



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2007년09월07일
(11) 등록번호 10-0756803
(24) 등록일자 2007년08월31일

(51) Int. Cl.

G06F 9/06 (2006.01)

(21) 출원번호 10-2005-0071672

(22) 출원일자 2005년08월05일

심사청구일자 2005년08월05일

(65) 공개번호 10-2006-0050256

공개일자 2006년05월19일

(30) 우선권주장

JP-P-2004-00231426 2004년08월06일 일본(JP)

(56) 선행기술조사문헌

JP 2005-216179 A

(뒷면에 계속)

전체 청구항 수 : 총 20 항

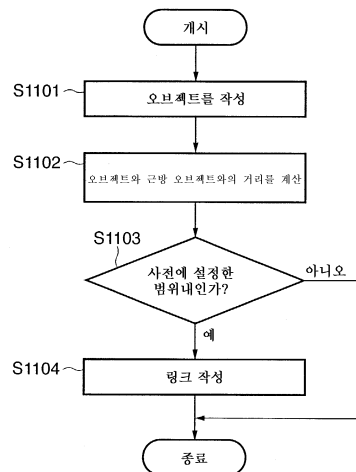
심사관 : 노지명

(54) 정보 처리 장치 및 정보 처리 방법

(57) 요약

본원 발명은, 자동 레이아웃 시스템에서 사용자가 설정 작업을 효율적으로 수행가능하도록 한다. 상기 목적을 달성하기 위해, 정보 처리 장치는 다음의 구성을 갖는다. 자동 레이아웃 기능을 구비하는 정보 처리 장치는, 컨테이너의 편집에서, 문서 상에 배치되는 제1 컨테이너와 문서 상에 배치되는 제2 컨테이너 간의 거리를 측정하여(단계 S1102), 상기 거리가 소정 거리 이하인지의 여부를 판정하며(단계 S1103), 상기 거리가 소정 거리 이하인 경우에, 상기 제1 컨테이너와 상기 제2 컨테이너 간에 링크를 자동 설정하는(단계 S1104) 기능을 구비한다.

대표도 - 도11



(56) 선행기술조사문헌

JP 2005-216180 A

KR 10-2001-0112686 A

KR 10-2004-0002427 A

US 2003/0229845 A1

KR 10-1999-0034152 A

KR 10-2002-0050295 A

특허청구의 범위

청구항 1

문서 상의 복수의 오브젝트를 연결하는 링크를 이용하여, 상기 오브젝트에 데이터가 유입되는 때의 상기 오브젝트의 레이아웃을 제어하는 제어 수단을 구비하는 정보 처리 장치에서, 상기 링크의 설정을 행하기 위한 정보 처리 방법으로서,

상기 링크를 설정하기 위한 거리를 설정하는 설정 공정과,

상기 문서 상에 배치되는 제1 오브젝트와, 상기 문서 상에 배치되어 있는 제2 오브젝트와의 거리를 계산하는 계산 공정과,

상기 설정 공정에서 설정된 거리와, 상기 계산 공정에서 계산된 상기 제1 및 제2 오브젝트 간의 거리에 기초하여, 링크를 생성할지의 여부를 판정하는 판정 공정과,

상기 판정 공정에서, 상기 링크를 생성해야 하는 것으로 판정된 경우에, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크를 작성하는 링크 작성 공정

을 포함하는 것을 특징으로 하는 정보 처리 방법.

청구항 2

제1항에 있어서,

상기 판정 공정은, 상기 제1 오브젝트가 상기 문서 상에 신규로 생성된 경우에, 상기 설정 공정에서 설정된 거리와 상기 계산 공정에서 계산된 거리에 기초하여, 상기 링크를 작성해야 할지의 여부를 판정하는 것을 특징으로 하는 정보 처리 방법.

청구항 3

제1항에 있어서,

상기 판정 공정은, 상기 제1 오브젝트가 상기 문서 상을 이동하는 경우에, 상기 설정 공정에서 설정된 거리와 상기 계산 공정에서 계산된 거리에 기초하여, 상기 링크를 작성해야 할지의 여부를 판정하는 것을 특징으로 하는 정보 처리 방법.

청구항 4

제1항에 있어서,

상기 판정 공정은, 상기 계산 공정에서 계산된 거리가 상기 설정 공정에서 설정된 거리 이하인지 여부를 판정하고,

상기 링크 작성 공정은, 상기 판정 공정에 의해, 상기 계산 공정에서 계산된 거리가 상기 설정 공정에서 설정된 거리 이하인 것으로 판정된 경우에, 상기 링크를 작성하는 것을 특징으로 하는 정보 처리 방법.

청구항 5

제1항에 있어서,

상기 링크 작성 공정은, 작성되는 링크의 종류를 설정할 수 있는 링크 설정 공정을 포함하는 것을 특징으로 하는 정보 처리 방법.

청구항 6

제1항에 있어서,

상기 링크 작성 공정은, 작성되는 링크의 사이즈를 설정할 수 있는 링크 사이즈 설정 공정을 포함하는 것을 특징으로 하는 정보 처리 방법.

청구항 7

제4항에 있어서,

상기 판정 공정에서, 상기 제1 오브젝트와 복수의 제2 오브젝트의 각각과의 거리가, 상기 설정 공정에서 설정된 거리 이하인 것으로 판정된 경우에는, 상기 링크 작성 공정은, 상기 제1 오브젝트와 상기 복수의 제2 오브젝트 간에 상기 링크를 작성하는 것을 특징으로 하는 정보 처리 방법.

청구항 8

제1항에 있어서,

상기 판정 공정에서, 상기 계산 공정에서 계산된 거리가 상기 설정 공정에서 설정된 거리 이하인 것으로 판정된 경우라도, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 링크 작성 공정은, 신규로 링크의 작성을 행하지 않는 것을 특징으로 하는 정보 처리 방법.

청구항 9

제1항에 있어서,

상기 판정 공정에서, 상기 계산 공정에서 계산된 거리가 상기 설정 공정에서 설정된 거리 이하가 아닌 것으로 판정된 경우로서, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 작성된 링크를 해제하는 링크 해제 공정을 더 포함하는 것을 특징으로 하는 정보 처리 방법.

청구항 10

제1항에 있어서,

상기 판정 공정에서, 상기 계산 공정에서 계산된 거리가 상기 설정 공정에서 설정된 거리 이하가 아닌 것으로 판정된 경우로서, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 작성된 링크를 유지하는 링크 유지 공정을 더 포함하는 것을 특징으로 하는 정보 처리 방법.

청구항 11

문서 상의 복수의 오브젝트를 연결하는 링크를 이용하여, 상기 오브젝트에 데이터가 유입되는 때의 상기 오브젝트의 레이아웃을 제어하는 제어 수단을 포함하며, 상기 링크의 설정이 가능한 정보 처리 장치로서,

상기 링크를 설정하기 위한 거리를 설정하는 설정 수단과,

상기 문서 상에 배치되는 제1 오브젝트와, 상기 문서 상에 배치되어 있는 제2 오브젝트와의 거리를 계산하는 계산 수단과,

상기 설정 수단에 의해 설정된 거리와, 상기 계산 수단에서 계산된 상기 제1 및 제2 오브젝트 간의 거리에 기초하여, 링크를 생성할지의 여부를 판정하는 판정 수단과,

상기 판정 수단에서 상기 링크를 작성해야 하는 것으로 판정된 경우에, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크를 작성하는 링크 작성 수단

을 포함하는 것을 특징으로 하는 정보 처리 장치.

청구항 12

제11항에 있어서,

상기 판정 수단은, 상기 제1 오브젝트가 상기 문서 상에 신규로 생성된 경우에, 상기 설정 수단에 의해 설정된 거리와 상기 계산 수단에 의해 계산된 거리에 기초하여, 상기 링크를 작성해야 할지의 여부를 판정하는 것을 특징으로 하는 정보 처리 장치.

청구항 13

제11항에 있어서,

상기 판정 수단은, 상기 제1 오브젝트가 상기 문서 상을 이동한 경우, 상기 설정 수단에 의해 설정된 거리와 상기 계산 수단에 의해 계산된 거리에 기초하여, 상기 링크를 작성해야 할지의 여부를 판정하는 것을 특징으로 하

는 정보 처리 장치.

청구항 14

제11항에 있어서,

상기 판정 수단은, 상기 계산 수단에 의해 계산된 거리가 상기 설정 수단에 의해 설정된 거리 이하인지 여부를 판정하고,

상기 링크 작성 수단은, 상기 판정 수단에 의해, 상기 계산 수단에 의해 계산된 거리가 상기 설정 수단에 의해 설정된 거리 이하인 것으로 판정된 경우에, 상기 링크를 작성하는 것을 특징으로 하는 정보 처리 장치.

청구항 15

제11항에 있어서,

상기 링크 작성 수단은, 작성되는 링크의 종류를 설정할 수 있는 링크 설정 수단을 포함하는 것을 특징으로 하는 정보 처리 장치.

청구항 16

제11항에 있어서,

상기 링크 작성 수단은, 작성되는 링크의 사이즈를 설정할 수 있는 링크 사이즈 설정 수단을 포함하는 것을 특징으로 하는 정보 처리 장치.

청구항 17

제14항에 있어서,

상기 판정 수단에 있어서, 상기 제1 오브젝트와 복수의 제2 오브젝트의 각각과의 거리가, 상기 설정 수단에 의해 설정된 거리 이하인 것으로 판정된 경우에는, 상기 링크 작성 수단은 상기 제1 오브젝트와 상기 복수의 제2 오브젝트 각각 사이에 상기 링크를 작성하는 것을 특징으로 하는 정보 처리 장치.

청구항 18

제11항에 있어서,

상기 판정 수단에 있어서, 상기 계산 수단에 의해 계산된 거리가 상기 설정 수단에 의해 설정된 거리 이하라고 판정된 경우라도, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 링크 작성 수단은, 신규로 링크의 작성을 행하지 않는 것을 특징으로 하는 정보 처리 장치.

청구항 19

제11항에 있어서,

상기 판정 수단에 있어서, 상기 계산 수단에 의해 계산된 거리가 상기 설정 수단에 의해 설정된 거리 이하가 아닌 것으로 판정된 경우로서, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 작성된 링크를 해제하는 링크 해제 수단을 더 포함하는 것을 특징으로 하는 정보 처리 장치.

