공적으로 생성하는데 실패할 것으로 예측되는 하나 이상의 데이터 슬라이스 중 적어도 하나를 포함함 - 와,

상기 디스플레이 디바이스 상에 표시하기 위해, 상기 선택한 데이터 슬라이스 및 상기 하나 이상의 관련된 데이터 슬라이스의 강조된 버전을 전달하는 단계

를 더 포함하는,

컴퓨터 판독가능 비일시적 저장 매체.

발명의 설명

배경기술

[0001] 가치있는 식견을 수집하기 위한 방대한 양의 데이터, 또는 소위 빅 데이터를 프로세싱하는 것은 먼저 데이터를 변환하는 것을 포함한다. 데이터는 하나 이상의 작업을 생성하고, 스케줄링하고 실행함으로써 대쉬보드와 같은, 비지니스 인텔리전스 엔드포인트에 의한 공개 또는 소비를 위해 사용가능한 형태로 변환된다. 이러한 상황에서, 하나의 작업(job)은 하나 이상의 변환 동작을 포함하는 데이터에 대한 작업 단위(unit of work)이다. 통상적으로, 작업은 데이터 개발자, 데이터 설계자, 비지니스 인텔리전스 설계자 등에 의해 수동으로 코딩된다. 후속하여, 작업이 스케줄링되고 실행될 수 있다.

발명의 내용

[0002] 이하는 개시된 청구 대상의 일부 측면의 기본적인 이해를 제공하기 위해 단순한 요약을 제시한다. 본 요약은 광범위한 개요가 아니다. 주요/중요한 구성요소를 식별하거나 청구된 청구 대상의 범위를 기술하기 위한 것이 아니다. 유일한 목적은 나중에 제시되는 더 자세한 설명의 서막으로서 단순한 형태로 일부 개념을 제시하기 위한 것이다.

[0003] 간략하게 설명하면, 본 개시는 작업 스케줄링 및 모니터링에 관한 것이다. 사용자가 작업 스케줄 및 실행을 보고 인터랙팅하는 것을 가능하게 하도록 생성된 시각화 및 실행을 위해 작업이 스케줄링될 수 있다. 하나의 측면에 따르면, 작업의 선택에 응답하여, 하나 이상의 관련된 작업이 데이터 의존성에 기초하여 자동으로 결정될 수 있다. 후속하여, 선택된 작업 뿐만 아니라 관련된 작업이 강조될 수 있다. 실행 상태가 또한 결정되고 제시되어서 예를 들어, 작업의 성공적인 실행 및 실패한 실행이 구분될 수 있다. 또한, 실패한 작업 수행의 선택에 응답하여, 하나 이상의 관련된 실패한 작업 수행 또는 실패할 것으로 예측되는 작업 수행이 식별될 수 있다. 후속하여 선택된 실패한 작업 수행 및 관련된 실패한 작업 수행 또는 실패할 것으로 예측되는 작업 수행이 강조될 수 있다.

[0004] 상술한 관련된 목적을 달성하기 위해, 청구된 청구 대상의 특정 구체적인 측면이 다음의 상세한 설명 및 첨부된 도면과 관련하여 설명된다. 이들 측면은 청구 대상이 실시될 수 있는 다양한 방식을 나타내고, 이들 모두는 청구된 청구 대상의 범위 내에 속하는 것이다. 다른 장점 및 신규한 특징들은 도면과 함께 고려될 때 다음의 상세한 설명으로부터 명백해질 수 있다.

도면의 간단한 설명

[0005] 도 1은 작업 시스템의 블록도이다.

도 2는 대표적인 사용자 인터페이스 컴포넌트의 블록도이다.

도 3은 대표적인 스케줄 컴포넌트의 블록도이다.

도 4는 대표적인 스케줄러 컴포넌트의 블록도이다.

도 5는 대표적인 모니터 컴포넌트의 블록도이다.

도 6은 작업 구동형 스케줄링 및 모니터링을 위한 인터페이스의 예시적인 스크린샷이다.

도 7은 관련된 작업을 구분하는 인터페이스의 예시적인 스크린샷이다.

도 8은 관련된 작업 실패를 구분하는 인터페이스의 예시적인 스크린샷이다.

도 9는 다이어그램 뷰를 생성하고 관련된 작업 및 데이터를 구분하는 인터페이스의 예시적인 스크린샷이다.

도 10은 데이터 구동형 스케줄링 및 모니터링을 위한 인터페이스의 예시적인 스크린샷이다.

도 11은 작업 시스템의 블록도이다.

도 12는 작업을 스케줄링 및 모니터링하는 방법의 플로우 차트 다이어그램이다.

도 13은 관련된 작업을 구분하는 방법의 플로우 차트 다이어그램이다.

도 14는 성공적인 작업 실행 및 실패한 작업 실행을 구분하는 방법의 플로우 차트 다이어그램이다.

도 15는 스케줄링 체인(scheduling chains) 트러블슈팅(troubleshooting)을 제공하는 방법의 플로우 차트 다이어그램이다.

도 16은 관련된 작업 및 데이터 세트를 구분하는 플로우 차트 다이어그램이다.

도 17은 작업의 데이터 구동형 프로세싱 방법의 플로우 차트 다이어그램이다.

도 18은 본 개시의 측면에 대해 적합한 운영 환경을 도시하는 개략적인 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0006]

이하의 상세한 설명은 일반적으로 작업 스케줄링 및 모니터링에 관한 것이다. 하나의 작업은 적어도 하나의 데이터 변환 동작에 대응한다. 예를 들어, 작업의 실행을 스케줄링하는 것 뿐만 아니라 모니터링하는 것과 함께 하나 이상의 작업이 인터페이스에서 디스플레이 상에 제시될 수 있다. 예를 들어, 작업이 실행될 때, 작업을 위한 실행 시간의 길이를 나타내는 타임라인 뷰 상에 작업이 제시될 수 있다. 관련된 작업이 작업 사이의 데이터의 존성에 기초하여 자동으로 식별되고 사용자에게 제시된 이후에, 작업이 선택될 수 있다. 관련된 작업은 선택된 작업에 의존하는 작업 및/또는 선택된 작업이 의존하는 작업을 포함할 수 있다. 이는 사용자가 효율적으로 작업 스케줄링에 관한 정보를 얻고 인터랙팅하는 것을 돋는다. 데이터의 존성 정보 이외에, 성공적으로 실행되거나 실행할 것으로 예측되는 작업이 실패하거나 실패할 것으로 예측되는 작업과 구분될 수 있도록 실행 상태가 캡처될 수 있다. 또한, 관련된 작업 수행이 실패하거나 실패할 것으로 예측된 이후에 실패한 작업 수행이 선택될 수 있고, 데이터의 존성, 상태, 및 스케줄 정보에 기초하여 자동으로 식별될 수 있다. 후속하여, 선택된 실패한 작업 수행, 관련된 실패한 작업 수행, 또는 실패한 것으로 예측되는 작업 수행이 강조될 수 있다. 그 결과로, 스케줄링 체인을 트러블슈팅하는 편리한 방식이 제공된다. 또한, 추가적인 뷰가 적어도 의존성 데이터를 이용할 수 있다. 예를 들어, 작업 또는 작업 파이프라인의 다이어그램은 의존성 정보를 활용하여 작업을 포함하는 작업 계보(lineage) 및 작업 및 선택 작업에 의해 활용되는 데이터 세트 및 선택 작업에 의존하는 데이터 소스를 식별할 수 있다. 여러가지 측면들은 적어도 작업 실행을 효율적으로 스케줄링하고 모니터링하는 것을 도울 뿐만 아니라 예러를 감소시킬 수 있다.

[0007]

본 개시의 다양한 측면들이 이제 첨부된 도면과 관련하여 더 자세하게 설명되고, 유사한 번호는 일반적으로 전반에서 유사한 또는 대응하는 구성요소를 지칭한다. 그러나, 도면 및 이에 관련된 상세한 설명은 개시된 특정 형태로 청구된 청구 대상을 제한하기 위한 것이 아님을 이해할 수 있다. 오히려, 청구된 청구 대상의 사상 및 범위 내에 속하는 모든 수정물, 등가물, 및 대체물을 커버하기 위한 의도이다.

[0008]

처음에 도 1을 참조하면, 작업 시스템(100)이 도시된다. 작업 시스템(100)은 데이터 변환 동작을 포함하는 작업의 스케줄링, 실행, 및 모니터링을 위한 수단을 제공한다. 작업 시스템은 사용자 인터페이스 컴포넌트(110), 데이터 저장소(120), 스케줄러 컴포넌트(130), 실행 컴포넌트(140), 및 모니터 컴포넌트(150)를 포함한다. 사용자 인터페이스 컴포넌트(110)는 사용자가 작업을 보고, 특정하고, 제어하는 것을 가능하게 하도록 구성된다. 하나

의 경우에, 사용자 인터페이스(110)는 작업 또는 작업 파이프라인을 정의하는 메커니즘을 제공하도록 구성된다. 이 정의의 일부로서, 작업이 의존하는 하나 이상의 데이터 세트 및 작업에 의해 생성되는 출력 데이터 세트를 포함하는 하나 이상의 관련성이 특정될 수 있다. 데이터 의존성 정보를 포함하는 작업은 컴퓨터 관통 가능한 저장 매체인 데이터 저장소(120)에 저장될 수 있다. 스케줄러 컴포넌트(130)는 작업 및 작업 사이의 관련성과 관련하여 특정된 스케줄에 관한 실행을 위해 작업을 스케줄링하도록 구성된다. 스케줄은 디스플레이를 위해 사용자 인터페이스에 제공될 수 있고, 사용자는 스케줄과 인터랙팅하여 특정 정보를 얻고 선택적으로 스케줄을 수정할 수 있다. 스케줄러 컴포넌트(130)는 실행 컴포넌트(140)에 의한 작업 프로세싱을 개시한다. 실행 컴포넌트(140)는 작업이 컴퓨터 상에서 실행되는 것을 가능하게 하는 소프트웨어 및 하드웨어 리소스의 수집을 포함할 수 있다. 작업 실행의 결과는 데이터 저장소(120)에 저장될 수 있다. 또한, 모니터 컴포넌트(150)는 실행 컴포넌트(140)에 의한 작업 실행을 모니터링할 수 있다. 예를 들어, 모니터 컴포넌트(150)는 실행이 실패하거나 성공적인 경우와 같이 작업 상태를 식별할 수 있다. 추가적으로, 모니터 컴포넌트(150)는 작업을 프로세싱하는 것과 관련하여 계산 리소스를 활용할 수 있다. 모니터 컴포넌트(150)에 의해 얻은 데이터 또는 정보는 데이터 저장소(120)에 저장될 수 있고 표시를 위한 사용자 인터페이스 컴포넌트(110)에 대해 사용가능하게 만들어질 수 있다.

[0009] 도 2는 대표적인 사용자 인터페이스 컴포넌트(110)를 더 자세하게 도시한다. 사용자 인터페이스 컴포넌트(110)는 저작(author) 컴포넌트(210) 및 스케줄 컴포넌트(220)를 포함한다. 저작 컴포넌트(210)는 두번재 작업으로 입력을 선택적으로 제공하는 첫번재 작업의 출력을 갖는 하나 이상의 관련된 작업의 세트를 포함하는 작업 및 파이프라인들을 사용자가 특정하는 것을 가능하게 하도록 구성된다. 실시예에 따르면, 저작 컴포넌트(210)는 작업 및 파이프라인들을 저작하기 위해 인터랙티브 비주얼 워크플레이스 또는 캔버스를 제공할 수 있다. 예를 들어, 데이터 세트는 원기둥으로 표현되고 데이터 세트를 소비하고 수정된 데이터 세트를 생성하는 작업을 나타내는 정육면체로의 화살표에 의해 연결될 수 있다. 처음에, 사용자는 데이터 세트와 작업 사이의 관련성의 다이어그램을 그릴 수 있다. 이는 관련성을 이해하는 것과 궁극적으로 파이프라인을 특정하는 것과 관련하여 시간을 절약하는 직관적인 경험이 된다. 다른 측면에 따르면, 저작 컴포넌트(210)는 프로그램 코드 또는 하나 이상의 다이얼로그 박스를 수용하여 작업 및 작업 사이의 관련성을 특정하는 코드 편집기로서 구현될 수 있다. 스케줄 컴포넌트(220)는 실행을 위한 작업을 스케줄링하는 것과 실행을 모니터링하는 것에 관한 시작화를 제시하도록 구성된다.

[0010] 도 3을 참조하면, 명확성 및 이해를 제공하기 위해 대표적인 스케줄 컴포넌트(220)가 더 자세하게 도시된다. 스케줄 컴포넌트(220)는 뷰 컴포넌트(310), 업데이트 컴포넌트(320), 관련된 작업 컴포넌트(330), 수정 컴포넌트(340), 및 로그 컴포넌트(350)를 포함한다. 뷰 컴포넌트(310)는 실행을 위해 스케줄링된 작업의 적어도 하나의 뷰를 제시하도록 구성된다. 하나의 측면에 따르면, 뷰는 스케줄링된 실행 및 완료된 실행을 포함하는 실행의 시간에 의해 순서화된 작업의 세트를 시작화하는 타임라인에 대응할 수 있다. 이러한 타임라인은 또한 작업 실행의 길이가 시작화되는 것을 가능하게 할 수 있다.

[0011] 업데이트 컴포넌트(320)는 뷰 컴포넌트(310)에 의해 생성되는 시작화를 업데이트하거나 작업 실행과 관련하여 최신으로 하도록 구성된다. 예를 들어, 업데이트 컴포넌트(320)는 실행되었던 작업을 반영하도록 시작화를 변경할 수 있다. 일 실시예에서, 업데이트 컴포넌트(320)는 뷰 컴포넌트(310)와 인터랙팅하여 스케줄링되었지만 실행되지 않은 작업을 회색으로 색칠하거나 비활성(gray out) 시키는 것에 의해 현재 시간을 나타내는 라인을 포함하고, 라인을 뒤에 놓을 수 있다. 추가로, 업데이트 컴포넌트(320)는 성공적으로 실행된 작업 대 실패하거나 실패할 것으로 예측되는 작업을 식별하고 구분하는데 활용될 수 있다. 예를 들어, 성공적으로 실행된 작업은 녹색이 될 수 있는 반면 실패는 적색으로 표시된다.

[0012] 특정 실시예에 따르면, 뷰 컴포넌트(310) 및 업데이트 컴포넌트(320)는 타임라인 및 계산 리소스 활용의 시작화를 포함하는 분할 뷰를 제시하도록 협업할 수 있다. 제 1 부분에서, 스케줄링된 작업의 타임라인은 상술한 바와 같이 제시될 수 있다. 제 2 부분에서, 예를 들어, 제 1 부분 아래에서, 그래프와 같은 시작화가 제시되어 작업 실행에 맞춰 활용되는 리소스를 나타낼 수 있다. 이 경우에, 업데이트 컴포넌트(320)는 리소스 활용 및 작업 실행에 관한 데이터를 얻고 이에 따라 그래프를 업데이트할 수 있다.

[0013] 관련 작업 컴포넌트(330)는 작업 사이의 관련성에 기초하여 작업을 구분하도록 구성된다. 관련성은 작업과 선택된 작업이 의존하는 또는 선택 작업에 의존하는 데이터를 포함할 수 있다. 하나의 경우에, 선택된 작업은 선택된 작업과 관련된 작업을 결정하는 것 및 뷰 컴포넌트(310)에 의해 제공되는 시작화에서 이들 관련된 작업을 시작적으로 구분하는 것을 개시할 수 있다. 예시로서, 작업을 선택하는 신호 및 시작화에서 이들의 표시를 수신한 이후에, 선택된 작업과 관련된 작업은 데이터 의존성에 기초하여 결정되고 강조될 수 있다. 관련된 작업은 작업

사이의 관련성을 저장한 데이터 저장소(120)로부터 식별될 수 있다. 일 실시예에 따르면, 방향성 그래프는 결절 점(vertices)인 작업 및 작업을 연결하는 방향성 엣지인 데이터 세트를 포함하여 저장될 수 있다. 관련된 작업을 결정하기 위해, 그래프는 선택된 작업을 나타내는 결절점을 앞뒤로 가로지를 수 있고, 앞으로 가로지르는 것(예를 들어, 선택된 작업으로부터 다운스트림)은 선택된 작업에 의존하는 데이터 세트 및 작업을 캡쳐하고, 뒤로 가로지르는 것(예를 들어, 선택된 작업으로부터 업스트림)은 선택된 작업이 의존하는 데이터 세트 및 작업을 식별한다. 작업 실행 또는 작업의 인스턴스는 또한 특성 또는 속성에 기초하여 구분될 수 있다. 예를 들어, 사용자가 성공적으로 실행하는데 실패한 작업 수행을 선택하면, 실패하거나 실패하는 것으로 예측되는 선택 작업 수행과 관련된 다른 작업 수행이 자동으로 식별되어 뷰에서 강조될 수 있다. 이러한 작업 수행은 작업 사이의 데이터 의존성에 관한 정보, 작업과 연관된 상태(예를 들어, 실패한, 성공한...), 및 실행 스케줄에 기초하여 식별될 수 있다. 여기서, 작업 수행이 성공적으로 실행하는데 실패한 작업 수행에 의존한다면 실패한 것으로서 예측될 수 있다.

[0014] 수정 컴포넌트(340)는 작업 스케줄링의 수정을 가능하게 하도록 구성된다. 수정 컴포넌트(340)는 뷰 컴포넌트(310)와 함께 작업하여 작업 실행 스케줄링의 수정에 관한 사용자 입력을 얻도록 구성된다. 예를 들어, 수정 컴포넌트(340)가 제시된 인터랙티브 뷰와 관련하여 사용자로부터 신호로서 수신된 하나 이상의 제스처에 기초하여 실행을 위한 작업의 리스케줄링을 적어도 초기화할 수 있다. 하나의 비제한적인 예시에 따르면, 사용자는 우클릭 또는 드래그 앤 드롭과 같은 일부 추가적인 제스처를 성공적으로 실행하고/거나 수행하는데 실패한 작업의 인스턴스를 선택하여 실행을 재스케줄링할 수 있다.

[0015] 로그 컴포넌트(350)는 실행 로그의 획득 및 디스플레이를 가능하게 하도록 구성된다. 일 실시예에 따르면, 로그 컴포넌트(350)는 오버레이된 검색 판 또는 판넬(overlaid search pane or panel)에 의해 검색 메커니즘을 제공할 수 있다. 이 시나리오에서, 사용자는 검색을 특정하고 제출하여 실행 로그와 관련한 결과를 수신할 수 있다. 다른 실시예에 따르면, 로그 컴포넌트(350)는 예를 들어, 특정 작업 수행의 선택에 기초하여 자동으로 질의를 생성하고 제출할 수 있다. 로그 파일 획득과 연관된 특정 제스처에 응답하여, 선택된 작업 수행에 대응하는 결과가 리턴될 수 있다. 이 방식으로, 사용자는 예를 들어, 시작점으로의 실패의 통지를 즉시 얻을 수 있다. 이 경우에, 실행 로그를 액세스 가능하게 만드는 것은 실패의 해결과 같은 트러블슈팅을 제공한다.

[0016] 도 4는 하나의 특정 구현예에 따른 대표적인 스케줄러 컴포넌트(130)를 도시한다. 스케줄러 컴포넌트(130)는 출력 슬라이스 선택 컴포넌트(410), 의존성 기간 결정 컴포넌트(420), 의존성 평가 컴포넌트(430), 및 실행 개시 컴포넌트(440)를 포함한다. 이 상황에서, 데이터 세트는 데이터 프로세싱 동작을 캡쳐하는 작업 또는 동작에 의해 생성되거나 소비되고 시간 축 상의 데이터의 집합에 대응한다. 특히, 데이터 세트는 데이터 슬라이스들로 지정되는 데이터 조각들을 기간과 상관시킨다. 다르게 말하면, 데이터 세트는 데이터 슬라이스의 연속체로 구성된다. 각각의 데이터 슬라이스는 시작 시간 및 종료 시간 뿐만 아니라 특정 슬라이스 길이를 가질 수 있다.

[0017] 출력 슬라이스 선택 컴포넌트(410)는 생성할 데이터 슬라이스를 결정하도록 구성된다. 결정은 선택적인 정책 뿐만 아니라 동작 기간에서의 상태에 기초한다. 각각의 데이터 슬라이스는 대응하는 데이터의 이용가능성을 나타내는 상태를 가질 수 있다. 상태는 아직 생성되지 않은 데이터를 의미하는 "실행 보류", 슬라이스가 생성되는 것을 나타내는 "진행 중", 데이터가 소비를 위해 준비된 것을 나타내는 "준비", 및 데이터를 생성하려는 하나 이상의 시도가 실패했다는 것을 표시하는 "실패한 실행"을 포함하는 다수의 값을 취할 수 있다. 출력 슬라이스 선택 컴포넌트(410)는 작업이 출력을 생성할 시간 윈도우를 특정하는 활성 기간 내에서 "실행 보류" 상태로 데이터 슬라이스를 식별한다. 이 데이터 슬라이스는 상태 및 활성 기간에 의해 필터링되는 데이터 슬라이스에 대한 질의를 실행함으로써 식별될 수 있다. 또한, 출력 슬라이스 선택 컴포넌트(410)는 또한 정책에 기초하여 슬라이스 실행, 또는 데이터 슬라이스의 생성을 순서화할 수 있다. 실행 보류 상태의 슬라이스는 벽시계 시간(wall clock time)과 비교되어 일부 순서로 꾸며질 수 있다. 예를 들어, 정책은 예를 들어, 상태가 "실행 보류"로 설정될 때에 기초하여, 가장 오래된 슬라이스가 먼저 생성되거나 가장 최근 슬라이스가 먼저 생성되는 것을 나타낼 수 있다. 결국, 출력 슬라이스 선택 컴포넌트(410)는 생성될 단일 슬라이스, 출력 슬라이스를 식별한다.

[0018] 의존성 기간 결정 컴포넌트(420)는 의존성 기간을 결정하도록 구성된다. 의존성 기간은 식별된 출력 슬라이스를 생성하는데 필요한 입력 데이터의 데이터 시간 범위이다. 의존성 기간은 작업과 관련하여 정의된 의존성 정보의 부분이다. 예를 들어, 작업은 제 1 소스로부터의 3시간의 데이터 그리고 제 2 소스로부터의 1시간의 데이터를 연산하도록 특정될 수 있다. 따라서, 모든 데이터 세트가 시간당 스케줄을 갖는다면 제 1 소스로부터의 3시간의 데이터 및 제 2 소스로부터의 1시간의 데이터는 1시간의 출력 슬라이스 데이터를 생성하는데 필요하다.

[0019]

의존성 평가 컴포넌트(430)는 출력 슬라이스의 의존성이 만족되는지 여부를 결정하도록 구성된다. 의존성 평가 컴포넌트(430)는 이전에 결정된 의존성 기간 내에서 입력 데이터 슬라이스를 식별할 수 있다. 또한, 각각의 입력 슬라이스의 상태를 얻고 각각의 슬라이스의 상태가 소비를 위해 슬라이스가 준비된 것임을 의미하는 "준비"인지 여부에 대한 결정이 이루어진다(예를 들어, 슬라이스가 성공적으로 생성되고 외부에서 사용가능하게 만들어진다). 의존성 기간 내에서 입력 슬라이스가 "준비" 상태라면, 의존성이 충족된다. 그렇지 않으면, 의존성이 충족되지 않는다.

[0020]

의존성 조건이 충족되면, 실행 개시 컴포넌트(440)는 작업의 실행을 개시하여 출력 슬라이스를 생성하도록 구성된다. 즉, 동작 실행이 실행 컴포넌트(140) 상에서 트리거되고 선택된 출력 데이터 슬라이스 생성이 시작된다. 실행이 개시된 이후에, 출력 데이터 슬라이스의 상태가 "실행 보류"에서 "진행 중"으로 변경된다. 최종 출력 슬라이스가 성공적으로 생성된다면, 출력 상태는 "준비"로 설정될 수 있다. 이는 슬라이스를 입력으로서 소비하는 다운스트림 동작이 이제 슬라이스를 사용할 수 있다는 것을 의미한다. 실행이 실패하면, 이 기간 동안 사전결정된 정책에 기초하여 여러번 재시도될 수 있고, 상태는 이전의 실패 및 실행 재시도를 표시하는 "재시도"로 설정될 수 있다. 실행이 후속하여 진행되지 않는다면, 상태는 "실패한 실행"으로 설정될 수 있다. 또한, 각각의 실행은 출력 데이터 슬라이스와 연관된 수행 기록으로 기록될 수 있다.

[0021]

스케줄러 컴포넌트(130)의 동작에 대한 명확성 및 이해를 제공하기 위해, 몇개의 예시적인 시나리오가 설명된다. 먼저, 데이터 변환 동작을 포함하는 작업은 하나의 입력 및 하나의 출력을 갖는 시나리오를 고려한다. 여기서, 실행은 직접적(straightforward)이다. 시간이 경과함에 따라, 의존성 기간이 준비되었다면 현재 수행 시간이 꾹업되고 생성되기 이전에 데이터 슬라이스는 "실행 보류" 상태이다. 작업이 하나의 입력 및 다수의 입력과 관련하여 동작하는 다음 시나리오를 고려한다. 이 경우에, 모든 출력으로부터의 데이터 슬라이스의 연합은 생성될 슬라이스의 풀로서 사용된다. 출력 데이터 슬라이스에 대한 상태의 변화는 모든 출력 데이터 세트에 대해 발생한다. 나머지 실행은 이전 경우와 다르지 않다. 다음으로, 다수의 입력 및 다수의 출력을 갖는 작업을 고려한다. 모든 출력으로부터의 데이터 슬라이스의 연합은 다시 생성될 슬라이스의 풀로서 사용된다. 생성될 "실행 보류" 출력 슬라이스를 위해, 의존성 기간이 결정되고, 활성 기간 내의 모든 슬라이스가 "준비" 상태를 갖는다면, 작업은 출력 슬라이스를 위해 실행될 수 있다. 출력 데이터 슬라이스를 위한 상태의 변화는 모든 출력 데이터 세트에 대해 발생하고, 실행은 첫번째 시나리오와 동일하다.

[0022]

도 5는 대표적인 모니터 컴포넌트(150)를 더 자세하게 도시한다. 모니터 컴포넌트(150)는 상태 컴포넌트(510) 및 실행 기록 컴포넌트(520)를 포함한다. 상태 컴포넌트(510)는 이들이 생성한 데이터 및/또는 작업의 상태를 모니터한다. 예를 들어, 상태 컴포넌트는 출력 데이터가 작업 실행에 의해 성공적으로 생성되는지 또는 작업 실행이 출력 데이터를 성공적으로 생성하는데 실패하는지를 모니터할 수 있다. 이전에 언급된 바와 같이, 실패 또는 성공적인 실행은 사용자 인터페이스에 의해 사용자에 제시될 수 있다. 상태 컴포넌트(510)는 또한 데이터를 생성한 작업이 실행 보류, 진행 중, 또는 소비를 위한 준비 등일 때를 포함하는 추가 상태 정보를 모니터할 수 있다. 실행 기록 컴포넌트(520)는 작업 실행 동안 컴퓨터 리소스 활용에 관한 데이터를 얻도록 구성된다. 어떤 경우에는, 정보를 요청하여 운영 시스템 또는 유사한 제어 메커니즘으로부터 얻을 수 있다. 후속하여 리소스 활용은 타임라인 및 타임라인에 맞춰진 데이터 활용을 포함하는 분할 뷰를 제시하도록 사용자 인터페이스에 의해 활용될 수 있다.

[0023]

도 6 내지 10은 작업 스케줄링 및 모니터링과 관련하여 사용자 인터페이스 컴포넌트(110)에 의해 생성되는 다양한 시각화를 도시하는 예시의 스크린샷이다. 이들 스크린샷은 본 개시의 측면에 대한 명확성 및 이해를 돋기 위한 것이고 청구된 청구 대상을 제한하기 위한 것은 아니다. 제공된 스크린샷은 단지 하나의 구현예만을 도시한다는 것이 이해될 것이다. 다양한 다른 조합 및 배열의 그래픽적인 요소 및 텍스트가 고려되며 첨부된 청구항의 범위 내에 속하는 것이다. 또한, 작업 스케줄링에 대한 사용자 이해를 돋기 위해 다양한 소리가 시각화와 함께 이용될 수 있음이 또한 이해될 것이다. 제한이 아닌 예시로서, 소리는 작업 또는 데이터의 선택 시에, 또는 실행 실패를 검출할 시에 재생될 수 있다.

[0024]

도 6은 사용자 인터페이스 컴포넌트(110)에 의해 생성될 수 있는 인터페이스(600)의 스크린샷이다. 도시된 바와 같이, 인터페이스(600)는 3개의 패널, 소스 패널(610), 공개 패널(620), 및 스케줄 패널(630)을 포함한다. 소스 패널(610)은 복수의 사용가능한 데이터 세트를 제시하고 소스가 이들로부터 삭제되거나 추가되는 것을 가능하게 한다. 소스 패널(610)에 도시된 데이터 세트는 임의의 데이터 소스가 될 수 있다는 것이 이해될 수 있다. 예를 들어, 일부 데이터 세트는 온 프레미스(on-premises) 데이터와 연관될 수 있는 반면 다른 데이터 소스는 네트워크 또는 클라우드 데이터 저장소와 연관된다. 또한, 데이터 세트는 실질적으로 임의의 구조체 또는 포맷이 될 수 있다. 공개 패널(620)은 원하는 변환을 수행된 이후에 공개된 또는 소비가능한 데이터 소스의 시각적 표현을

제공한다.

[0025] 스케줄 패널(630)은 작업 실행의 스케줄 뿐만 아니라 모니터링된 실행의 결과를 시작화한다. 더 구체적으로, 스케줄 패널(630)은 작업의 시작 및 중지 실행 시간을 포함하는 갠트(Gantt) 차트로서 디스플레이되는 타임라인 뷰(640) 및 작업 실행에 맞춰진 리소스 소비를 도시하는 라인 그래프를 포함하는 리소스 활용 뷰(650)를 포함하는 분할 뷰를 제시한다. 작업 스케줄링은 그래픽에 기초하여 결정하기 쉽다. 예를 들어, 중복 제거 동작을 수행하는 첫번째 작업은 매일 실행을 위해 스케줄링되고, 조건부 분할을 수행하는 두번째 작업은 격일 실행을 위해 스케줄링된다. 라인(660)은 실행과 관련한 현재 시간을 나타낸다는 것에 유의해야한다. 라인 이전의 작업은 이미 실행된 작업이고, 라인을 넘은 작업은 앞으로 언젠가 실행하기 위해 스케줄링된다. 이 구분을 강조하기 위해, 스케줄링되었지만 아직 실행되지 않은 동작은 회색으로 도시되고, 또는, 즉, 동작이 비활성화된다. 이들이 한번 실행되면, 작업은 더 이상 비활성화되지 않을 것이다. 또한, 작업 실행과 관련한 색상 차이는 추가적인 정보를 나타낼 수 있다. 여기서, 예를 들어, 검정으로 색칠된 작업은 실패한 실행을 나타낸다. 하나의 측면에 따르면, 사용자는 검정으로 색칠된 작업 실행을 선택하고 작업의 실행을 스케줄링할 수 있다.

[0026] 도 7은 사용자 인터페이스 컴포넌트(110)에 의해 생성될 수 있는 인터페이스(700)의 스크린샷이다. 도 6의 인터페이스(600)와 유사하게, 이전에 도시된 바와 같이, 인터페이스(700)는 소스 패널(610), 공개 패널(620), 및 타임라인 뷰(640) 및 리소스 활용 뷰(650)를 포함하는 스케줄 패널(630)을 포함한다. 그러나, 인터페이스(700)는 작업의 선택 및 선택과 관련된 작업의 강조를 보여준다. 여기서, 조건부 분할을 수행하는 두번째 작업(710)은 예를 들어, 클릭, 터치, 또는 다른 제스처에 의해 사용자에 의해 선택된다. 작업이 선택된 이후에, 선택된 작업과 관련된 작업이 자동으로 식별되고 다른 작업과 시작적으로 구분되어 제시된다. 이 예시에서, 중복 제거 동작을 수행하는 첫번째 작업(712), 판매 데이터의 정화(cleansing)를 수행하는 다섯번째 작업(714), 연합을 수행하는 일곱번째 작업(716), 및 정렬 동작을 수행하는 여덟번째 작업(718)이 관련된 작업으로 식별되고 다른 작업과 관련하여 강조된다. 특히, 선택된 작업 뿐만 아니라 관련된 작업은 백색 배경과 함께 제시되는 반면 모든 다른 비관련 작업은 비활성화된다. 관련된 작업은 선택된 작업에 의존하는 작업 및 선택된 작업이 의존하는 작업을 포함한다. 여기서, 다섯번째 작업(714), 일곱번째 작업(716), 및 여덟번째 작업(718)은 선택된 작업에 의존하는 반면, 선택된 작업은 첫번째 작업(712)에 의존한다. 이러한 의존성은 작업의 위치에 기초하여 식별될 수 있고, 선택된 작업 이후에 제시되는 작업이 선택된 작업에 의존하는 반면 선택된 작업 이전에 위치된 작업은 선택된 작업이 의존하는 작업이다.