청구항 20

제11항에 있어서,

상기 판정 수단에 있어서, 상기 계산 수단에 의해 계산된 거리가 상기 설정 수단에 의해 설정된 거리 이하가 아닌 것으로 판정된 경우로서, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크가 이미 작성되어 있는 경우에는, 상기 작성된 링크를 유지하는 링크 유지 수단을 더 포함하는 것을 특징으로 하는 정보 처리 장치.

청구항 21

삭제

명세서

발명의 상세한 설명

발명의 목적

종래기술의 문헌 정보

<52> [특허 문헌 1] 특개평7-129658호 공보(0049, 도 8)

발명이 속하는 기술 및 그 분야의 종래기술

- <53> 본 발명은, 문서 상의 복수의 오브젝트에 데이터를 유입시켜, 상기 오브젝트의 레이아웃을 해당 데이터에 따라 소정의 제어 조건 하에서 제어함으로써, 문서를 생성하는 정보 처리 장치에서, 상기 오브젝트의 레이아웃의 제어 조건을 설정하기 위한 정보 처리 기술에 관한 것이다.
- <54> 최근, CRM(Customer Relationship Management) 및 One-to-One 마케팅의 필요성이 주목받고 있다. 이것은 상품의 다품종화로 상품 수명이 짧아지고 있는 점, 인터넷 이용의 보급에 의한 소비자의 커스터마이징 서비스 지향 등의 이유에서이다. 이들의 방법은, 고객 만족도를 높여, 고객의 개척이나 확보를 목표로 한다는 목적에 대하여 매우 효과적인 것이다.
- <55> One-to-One 마케팅은 데이터베이스 마케팅의 일종으로, 고객의 연령, 성별, 취미, 기호, 구매 이력 등의 고객 개인 속성 정보를 데이터베이스화하여, 그 정보의 내용을 분석, 고객의 니즈(needs)에 맞는 제안을 행하는 것이다. one-to-one 마케팅의 대표적인 방법으로서 가변 프린트를 들 수 있다. 여기 최근에는 DTP(데스크탑 퍼블리싱) 기술의 진전과 디지털 인쇄 장치의 보급에 수반하여, 문서를 고객마다 커스터마이징하여 출력하는 가변 프린트 시스템이 개발되었다. 따라서, 고객마다 상이한 양의 콘텐츠를 최적으로 레이아웃하는 것이 요구되게 되었다.
- <56> 일반적으로, 가변 프린트 시스템에서 그와 같은 커스터마이징 문서를 작성할 때에는, 문서 상에 컨테이너(container)를 레이아웃한다. 컨테이너란 데이터베이스로부터의 콘텐츠(묘화 내용)를 묘화하기 위한 부분 영역인 오브젝트를 말한다(필드 영역이라 부르는 경우도 있음). 즉, 문서 상에 이러한 컨테이너를 레이아웃하여, 데이터베이스와 컨테이너를 관련시키는(데이터베이스의 콘텐츠와 컨테이너를 관련시킴) 등의 작업에 의해, 커스터마이징 문서를 작성한다.
- <57> 그러나, 가변 프린트 시스템의 경우, 텍스트 및 이미지의 컨테이너의 사이즈가 고정되어 있다. 따라서, 데이터베이스로부터의 데이터가 컨테이너에 삽입될 때에, 데이터량이 컨테이너 사이즈보다 크면 텍스트의 오버랩 또는 이미지의 클리핑이 발생한다. 데이터량이 컨테이너 사이즈보다 작으면 컨테이너에 간극(gap)을 갖게 되는 문제가 있었다.
- <58> 이러한 문제를 해결하기 위해, 자동 레이아웃 시스템이 제안되고 있다. 자동 레이아웃 시스템이란, 콘텐츠를 사용하여 컨테이너의 레이아웃을 동적으로 변경하는 시스템이다. 삽입되는 텍스트 또는 이미지의 데이터량에 따라 컨테이너 사이즈를 가변으로 설정하는 것이 가능하다.
- <59> 구체적으로는, 자동 레이아웃 시스템은, 컨테이너의 사이즈를 가변으로 하여, 삽입되는 데이터량에 따라 컨테이너의 사이즈를 크게 하여 레이아웃한다. 텍스트인 경우에는, 컨테이너 내의 폰트의 사이즈를 가변으로 하여, 삽입되는 데이터량에 따라 폰트의 사이즈를 축소하여 레이아웃한다. 따라서, 컨테이너 용량을 초과하는 데이터량의 텍스트가 컨테이너에 삽입된 경우라도, 컨테이너 내에 모든 텍스트를 표시할 수 있다.
- <60> 그러나, 컨테이너의 사이즈를 가변으로 하면 데이터량이 큰 컨테이너의 경우, 동일한 문서 상의 다른 컨테이너의 상부에 놓일 수 있다. 또 다른 경우, 컨테이너의 사이즈를 고정하여 폰트의 사이즈를 가변으로 하면, 텍스트의 데이터량이 큰 경우, 폰트 사이즈가 너무 작게 레이아웃되는 등의 경우가 발생할 수 있다.
- <61> 이 때문에, 상기 자동 레이아웃 시스템에는, 인접하는 컨테이너끼리 관련시키는 링크 기능(오브젝트에 데이터를 유입시켜 얻어진 오브젝트의 레이아웃이, 상기 오브젝트에 인접하는 또 다른 오브젝트의 레이아웃에 기초하여 정해지도록, 인접하는 오브젝트를 간을 관련시키기 위한 기능)을 포함한다. 이것은 사이즈가 커진 특정한 컨테이너에 인접하는 컨테이너의 사이즈를 자동적으로 크게함으로써 상기 문제를 해결한다(예를 들면, 상기 특허 문헌 1 참조).

발명이 이루고자 하는 기술적 과제

<62> 그러나, 상기 자동 레이아웃 시스템에서는, 문서 상에 복수의 컨테이너를 배치하고, 각각의 컨테이너끼리 관련 시킴에 있어서 유저는 해당 컨테이너들 간의 링크를 각각 수작업으로 설정해야 한다. 그러나, 전술한 One-to-One 마케팅에서는 고객의 니즈에 맞게 카탈로그나 팜플렛을 작성하기 때문에, 고객에 따라서는 많은 정보를 게재할 필요가 있다. 따라서, 콘텐츠가 유입되는 컨테이너 수의 증가에 수반하여, 컨테이너 간을 관련시키는 링크의 수도 증가하는 것을 생각할 수 있다. 이 경우에, 상기 기술과 같이 컨테이너 간을 관련시키는 링크의 설정을 하나 하나 수작업으로 행하고 있어서는, 해당 설정 작업의 부하가 증대되는 등의 문제가 있다.

<63> 본 발명은, 상기 과제를 감안하여 이루어진 것이다. 본 발명의 목적은, 각 컨테이너 간을 관련시키는 링크를 설정할 때에, 유저가 미리 설정한 소정의 거리 내에 복수의 오브젝트가 포함된 경우에 자동적으로 링크를 설정함으로써, 레이아웃 시스템에서의 유저의 설정 작업의 효율화를 도모하는 것이다.

발명의 구성 및 작용

<64> 상기의 목적을 달성하기 위해 본 발명에 따른 정보 처리 방법은 이하와 같은 구성을 구비한다.

<65> 즉, 문서 상의 복수의 오브젝트를 연결하는 링크를 이용하여, 상기 오브젝트에 데이터를 유입시켰을 때의 상기 오브젝트의 레이아웃을 제어하는 제어 수단을 구비하는 정보 처리 장치에서, 상기 링크의 설정을 행하기 위한 정보 처리 방법으로서, 상기 문서 상에 배치되는 제1 오브젝트와, 상기 문서 상에 배치되는 제2 오브젝트와의 거리를 계산하는 계산 공정과, 상기 계산 공정에서 계산된 상기 제1 및 제2 오브젝트 간의 거리에 기초하여, 링크의 작성을 행할지 여부를 판정하는 판정 공정과, 링크를 작성하라고 판정된 경우에, 상기 제1 오브젝트와 상기 제2 오브젝트 간에 상기 링크를 작성하는 링크 작성 공정을 포함한다.

<66> <실시예>

<67> 우선, 본 실시예를 상세히 설명한다. 일반적으로 자동 레이아웃 시스템에서의 문서의 작성은, 레이아웃 편집 처리와 자동 레이아웃 처리로 구분할 수 있다. 전자는 컨테이너를 문서 상에 배치함과 함께, 데이터를 유입시킨 경우에 어떻게 컨테이너의 레이아웃을 제어할지, 그 제어 조건을 설정하기 위한 처리이다. 후자는 컨테이너에 데이터를 유입시키고, 설정된 제어 조건에 따라, 해당 유입된 데이터에 따라 레이아웃을 제어하여, 문서를 생성하기 위한 처리이다.

<68> 자동 레이아웃 시스템을 구성하는 본 실시예에 따른 호스트 컴퓨터(정보 처리 장치)에서는, 레이아웃 편집 처리에서의 제어 조건의 설정 항목 중 하나인 링크의 설정을 자동적으로 행하는 자동 링크 기능을 구비함으로써, 유저의 설정 작업의 효율화를 도모하고 있다. 구체적으로는, 종래의 레이아웃 편집 처리에서는, 문서 상에 컨테이너를 배치한 후에, 해당 컨테이너들 간에 각 링크를 설정한다. 본 발명에 따른 자동 링크 기능을 이용하면, 서로 소정의 거리 이하에 컨테이너를 배치한 것만으로, 자동적으로 링크를 설정하는 것이 가능하게 된다. 이로 인해 링크 설정에서 유저에게 부과되는 부하가 매우 감소하게 된다.

<69> 1. 자동 레이아웃 시스템의 구성예

<70> 먼저, 도 1a 및 도 1b를 참조하여, 본 발명의 일 실시예에 따른 호스트 컴퓨터(정보 처리 장치)를 구비하는 자동 레이아웃 시스템의 구성을 설명한다. 도 1a는 자동 레이아웃 시스템(100)의 구성예를 도시하는 블록도이다. 또한, 도 1b는 도 1a에 도시한 호스트 컴퓨터(정보 처리 장치)(101)의 구성을 보다 상세하게 도시하는 블록도이다.

<71> 본 실시예에서 설명되는 레이아웃 편집 처리 및 자동 레이아웃 처리는, 호스트 컴퓨터(101)(범용 컴퓨터 모듈로 구성됨)에 의해 실행된다. 자동 레이아웃 시스템(100) 상에서 실시 가능하게 되는 레이아웃 편집 어플리케이션 프로그램(121)은 호스트 컴퓨터(101)에서, 그 소프트웨어의 전체, 혹은 일부분이 실행된다. 이에 따라, 상기 레이아웃 편집 처리 및 자동 레이아웃 처리가 실현된다.