[0027] 도 8은 실패한 작업 수행의 선택에 응답하여 사용자 인터페이스 컴포넌트(110)에 의해 제시될 수 있는 인터페이스(800)의 스크린샷이다. 도 7 및 도 8의 스크린샷과 유사하게, 이전에 논의된 바와 같이, 인터페이스(800)는 소스 패널(610), 공개 패널(620), 및 타임라인 뷰(640) 및 리소스 활용 뷰(650)를 포함하는 스케줄 패널(630)을 포함한다. 인터페이스(800)는 또한 단일한 검정색으로 캡처된 성공적으로 실행하는데 실패한 특정 작업 수행의 선택을 나타낸다. 여기서, 조건부 분할 작업의 수행은 (810)에서 도시된다. 실패한 작업 수행의 선택은 선택된 작업 수행의 실패에 기초하여, 선택된 작업 수행이 실패하게 할 수 있는 다른 관련된 실패한 작업 수행 또는 실패할 수 있거나, 실패할 것으로 예측되는 다른 작업 수행을 식별하는 것을 트리거한다. 이들 관련된 작업은 작업 사이의 의존성, 실행 상태(예를 들어, 실패함, 성공함), 및 스케줄에 기초하여 식별될 수 있다. 의존성, 상태, 및 스케줄 정보의 분석 시에, (810)에서 언급된 선택된 "조건부 분할" 작업 수행의 실패 가능성으로서, (820)에 도시된 바와 같이, 중복 제거 작업 수행이 식별될 수 있다. 또한, (830)에서 도시된 바와 같이, "판매 정보 정화" 작업의 수행의 실패는 "조건부 분할" 작업의 실패의 결과로서 예측될 수 있다. 또한, (840)에서 언급된 "연합" 작업의 수행의 실패는 "조건부 분할" 작업 및 "판매 변환 정화" 작업 중 하나 또는 둘다의 실패에 기초하여 예측될 수 있다. 유사하게, (850)에서 도시된 "정렬" 작업의 실패는 "조건부 분할" 작업, "데이터 변환 정화" 작업, 또는 "연합" 작업의 실패의 기능으로서 예측될 수 있다. 인터페이스(800)는 선택된 실패한 작업 수행 및 관련된 실패한 작업 수행을 다른 작업 수행과 구분한다. 즉, 선택되고 관련된 실패한 작업 수행이 강조된다. 여기서, 선택되고 관련된 실패한 작업 수행을 제외한 모든 작업 수행은 비활성화된다. 물론, 그 반대가 가능하고, 다른 색상, 다른 폰트, 다른 크기 등이 구별의 목적을 위해 활용될 수 있다. 이 간편한 메커니즘은 사용자가 스케줄링 체인을 트러블슈팅하는 것을 돋는다. 이를 위해, 선택 시에 또는 실패한 작업 수행들 중 하나에 관한 다른 제스처 시에 다이얼로그 검색 판이 제시될 수 있어서(도시되지 않음) 작업 실행과 연관된 실행로그, 또는 로그 파일의 검색을 가능하게 할 수 있다. 이 방식으로, 사용자는 로그 파일 내의 원점 및 실패의 식별로부터의 빠른 전환이 가능하다. 또한, 사용자는 실패한 수행에 대해 선택하거나 제스처하고 작업의 실행을 재스케줄링한다. 예를 들어, 사용자는 실패한 작업 수행을 시간의 다른 포인트로 드래그 앤 드롭하여 실행을 재스케줄링한다. 다른 예시로서, 사용자는 다이얼로그 박스를 불러오기 위해 실패한 작업 수행에 대해 우클릭하여

사용자가 작업을 재스케줄링하는 것을 가능하게 할 수 있다.

[0028] 도 9는 사용자 인터페이스 컴포넌트(110)에 의해 생성될 수 있는 인터페이스(900)의 스크린샷을 도시한다. 이전 스크린샷과 유사하게, 인터페이스(900)는 복수의 사용가능한 데이터 세트를 제시하고 소스가 이들로부터 삭제되거나 추가되는 것을 가능하게 하는 소스 패널(610), 및 원하는 변환이 수행된 이후에 공개된 또는 소비가능한 데이터 세트의 시각적 표현을 제공하는 공개 패널(620)을 포함한다. 워크스페이스 패널(910)은 디어그램에 의한 작업 및 파이프라인의 시각적인 저작을 가능하게 한다. 예를 들어, 사용자는 소스 패널(610)로부터의 데이터 세트의 시각적 표현을 드래그 앤 드롭함으로써 데이터 세트를 얻을 수 있다. 다음으로, 예를 들어, 작업이 하나 이상의 데이터 변환 동작(예를 들어, 정렬, 그룹, 퍼벗, 분할, 필터...)을 소비하고 수행하는 입력을 데이터 세트가 제공한다는 것을 나타내도록 데이터 세트 표현으로부터 작업의 정육면체 표현으로 화살표를 당김으로써 데이터 세트는 이전에 저작된 작업에 연결될 수 있다(예를 들어, 데이터 프리뷰에 의해 자동으로 생성됨 및/또는 수동으로 코딩됨). 추가로, 변환된 출력의 표현은 워크스페이스 상에서 작업의 표현에 연결될 수 있다. 그 결과로서, 데이터 소스로부터 입력을 수신하고 작업의 하나 이상의 변환 동작의 적용예를 반영하는 새로운 데이터 소스를 출력하는 작업의 디어그램이 디스플레이된다. 인터페이스(900)는 작업 저작을 위한 이 디어그램 뷰와 작업 스케줄링 및 모니터링을 위한 타임라인 뷰 사이의 전환을 가능하게 한다. 또한, 데이터를 스케줄링하는 것과 모니터링하는 것은 워크스페이스 패널(910)에서의 이 디어그램 뷰 내에서 제시될 수 있다. (920)에서 도시된 바와 같이, "정렬" 작업의 표현은 리소스 활용의 데이터 및 그래프를 스케줄링하는 것을 포함하는 분할 뷰를 포함한다. 또한, 이전에 타임라인과 관련하여 또는 워크스페이스 패널(910)에서 제시된 작업 표현을 선택하는 신호를 수신한 이후에, 관련된 작업 및 데이터 세트가 결정되고 강조된다. 여기서, (930)에서 스크린샷은 "조건부 분할" 작업의 선택을 도시한다. 선택이 수신된 이후에, 작업 및 데이터에 관하여 수집된 의존성 데이터는 선택된 작업이 의존하는 작업 및 데이터와 선택된 작업에 의존하는 작업 및 데이터를 식별하도록 활용될 수 있다. 즉, 업스트림 및 다운스트림 작업 및 데이터를 포함하는 작업 계보가 결정되어 그 다음 디스플레이될 수 있다. 또한, 관련된 데이터 및 작업은 다른 데이터 및 작업과 관련하여 시각적으로 구분되거나 강조될 수 있다. 여기서, 관련된 작업 및 데이터는 흰색으로 색칠되는 반면 다른 데이터 및 작업은 회색으로 색칠되거나 비활성화된다.

[0029] 도 10은 사용자 인터페이스 컴포넌트(100)에 의해 생성될 수 있는 인터페이스(1000)의 스크린샷을 도시한다. 도 6의 인터페이스(600)와 유사하게, 인터페이스(1000)는 복수의 사용가능한 데이터 세트를 제시하고 소스가 이들로부터 삭제되거나 추가되는 것을 가능하게 하는 소스 패널(610), 및 원하는 변환이 수행된 이후에 공개된 또는 소비가능한 데이터 세트의 시각적 표현을 제공하는 공개 패널(620)을 포함한다. 인터페이스(1000)는 또한 타임라인 뷰 뿐만 아니라 리소스 활용 뷰(650)를 포함하는 분할 뷰를 포함하는 스케줄 패널(630)을 포함한다. 그러나, 여기서, 타임라인 뷰(1020)는 작업 실행을 위한 프로세싱 시간과는 대조적으로 데이터 슬라이스 시간 단위로 제시된다. 즉, 타임라인 뷰(1020)는 프로세싱 시간 보다는 데이터 시간을 도시한다. 예를 들어, 이벤트에 대해 시간별 프로세싱이 수행되는 시나리오를 고려한다. 일부 변환 동작에 후속하여 프로세싱은 1시간이 걸리는 이벤트를 시작한다. 더 구체적으로, 오전 9시부터 오전 10시까지의 시간의 프로세싱은 오전 10시 15분에 시작할 수 있다. 후속하여, 일부 집성을 오전 11시에 수행될 수 있고, 다른 데이터와의 병합은 오후 12시에 수행될 수 있어서, 결과는 오후 1시에 완료될 수 있다. 따라서, 타임라인 뷰(1020)는 작업을 실행하는데 얼마나 걸리는지에 관한 작업 실행 시간과는 대조적으로 기간(예를 들어, 일, 주...) 전반에서 프로세싱되는 방법 및 데이터 시간의 뷰를 제공한다. 스케줄 패널(630)은 또한 데이터 프로세싱에 관한 현재 시간을 나타내는 라인(660)을 포함한다. 라인 이전의 데이터는 이미 생성된 데이터를 나타내고, 라인 이후의 데이터는 앞으로 언젠가 생성하기 위해 스케줄링되는 데이터를 나타낸다. 이 구분을 강조하기 위해, 스케줄링되었지만 아직 생성되지 않은 데이터는 회색으로 도시되고, 또는, 즉, 동작이 비활성화된다. 데이터가 생성된 이후에, 작업은 더 이상 비활성화되지 않을 것이다. 또한, 데이터와 관련한 색상 차이는 추가적인 정보를 나타낼 수 있다. 예를 들어, 검정으로 색칠된 데이터는 연관된 데이터의 실패한 생성을 나타낸다. 하나의 측면에 따르면, 사용자는 검정으로 색칠된 데이터를 선택하고 데이터를 생성하는 작업을 재스케줄링할 수 있다. 도시되지 않았지만, "실행 보류", "준비", "실행 중", 및 "실패" 등과 같은 데이터 상태를 포함하는 데이터에 관한 색상 또는 다른 시각적 특징에 의해 다른 정보가 또한 전달될 수 있다.

[0030] 도 11은 작업 시스템(1100)을 도시한다. 시스템(110)은 데이터 변환 작업을 포함하는 입력을 수신하고 데이터 변환 작업의 뷰를 생성하기 위한 수단을 제공하는 뷰 생성 수단(1110)을 포함한다. 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합이 데이터 변환 작업을 포함하거나 수반하는 시각화인 뷰의 생성에서 활용될 수 있다. 뷰 생성 수단(1110)에 의해 생성된 뷰는 제시를 위한 디스플레이에 제공될 수 있다. 하나의 경우에, 뷰 생성 수단(1110)은 그래픽 사용자 인터페이스가 되거나 일부를 형성할 수 있다. 일 실시예에 따르면, 생성된

뷰는 두번째 작업으로 입력을 선택적으로 제공하는 첫번째 작업의 출력을 갖는 하나 이상의 관련된 작업의 세트를 포함하는 데이터 변환 작업 또는 작업 파이프라인의 다이어그램이 될 수 있다.

[0031] 선택 수단(1120)은 뷰에서 작업의 선택을 나타내는 신호를 생성하는 메커니즘이다. 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합은 선택 수단(1120)을 구현하는데 활용될 수 있다. 사용자는 작업을 선택하거나 식별하는 입력 메커니즘을 활용할 수 있고 예를 들어, 사용자 입력의 위치를 뷰에서의 작업의 위치와 비교함으로써 선택 수단(1120)은 입력을 수신하고 선택된 작업을 식별할 수 있다. 다양한 입력 메커니즘은 사용자에 의해 활용될 수 있고 터치 패드, 마우스, 터치 스크린, 카메라, 또는 마이크로폰을 포함하지만 이에 제한되지 않는다.

[0032] 관련된 작업 수단(1130)은 선택된 작업에 관련된 작업을 자동으로 식별하는 메커니즘을 제공한다. 선택 수단(1120)에 의해 제공된 선택된 작업을 고려하면, 관련된 작업 수단은 작업에 관한 의존성 정보를 수신, 검색, 또는 획득 또는 얻을 수 있다. 후속하여, 관련된 작업은 예를 들어, 선택된 작업을 검색하고 선택된 작업과 관련된 작업을 식별함으로써 관련된 작업을 식별할 수 있고, 관련된 작업은 선택된 작업에 의존하는 작업 또는 선택된 작업이 의존하는 작업이다. 하나의 측면에 따르면, 의존성은 작업이 의존하는 입력 데이터 및 작업에 의해 생성되는 출력 데이터를 포함하는 작업 데이터의 관점에서 캡쳐될 수 있다. 일 실시예에서, 의존성 데이터는 결절점인 작업 및 작업을 연결하는 방향성 엣지인 데이터 세트를 포함하는 방향성 그래프에서 캡쳐될 수 있다. 관련된 작업을 결정하기 위해, 그래프는 선택된 작업을 나타내는 결절점을 앞뒤로 가로지를 수 있고, 앞으로 가로지르는 것(예를 들어, 선택된 작업으로부터 다운스트림)은 선택된 작업에 의존하는 데이터 세트 및 작업을 캡쳐하고, 뒤로 가로지르는 것(예를 들어, 선택된 작업으로부터 업스트림)은 선택된 작업이 의존하는 데이터 세트 및 작업을 식별한다. 관련된 작업 수단(1130)은 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합을 이용하여 예를 들어, 컴퓨터 실행가능한 소프트웨어 컴포넌트 또는 펌웨어와 같은 관련된 작업(뿐만 아니라 데이터 세트)을 식별한다.

[0033] 뷰 업데이트 수단(1140)은 뷰에 대한 변경을 수행하는 것을 가능하게 하는 메커니즘이다. 하나의 경우에, 뷰 업데이트 수단은 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합을 포함하여 선택된 작업을 수신, 검색, 또는 획득 또는 얻을 수 있고, 하나 이상의 관련된 작업 및 선택적으로 관련된 데이터 세트는 관련된 작업 수단(1130)으로부터 직접적이거나 위치(예를 들어, 메모리, 저장소...)로부터 간접적이고 뷰에서 제시된 다른 작업 및 데이터 소스로부터 선택된 작업 및 관련된 작업 및 데이터 소스를 시각적으로 구분한다. 즉, 뷰 업데이트 수단(1140)은 뷰에서 선택된 작업 및 관련된 작업 및 데이터 소스를 강조할 수 있다.

[0034] 스케줄링 수단(1150)은 작업 실행에 관한 스케줄링 기능을 제공한다. 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합으로서 구현되어, 스케줄링 수단(1150)은 하나 이상의 작업을 수신, 검색, 또는 획득 또는 얻을 수 있고 이와 함께 연관된 작업 또는 정보의 정의에 따른 작업 실행을 위한 스케줄을 생성하여 스케줄링 정보를 제공한다. 예를 들어, 작업이 매일 실행을 지시하면, 작업 또는 작업 수행은 매일 스케줄링될 수 있다. 유사하게, 작업이 격일 실행을 특정한다면 작업 또는 작업 수행은 격일로 스케줄링될 수 있다. 물론, 스케줄은 또한 다양한 데이터 의존성을 설명하여 작업은 입력 데이터가 사용가능한 이후에 실행을 위해 스케줄링된다.

[0035] 스케줄링 수단(1150)은 스케줄을 뷰 생성 수단에 대해 사용가능하게 만들 수 있다. 따라서, 뷰 생성 수단은 스케줄링된 작업의 타임라인으로서 스케줄의 뷰를 생성할 수 있다. 이 경우에, 선택 수단(1120)은 스케줄의 타임라인 뷰로부터 작업의 선택을 가능하게 할 수 있다. 후속하여, 관련된 작업 수단은 선택된 작업과 관련된 작업을 결정할 수 있고 뷰 업데이트 수단(1140)은 선택된 작업 및 관련된 작업을 강조하기 위해 스케줄의 타임라인 뷰를 업데이트할 수 있다.