<72> 레이아웃 편집 어플리케이션 프로그램(121)은 컴퓨터의 판독 가능 매체에 저장고, 그 컴퓨터의 판독 가능 매체로부터 호스트 컴퓨터(101)의 메모리(136)에 로드되고, 실행된다. 그와 같은 소프트웨어나 컴퓨터 프로그램을 저장한 컴퓨터의 판독 가능 매체는 컴퓨터 프로그램 제품이다. 컴퓨터에서 그 컴퓨터 프로그램을 사용함으로써, 문서의 작성이나 작성된 문서의 인쇄에 적합한 장치가 제공되게 된다.

<73> 도 1b에 도시한 바와 같이, 호스트 컴퓨터(101)에는, 입출력 인터페이스(143)를 통해 키보드(132)나, 마우스

(133)와 같은 포인팅 디바이스 등이 입력 장치로서 접속되어 있다. 또한, 출력 장치로서의 디스플레이 장치(144)가 비디오 인터페이스(137)를 통해 접속되어 있다. 또한, 로컬 프린터(145) 등을 입출력 인터페이스(138)를 통해 접속하는 것도 가능하다. 또한, 입출력 인터페이스(138)는 컴퓨터 모듈(101)을 네트워크(107)에 접속하는 기능도 갖는다. 이에 따라, 네트워크를 통해 다른 컴퓨터 장치에 호스트 컴퓨터(101)를 접속할 수 있다. 네트워크(107)의 전형적인 예로서는, 근거리 통신망(LAN), 혹은 원거리 통신망(WAN)을 들 수 있다.

<74> 또한, 도 1b에 도시한 바와 같이, 호스트 컴퓨터(101)는 적어도 하나의 프로세서 유닛(135), 예를 들면 반도체의 랜덤 액세스 메모리(RAM)나 리드 온리 메모리(ROM)로 구성되는 메모리 유닛(136)을 포함하고 있다. 저장 디바이스(139)는, 프로그램 등을 저장하는 컴퓨터 판독 가능 매체 사이에서 데이터의 교환이 가능한 하드디스크 드라이브(140)나 플로피(등록 상표) 디스크 드라이브(141)를 포함한다. 또한, 도 1b에는 나타내고 있지 않지만, 자기 테이프 드라이브 등도 저장 디바이스(139)로서 사용 가능하다. CD-ROM 드라이브(142)는 불휘발성 데이터 소스로서 제공된다(물론, CD-ROM에 의해 컴퓨터 프로그램을 제공하여도 됨).

<75> 호스트 컴퓨터(101)는, GNU/LINUX나 마이크로소프트 Windows(등록 상표)와 같은 오퍼레이팅 시스템이나, 전형적으로는 오퍼레이팅 시스템에 따른 형태나, 혹은 관련이 있는 기술로 알고 있는 것에 의해 형성된 컴퓨터 시스템의 종래의 오퍼레이션 모드에 의한 방법에 의해, 상호 접속 버스(134)를 통해 통신을 행하는 컴퓨터 모듈(101)의 컴포넌트(135 내지 143)를 이용한다. 즉, 전술한 135 내지 143으로 나타내는 각 구성은, 버스(134)를 통해 통신 가능하게 접속되어 있으며, 호스트 컴퓨터(101)에 인스톨된 오퍼레이팅 시스템에 의해 이용된다.

<76> 또한, 도 1b에 도시한 호스트 컴퓨터(101)의 예로서는, IBM 호환 PC이나 SUN의 Sparcstation, 혹은 이들을 포함한 컴퓨터 시스템이 있다.

<77> 본 실시예에서는, 레이아웃 편집 어플리케이션 프로그램(121)은 하드디스크 드라이브(140)에 상주한다. 프로세서(135)는 레이아웃 편집 어플리케이션 프로그램(121)의 실행 및 로딩을 제어한다. 또한, 레이아웃 편집 어플리케이션 프로그램(121)의 매체 기억 장치와 네트워크(107)로부터 켓치(fetch)되는 데이터에 대하여는 하드디스크 드라이브(140)에 호응하여 반도체 메모리(136)가 사용된다.

<78> 하나의 예에서는, 레이아웃 편집 어플리케이션(121)의 인코딩된 프로그램은, CD-ROM이나 플로피(등록 상표) 디스크 상에 저장된다. 그 프로그램은 대응하는 드라이브(142나 141)를 통해 로딩되고, 하드디스크 드라이브(140)에 인스톨된다. 다른 예로서, 레이아웃 편집 어플리케이션 프로그램(121)은 네트워크(107)로부터 호스트 컴퓨터(101)로 로딩되고, 하드디스크 드라이브(140)에 인스톨되어도 된다.

<79> 또한 소프트웨어는, 자기 테이프 또는 ROM 또는 집적 회로, 광 자기 디스크, 또는, 호스트 컴퓨터(101)와 그 밖의 디바이스 간에서의 적외선 등의 무선 통신, PCMCIA 카드와 같은 컴퓨터 판독 가능 카드, 그리고 전자 메일 통신이나 WEB 사이트 상의 기록 정보를 갖는 인터넷이나 인트라넷을 포함하는 다른 적당한 컴퓨터로부터 호스트 컴퓨터(101) 내로 로드되어도 된다. 이들은, 컴퓨터 판독 가능 매체의 예이며, 다른 컴퓨터 판독 가능 매체가 사용되어도 되는 것은 분명하다.

<80> 도 1a에서, 레이아웃 편집 어플리케이션(121)은 컴퓨터에 자동 레이아웃 처리 및 레이아웃 편집 처리를 행하게 하는 것이다. 레이아웃 편집 어플리케이션(121)은 2개의 소프트웨어 컴포넌트, 즉 레이아웃 엔진(105)과 유저 인터페이스(103)를 포함하고 있다.

<81> 레이아웃 엔진(105)은, 부분 영역인 오브젝트(사각형의 범위, 예를 들면 컨테이너)의 사이즈나 위치의 설정에 따라, 후술하는 바와 같이, 자동 레이아웃 처리를 실행하는 소프트웨어 컴포넌트이다. 레이아웃 엔진은 데이터 베이스(119)에 저장되어 있는 데이터로부터 각 레코드를 하나씩 로딩하고, 로딩된 데이터와 오브젝트의 설정을 기초로, 판독한 데이터가 유입되는 오브젝트의 사이즈나 위치를 계산한다. 또한, 본 실시예에서는, 레이아웃 엔진(105)은, 또한, 오브젝트에 할당된 데이터를 묘화하여, 문서의 이미지를 생성하는 처리도 행한다. 단, 본 발명은 이것에 한정되는 것은 아니다. 레이아웃 엔진(105)은 각 부분 영역(오브젝트)의 사이즈와 위치를 결정하는 어플리케이션으로서 동작할 수 있다. 또한, 레이아웃 엔진(105)은 프린터 드라이버(도시 생략)로 묘화 정보를 출력함으로써, 프린터 드라이버가 문서의 이미지 묘화 처리를 행하여, 인쇄 데이터를 생성하도록 하여도 된다.

<82> 한편, 유저 인터페이스(103)는, 유저에 의한 레이아웃 편집 처리를 가능하게 한다. 구체적으로는, 유저는 인터페이스(103)를 통해 오브젝트의 작성 및 이동을 행하여, 문서 템플릿을 작성한다. 유저 인터페이스(103)는 문서 템플릿 내의 각 오브젝트와 데이터를 관련시키는 메카니즘을 제공한다. 또한, 유저 인터페이스(103)는 문서 템플릿 내의 각 오브젝트에 대한 각종 설정(링크 설정도 포함함)을 가능하게 한다. 유저 인터페이스(103)와 레

이아웃 엔진(105)은 커뮤니케이션 채널(123)을 통해 통신한다.

<83> 2. 다른 시스템 구성의 예

<84> 도 2는, 도 1a와 유사의 블록도이지만, 엔진 서버(227)가 추가되어 있는 점이 상이하다. 엔진 서버(227)에 저장되어 있는 레이아웃 엔진(225)은, 레이아웃 엔진(105)의 분리 버전이다. 엔진 서버(227)에는 일반적인 컴퓨터가 이용된다. 레이아웃 엔진(225)은, 인쇄나 그 밖의 목적에 따라 문서를 생성하기 위해, 파일 서버(115)에 보존된 문서 템플릿과 데이터베이스(119)에 보존된 데이터를 결합한다. 그와 같은 오퍼레이션은 유저 인터페이스(103)를 통해 요구된다.

<85> 3. 레이아웃 편집 어플리케이션의 설명

<86> 이하, 상기 레이아웃 편집 어플리케이션(121) 중, 유저 인터페이스(103)를 통해 실현되는 레이아웃 편집 처리의 상세에 대하여 설명한다.

<87> 3-1 메인 윈도우

<88> 유저 인터페이스(103)는, 조작 시에 도 3에 도시한 바와 같은 어플리케이션 윈도우(301)로 형성된 유저 인터페이스 화면을 비디오 디스플레이(144)에 표시시킨다. 이 윈도우(301)는, 메뉴 바(302), 툴 바(303), 워크 에리어(306)와 옵션의 팔레트(pallet; 311)를 갖는다. 윈도우(301)는 메뉴 바(302)와 툴 바(303)를 비표시로 하게 하거나, 스크린 상의 여러가지의 위치로 이동시키는 것이 가능하다. 또한, 워크 에리어(306)는 마우스(133)의 조작에 의해 그 위치를 이동시키는 것이 가능하다. 또한, 팔레트(311)는 옵션이고, 커서/포인터 디바이스(313)는 마우스(133)가 지시하는 위치를 나타낸다.

<89> 메뉴 바(302)는, 주지의 기술로서 알려져 있듯이, 메뉴 동작 옵션의 계층의 아래에 확장되는 많은 메뉴 아이템(304)을 갖는다.

<90> 툴 바(303)는, 어플리케이션이 특별한 모드에 의해 비표시 상태로 하거나, 또는 표시 상태로 하는 것이 가능한 많은 툴 버튼과 위젯(widget; 305)을 갖는다.

<91> 룰러(ruler; 308)는 옵션으로, 워크 에리어 내의 포인터, 페이지, 라인, 마진 가이드, 오브젝트의 위치를 나타내기 위해 사용된다.