[0036] 실행 수단(1160)은 스케줄링 수단(1150)에 의해 제공되거나 사용가능하게 만들어진 작업을 실행한다. 실행 수단(1160)은 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합을 이용하여 하나 이상의 입력 데이터 세트에 대한 데이터 변환 작업을 실행하고 작업에 의해 특정된 데이터 변환 동작의 적용예를 반영하는 하나 이상의 출력 데이터 세트를 생성할 수 있다. 실행 수단(1160)은 제 1 지정 위치로부터 입력 데이터를 판독할 수 있고 후속하여 제 2 지정 위치로 출력 데이터를 기록할 수 있고, 위치는 메모리 또는 저장 디바이스 위치가 될 수 있다.

[0037] 모니터링 수단(1170)은 하드웨어, 소프트웨어, 또는 하드웨어 및 소프트웨어의 조합을 포함하는 메커니즘이고 실행 수단(1160)에 의해 작업의 실행을 모니터링한다. 하나의 실시예에 따르면, 모니터링 수단(1170)은 작업이 성공적으로 실행되는지 또는 성공적으로 실행하는데 실패하는지를 검출 또는 결정할 수 있다. 이는 실행의 실패 또는 성공에 관한 잠재적인 메시지를 포함하는 실행 수단(1160)의 출력을 모니터링함으로써 달성될 수 있다. 하

나의 경우에, 모니터링 수단(1170)은 작업의 실행이 스케줄링된 것으로 개시되었는지 여부 및 실행이 성공적이지 또는 성공적이지 않은지를 결론 내리는 것에 관한 작업의 출력이 생성되는지 여부를 결정할 수 있다. 다른 실시예에서, 모니터링 수단(1170)은 작업 실행에 관한 리소스 활용을 모니터링할 수 있다. 하나의 경우에, 모니터링 수단(1170)은 프로세서, 메모리, 저장소, 및 네트워크 활용을 포함하는 리소스를 직접 모니터링하는 메커니즘으로 구현될 수 있다. 대안으로, 모니터링 수단(1170)은 리소스 활용을 모니터링하는 운영 시스템 또는 다른 시스템 또는 컴포넌트로부터 이러한 정보를 요청하여 수신할 수 있다.

[0038] 모니터링 수단(1170)은 뷰를 생성하는데 사용하기 위해 획득된 데이터 또는 정보를 뷰 생성 수단(1110)에 대해 사용가능하게 만들 수 있다. 하나의 경우에, 뷰 생성 수단(1110)은 모니터링 수단(1170)에 의해 제공된 정보에 기초하여 스케줄링된 작업 수행이 성공했는지 또는 실패했는지 여부를 나타내는 스케줄의 타임라인 뷰를 생성할 수 있다. 이 경우에, 선택 수단(1120)은 실패한 수행의 선택을 시그널링하는 메커니즘을 제공할 수 있다. 후속하여, 관련된 작업 수단(1130)은 관련된 작업 수행을 식별하고 실패한 더 특별히 관련된 작업 실행을 식별하거나 실패한 작업 수행에 대한 의존성에 기초하여 실패할 것을 예측할 수 있다. 하나의 실시예에 따르면, 데이터 의존성 정보 이외에, 관련된 작업 수단은 무엇보다도 작업이 성공적으로 실행되었는지 또는 성공적으로 실행하는데 실패했는지를 포함하는 동작 정보를 수신할 수 있다. 하나의 경우에, 동작적인 정보는 의존성 정보와 함께 저장될 수 있다. 대안으로, 동작적인 정보는 개별적으로 저장되고 액세스될 수 있다. 후속하여 뷰 업데이트 수단(1140)은 실패한 또는 실패할 것으로 예측되는 관련된 작업 실행 뿐만 아니라 선택된 실패한 작업 실행을 강조할 수 있다. 추가적으로, 뷰 생성 수단(1110)은 작업 수행에 맞춰진 모니터링 수단(1170)에 의해 제공되는 리소스 활용 정보의 시각화를 생성하는 것을 포함할 수 있다. 예를 들어, 뷰 생성 수단(1110)은 리소스 활용을 나타내는 라인 그래프와 같은 그래프를 생성한다.

[0039] 전술한 시스템, 아키텍처, 환경, 및 유사한 것은 다수의 컴포넌트들 사이의 인터랙션과 관련하여 설명되었다. 이러한 시스템 및 컴포넌트는 본원에서 특정된 이를 컴포넌트 또는 서브컴포넌트, 특정된 컴포넌트 또는 서브컴포넌트 중 일부, 및/또는 추가적인 컴포넌트들을 포함할 수 있다. 서브 컴포넌트는 또한 부모 컴포넌트 내에 포함되기 보다는 다른 컴포넌트에 통신가능하게 연결된 컴포넌트로서 구현될 수 있다. 그러나, 하나 이상의 컴포넌트 및/또는 서브 컴포넌트는 집성 기능을 제공하는 단일 컴포넌트로 통합될 수 있다. 시스템, 컴포넌트 및/또는 서브 컴포넌트 사이의 통신은 푸쉬 및/또는 폴 모델 중 하나에 따라 달성될 수 있다. 컴포넌트는 또한 간결성을 위해 본원에서 특별하게 설명되지 않았지만, 당업자에 의해 알려진 하나 이상의 다른 컴포넌트와 인터랙팅할 수 있다.

[0040] 또한, 개시된 위의 시스템 및 이하의 방법의 다양한 부분은 인공 지능, 머신 러닝, 또는 지식 또는 규칙 기반 컴포넌트, 서브 컴포넌트, 프로세서, 수단, 방법론, 또는 메커니즘을 포함하거나 이용할 수 있다(예를 들어, 벡터 머신, 신경망, 전문가 시스템, 베이지안 벌리프 네트워크(Bayesian belief networks), 퍼지 논리(fuzzy logic), 데이터 융합 엔진, 분류자(classifier)..를 지원). 이러한 컴포넌트들은, 그 중에서도, 수행된 특정 메커니즘 또는 프로세스를 자동화시켜 시스템 및 방법의 일부를 효율적이고 지적일 뿐만 아니라 더 적응적으로 만들 수 있다. 제한이 아닌 예시로서, 이전 인터랙션 및 다른 상황 정보에 기초하여, 스케줄러 컴포넌트(130) 및 사용자 인터페이스(110)는 이러한 메커니즘을 이용하여 작업 스케줄 및 데이터 제시를 각각 결정하거나 추론할 수 있다.

[0041] 상술한 예시적인 시스템을 참조하면, 개시된 청구 대상에 따라 구현될 수 있는 방법론은 도 12 내지 17의 흐름도를 참조하여 더 잘 이해될 수 있다. 설명의 단순함의 목적을 위해, 방법론은 일련의 블록으로서 도시되고 설명되었지만, 일부 블록은 본원에서 도시되고 설명된 것과는 상이한 순서로 발생할 수 있고/거나 다른 블록과 동시에 발생할 수 있기 때문에, 청구된 청구 대상은 블록의 순서에 의해 제한되지 않는다는 것이 이해될 수 있다. 또한, 모든 도시된 블록이 이후에 설명된 방법을 구현하기 위해 필요하지 않을 수 있다.

[0042] 도 12를 참조하면, 작업 스케줄링 및 모니터링의 방법(1200)이 도시된다. 참조 번호(1210)에서, 적어도 작업이 의존하는 데이터 및 작업에 의해 생성되는 데이터의 식별을 포함하는 각각의 작업에 대한 의존성을 포함하는 작업의 세트가 수신된다. 즉, 각각의 작업에 대해 특정된 데이터 의존성을 포함하는 작업의 세트가 수신된다. 참조 번호(1220)에서, 각각의 작업에 의해 정의된 실행 시간 및 빈도에 관한 다른 정보 및 의존성에 기초하여 작업의 세트를 프로세싱하기 위해 스케줄이 생성된다. 참조 번호(1230)에서, 생성된 스케줄이 디스플레이된다. 하나의 측면에 따르면, 스케줄은 작업에 의해 생성된 작업 또는 데이터의 시작 및 중단 실행 시간을 포함하는 겐트 차트로 제시될 수 있다. 참조 번호(1240)에서, 작업의 프로세싱 또는 작업 수행이 스케줄에 따라 개시될 수 있다. 예를 들어, 작업은 프로세싱을 위해 실행 컴포넌트에 제공될 수 있다. 참조 번호(1250)에서, 실행이 개시된 작업에 대해 프로세스 성공 또는 실패가 판정된다. 이러한 판정은 단지 실행 상태에 관한 실행 컴포넌트로부터

터의 통지를 수용하거나 메모리 또는 디스크와 같은 위치로부터 실행 상태를 검색하는 것에 의해 수행될 수 있다. 참조 부호(1260)에서, 작업 수행의 성공 또는 실패는 스케줄과 관련하여 디스플레이된다. 예를 들어, 성공적으로 실행된 작업 수행은 녹색으로 색칠될 수 있고, 작업 수행의 실패는 빨간색으로 색칠되어 작업 수행의 성공 또는 실패에 관한 명확한 지시를 제공할 수 있다. 또한, 작업은 따라서 이들의 의존성 및 색상에 기초하여 성공하거나 실패할 것으로 예측될 수 있다.

[0043] 도 13은 관련된 작업을 구분하는 방법(1300)을 도시한다. 참조 부호(1310)에서, 스케줄로부터의 하나 이상의 작업 수행을 포함하는 작업을 선택하는 신호가 수신된다. 예를 들어, "중복 제거"와 같은 작업은 예를 들어, 작업에 대한 좌클릭 또는 터치에 의해 실행을 위해 스케줄링된 하나 이상의 작업 및 작업 수행을 도시하는 타임라인 뷰로부터 선택될 수 있다. 참조 부호(1320)에서, 선택된 작업에 의존하는 작업이 식별된다. 참조 부호(1330)에서, 선택된 작업이 의존하는 작업이 식별된다. 작업의 식별은 의존성의 분석에 기초할 수 있다. 하나의 실시예에 따르면, 의존성은 결절점인 작업 및 작업을 연결하는 방향성 엣지인 데이터 세트의 방향성 그래프로서 캡쳐될 수 있다. 이 경우에, 선택된 작업은 이 그래프 내에서 식별될 수 있고, 분석은 선택된 작업에 의존하는 작업을 식별하기 위해 선택된 작업으로부터 앞으로 진행하고 선택된 작업이 의존하는 작업을 식별하기 위해 뒤로 진행할 수 있다. 참조 부호(1340)에서, 선택된 작업 및 식별된 작업이 강조된다. 즉, 스케줄 상에서, 예를 들어, 선택된 작업 및 식별된 작업이 디스플레이된 다른 작업으로부터 시작적으로 구분된다. 이 방식으로, 사용자는 작업 사이의 관련성을 빠르게 이해할 수 있다.

[0044] 도 14는 성공적인 작업 실행 및 실패한 작업 실행을 구분하는 방법(1400)을 도시한다. 참조 부호(1410)에서, 작업 수행의 실행이 개시된다. 여기서, 작업 수행은 작업의 인스턴스이고, 특정 시간에 실행하도록 스케줄링된다. 참조 부호(1420)에서, 실행이 개시된 작업 수행에 의존적인 작업 수행이 식별된다. 의존성 정보는 의존적인 작업 수행을 식별하기 위해 활용될 수 있다. 예를 들어, 의존성 정보는 결절점인 작업 및 작업을 연결하는 방향성 엣지인 데이터 세트의 방향성 그래프로 기록될 수 있다. 실행이 개시되는 작업이 그래프에서 식별될 수 있고 의존적인 작업은 작업으로부터 그래프를 아래로 가로지름으로써 식별될 수 있다. 후속하여, 의존적인 작업의 스케줄링된 실행은 스케줄링된 실행이 각각의 작업에 대해 기록된다면 스케줄로부터 또는 그래프로부터 결정될 수 있다. 참조 부호(1430)에서, 작업 수행의 실행에 대한 실패가 존재하는지 여부에 대한 결정이 이루어질 수 있다. 실패가 존재하지 않으면("아니오"), 방법은 성공적인 실행이 작업 수행에 대해 기록되는 (1440)으로 진행한다. 하나의 측면에 따르면, 의존성 정보 외에 작업 상태를 포함하는 동작적인 의존성 정보가 기록된다. 성공적인 실행은 이와 같이 저장될 수 있는 작업 상태이다. 다음으로, (1450)에서, 실행된 작업과 관련하여 그리고 성공적인 실행의 예측으로서 의존적인 작업에 대해 선택적으로 성공적인 실행이 디스플레이된다. 예를 들어, 스케줄 상에 디스플레이된 작업 수행은 녹색으로 색칠될 수 있다. (1430)에서, 실행 실패가 결정되면("예"), 방법은 실행 실패가 기록되는 (1460)에서 시작한다. 실패는 위에서 언급된 바와 같이 동작적인 의존성 정보를 보존하는 것에 대해 저장될 수 있는 작업 상태이다. 그 후 방법은 (1470)으로 진행하고 실행된 작업과 관련한 실패의 디스플레이가 예를 들어, 스케줄 상에서 나타난다. 또한, 실행된 작업에 따른 작업은 또한 실행된 작업이 실패하면 실패할 가능성이 있다. 따라서, 독립적인 작업 수행은 실패가 예측되는 것을 나타내는 방식으로 디스플레이될 수 있다. 예를 들어, 실행된 작업 수행 및 독립 작업 수행은 적색으로 색칠되어 각각 실패를 나타내고 실패를 예측할 수 있다.

[0045] 도 15는 스케줄링 체인 트리블슈팅을 제공하는 방법(1500)을 도시한다. 참조 부호(1510)에서, 실패하거나 실패할 것으로 예측되는 작업 수행을 선택하는 신호를 수신하고, 또는 더 구체적으로, 실패는 예를 들어, 스케줄과 관련하여 나타낸다. 예를 들어, 스케줄은 복수의 작업을 포함할 수 있고, 각각의 작업에 대해, 작업 수행은 언젠가 실행하기 위해 스케줄링된다. 또한, 각각의 작업 수행에 대해 이전에 실행되거나 성공 또는 실패할 것이라고 예측되면, 또는 아직 실행되지 않으면, 작업 수행이 성공적이었거나 실패했음을 나타내는 시각적 표시가 제공될 수 있다. 선택 신호는 실패를 나타내는 작업 수행 중 하나와 연관될 수 있고 예를 들어, 사용자가 작업 수행의 시각적 표현을 클릭하거나 터치할 시에 생성될 수 있다. 참조 부호(1520)에서, 실패하거나 실패할 것으로 예측되는 관련된 작업 수행이 식별된다. 관련된 작업 수행은 선택된 작업 수행에 의존하는 작업 수행 및 선택된 작업 수행이 의존하는 작업 수행을 포함한다. 관련된 작업은 작업에 관한 의존성 정보를 분석함으로써 식별될 수 있다. 예시로서, 선택된 작업은 결절점인 작업 및 작업을 연결하는 방향성 엣지로서 연결되는 데이터 세트를 포함하는 방향성 그래프에서 식별될 수 있다. 후속하여, 그래프는 선택된 작업으로부터 업스트림 및 선택된 작업으로부터 다운스트림으로 가로질러 관련된 작업을 식별할 수 있다. 하나의 실시예에 따르면, 작업 수행 및 상태는 의존성 정보와 함께 기록될 수 있다. 따라서, 관련된 작업의 식별시에, 작업 수행이 식별될 수 있고 실제의 또는 예측되는 실패 상태를 포함하는 작업 수행이 식별될 수 있다. 물론, 작업 수행을 포함하는 일부 정보는 스케줄과 관련하여 저장될 수 있고 관련된 작업 수행을 결정하기 위해 활용될 수 있다. 참조 부호(1530)에서,

선택된 작업 수행 및 관련된 작업 수행이 강조된다. 다르게 말하면, 선택된 작업 수행 및 관련된 작업 수행은 다른 작업 수행들과 시각적으로 구분될 수 있다. 이는 사용자가 트러블슈팅을 위해 실패 및 연속 실패에 초점을 맞추는 것을 가능하게 한다.

[0046] 도 16은 관련된 작업 및 데이터 세트를 구분하는 플로우 차트 다이어그램이다. 참조 부호(1610)에서, 작업(또는 데이터 세트)을 선택하는 신호가 수신된다. 예를 들어, 사용자가 작업 및 데이터 세트의 스케줄 뷰에서 또는 개략도에서 하나 이상의 사전결정된 제스처에 의해 작업을 선택할 시에 신호가 수신될 수 있다. 참조 부호(1620)에서, 관련된 작업 데이터 세트가 식별된다. 실시예에 따르면, 작업 및 데이터 세트에 관한 의존성 정보가 저장될 수 있고 선택된 작업에 의존하는 작업 및 데이터 세트(예를 들어, 다운스트림) 및 선택된 작업이 의존하는 작업 및 데이터 세트(예를 들어, 업스트림)를 포함하는 관련된 작업 및 데이터 세트를 식별하도록 참조될 수 있다. 하나의 경우에, 방향성 그래프는 결절점인 작업 및 작업을 연결하는 방향성 엣지인 데이터 세트를 포함할 수 있다. 선택된 작업(또는 데이터 세트)이 그래프에서 식별될 수 있고 그래프가 선택된 작업에 의존하는 작업 및 데이터 세트 및 선택된 작업(또는 데이터 세트)가 의존하는 작업 및 데이터 세트를 식별하도록 가로지른다. 참조 부호(1630)에서, 선택된 작업(또는 데이터 세트) 및 관련된 작업 및 데이터 세트가 시각적 디스플레이 상에서 강조된다. 즉, 선택된 및 관련된 작업 및 데이터 세트는 다른 작업 및 데이터 세트와 시각적으로 구분된다. 예를 들어, 특정 작업 및 데이터 세트의 개략도에서, 관련된 작업 및 데이터 세트의 서브세트가 시각적으로 구분될 수 있다. 이는 특히 대량의 작업 및 데이터 세트 및 복합 파이프라인 또는 체인이 존재하는 경우에 관련성을 이해하는데 도움이 된다.

[0047] 도 17은 작업의 데이터 구동형 프로세싱의 방법(1700)을 도시한다. 참조 부호(1710)에서, 생성될 출력 데이터 슬라이스가 결정되고, 데이터 슬라이스는 이들이 연관된 기간에 의한 데이터 세트 슬라이스에서의 데이터의 조각이다. 하나의 특정 구현예에서, 출력 데이터 슬라이스는 데이터 슬라이스의 상태를 분석하여 식별될 수 있다. 상태가 "실행 보류"이거나 유사한 것이면, 데이터 슬라이스는 생성될 출력 데이터 슬라이스가 되도록 결정될 수 있다. 참조 부호(1720)에서, 출력 데이터 슬라이스가 생성을 위해 선택된다. 여기서, 생성될 복수의 출력 데이터 슬라이스로부터의 선택은 정책 기반이 될 수 있다. 예를 들어, 작업 실행을 위한 정책은 상태가 "실행 보류" 또는 유사한 것으로 설정되었을 때에 기초하여 가장 오래된 순 또는 가장 최신 순으로 생성하기 위해 출력 데이터 슬라이스를 선택하는 것을 특정할 수 있다. 참조 부호(1730)에서, 의존성 기간이 결정된다. 의존성 기간은 출력 데이터 슬라이스를 생성하는데 필요한 입력 데이터의 시간 범위이다. 이는 출력 데이터 슬라이스를 생성하는데 작업과 관련하여 정의된 의존성 정보에 기초하여 선택 출력 데이터 슬라이스에 대해 결정될 수 있다. 예를 들어, 작업은 제 1 소스로부터의 3시간의 데이터 그리고 제 2 소스로부터의 1시간의 데이터를 연산하도록 특정될 수 있다. 따라서, 모든 데이터 세트가 시간당 스케줄을 갖는다면(예를 들어, 슬라이스가 1시간임) 제 1 소스로부터의 3시간의 데이터 및 제 2 소스로부터의 1시간의 데이터는 1시간의 출력 슬라이스 데이터를 생성하는데 필요하다. 참조 부호(1740)에서, 의존성 기간에 모든 필요한 입력 데이터 슬라이스가 준비되는지 여부에 대한 결정이 수행된다. 이는 입력 데이터 슬라이스의 상태에 대해 질의함으로서 결정될 수 있다. 입력 데이터 슬라이스의 각각은 데이터가 소비를 위해 준비된 것을 의미하는 "준비" 상태이면, 방법은 참조 부호(1750)으로 진행할 수 있다. 그렇지 않으면, 방법은 루프하기 시작하고 준비가 될 입력 데이터 슬라이스에 대해 대기할 수 있다. 참조 부호(1750)에서, 선택된 출력 슬라이스를 생성하는 작업의 실행이 트리거되거나 개시된다. 참조 부호(1760)에서, 출력 슬라이스의 상태가 설정될 수 있다. 데이터가 생성되고 있는 동안, 상태는 데이터 슬라이스가 작업을 위해 생성되고 있다는 것을 나타내는 "진행 중"으로 설정될 수 있다. 실행이 완료되면(또는 수행이 실패함), 출력 슬라이스를 생성하는 시도 실패하면, 상태는 "실패함"으로 변경될 수 있고, 또는 생성하는 시도가 성공하고 데이터 슬라이스가 소비를 위해 준비되면 "준비"로 변경될 수 있다. 참조 부호(177)에서, 실행에 관한 정보가 기록된다. 예를 들어, 출력 슬라이스의 상태 뿐만 아니라 실행의 시작 및 종료 시간이 기록될 수 있다. 이러한 데이터는 작업 실행의 이해 및 제어를 제공하기 위해 사용자에게 제시하기 위한 정보 및 시각화를 생성하는 것과 관련하여 활용될 수 있다.

[0048] 본 개시는 작업 스케줄링 및 모니터링에 관한 다양한 동작을 수행하거나 수행하도록 구성된 다양한 제품 및 프로세스를 지원한다. 후속하는 것은 하나 이상의 예시적인 방법 및 시스템이다.

[0049] 방법은 인터페이스의 디스플레이 상에서 하나 이상의 데이터 변환 작업의 뷰를 제시하는 단계와, 인터페이스에 의해 하나 이상의 데이터 변환 작업으로부터 하나의 작업을 선택하는 사용자로부터 제 1 신호를 수신하는 단계와, 작업 사이의 데이터 의존성에 기초하여 작동으로 선택된 작업에 대해 하나 이상의 관련된 작업을 결정하는 단계와, 선택된 작업 및 하나 이상의 다른 작업로부터 시각적으로 구분되는 하나 이상의 관련된 작업을 제시하는 단계를 포함한다. 방법은 또한 인터페이스의 디스플레이 상에서 실행하기 위해 스케줄링된 하나 이상의 작업

을 제시하는 단계를 포함한다. 방법은 또한 인터페이스의 디스플레이 상에 하나 이상의 이전에 실행된 작업을 제시하는 단계를 포함한다. 방법은 또한 인터페이스의 디스플레이 상에서 스케줄링된 작업으로부터 시작적으로 구분되는 이전에 실행된 작업을 제시하는 단계를 포함한다. 방법은 또한 하나 이상의 작업의 실행 동안 계산 부하를 결정하는 단계와, 인터페이스의 디스플레이 상에 하나 이상의 작업에 맞춰진 계산 부하를 제시하는 단계를 포함한다. 방법은 인터페이스의 디스플레이 상에 적어도 하나의 작업이 성공적으로 실행하는데 실패했다는 표시를 제시하는 단계를 포함한다. 방법은 또한 성공적으로 실행하는데 실패하거나 성공적으로 실행하는데 실패한 작업 수행에 대한 의존성에 기초하여 실패할 것으로 예측되는, 선택된 작업 수행에 관련된 하나 이상의 관련된 작업 수행을 자동으로 결정하는 단계와, 인터페이스의 디스플레이 상에서 선택된 작업 수행 및 하나 이상의 관련된 작업 수행을 강조하는 단계를 포함한다. 방법은 또한 성공적으로 실행하는데 실패한 적어도 하나의 작업 중 하나를 선택하는 제 2 신호를 수신하는 단계와, 성공적으로 실행하는데 실패하거나 성공적으로 실행하는데 실패한 작업 수행에 대한 의존성에 기초하여 실행할 것으로 예측되는, 선택된 작업 수행에 관련된 하나 이상의 관련된 작업 수행을 자동으로 결정하는 단계와, 인터페이스의 디스플레이 상에서 선택된 작업 수행 및 하나 이상의 관련된 작업 수행을 강조하는 단계를 포함한다. 방법은 작업 및 데이터 세트의 표현 뿐만 아니라 작업과 데이터 세트 사이의 연결을 포함하는 다이어그램에서 선택된 작업 및 하나 이상의 관련된 작업을 제시하는 단계를 포함한다.

[0050]

방법은 메모리에 저장된 컴퓨터 실행가능한 명령어를 실행하여 다음의 단계들을 수행하도록 구성된 적어도 하나의 프로세서를 이용하는 단계와, 그래픽 사용자 인터페이스에 의해 데이터 변환 작업의 선택을 검출하는 단계와, 작업의 선택을 검출하는 것에 응답하여 작업 사이의 데이터 의존성에 기초하여 자동으로 하나 이상의 관련된 작업을 결정하는 단계와, 선택된 작업 및 그래픽 사용자 인터페이스 상에 디스플레이되는 다른 작업과 시작적으로 구분되는 하나 이상의 관련된 작업을 제시하는 단계를 포함한다. 방법은 또한 작업 실행과 함께 컴퓨터 리소스 활용을 모니터링하는 단계와, 하나 이상의 개별적인 작업에 맞춰진 리소스 활용의 시각화를 제시하는 단계를 포함한다. 방법은 또한 작업 수행의 실행의 성공 또는 실패를 결정하는 단계와, 성공적인 실행 수행과 상이한 실패한 실행 수행을 제시하는 단계를 포함한다. 방법은 또한 성공적으로 실행하는데 실패한 작업 수행과 관련하여 수신된 신호에 기초하여 작업 수행의 실행을 재스케줄링하는 단계를 포함한다. 방법은 또한 선택된 실패 작업 수행 및 하나 이상의 관련된 실패 작업 수행 또는 선택된 실패 작업 수행을 식별하는 신호의 수신 이후에 실패 작업 수행에 대한 의존성에 기초하여 실패가 예측되는 실행을 강조하는 단계를 포함한다. 방법은 또한 작업 및 데이터 세트의 표현 및 작업과 데이터 세트 사이의 접속을 포함하는 다이어그램에서 선택된 작업 및 하나 이상의 관련된 작업을 제시하는 단계를 포함한다.

[0051]

시스템은 메모리에 연결된 프로세서-프로세서는 메모리에 저장된 다음의 컴퓨터 실행가능한 컴포넌트를 실행하도록 구성됨-와, 작업에 의해 특정되는 필요조건에 기초하여 실행하기 위한 데이터 변환 작업을 스케줄링하도록 구성되는 제 1 컴포넌트와, 스케줄에 따라 타임라인 다이어그램 상에 상기 작업을 제시하도록 구성되는 제 2 컴포넌트와, 상기 타임라인 다이어그램 상에서 선택 작업을 식별하는 신호에 응답하여 데이터 의존성에 기초하여 상기 선택 작업과 관련된 하나 이상의 작업을 강조하도록 구성된 제 3 컴포넌트를 포함한다. 상기 제 3 컴포넌트는 또한 상기 선택 작업에 의존하는 하나 이상의 작업을 강조하도록 구성된다. 상기 제 3 컴포넌트는 또한 상기 선택 작업이 의존하는 하나 이상의 작업을 강조하도록 구성된다. 시스템은 작업 실행이 성공적인지 또는 성공적이지 않은지 검출하도록 구성된 제 4 컴포넌트를 더 포함한다. 시스템은 타임라인 상에서 성공적이지 않은 작업 실행의 표현을 선택하는 것에 응답하여 성공적이지 않은 작업 실행에 대한 의존성에 기초하여 관련된 성공적이지 않은 작업 실행 또는 예측되는 성공적이지 않은 작업 실행을 강조하도록 구성되는 제 5 컴포넌트를 더 포함한다.

[0052]

시스템은 데이터 변환 작업의 세트의 뷰를 생성하는 수단과, 데이터 변환 작업의 세트로부터 작업의 선택, 선택된 작업을 수신하는 수단과, 의존성 정보에 기초하여 선택된 작업에 대해 하나 이상의 관련된 작업을 자동으로 결정하는 수단과, 뷰에서 선택된 작업 및 하나 이상의 관련된 작업을 강조하기 위한 수단을 포함한다. 시스템은 데이터 변환 작업의 세트를 실행하기 위한 스케줄을 생성하는 수단을 더 포함한다. 시스템은 스케줄의 뷰를 생성하는 뷰 생성 수단을 더 포함한다. 시스템은 스케줄에 따라 데이터 변환 작업의 세트 중 하나를 실행하는 수단을 더 포함한다. 시스템은 데이터 변환 작업의 세트 중 하나의 수행의 성공 또는 실패를 검출하는 수단을 더 포함한다. 작업 수행 성공 또는 실패의 식별을 포함하는 스케줄의 뷰를 생성하는 뷔 생성 수단을 더 포함한다. 실패한 수행에 대한 의존성에 기초하여 실패했거나 실패할 것으로 예측되는 하나 이상의 관련된 작업 수행을 강조하는 강조 수단을 더 포함한다.

[0053]

본 개시의 측면들은 데이터 변환에 대한 것이거나, 즉, 분석 도구에 의한 후속 공개 또는 소비를 위해 사용가능한 형태로 데이터를 두는 것이다. 더 구체적으로, 측면들은 데이터 변환 작업을 스케줄링하고 모니터링하는 기술적인 문제에 관한 것이다. 이 문제를 해결하기 위해 이용되는 기술적 수단은 사용자 선택 작업과 관련된 작업

을 결정하는 것, 의존성에 기초하여, 선택된 작업에 의존하는 작업 및/또는 선택된 작업이 의존하는 작업을 포함하는 것을 포함한다. 작업의 선택 뿐만 아니라 선택된 작업 및 결정된 관련된 작업을 강조하는 것은 사용자가 작업 사이의 관련성을 이해하는 것과 트러블슈팅을 돋는 그래픽 사용자 인터페이스와 함께 구현된다. 그 결과로, 기술적인 효과는 작업 스케줄링 및 에러 감소와 관련한 향상된 사용자 효율성을 포함하지만 이에 제한되지 않는다.

[0054] 단어 "예시적인" 또는 이들의 다양한 형태는 본원에서 일례, 사례, 또는 예시로서 역할을 하는 것을 의미하도록 사용되었다. "예시적인" 것으로서 본원에서 설명된 임의의 측면 또는 설계는 반드시 다른 측면 또는 설계 보다 선호되거나 유리한 것으로서 해석되는 것은 아니다. 또한, 예시들은 단지 명료성 및 이해의 목적을 위해 제공되고 임의의 방식으로 청구된 청구 대상 또는 본 개시의 관련 부분을 제한하거나 한정하는 것을 의미하지 않는다. 각기 다른 범위의 수많은 추가적인 또는 대안의 예시들이 제시될 수 있지만, 간결성의 목적을 위해 생략되었다는 것을 이해할 수 있다.

[0055] 본원에서 사용된 바와 같이, 용어 "컴포넌트" 및 "시스템" 뿐만 아니라 이들의 다양한 형태(예를 들어, 컴포넌트, 시스템, 서브시스템...)는 컴퓨터 관련 엔티티, 하드웨어, 하드웨어 및 소프트웨어의 조합, 소프트웨어 또는 실행 중인 소프트웨어를 지칭하기 위한 것이다. 예를 들어, 컴포넌트는 프로세서 상에서 실행 중인 프로세스, 프로세서, 객체, 인스턴스, 실행가능한 실행 쓰레드, 프로그램 및/또는 컴퓨터가 될 수 있지만 이에 제한되지 않는다. 예시로서, 컴퓨터 상에서 실행 중인 애플리케이션 및 컴퓨터 모두가 컴포넌트가 될 수 있다. 하나 이상의 컴포넌트는 프로세스 내에 존재할 수 있고/거나 실행 쓰레드 및 컴포넌트는 하나의 컴퓨터 상에서 국부화되고/거나 둘 이상의 컴퓨터 사이에서 분산될 수 있다.

[0056] 달리 특정되거나 문맥으로부터 분명하지 않다면, 본 상세한 설명 및 첨부된 청구항에서 사용되는 접속사 "또는"은 배타적인 "또는"이라기 보다는 포괄적인 "또는"을 의미하는 것이다. 즉, "'X' 또는 'Y'"는 "X" 및 "Y"의 임의의 포괄적인 순열 조합을 의미하는 것이다. 예를 들어, "'A'가 'X'를 이용"하거나, "'A'가 'Y'를 이용"하거나 "'A'가 'X' 및 'Y'를 이용"하면, "'A'가 'X' 또는 'Y'를 이용한다"는 것이 전술한 경우 중 어떠한 경우에서도 만족된다.