<92> 팔레트(311)는 이동, 리사이즈(resizing), 클로즈를 하기 위한 윈도우 컨트롤(312)을 갖는다. 팔레트(311)는 옵션으로, 워크 에리어의 전면에 표시되거나, 혹은 오브젝트의 배면에 숨겨진다. 팔레트(311)는 어플리케이션 윈도우(301)의 범위 내에만 표시되도록 할 수도 있거나, 혹은 어플리케이션 윈도우(301)의 외측에 그 일부 혹은 전체가 표시되도록 할 수도 있다.

<93> 툴 바(303)에는 도 4에 도시한 바와 같은, 유저 선택 가능한 "버튼"이 배치되어 있다.

<94> (1) 선택 툴 버튼(403): 오브젝트의 변을 선택, 이동, 사이즈 변경, 리사이즈 그리고 로크(lock)/로크 해제(unlock)를 위해 사용된다. 오브젝트의 선택은, 오브젝트의 주위에 선택 박스를 드래그함으로써 이루어진다. 또한, CTRL 키를 억지하면서, 복수의 오브젝트에 대하여 선택 조작을 함으로써, 복수의 오브젝트를 선택 가능하다.

<95> (2) 텍스트 오브젝트 툴 버튼(404): 스태틱 혹은 밸류어블(valuable) 텍스트를 갖는 오브젝트를 작성하기 위해 사용된다.

<96> (3) 이미지 오브젝트 툴 버튼(405): 스태틱 혹은 밸류어블 이미지를 갖는 오브젝트를 작성하기 위해 사용된다.

<97> (4) 링크 툴 버튼(406): 오브젝트들 간을 관련시키는 링크를 작성하기 위해 사용되며, 링크의 거리를 컨트롤하기 위해서도 사용된다.

<98> 3-2 문서 템플릿

<99> 도 3에서, 워크 에리어(306)는 문서 템플릿의 디자인을 표시 및 편집하기 위해 사용된다. 이에 따라, 유저는 사전 인쇄된(preprinted) 문서의 개관을 디자인하는 것이 가능하게 된다.

<100> 워크 에리어(306)는 스크롤 바(307)와 옵션 룰러(308)와 문서 템플릿(309)을 구비한다. 문서 템플릿(309)은 페이지가 복수개 있는 것을 나타낼 수 있다.

- <101> 공급된 문서 템플릿의 페이지 사이즈는, 유저에 의해 지정된다. 각각의 문서에서의 실제의 페이지 수는, 유입되는 데이터의 데이터량에 따라 변화된다. 데이터가 1 페이지 내에 맞게할 수 없을 때, 추가의 페이지가 자동적으로 작성된다.
- <102> 각각의 페이지 내의 경계선(310)은, 페이지 상의 인쇄 가능한 오브젝트의 최대 폭을 나타내는 임의의 페이지 마진이다.
- <103> 또한, 도 4는 1 페이지의 문서 템플릿(309) 상에 표시하는 것이 가능한 오브젝트의 예이다. 템플릿은, 복수의 오브젝트(407, 408)와, 임의로 적용가능한 앵커(anchor) 아이콘(409) 및 링크(412), 그리고 슬라이더(413)를 갖는다.
- <104> 3-3 오브젝트
- <105> 여기서, 오브젝트에 대하여 설명한다. 오브젝트란, 문서 템플릿 내에 데이터 파일로부터 고정 혹은 가변의 텍스트/이미지가 유입됨으로써, 묘화되는 스페이스(이것을 부분 영역이라고 부른다)이다. 도 4에 도시한 바와 같이, 오브젝트는 다른 오브젝트와 함께 문서 템플릿 내에 배치된다. 유저 인터페이스 화면을 통해, 유저로부터의 조작 지시, 즉, 오브젝트는 마우스(133)의 조작에 의해 이동되고, 사이즈 조정되며, 제작성된다. 하기에 본 실시예에서의 오브젝트가 정의된다.
- <106> (1) 오브젝트는 고정 혹은 가변의 콘텐츠를 갖는다. 가변 콘텐츠는, 데이터 소스로부터 취득한 데이터가 문서마다, 즉 레코드마다 상이할 가능성이 있다는 의미에서 다이내믹하다고 말할 수 있다. 단, 본 실시예의 가변 콘텐츠는, 애니메이션화된 것, 혹은 다른 방법에 의해 시간적으로 변화되는 콘텐츠는 의도하지 않는다. 마찬가지로, 고정 콘텐츠는 오브젝트를 사용하여 생성되는 모든 문서에서, 유사하게 표시된다. 그러나, 가변 콘텐츠와 링크가 설정되어 있는 경우, 고정 콘텐츠는 각각의 문서로 위치가 상이할 가능성이 있다.
- <107> (2) 오브젝트는, 콘텐츠에 적용되는 배경색, 경계, 폰트 스타일과 같은 텍스트 설정과 마찬가지로의 장식 기능을 가지고 있다. 이러한 설정을 오브젝트 속성이라 한다. 오브젝트 속성은, 각 오브젝트마다 설정 가능하다. 그러나, 오브젝트가 다른 오브젝트와 동일한 오브젝트 속성을 갖도록 설정을 행하는 것도 가능하다.
- <108> (3) 오브젝트는 문서를 생성할 때에 데이터 소스로부터의 데이터와 머지(merge)된다. 장식 기능은, 어떠한 고정 콘텐츠라도, 인쇄된 출력물에서 가시적이다. 가변 콘텐츠는 데이터 소스로부터의 특정한 데이터의 표시를 제공한다. 오브젝트의 이 표현은 예를 들면 인쇄되거나 및/또는 비디오 디스플레이(144)의 스크린 상에 표시될 수 있다.
- <109> (4) 오브젝트는, 도 4에 도시한 바와 같이, 시각적인 단서로서의 유저 인터페이스를 갖고 있다. 예를 들면 오브젝트의 편집과 표시 및 설정을 위한 인터랙티브한 그래픽컬 유저 인터페이스(GUI)를 갖는다. GUI의 요소는 비디오 디스플레이(144)의 스크린 상에 표시되지만, 문서로서는 인쇄되지 않는다. 레이아웃 편집 어플리케이션(121)의 유저 인터페이스(103)는, 배경색이나 폰트와 같은 오브젝트의 장식 기능 중 몇가지를 표시하며, 또한 오브젝트의 설정의 편집이나 표시를 가능하게 하기 위한 기능을 갖고 있다.
- <110> 유저는 오브젝트의 사이즈 및 위치를 지정하는 것이 가능하다. 오브젝트는 각각의 문서로 표시되는 콘텐츠를 어떻게 결부시킬지의 제어에 관한 제약(제어 조건)이 있다. 이들 제약(고정/가변 콘텐츠를 오브젝트와 결부시키는 것을 포함함)은, 유저가 하나의 문서 템플릿으로부터 다수의 문서를 컨트롤하는 주요한 방법이다.
- <111> 1개의 오브젝트의 변은, 관련된 콘텐츠가 문서 내에서 표시되는 가상의 경계선을 정의한다. 따라서, 오브젝트의 좌변을 논하는 것은, 관련된 콘텐츠가, 각 문서에서, 표시 가능한 에리어 내의 가장 좌측의 변을 논하는 것과 동일하다. 마찬가지로, 오브젝트의 높이를 논하는 것은, 생성된 문서에서 관련된 콘텐츠의 높이의 제약을 논하는 것으로서 이해된다. 본 명세서에서는, 유저 인터페이스(103)를 참조하여 오브젝트의 변 혹은 사이즈를 논할 때에, 이 구별은 명백해질 것이다.
- <112> 3-4 신규 오브젝트의 작성 방법
- <113> 도 4에 도시한 바와 같이, 신규의 텍스트 오브젝트 혹은 이미지 오브젝트는, 텍스트 오브젝트 툴(404) 혹은 이미지 오브젝트 툴(405)을 마우스(133)로 클릭하고, 문서 템플릿(309) 상에 사각형을 드래그함으로써, 해당 문서 템플릿(309) 상에 작성된다.
- <114> 혹은, 오브젝트는, 적절한 툴(404 또는 405)을 액티브로 한 후에, 문서 템플릿(309) 상에서 단순히 클릭함으로써 작성되도록 하여도 된다. 이 경우, 마우스(133)의 클릭 조작에 따라 디폴트 사이즈의 오브젝트가 템플릿 상

에 삽입된다. 또한, 해당 신규 오브젝트의 사이즈 등을 설정하기 위한 다이얼로그 박스 혹은 다른 프롭र्ट가 제공된다. 또한, 오브젝트의 사이즈는 자동적으로 미리 정의되도록 하여도 되거나, 혹은, 계산된 방식에 의해 작성 및 배치되도록 하여도 되며, 여러가지의 방법이 생각된다. 여기서 생성된 오브젝트를 마우스 등의 입력 장치에 의해 선택한다. 우측 클릭에 의해 프로퍼티(property)를 지시하는 등의 조작을 행한다. 오브젝트의 프로퍼티 다이얼로그가 표시되어, 오브젝트의 제약을 설정할 수 있다. 오브젝트의 프로퍼티 다이얼로그 UI에서는, 전술한 각종 제약을 설정할 수 있다. 또한, 오브젝트의 프로퍼티 다이얼로그에서는, 오브젝트의 사이즈(폭 및 높이)나 위치를 결정할 수 있다. 가변 사이즈로 설정하는 경우에는, 오브젝트의 기본 패턴(기본 사이즈와 기준 위치)을 설정하고, 또한, 최대 오브젝트 사이즈(폭 및 높이)와 최소 오브젝트 사이즈(폭 및 높이)를 설정하는 것이 가능하게 되어 있다.

<115>

3-5 오브젝트의 표시 방법

<116>

어플리케이션(121)에서, 오브젝트의 변은 실선(414)으로 표시된다(단, 다른 방법으로 변을 나타내어도 됨). 어플리케이션(121)은 또한, 앵커(409)(변의 근처에 묘화된 선, 형상 또는 아이콘), 핸들(411)(변의 이동 또는 수정을 위해 묘화된 컨트롤 점) 및 슬라이더(413)(변의 양 변에 묘화된 짧은 병행선)를 구비한다.

<117>

이들 아이콘 및 변 전부는, 어떤 툴 또는 어떤 오브젝트를 선택, 하이라이트 혹은 액티브로 할지에 따라, 묘화되거나 묘화되지 않기도 한다. 일반적으로, 오브젝트의 변 및 아이콘은 단지 문서 템플릿의 디자인에 도움을 주고, 인쇄물에는 묘화되지 않는다.