[0057] 또한, 용어 "포함하는", "수반하는", "갖는", "가짐", 또는 이들 형태의 변형은 상세한 설명 또는 청구항에서 사용되는 결과로, 이러한 용어들은 용어 "포함함"이 청구항에서 전이어(transitional word)로서 이용될 때 해석되는 것과 유사한 방식으로 포괄적인 것이다.

[0058] 청구된 청구 대상을 위한 상황을 제공하기 위해, 도 18 뿐만 아니라 다음의 논의는 청구 대상의 다양한 측면들이 구현될 수 있는 적합한 환경의 간략한 일반적 설명을 제공하기 위한 것이다. 그러나, 적합한 환경은 단지 일례이고 사용 또는 기능의 범위에 대한 임의의 제한을 암시하는 것은 아니다.

[0059] 위에서 개시된 시스템 및 방법은 하나 이상의 컴퓨터 상에서 구동하는 프로그램의 컴퓨터 실행가능한 명령어의 일반적인 상황에서 설명될 수 있지만, 당업자는 측면들이 또한 다른 프로그램 모듈 또는 유사한 것과의 조합으로 구현될 수 있다는 것을 인식할 것이다. 일반적으로, 프로그램 모듈은, 특정 태스크를 수행하고/거나 특정 추상 데이터 타입을 구현하는 루틴, 프로그램, 컴포넌트, 데이터 구조 등을 포함한다. 또한, 당업자는 위의 시스템 및 방법이 단일 프로세서, 멀티 프로세서, 또는 멀티 코어 프로세서 컴퓨터 시스템, 미니 컴퓨팅 디바이스, 메인프레임 컴퓨터 뿐만 아니라 개인용 컴퓨터, 핸드헬드 컴퓨팅 디바이스(예를 들어, 개인용 디지털 보조장치(PDA), 전화기, 시계...), 마이크로프로세서 기반 또는 프로그래밍 가능한 소비자 또는 산업용 전자기기 등을 포함하는 다양한 컴퓨터 시스템 구성에 의해 실시될 수 있다는 것을 이해할 수 있다. 측면들은 또한 태스크가 통신 네트워크를 통해 연결된 원격 프로세싱 디바이스에 의해 수행되는 분산 컴퓨팅 환경에서 실시될 수 있다. 그러나, 청구된 청구 대상의 일부, 그렇지 않다면, 모든 측면들이 스템드 얼론 컴퓨터 상에서 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 로컬 및 원격 메모리 디바이스 중 하나 또는 모두에 위치될 수 있다.

[0060] 도 18을 참조하면, 예시의 범용 컴퓨터 또는 컴퓨팅 디바이스(1802)(예를 들어, 데스크탑, 랩탑, 태블릿, 시계, 서버, 핸드헬드, 프로그래밍 가능한 소비자 또는 산업용 전자기기, 셋탑 박스, 게임 시스템, 계산 노드...)가 도시된다. 컴퓨터(1802)는 하나 이상의 프로세서(1820), 메모리(1830), 시스템 버스(1840), 대용량 저장 디바이스(1850), 및 하나 이상의 인터페이스 컴포넌트(1870)를 포함한다. 시스템 버스(1840)는 적어도 상기 시스템 구성 요소를 통신 가능하게 연결한다. 그러나, 이의 가장 단순한 형태로 컴퓨터(1802)는 메모리(1830)에 저장된 다양한 컴퓨터 실행 가능한 동작, 명령어 및/또는 컴포넌트를 실행하는 메모리(1830)에 연결된 하나 이상의 프로세서(1820)를 포함할 수 있다는 것을 이해할 수 있다.

- [0061] 프로세서(1820)는 본원에 설명된 기능을 수행하도록 설계된 범용 프로세서, 디지털 신호 프로세서(DSP), 애플리케이션 특정 집적 회로(ASIC), 필드 프로그래밍 가능한 게이트 어레이(FPGA) 또는 다른 프로그램 가능한 로직 디바이스, 개별 게이트 또는 트랜지스터 로직, 개별 하드웨어 컴포넌트, 또는 이들의 임의의 조합으로 구현될 수 있다. 범용 프로세서는 마이크로프로세서가 될 수 있지만, 대안으로, 프로세서는 임의의 프로세서, 제어기, 마이크로제어기, 또는 상태 머신이 될 수 있다. 프로세서(1820)는 또한 컴퓨팅 디바이스의 조합, 예를 들어, DSP 및 마이크로프로세서의 조합, 복수의 마이크로프로세서, 멀티코어 프로세서, DSP 코어와 협업하는 하나 이상의 마이크로프로세서, 또는 임의의 다른 이러한 구성으로 구현될 수 있다. 하나의 실시예에서, 프로세서는 그 래피 프로세서가 될 수 있다.
- [0062] 컴퓨터(1802)는 컴퓨터(1802)의 제어를 제공하기 위해 다양한 컴퓨터 관독가능 매체를 포함하거나 인터랙팅하여 청구된 청구 대상의 하나 이상의 측면들을 구현할 수 있다. 컴퓨터 관독가능한 매체는 컴퓨터(1802)에 의해 액세스될 수 있는 임의의 사용가능한 매체가 될 수 있고 휘발성 및 비휘발성 매체, 및 제거가능한 및 제거불가능한 매체를 포함한다. 컴퓨터 관독가능 매체는 두 가지 구분된 상호 배타적인 타입, 즉 컴퓨터 저장 매체 및 통신 매체를 포함할 수 있다.
- [0063] 컴퓨터 저장 매체는 컴퓨터 관독가능한 명령어, 데이터 구조체, 프로그램 모듈 또는 다른 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현되는 휘발성 및 비휘발성, 제거가능한 및 제거불가능한 매체를 포함한다. 컴퓨터 저장 매체는 메모리 디바이스(예를 들어, 랜덤 액세스 메모리(RAM), 관독 전용 메모리(ROM), 전기로 삭제가능한 프로그래밍 가능한 관독 전용 메모리(EEPROM)...), 자기 저장 디바이스(예를 들어, 하드 디스크, 플로피 디스크, 카세트, 테이프...), 광학 디스크(예를 들어, 컴팩트 디스크(CD), 디지털 범용 디스크(DVD)...), 및 솔리드 스테이트 디바이스(예를 들어, 솔리드 스테이트 드라이브(SSD), 플래쉬 메모리 드라이브(예를 들어, 카드, 스틱, 키 드라이브...)) 또는 임의의 다른 유사한 매체와 같은 저장 디바이스를 포함하고, 컴퓨터(1802)에 의해 액세스가능한 원하는 정보를 전송 또는 전달하는 것과 반대된다. 따라서, 컴퓨터 저장 매체는 통신 매체를 포함하는 것 및 변조된 데이터 신호를 배제한다.
- [0064] 통신 매체는 컴퓨터 관독가능 명령어, 데이터 구조체, 프로그램 모듈, 또는 반송파 또는 다른 전달 메커니즘과 같은 변조된 데이터 신호의 다른 데이터를 구현하고 임의의 정보 전달 매체를 포함한다. 용어 "변조된 데이터 신호"는 하나 이상의 특성 세트를 갖거나 신호에서 정보를 인코딩하는 방식으로 변경되는 신호를 의미한다. 제한이 아닌 예시로서, 통신 매체는 유선 네트워크 또는 직도선(direct-wired) 접속, 및 음향, RF, 적외선 및 다른 무선 매체와 같은 무선 매체를 포함한다.
- [0065] 메모리(1830) 및 대용량 저장 디바이스(1850)는 컴퓨터 관독가능한 저장 매체의 예시들이다. 컴퓨팅 디바이스의 정확한 구성 및 타입에 따라, 메모리(1830)는 휘발성(예를 들어, RAM), 비휘발성(예를 들어, ROM, 플래쉬 메모리...), 또는 둘의 일부 조합이 될 수 있다. 예시로서, 기동하는 동안, 컴퓨터(1802) 내의 구성요소들 사이에서 정보를 전달하는 기본 루틴을 포함하는, 기본 입력/출력 시스템(BIOS)은 비휘발성 메모리에 저장될 수 있지만, 반면 무엇보다도 휘발성 메모리는 외부 캐시 메모리로서 동작하여 프로세서(1820)에 의한 프로세싱을 제공할 수 있다.
- [0066] 대용량 저장 디바이스(1850)는 메모리(1830)에 비해 상대적으로 대용량의 데이터의 저장을 위한 제거가능한/제거불가능한, 휘발성/비휘발성 컴퓨터 저장 매체를 포함한다. 예를 들어, 대용량 저장 디바이스(1850)는, 자기 또는 광학 디스크 드라이브, 플로피 디스크 드라이브, 플래쉬 메모리, 솔리드 스테이트 드라이브, 또는 메모리 스틱과 같은 하나 이상의 디바이스를 포함하지만 이에 제한되지 않는다.
- [0067] 메모리(1830) 및 대용량 저장 디바이스(1850)는 운영 시스템(1860), 하나 이상의 애플리케이션(1862), 하나 이상의 프로그램 모듈(1864), 및 데이터(1866)를 포함하거나 내부에 저장할 수 있다. 운영 시스템(1860)은 컴퓨터(1802)의 리소스를 제어하고 할당하도록 동작한다. 애플리케이션(1862)은 시스템 및 애플리케이션 소프트웨어 중 하나 또는 모두를 포함하고 프로그램 모듈(1864)을 통한 운영 시스템(1860)에 의한 리소스 및 메모리(1830) 및/또는 대용량 저장 디바이스(1850)에 저장된 데이터(1866)의 관리를 활용하여 하나 이상의 동작을 수행할 수 있다. 따라서, 이렇게 제공된 로직에 따라 애플리케이션(1862)은 범용 컴퓨터(1802)를 특수 머신으로 바꿀 수 있다.
- [0068] 청구된 청구 대상 중 전부 또는 일부는 표준 프로그래밍 및/또는 엔지니어링 기술을 사용하여 구현되어 개시된 기능을 실현하도록 컴퓨터를 제어하는 소프트웨어, 펌웨어, 하드웨어, 또는 이들의 임의의 조합을 생성한다. 제한이 아닌 예시로서, 작업 시스템(100) 또는 이들의 일부는, 애플리케이션(1862)이 되거나 일부를 형성하고, 메모리 및/또는 대용량 저장 디바이스(1850)에 저장되는 하나 이상의 모듈(1864) 및 데이터(1866)을 포함하며 이

들의 기능은 하나 이상의 프로세서(1820)에 의해 실행될 때 실현될 수 있다.

[0069]

특정 실시예에 따르면, 프로세서(1820)는 단일 집적 회로 기판 상에 하드웨어 및 소프트웨어 모두를 포함시키거나, 즉, 집적시키는, 시스템 온 칩(SOC) 또는 유사한 아키텍쳐에 대응할 수 있다. 여기서, 프로세서(1820)는 하나 이상의 프로세서 뿐만 아니라 프로세서(1820) 및 메모리(1830)와 적어도 유사한 메모리 등을 포함할 수 있다. 통상적인 프로세서는 최소량의 하드웨어 및 소프트웨어를 포함하고 외부 하드웨어 및 소프트웨어에 광범위하게 의존한다. 대조적으로, 특정 기능을 최소로 가능하게 하는 하드웨어 및 소프트웨어를 내부에 내장시키거나 외부 하드웨어 및 소프트웨어에 의존하지 않기 때문에, 프로세서의 SOC 구현은 더 강력하다. 예를 들어, 작업 시스템(100) 및/또는 연관된 기능은 SOC 아키텍처의 하드웨어 내에 내장될 수 있다.

[0070]

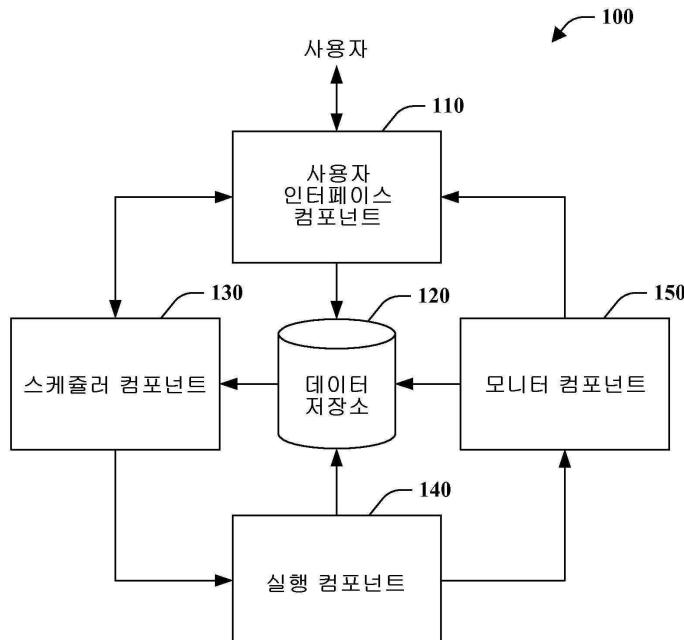
컴퓨터(1802)는 또한 시스템 버스(1840)에 통신가능하게 연결되고 컴퓨터(1802)와의 인터랙션을 제공하는 하나 이상의 인터페이스 컴포넌트(1870)를 포함한다. 예시로서, 인터페이스 컴포넌트(1870)는 포트(예를 들어, 직렬, 병렬, PCMCIA, USB, 파이어와이어...) 또는 인터페이스 카드(예를 들어, 음향, 비디오...) 또는 유사한 것이 될 수 있다. 일례의 구현예에서, 인터페이스 컴포넌트(1870)는 사용자 입력/출력 인터페이스로서 구현되어 예를 들어, 하나 이상의 입력 디바이스(예를 들어, 마우스, 트랙볼, 스타일러스, 터치 패드, 키보드, 마이크로폰, 조이스틱, 게임 패드, 위성 접시, 스캐너, 카메라, 다른 컴퓨터와 같은 포인팅 디바이스)를 통한 하나 이상의 제스처 또는 음성 입력에 의해, 사용자가 컴퓨터(1802)로 명령어 및 정보를 입력하는 것을 가능하게 한다. 다른 예시의 구현예에서, 인터페이스 컴포넌트(1870)는 출력 주변 인터페이스로서 구현되어 출력을 디스플레이(예를 들어, LCD, LED, 플라즈마...), 스피커, 프린터, 및/또는 다른 컴퓨터 등에 공급한다. 또한, 인터페이스 컴포넌트(1870)는 네트워크 인터페이스로서 구현되어 유선 또는 무선 통신 링크를 통해, 다른 컴퓨팅 디바이스(도시되지 않음)와의 통신을 가능하게 할 수 있다.

[0071]

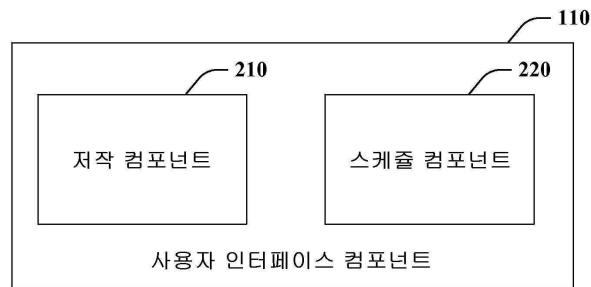
상술한 것은 청구된 청구 대상의 측면들의 예시를 포함한다. 물론, 청구된 청구 대상을 설명하는 목적을 위해 컴포넌트 또는 방법론의 모든 도출가능한 조합을 설명하는 것이 불가능하지만, 당업자는 개시된 청구 대상의 많은 추가 조합 및 순열 조합이 가능하다는 것을 인식할 수 있다. 따라서, 개시된 청구 대상은 첨부된 청구항의 사상 및 범위 내에 속하는 모든 이러한 변경, 수정, 및 변형을 포함하는 것이다.

도면

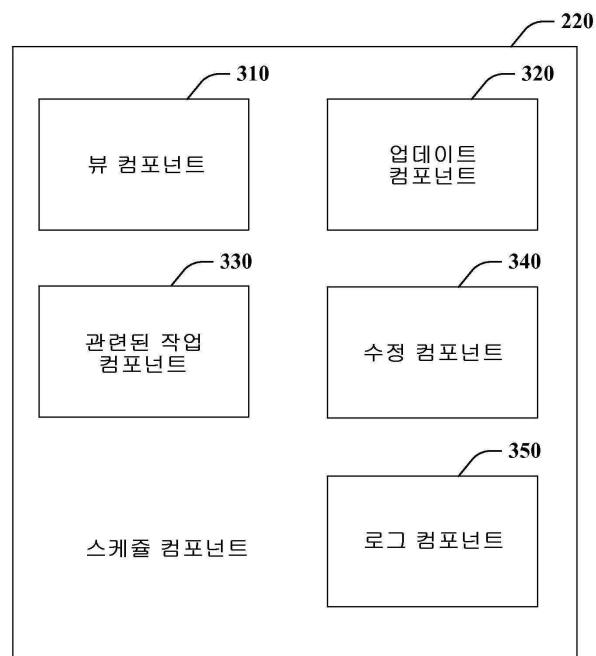
도면1



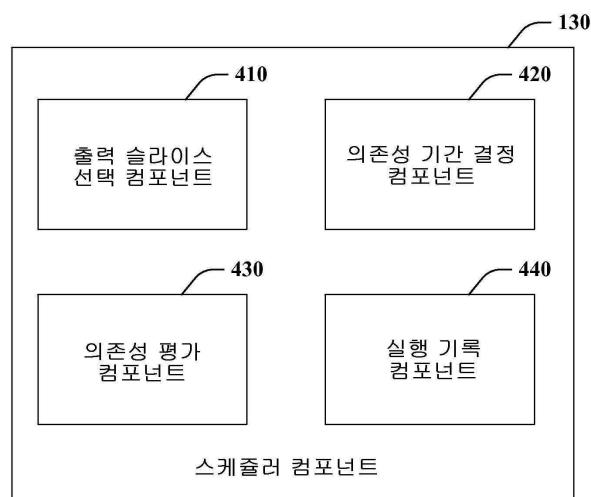
도면2



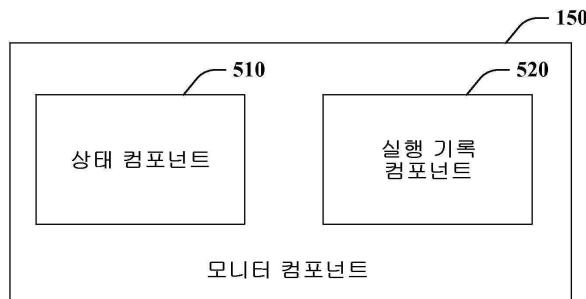
도면3



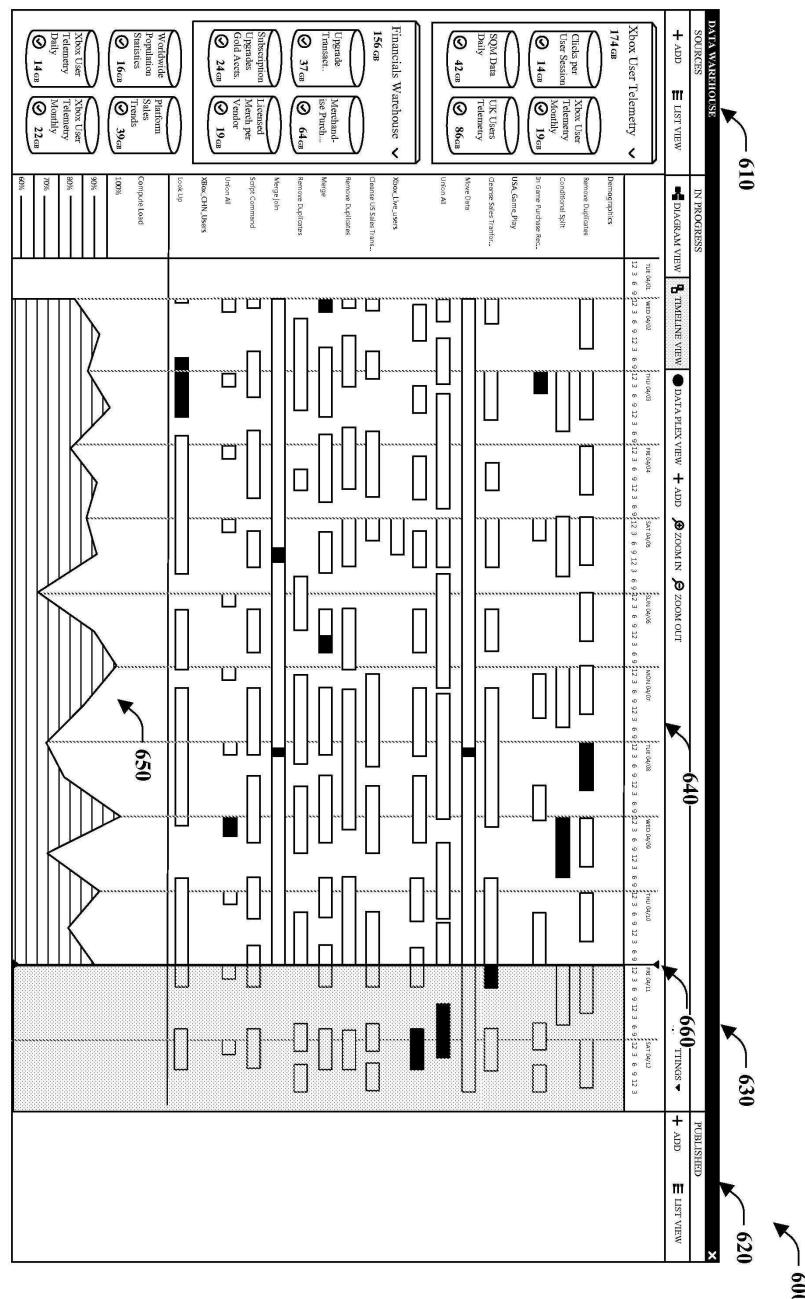
도면4



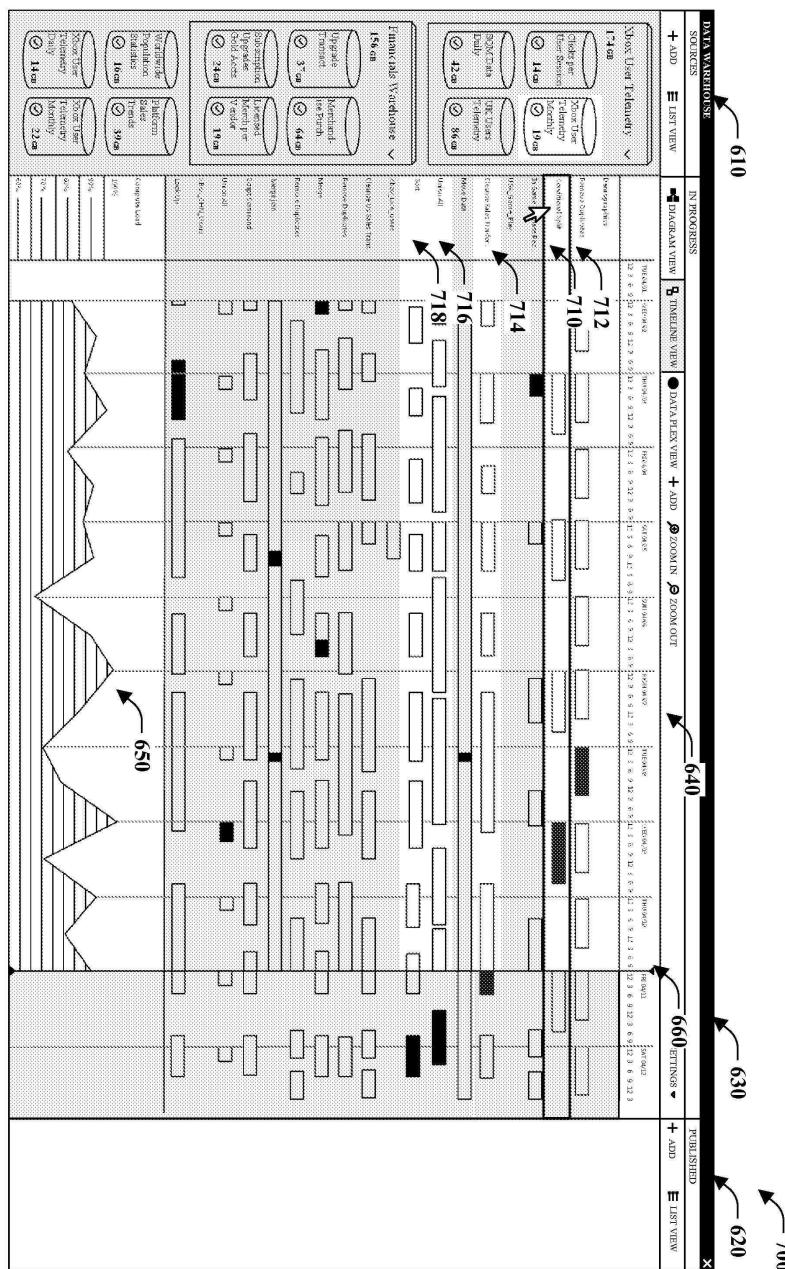
도면5



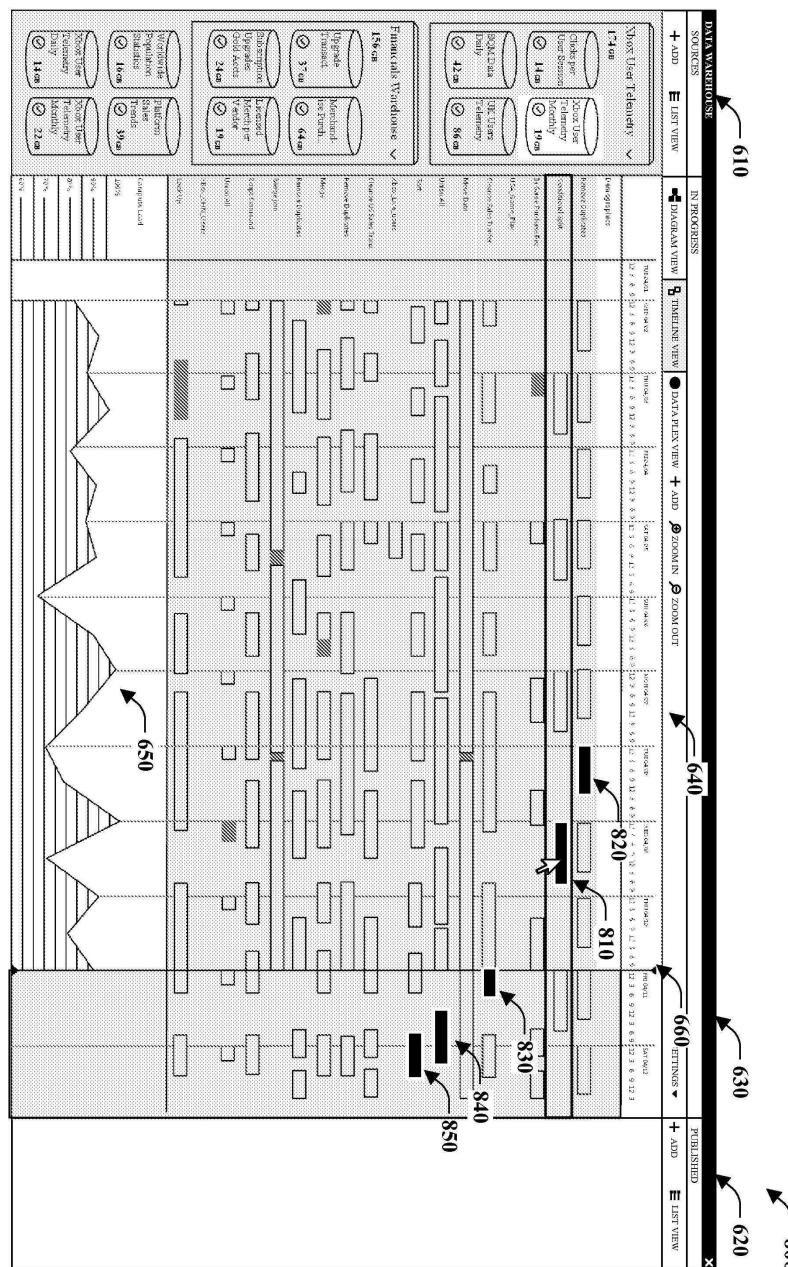
도면6



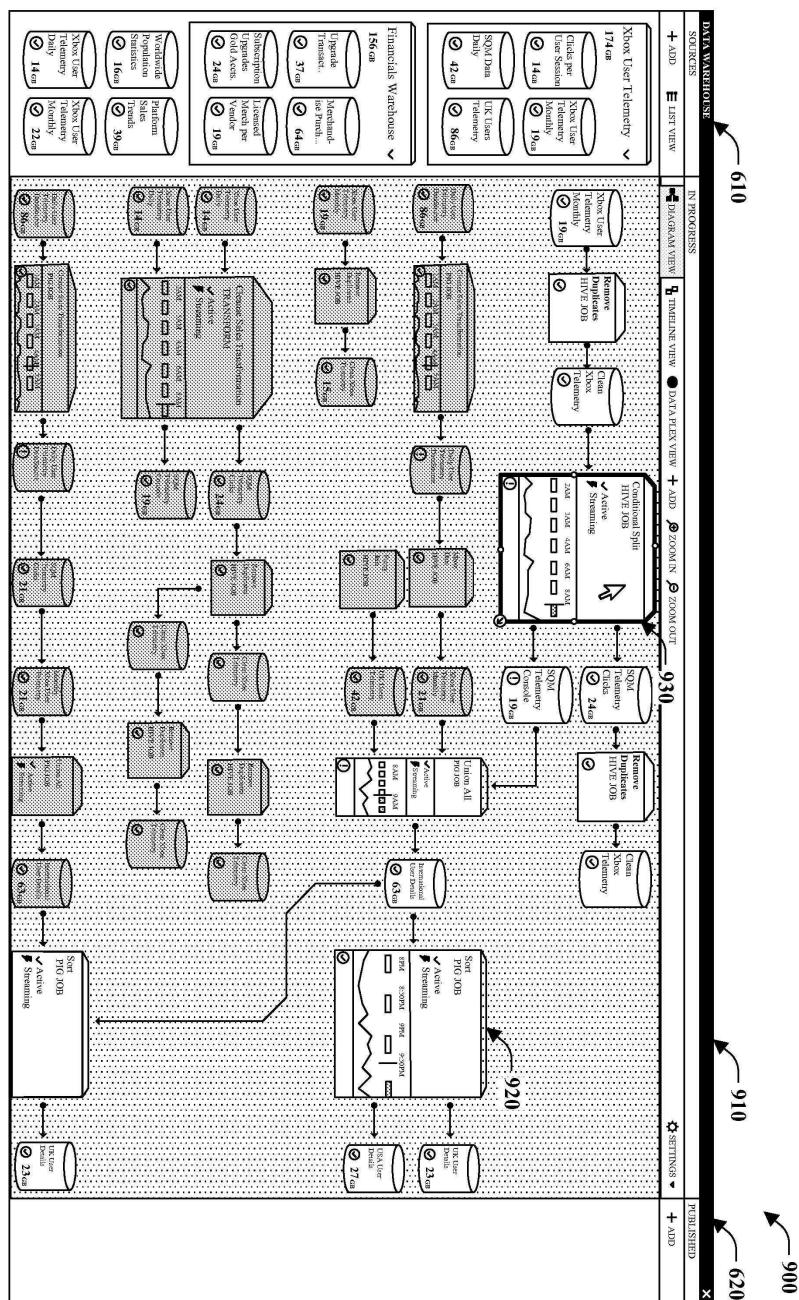
도면7



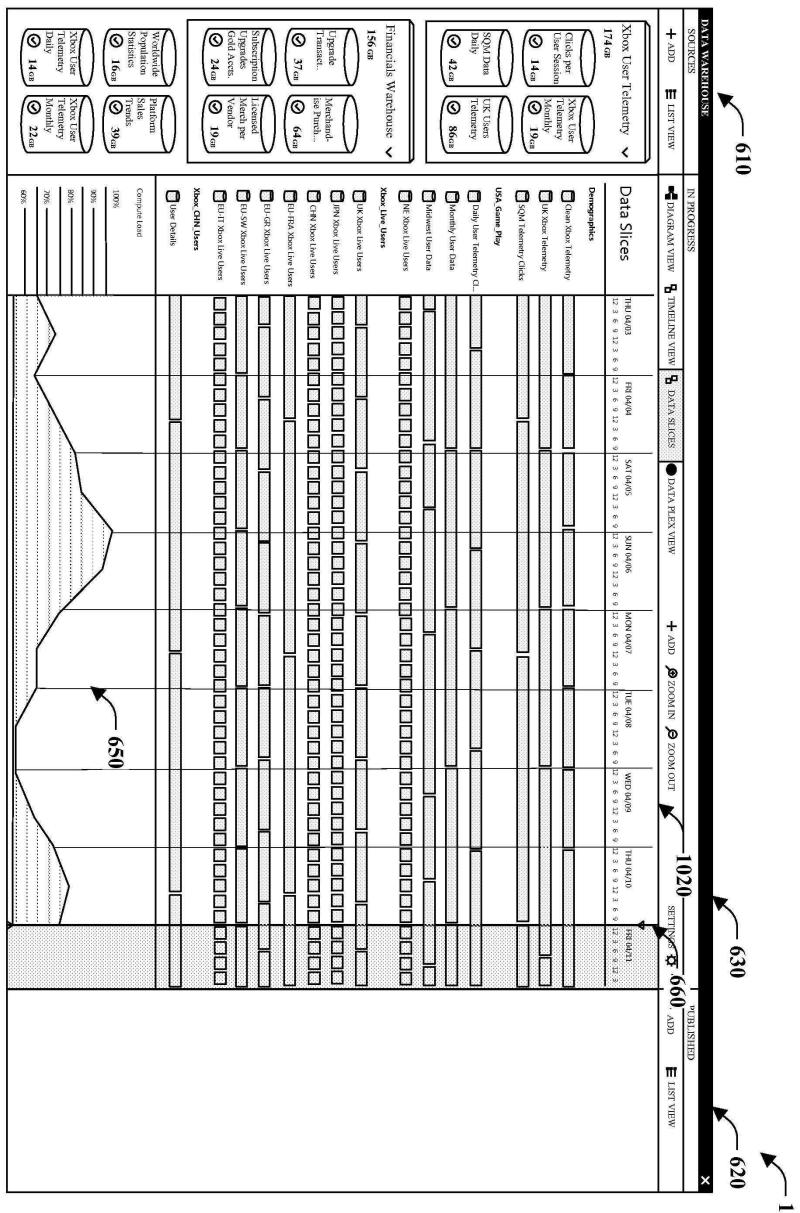
도면8



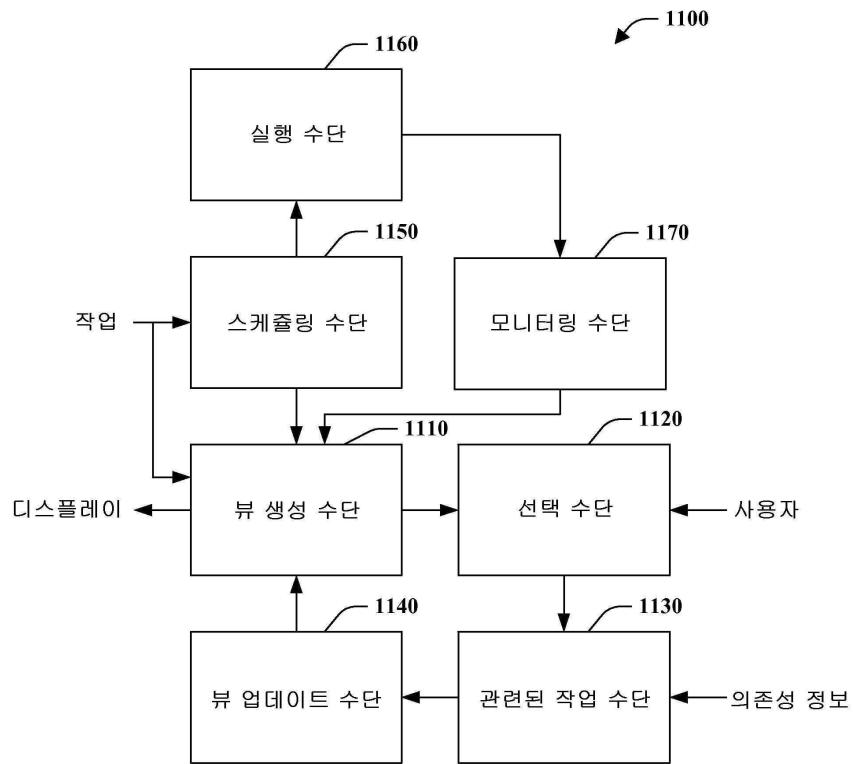