<118>

3-6 링크

<119>

링크는, 해당 오브젝트들 간의 관련을 나타내고 있다. 링크에 의해 서로 관련된 오브젝트 각각은, 서로의 레이아웃 변경의 영향을 받아 레이아웃을 계산한다. 도 4의 412로 나타내고 있는 것이 링크이며, 이 도면에서는 오브젝트(407 및 408)를 서로 관련시키고 있다.

<120>

3-7 링크의 설정 방법

<121>

다음으로, 오브젝트를 서로 관련시키기 위한 링크의 설정 방법에 대하여 설명한다. 도 7a 내지 도 7c는 링크 설정 시의 UI의 일례를 나타내는 도면이다. 도 6, 도 7a 및 도 7c를 이용하여 오브젝트에 링크를 설정하는 방법에 대하여 설명한다.

<122>

먼저, 링크를 설정하기 위해서는, 링크를 설정하기 위한 오브젝트(최저 2개)를 작성한다(단계 S601). 도 7a는, 작성된 2개의 오브젝트(701, 702)를 나타낸다. 참조 부호 703 및 704는 앵커를, 참조 부호 705는 마우스 포인터를 각각 나타내고 있다.

<123>

단계 S602에서는, 링크 툴 버튼(406)이 선택된다. 단계 S603에서는, 링크를 설정하는 한 쪽의 오브젝트(여기서는, 오브젝트(701))를 마우스 포인터(705)로 클릭하여 선택한다.

<124>

단계 S604에서는, 단계 S603에서 선택된 오브젝트(701) 간에 링크를 설정하고자 하는 오브젝트(702)를 선택한다. 구체적으로는 마우스 포인터(705)를 이동하여, 오브젝트(702)를 클릭한다. 도 7b는, 오브젝트(701)를 클릭한 후, 오브젝트(702)를 클릭할 때까지의 기간 동안의 상태를 나타내는 도면이다. 도 7b에 도시한 바와 같이, 단계 S603에서 클릭한 위치와, 현재의 마우스 포인터(705)의 위치 사이를 연결한 선(706)이 표시된다.

<125>

단계 S604에서, 오브젝트(702)가 선택되면 오브젝트(701) 사이에 링크가 표시된다. 도 7c는, 오브젝트(702)가 클릭됨으로써, 링크(707)가 표시된 상태를 나타내고 있다.

<126>

3-8 길이가 가변인 링크의 설정

<127>

도 28에서는 가변 링크의 설정을 위한 UI를 나타내고 있다. 도 28의 상태에서는, 문서 템플릿(309) 상에 컨테이너(2803)와 컨테이너(2804)가 존재한다. 각각의 컨테이너는 앵커 아이콘(2801), 앵커 아이콘(2802)과 고정된 변(2805), 변(2806)을 포함하여 구성되어 있다. 컨테이너(2803 및 2804)의 사이에는 가변 사이즈의 링크(2809)가 있으며, 컨테이너(2803)와 컨테이너(2804)를 연결하고 있다. 컨테이너(2803)와 컨테이너(2804)의 사이에는 링크가 설정되어 있기 때문에, 컨테이너(2803)의 우변(2807)과 컨테이너(2804)의 좌변(2808)은 점선으로 표현되어 있다. 이 때문에 각 컨테이너에 인디케이터(indicator; 2810), 인디케이터(2811)가 표시되며, 각각 변(2807)과 변(2808)이 가변인 것을 나타내고 있다.

<128>

또한, 도 29는, 링크 설정 수단에서의 UI 화면이며, 링크(2809)의 정보를 설정하기 위한 다이얼로그 윈도우

(2901)의 예이다. 이 다이얼로그는, 타이틀 바(2902), 툴 버튼(2903), 다이얼로그 윈도우의 개폐를 행하는 버튼(2904), 각종 정보를 설정하는 에리어(2909)로 구성되어 있다. 이 다이얼로그 윈도우에서는 링크 타입이 가변 길이(2907)의 링크인지, 혹은 고정 길이(2906)의 링크인지의 택일적인 선택을 행할 수 있다. 링크 타입이 가변인 경우에는 링크의 길이의 최소값(Min. Distance(2910)), 최대값(Max. Distance(2912)), 및 기준값(Distance(2911))을 설정할 수 있다. 이 설정은 2개의 컨테이너 간에 링크를 설정한 후에, 이 설정된 링크를 클릭 등의 조작에 의해 선택하였을 때에 표시된다. 혹은, 링크를 설정한 직후에, 상기 링크에 관한 다이얼로그 윈도우(2901)가 자동적으로 표시되도록 하여도 된다. 여기서 각 컨테이너간의 거리의 기준값(2911)은, 데이터를 유입시켰을 때에 각 컨테이너의 사이즈가 변경되지 않는 경우에 이용되는 링크의 길이이다.

<129> 도 30은 고정 사이즈의 링크를 사용한 경우의 레이아웃 결과를 나타내고 있다. 레이아웃 계산 방법은 전술한 바와 같이 행해진다. 예를 들면 도 28에서 컨테이너(2803)와 컨테이너(2804)에 각각 다른 사이즈의 이미지 데이터가 삽입된 경우를 생각한다. 이 경우, 각각의 컨테이너는 데이터의 크기를 최적이라고 간주한다. 컨테이너(2803)는 삽입된 이미지 사이즈가 되는 프레임(3004)(최적 컨테이너 사이즈)에 도달하도록 하기 위해서는 우측 방향으로 사이즈를 변경하고자 한다. 마찬가지로 컨테이너(2804)는 삽입된 이미지 사이즈가 되는 프레임(3005)(최적 컨테이너 사이즈)에 도달하도록 하기 위해서는 좌측 방향으로 사이즈를 변경하고자 한다. 그러나 컨테이너(2803)와 컨테이너(2804)는 앵커(2801)와 앵커(2802)에 의해 각각 좌변(2812)과 우변(2813)의 이동이 가능하지 않다. 따라서, 상기한 바와 같이 사이즈를 변경하고자 하면 양자의 간격을 좁힐 수밖에 없다. 그러나, 컨테이너 간에는 고정 사이즈의 링크(3003)가 설정되어 있다. 레이아웃 계산 시에 그 길이가 유지되기 때문에, 컨테이너(2803)와 컨테이너(2804)의 사이즈가 변경되게 된다.

<130> 그 결과, 컨테이너(2803)와 컨테이너(2804)는 데이터의 중형비에 대응하는 최적의 사이즈를 확보할 수 없다. 최종적으로, 도 30에 도시한 바와 같이 최적의 사이즈(프레임(3004), 프레임(3005))보다도 작아지게 된다. 즉, 링크(3003)의 사이즈가 고정이기 때문에 컨테이너(2803)와 컨테이너(2804)는 최적 사이즈를 달성할 수 없다(도 30에서, 각 컨테이너 내의 일점 차선으로 나타낸 범위가 데이터가 갖는 중형비이다).

<131> 한편, 도 31은 도 30과 마찬가지로의 상태에서 링크를 가변 사이즈로 한 경우를 나타내고 있다. 이 경우, 상기의 예에서 컨테이너(2803)와 컨테이너(2804)의 사이에는 도시한 바와 같이 가변 사이즈의 링크가 설정되어 있다. 따라서, 컨테이너(2803)와 컨테이너(2804)의 사이즈가 변경될 때에는, 링크 사이즈가 축소됨으로써 컨테이너(2803)와 컨테이너(2804)의 사이즈를 도 30의 예보다 크게 할 수 있다. 이 결과, 삽입되는 데이터 사이즈에 적합한 최적의 사이즈를 달성할 수 있다, 혹은 보다 삽입 데이터 사이즈(최적 사이즈)에 가까운 사이즈의 컨테이너의 프레임을 설정할 수 있다. 도 31은 이 결과를 나타내고 있다. 가변 링크(2809)는 레이아웃 계산의 결과, 가변 링크(3103)에 나타낸 바와 같은 사이즈 상태로 된다. 또한, 이 경우 컨테이너(2803)와 컨테이너(2804)는 각각 최적의 사이즈(데이터 사이즈에 맞는 크기)로 되어 있다.

<132> 4. 자동 링크 기능의 설명

<133> 다음으로 본 실시예에 따른 호스트 컴퓨터(101)가 구비하는 레이아웃 편집 어플리케이션(121)의 특징적인 기능인, 자동 링크 기능에 대하여 설명한다. 본 실시예에 따른 호스트 컴퓨터(101)는, 오브젝트 간의 링크 설정을 유저가 수동으로 행하는 기능(상기 3 내지 6) 외에, 자동으로 링크 설정을 행하는 기능을 더 구비하고 있다. 자동 링크 기능에 대해서는 메뉴 바에 있는 파일이나 표시 메뉴로부터 자동 링크 기능의 UI를 표시시킨다. 다음에, 자동 링크를 행하는 것을 설정한다. 자동 링크 기능을 이용하는 경우에는, 링크의 상세에 대하여 설정할 수 있다. 이들에 대해서는 도 8에서 후술한다. 이하, 해당 자동 링크 기능의 상세에 대하여 설명한다.

<134> 4-1 자동 링크 기능을 이용하여 레이아웃 편집 처리를 행하기 위한 윈도우

<135> 도 5는, 자동 링크 기능을 구비하는 레이아웃 편집 어플리케이션(121)을 이용하여 레이아웃 편집 처리를 행하는 경우에 표시되는 유저 인터페이스의 일례를 나타내는 도면이다. 도 5에서, 참조 부호 501은 어플리케이션 윈도우, 참조 부호 502는 오브젝트를 나타낸다. 참조 부호 503 및 참조 부호 504는 오브젝트 간에 설정되어 있는 링크 및 마우스 포인터를 각각 나타내고 있다.

<136> 또한, 본 명세서 중에 오브젝트의 위치를 설명한다. 오브젝트의 위치는, 문서 내 좌측 위를 원점으로 하여, 수평 방향을 X 방향, 수직 방향을 Y 방향으로 정의한다. 자동 링크 기능을 실행함에 있어서는, 오브젝트는 가변인 것을 전제로 한다. 앵커 설정이 되어 있는 등 오브젝트가 고정인 경우에는, 자동 링크 기능은 동작하지 않는다.

<137> 4-2 설정 윈도우

- <138> 도 8은 오브젝트 간의 거리를 설정하기 위한 다이얼로그 윈도우(801)의 일례를 나타내는 도면이다. 도 8에 도시한 바와 같이, 다이얼로그 윈도우(801)는 타이틀 바(802), 툴 버튼(803), 다이얼로그 윈도우의 개폐를 행하는 버튼(804), 오브젝트 간의 거리 정보를 설정하는 설정 란(807)을 구비한다.
- <139> 유저는, 다이얼로그 윈도우(801)에서, 링크를 자동적으로 작성하는 오브젝트 간의 거리(806)를 설정한다(이하, 806을 "설정 거리"라 함). 또한 자동적으로 작성하는 링크의 길이인 LinkLength(808)를 지정하는 것도 가능하다. LinkLength(808)를 지정하지 않는 경우, 설정되는 링크의 길이는 오브젝트 간의 거리로 된다. 다이얼로그 윈도우 개폐 버튼(804)의 OK 버튼을 가압함으로써 설정을 적용하면, 다이얼로그 윈도우(801)의 설정 정보는 메모리(136)에 저장된다. 또한, 도 8에서 설정하는 링크는 고정 링크 종류인 것을 가정한다. 그러나, 여기서 링크의 종류로서 고정 링크 또는 가변 링크를 선택할 수 있도록 하여도 무방하다. 또한, 여기서 가변 링크의 설정이 선택된 경우, 도 29의 가변 링크의 설정 UI가 표시되게 된다.
- <140> 4-3 신규로 오브젝트를 작성한 경우의 자동 링크 기능의 동작
- <141> 도 11은 오브젝트를 신규 생성한 경우의 자동 링크 기능의 일반적인 처리의 흐름을 도시하는 플로우차트이다. 또한, 도 11의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다.
- <142> 단계 S1101에서, 유저가 텍스트 오브젝트 툴(404), 또는 이미지 오브젝트 툴(405)을 이용하여 신규로 오브젝트를 생성한다. 레이아웃 편집 어플리케이션(121)은 생성된 오브젝트의 위치 정보를 메모리(136)에 저장한다.
- <143> 단계 S1102에서, 레이아웃 편집 어플리케이션(121)은, 생성한 오브젝트와 그 근방에 있는 오브젝트와의 거리를 계산한다. 또한, 본 명세서에서, 오브젝트 간의 거리란 대상으로 되는 오브젝트 간의 가장 가까운 변끼리의 거리를 말한다. 근방에 복수의 오브젝트가 존재하는 경우에는 각각에 대하여 거리를 구한다.
- <144> 단계 S1103에서, 레이아웃 편집 어플리케이션(121)은, 얻어진 거리와 사전에 설정된 설정 거리(806)를 비교한다. 얻어진 거리가 설정 거리 이하인 경우에는 상기 오브젝트 간에 링크를 작성한다(단계 S1104). 그후, 처리를 종료한다. 한편, 얻어진 거리가 설정 거리보다 큰 경우에는, 링크 작성을 행하지 않고 처리를 종료한다. 또한, 링크 작성 처리의 상세는 후술한다.
- <145> 4-4 오브젝트를 이동시킨 경우의 자동 링크 기능의 동작
- <146> 도 12는 레이아웃된 오브젝트를 이동시킨 경우의 자동 링크 기능의 일반적인 처리의 흐름을 도시하는 플로우차트이다. 도 12의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다.
- <147> 단계 S1201에서, 유저는 이동될 오브젝트를 선택한다. 다음에, 레이아웃 편집 어플리케이션(121)은, 유저에 의해 선택된 해당 오브젝트를 인식한다.
- <148> 단계 S1202에서는, 인식한 오브젝트를 유저에게 시각적으로 나타내기 위해 UI를 변경한다. 도 9는 오브젝트 E가 선택된 것을 도시하고 있다. 마우스 포인터(903)로 직사각형(902)을 드래그하여 직사각형 내에 오브젝트를 넣음으로써 선택하는 것이 가능하다. 또한, 마우스 포인터(903)에서 오브젝트를 누름으로써 선택할 수도 있다. 또한, 선택 방법은 특별히 이들 방법에 한정되는 것이 아니라, 다른 방법이어도 된다. 도 10은 상기 오브젝트 E가 선택된 것을 유저에게 명시하기 위해 오브젝트 E의 표시가 변경된 예를 도시하고 있다(1002). 또한, 이 예에서는 오브젝트에 사선을 넣었다. 다른 방법에 의해 선택된 것을 나타내어도 된다.
- <149> 단계 S1203에서, 유저는 선택된 오브젝트를 마우스(133)로 드래그하여 원하는 위치로 이동시킨다. 다음에, 레이아웃 엔진(105)은 목적지에서 오브젝트의 위치 정보를 취득한다. 도 13은 선택된 오브젝트 E를 마우스 포인터(1303)로 드래그함으로써 이동시키고 있는 모습을 도시하고 있다(1302).
- <150> 여기서, 이동시킨 오브젝트가 마우스(133)에 의해 드래그된 상태 그대로 있으면, 오브젝트의 신규 생성의 경우와 마찬가지로, 어플리케이션(121)은 이동된 오브젝트와 근방에 있는 오브젝트와의 거리를 계산한다(단계 S1204). 이동된 오브젝트 근방에 복수의 오브젝트가 존재하는 경우에는 각각에 대하여 거리를 구한다.
- <151> 단계 S1205에서, 레이아웃 편집 어플리케이션(121)은, 얻어진 거리와 설정 거리(806)를 비교한다. 얻어진 거리가 설정 거리 이하인 경우에는, 단계(S1206)로 진행하여, 유저에게 시각적으로 나타내기 위한 UI를 표시한다. 도 14는 상기 이동시킨 오브젝트가 설정 거리 이하인 것을 유저에게 나타내기 위한 UI의 예를 도시하고 있다(1402). 이 예에서는, 링크를 점선으로 나타내고 있다. 그러나, 다른 방법을 사용하여 링크를 나타내도 된다.
- <152> 단계(S1207)에서는, 상기 UI(1402)가 표시된 상태에서 유저가 마우스(133)를 드롭하면, 레이아웃 편집 어플리케이션

이션(121)은 오브젝트의 선택이 해제된 것으로 인식한다. 단계(S1208)에서, 레이아웃 편집 어플리케이션(121)은, 단계(S1207)에서의 처리에 따라, 도 16에서 설명된 바와 같이 단계(S1205)에서 사전 설정된 범위 내에 있도록 결정되는 오브젝트들 간에 링크를 작성한다. 도 19는 도 14의 상태에서부터 마우스를 드롭하고, 링크의 UI를 통상의 표시(1902)로 변경하여 오브젝트들 간에 링크가 작성된 것을 나타내고 있다. 이 때, 선택한 오브젝트의 메모리(136) 상의 위치 정보가, 이동한 위치로 갱신된다.

- <153> 한편, 마우스(133)를 드롭하지 않고 더 이동시키면, 레이아웃 편집 어플리케이션(121)은 이동한 오브젝트와 근방 오브젝트와의 거리를 다시 계산한다(즉, 단계(S1207)에서 "아니오"로 되어, 단계(S1204)로 되돌아간다).
- <154> 4-5 링크 작성 처리(오브젝트를 신규 생성한 경우)
- <155> 도 15는 오브젝트 신규 생성 시 실행되는 링크 작성 처리(단계(S1104))의 상세를 도시하는 플로우차트이다. 또한, 도 15의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다.
- <156> 단계(S1501)에서, 레이아웃 엔진(105)은 우선, 신규 생성한 오브젝트의 위치 정보를 취득한다. 상술한 바와 같이, 본 명세서에서 오브젝트들 간의 거리는, 오브젝트 간에서의 변끼리의 거리를 말한다. 따라서, "취득하는 위치 정보"란 오브젝트의 각 변의 X 방향 및 Y 방향의 위치 정보이다.
- <157> 단계(S1502)에서, 레이아웃 편집 어플리케이션(121)은 상기 오브젝트 이외에서 레이아웃되어 있는 오브젝트가 있는지 확인한다. 상기 오브젝트 이외에서 레이아웃되어 있는 오브젝트가 있었던 경우, 레이아웃 편집 어플리케이션(121)은 그 갯수 및 각 위치 정보를 취득한다. 또, 취득한 각 오브젝트의 위치 정보는 서열(sequence)로 저장된다.
- <158> 단계(S1503)에서, 레이아웃 편집 어플리케이션(121)은 링크 설정의 처리를 행한다. 링크 설정의 처리가 종료되고, 그 외에 레이아웃되어 있는 오브젝트가 없는 경우, 처리는 종료된다.
- <159> 4-6 링크 설정 처리(오브젝트를 신규 생성한 경우)
- <160> 도 16a 및 도 16b는 도 15의 링크 설정 처리(단계(S1503))의 상세한 흐름을 도시하는 플로우차트이다. 또한, 도 16a 및 도 16b의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다. 링크의 설정은, 근방에 있는 오브젝트를 추출하고, 그 오브젝트 간의 링크를 설정한다. 링크는 근방의 오브젝트를 추출하고 그 오브젝트에 링크를 설정함으로써 설정된다. 오브젝트 간의 거리가 상기 설정 거리 이하로 된 경우에 오브젝트 간에 링크를 설정한다. 상기 설정 거리 이하로 되는 오브젝트가 복수 발견된 경우에는, 오브젝트간 거리가 설정 거리 이하인 복수의 컨테이너에 대하여 링크를 설정한다.
- <161> 단계(S1601)에서, 레이아웃 편집 어플리케이션(121)은 링크를 자동적으로 작성하는 오브젝트 간의 거리(806)를 Dist_AutoLink에 대입한다. 다음으로, 레이아웃 편집 어플리케이션(121)은 i에 1을 대입한다.
- <162> 단계(S1602)에서는, 생성한 오브젝트와 i번째의 오브젝트의 X 방향의 위치가 중첩되는지의 여부를 체크한다. 본 실시 형태에 따른 자동 링크 기능에서는, 작성되는 링크는, 오브젝트끼리가 동일 수평 방향 또는 수직 방향에 있다는 것을 전제로 하고 있다. 도 18은 링크 작성의 대상으로 되는 영역을 도시하고 있다. 생성한 오브젝트(1801)의 Y 방향의 변의 연장선(1807)과 X 방향의 변으로 둘러싸인 영역(1808 또는 1811) 또는 오브젝트(1801)의 X 방향의 변의 연장선(1807)과 Y 방향의 변으로 둘러싸인 영역(1809 또는 1810) 내에 오브젝트의 일부 또는 전부가 존재하는 경우에 링크 작성의 대상으로 된다. 이 때, 오브젝트(1801)의 Y 방향의 변의 연장선(1807)이나 오브젝트(1801)의 X 방향의 변의 연장선(1806)과 다른 오브젝트의 링크는 작성되지 않는다. 도 18에서는, 오브젝트(1801)와 Y 방향의 링크 대상으로 되는 것은 오브젝트(1803)이다. 오브젝트(1801)와 X 방향의 링크의 대상으로 되는 것은 오브젝트(1802, 1805)이다. 오브젝트(1804)는 링크 작성의 대상 밖이다.
- <163> 단계(S1602)에서 생성된 오브젝트의 위치가 Y방향으로 i번째 오브젝트의 위치와 중첩된 것으로 판단된 경우에는, 단계(S1603)으로 진행하여, Y 방향의 위치가 서로 중첩되는지의 여부를 판정한다. 생성된 오브젝트의 위치가 X 방향에서 i번째 오브젝트의 위치와 중첩되지 않는다고 판단된 경우에는, 단계(S1614)로 진행한다.
- <164> 단계(S1614)에서는, 상기 어플리케이션은 i번째의 오브젝트의 X 방향의 변 중 생성한 오브젝트에 가까운 쪽의 변에 앵커 설정이 되어 있는지의 여부를 체크한다. 상기 변에 앵커 설정이 되어 있는 경우에는, 그 변은 고정이기 때문에 링크 설정은 행하지 않는다. 상기 변에 앵커 설정이 되어 있지 않으면, 단계(S1604)로 진행하여, 생성한 오브젝트와 i번째의 오브젝트의 X 방향의 변 중 가까운 변끼리의 거리를 계산한다. 그 결과를 DistX에

대입한다.