도면9



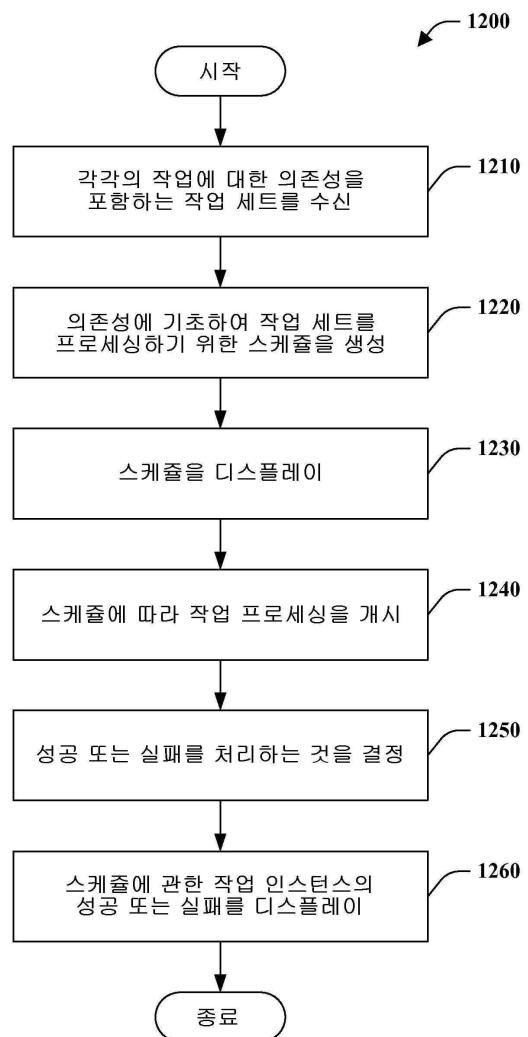
도면10



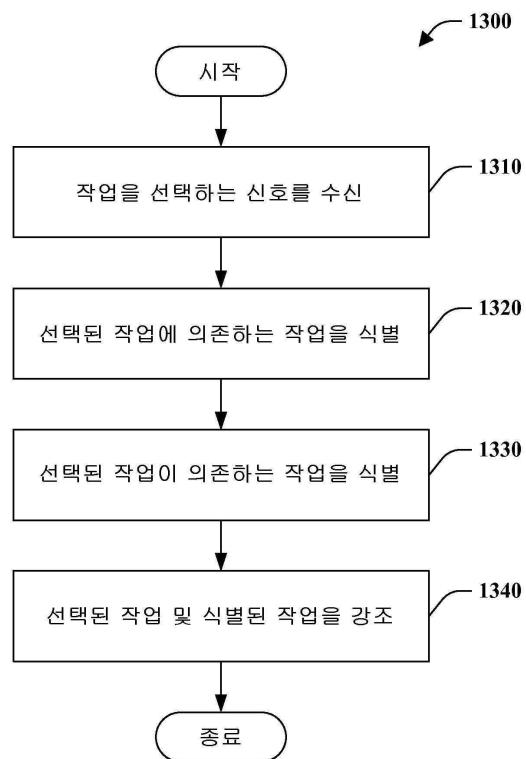
도면11



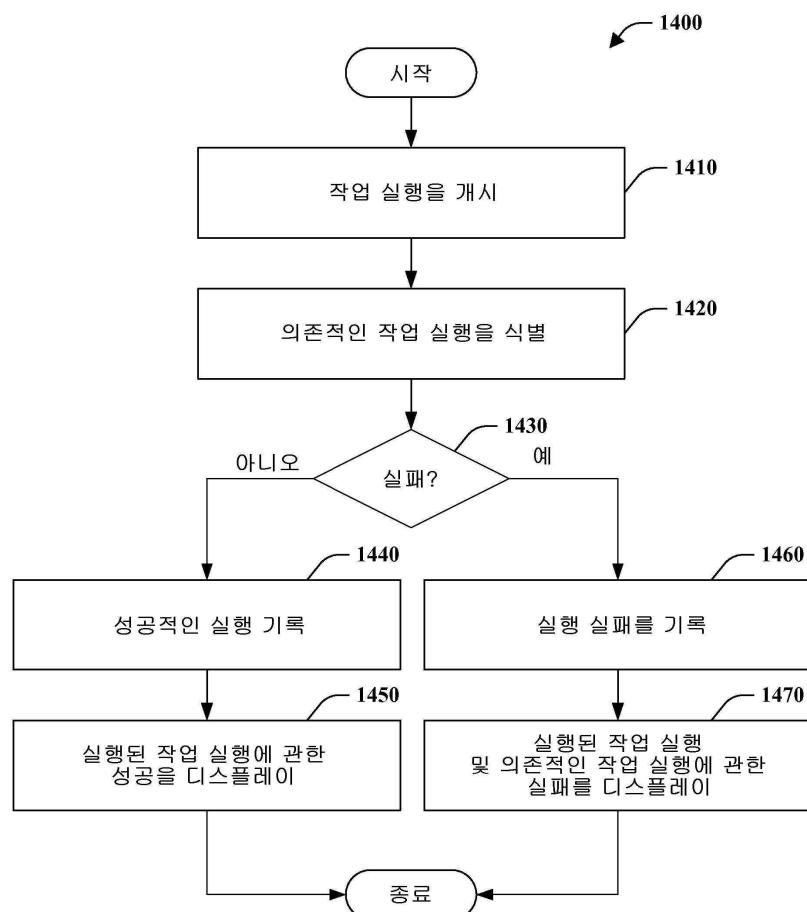
도면12



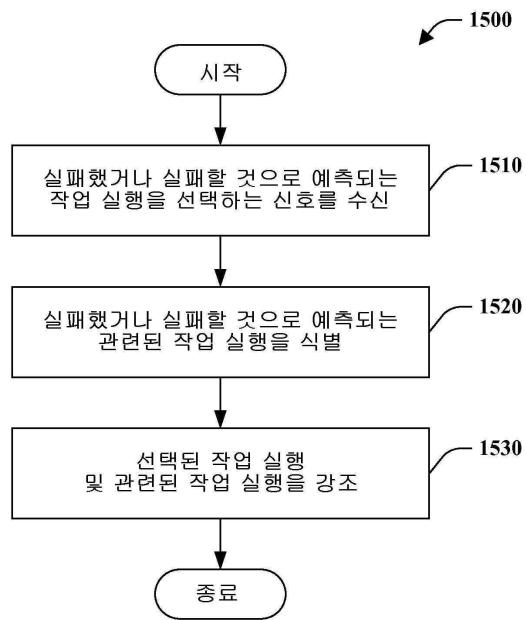
도면13



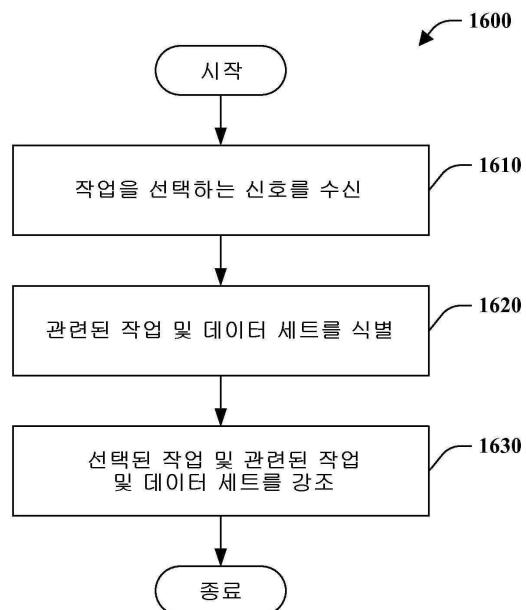
도면14



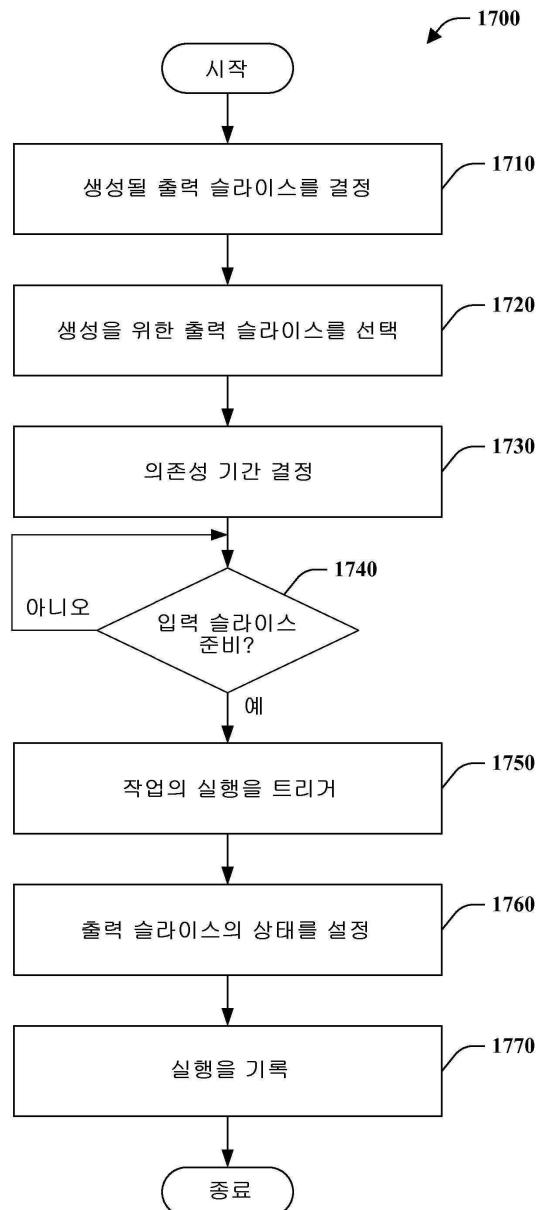
도면15



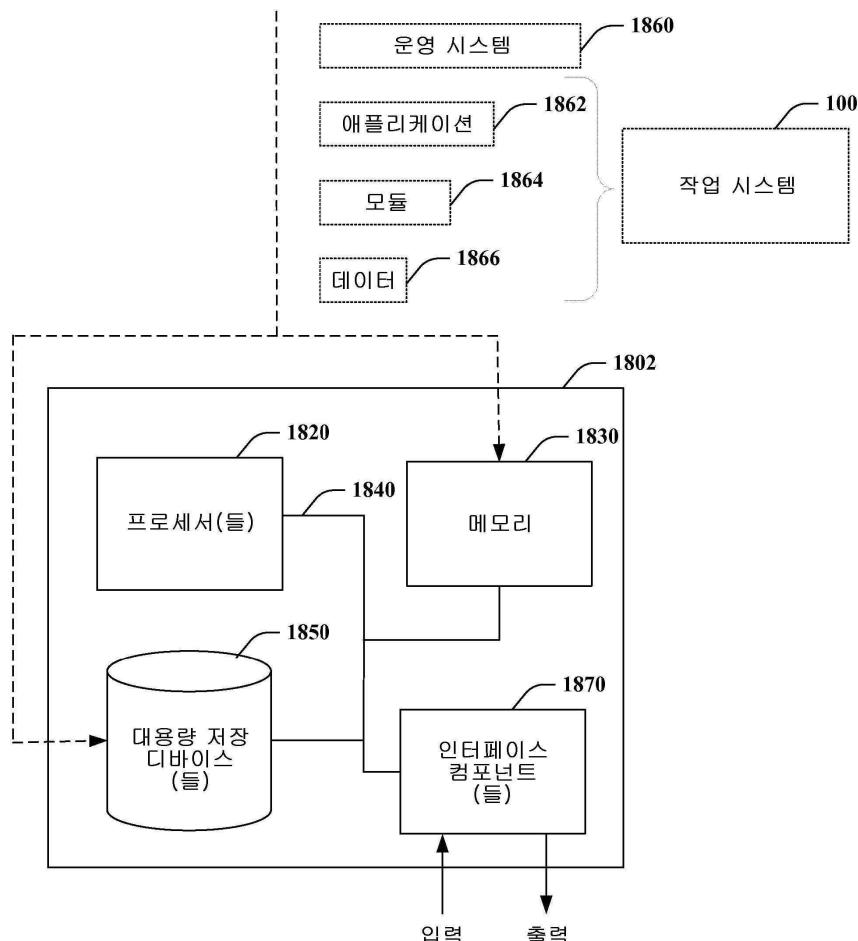
도면16



도면17



도면18



(예를 들어, 디스플레이, 스피커, 컴퓨터...)