- <165> 단계(S1605)에서, 레이아웃 편집 어플리케이션(121)은, 단계(S1604)에서 계산된 DistX와 자동 링크를 생성하는 설정 거리가 저장되어 있는 Dist_AutoLink를 비교한다. 만약에 DistX가 Dist_AutoLink보다 작으면, 즉, 계산된 DistX의 값이 설정 거리 값 이하이면, 단계(S1606)으로 진행하여, 상기 오브젝트 간에 X 방향의 링크를 작성한다.
- <166> 한편, 단계(S1603)에서 위치가 중첩된다고 판단된 경우는, 오브젝트끼리가 중첩되어 있는 것을 의미하고 있다. 도 20은 오브젝트(2001)에 중첩되는 위치에 오브젝트(2003)를 생성한 예를 도시하고 있다. 이 경우, 단계(S1607)에서, 레이아웃 편집 어플리케이션(121)은 유저에 대하여 링크를 설정할지의 여부의 확인을 행한다(상세는 후술).
- <167> 단계 S1606 및 S1607을 처리한 후나, 단계(S1614)에서 변에 앵커가 설정되어 있는 경우, 또는 단계(S1605)에서 DistX가 Dist_AutoLink보다 크다고 판단된 경우에는, 단계(S1608)로 진행하여, 그 다음의 근방 오브젝트를 체크하기 위해 i를 증분한다.
- <168> 이상은 X 방향의 링크의 작성에 대한 처리이다. 단계(S1602)에서 위치들이 중첩되지 않는다고 어플리케이션이 판단한 경우에는, 도 16b의 단계(S1610)으로 진행한다. 레이아웃 편집 어플리케이션(121)은 Y 방향의 링크에 대해 체크를 행하고, 오브젝트들이 X 방향으로 중첩되는지를 판단한다.
- <169> 단계(S1610)에서 위치들이 중첩된다고 판단된 경우에는, 단계(S1615)로 진행하여, i번째의 오브젝트의 Y 방향의 변 중 생성한 오브젝트에 가까운 쪽의 변에 앵커 설정이 되어 있는지의 여부를 체크한다. 상기 변에 앵커 설정이 되어 있지 않으면, 단계(S1611)로 진행하여, 생성한 오브젝트와 i번째의 오브젝트의 Y 방향의 변 중 가까운 변끼리의 거리를 계산한다. 그 결과를 DistY에 대입한다.
- <170> 또한 단계(S1612)에서, 어플리케이션은 단계(S1611)에서 계산된 DistY와 Dist_AutoLink를 비교한다. 만약에 DistY가 Dist_AutoLink보다 작으면, 즉, 계산된 DistX의 값이 설정 거리 값 이하라고 판단된 경우, 단계(S1613)로 진행하여, 상기 오브젝트 간에 Y 방향의 링크를 작성한다.
- <171> 단계(S1613)를 처리한 후나, 어플리케이션이 단계(S1610)에서 위치들이 중첩되지 않았다고 판단된 경우, 또는 단계(S1615)에서 변에 앵커가 설정되어 있는 경우, 또는 단계(S1612)에서 DistY가 Dist_AutoLink보다 크다고 판단된 경우에는, 단계(S1608)로 진행하여, 레이아웃 편집 어플리케이션(121)은 그 다음 근방 오브젝트를 체크하기 위해 i를 증분한다.
- <172> 단계(S1609)에서, 어플리케이션은 배치되어 있는 오브젝트 전체에 대하여 체크를 행하였는지의 여부를 판단한다. 모든 오브젝트를 체크하지 않았다면, 단계(S1602)로 진행하여 판독 처리를 반복한다. 모든 오브젝트를 체크하였다면, 링크 설정 처리를 종료한다. 이와 같이, 오브젝트를 새롭게 생성한 시점에서 유저가 설정한 소정 거리 내에 다른 오브젝트가 존재하는 것을 어플리케이션이 확인하면, 자동적으로 링크가 작성된다. 따라서, 오브젝트의 수가 증가해도 유저의 작업 효율은 저하되지 않는다.
- <173> 4-7 링크 작성 처리(오브젝트를 이동시킨 경우)
- <174> 도 21은 오브젝트 이동 시 실행되는 링크 작성 처리(단계(S1208))의 상세를 도시하는 플로우차트이다. 도 21의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다.
- <175> 단계(S2101)에서, 레이아웃 엔진(105)은 선택한 오브젝트의 위치 정보를 취득한다. 다음으로, 단계(S2102)에서, 레이아웃 편집 어플리케이션(121)은 상기 오브젝트 이외에서 배치되어 있는 오브젝트가 있는지 확인한다. 단계(S2102)에서, 그 외에 배치되어 있는 오브젝트가 없다고 판정된 경우에는, 단계(S2108)에서 오브젝트를 이동시키고 처리는 종료한다.
- <176> 한편, 단계(S2102)에서 어플리케이션이 그 외에 배치되어 있는 오브젝트가 있다고 판정한 경우에는, 단계(S2103)로 진행한다. 다음에, 어플리케이션(121)은 선택한 오브젝트에 자동 링크를 행하지 않는 속성이 부여되어 있는지를 확인한다. 또한, 자동 링크를 행하지 않는 속성이란, 자동 링크 기능을 무효로 하는 속성으로, 오브젝트 간의 거리가 상기 설정 거리(806) 이하로 된 경우라도 오브젝트 간에 링크를 작성하지 않는다.
- <177> 도 27은 각 오브젝트에 자동 링크를 설정하기 위해 사용되는 다이얼로그 윈도우를 도시한 도면이다. 오브젝트를 선택하여 마우스의 우측 클릭 또는 키보드의 특정 키에 의해 프로퍼티(property) 다이얼로그 윈도우(2701)가 표시된다. 자동 링크를 행하는 설정이지만, 자동 링크를 행하지 않는 속성(2705)을 선택함으로써, 자동 링크

기능을 무효화할 수 있다. 오브젝트에 상기 자동 링크를 행하지 않는 속성이 부여되어 있는 경우, 다른 오브젝트와의 사이에 링크를 작성하지 않고 임의로 이동하는 것이 가능하게 된다. 이미 다른 오브젝트와 링크가 설정되어 있는 경우에는, 오브젝트를 이동하여 사용자가 지정한 소정 거리도 커졌다고 해도 그 링크를 유지하면서, 그 이상의 링크를 작성하지 않고 오브젝트를 이동할 수 있다.

<178> 도 23은 이미 링크(2304)가 설정되어 있는 오브젝트(2302)가 선택되어 있는 예를 도시하고 있다. 오브젝트(2302)에 상기 자동 링크를 행하지 않는 속성이 부가되어 있는 경우에, 도 23의 상태에서부터 오브젝트(2302)를 이동시켜 도 24에 도시한 바와 같이 오브젝트(2402, 2405)의 거리가 상기 설정 거리(806) 이하로 된 경우에도, 오브젝트(2402, 2405) 사이에 링크는 작성되지 않는다. 한편, 오브젝트(2402, 2406)간의 거리가 상기 설정 거리(806)보다 커진 경우라도 링크(2404)는 설정된 상태 그대로 된다.

<179> 도 21을 다시 참조한다. 단계(S2103)에서 선택한 오브젝트에 상기 자동 링크를 행하지 않는 속성이 부여되어 있지 않은 경우에는, 단계(S2104)로 진행한다. 레이아웃 편집 어플리케이션(121)은 그 외에 배치되어 있는 각 오브젝트의 갯수 및 각 위치 정보를 취득한다. 취득한 각 오브젝트의 위치 정보는 서열로 저장된다.

<180> 단계(S2105)에서는, 사용자가 오브젝트를 이동시킬 때에 수행되는 동작으로서 이미 설정되어 있는 링크를 유지한다고 선택하였는지, 파괴한다고 선택하였는지를 판정한다. 링크의 유지 또는 파괴는, 예를 들면 키보드(132)의 특정한 키를 누르면서 오브젝트를 이동시킴으로써 선택될 수 있다. 그러나, 본 발명은 상기 방법에 한정되는 것은 아니다. 다른 방법을 사용하여 유지/파괴를 선택해도 된다.

<181> 단계 S2105에서, 어플리케이션은 사용자가 기존 링크의 유지를 선택하였다고 판정한 경우, 기존의 링크 정보를 유지한 상태 그대로 오브젝트의 이동을 행한다(단계(S2107 및 S2108)), 레이아웃 편집 어플리케이션(121)은 링크가 아직 작성되어 있지 않은 오브젝트만을 대상으로 하여 링크 작성의 처리를 행한다(단계(S2109)). 한편, 어플리케이션이 단계(S2105)에서 사용자가 기존 링크의 파괴를 선택하였다고 판정한 경우에는, 기존의 링크 정보를 파괴한 후에 오브젝트의 이동을 행한다(단계(S2106 및 S2108)). 다음에, 어플리케이션은 레이아웃된 모든 오브젝트를 대상으로 하여 링크 작성 처리를 행한다(단계(S2109)).

<182> 도 25는 도 23에서 오브젝트(2302)에 상기 자동 링크를 행하지 않는 속성이 부여되어 있지 않은 경우에, 사용자가 기존 링크의 유지를 선택하고, 오브젝트(2502 및 2505)간의 거리가 상기 설정 거리(806) 이하로 되도록 오브젝트(2502)를 이동시킨 경우의 예를 도시하고 있다. 오브젝트(2502 및 2507)간의 거리가 상기 설정 거리(806)보다 긴 경우라도 링크(2504)는 설정된 상태 그대로이다.

<183> 도 26은 도 23에서 오브젝트(2302)에 상기 자동 링크를 행하지 않는 속성이 부여되어 있지 않은 경우에, 사용자가 기존 링크의 파괴를 선택하고, 오브젝트(2602 및 2605)간의 거리가 상기 설정 거리(806) 이하로 되도록 오브젝트(2602)를 이동시킨 예를 도시하고 있다. 오브젝트(2602 및 2607)간의 거리가 상기 설정 거리(806)보다 길더라도, 이미 설정되어 있던 링크(2404)는 해제되어, 화면으로부터 소거된다. 상술한 바와 같이, 오브젝트를 이동시킨 경우에는 마우스의 드롭 시에 링크가 작성된다(단계(S2110)).

<184> 4-8 링크 설정 처리(오브젝트를 이동시킨 경우)

<185> 도 22a 및 도 22b는 도 21의 링크 설정 처리(단계(S2109))의 상세한 처리의 흐름을 도시하는 플로우차트이다. 또한, 도 22a 및 도 22b의 처리는, 호스트 컴퓨터(101) 내의 프로세서(135)가 제어함으로써 실현된다. 각 단계는 오브젝트의 신규 생성 시의 단계와 거의 동일 처리이다. 다른 것은, 단계(S2206 및 S2213)에서의 처리가, 도 21의 단계(S2110)에서 최종적으로 링크를 확정하기 위한 일시적인 링크를 작성한다고 하는 점이다. 그 밖의 단계는 오브젝트의 신규 생성 시와 동일하기 때문에, 여기서는 설명을 생략한다. 이와 같이 오브젝트를 이동한 후, 오브젝트 간의 거리가 소정 거리 이하인 것으로 판정된 경우, 자동적으로 링크가 작성된다. 이 때문에 오브젝트의 편집에서도 유저의 작업 효율의 저하를 막을 수 있다.

<186> 4-9 에러 처리

<187> 도 17은 오브젝트를 생성 또는 이동하였을 때에 다른 오브젝트와 중첩된 경우(도 20의 경우)에 표시되는 다이얼로그 윈도우(1701)의 예이다. 이 다이얼로그 윈도우는 링크 작성을 행할지를 유저에게 확인하기 위한 것으로 사용된다. 일반적인 처리를 계속하는 버튼(1703)을 선택하면 링크가 작성된다. 처리를 중지하는 버튼(1704)이 선택되면 링크는 작성되지 않는다.

<188> 이상의 설명으로부터 명백해지는 바와 같이, 본 실시 형태에 따른 자동 레이아웃 시스템에 따르면, 인접하는 오브젝트 간의 거리가 소정 거리 이하로 되면, 자동적으로 링크를 작성한다. 이 때문에, 오브젝트의 수가 증가해

도, 링크 설정을 위한 작업 부하가 증대되지 않게 된다. 따라서, 자동 레이아웃 시스템에서의 유저의 설정 작업의 효율화를 실현할 수 있다.

- <189> [다른 실시 형태]
- <190> 또한, 본 발명은, 복수의 기기(예를 들면 호스트 컴퓨터, 인터페이스 기기, 리더(reader), 프린터 등)로 구성되는 시스템에 적용하거나, 하나의 기기로 이루어지는 장치(예를 들면, 복사기, 팩시밀리 장치 등)에 적용해도 된다.
- <191> 또한, 본 발명의 목적은, 상술한 실시 형태의 기능을 실현하는 소프트웨어의 프로그램 코드를 기록한 기억 매체를, 시스템 또는 장치에 공급하고, 그 시스템 또는 장치의 컴퓨터(또는 CPU나 MPU)가 기억 매체에 저장된 프로그램 코드를 판독 실행함으로써, 달성되는 것은 물론이다.
- <192> 이 경우, 기억 매체로부터 판독된 프로그램 코드가 상술한 실시 형태의 기능을 실현하게 된다. 프로그램 코드를 기억한 기억 매체는 본 발명을 구성하는 것으로 된다.
- <193> 프로그램 코드를 공급하기 위한 기억 매체로서는, 예를 들면, 플로피(등록상표) 디스크, 하드디스크, 광 디스크, 광 자기 디스크, CD-ROM, CD-R, 자기 테이프, 불휘발성 메모리 카드, ROM 등을 사용할 수 있다.
- <194> 상술한 실시 형태의 기능은 컴퓨터가 판독된 프로그램 코드를 실행하는 대신 다음과 같이 실현될 수 있다. 프로그램 코드의 지시에 기초하여, 컴퓨터 상에서 가동하고 있는 OS(오퍼레이팅 시스템) 등이 실제의 처리의 일부 또는 전부를 행한다.
- <195> 또한, 상술한 실시 형태의 기능은 다음과 같이 실현될 수 있다. 기억 매체로부터 판독된 프로그램 코드가, 컴퓨터에 삽입된 확장 보드나 컴퓨터에 접속된 확장 유닛에 구비되는 메모리에 기록된다. 그 후, 그 프로그램 코드의 지시에 기초하여, 그 확장 보드나 확장 유닛에 구비되는 CPU 등이 실제의 처리의 일부 또는 전부를 행한다.
- <196> 본 발명은 상술한 실시 형태로 한정되는 것이 아니며, 본 발명의 정신 및 범위 내에서 다양한 변경 및 수정이 이루어질 수 있다. 그러므로, 본 발명의 권리 범위는 첨부한 특허 청구의 범위에 의해 정해진다.

발명의 효과

- <197> 본 발명에 따르면, 복수의 오브젝트를 관련시키는 링크를 자동적으로 설정할 수 있게 되기 때문에, 자동 레이아웃 시스템에서의 유저의 작업의 효율화를 도모하는 것이 가능하게 된다.

도면의 간단한 설명

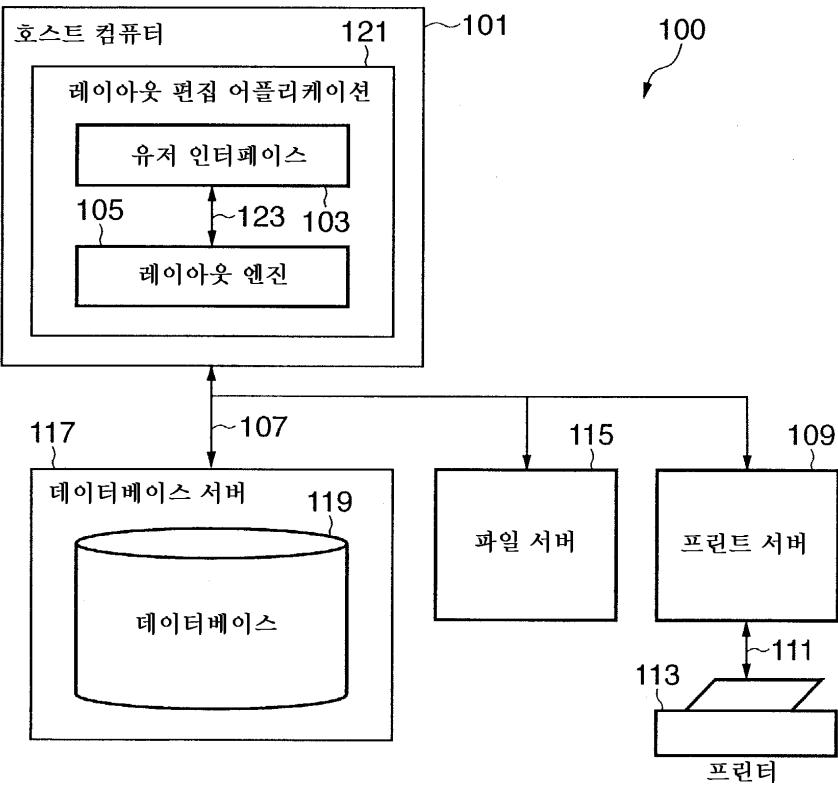
- <1> 도 1a는 본 발명의 일 실시 형태에 따른 호스트 컴퓨터를 구비하는 자동 레이아웃 시스템의 구성을 도시한 도면.
- <2> 도 1b는 본 발명의 일 실시 형태에 따른 호스트 컴퓨터의 구성을 더욱 상세하게 도시하는 블록도.
- <3> 도 2는 본 발명의 일 실시 형태에 따른 호스트 컴퓨터를 구비하는 자동 레이아웃 시스템의 구성을 도시한 도면.
- <4> 도 3은 레이아웃 편집 어플리케이션 프로그램의 메인 윈도우를 도시한 도면.
- <5> 도 4는 레이아웃 편집 어플리케이션 프로그램의 메인 윈도우를 도시한 도면.
- <6> 도 5는 자동 링크 기능을 구비하는 레이아웃 편집 어플리케이션을 이용하여 레이아웃 편집 처리를 행하는 경우에 표시되는 유저 인터페이스의 일례를 도시한 도면.
- <7> 도 6은 링크 설정 처리의 흐름을 도시하는 플로우차트.
- <8> 도 7a 내지 도 7c는 링크 설정 시의 UI의 일례를 도시한 도면.
- <9> 도 8은 자동 링크 기능을 동작시킬 때에, 오브젝트들 간의 거리를 설정하기 위해 사용되는 다이얼로그 윈도우의 일례를 도시한 도면.
- <10> 도 9는 오브젝트 선택의 UI의 일례를 도시한 도면.
- <11> 도 10은 선택된 오브젝트의 UI 변경의 일례를 도시한 도면.

- <12> 도 11은 오브젝트를 신규 생성한 경우 실행되는 자동 링크 기능의 일반적인 처리의 흐름을 도시하는 플로우차트.
- <13> 도 12는 오브젝트를 이동시킨 경우 실행되는 자동 링크 기능의 일반적인 처리의 흐름을 도시하는 플로우차트.
- <14> 도 13은 선택된 오브젝트를 마우스 포인터로 드래그함으로써 이동시키고 있는 모습을 도시한 도면.
- <15> 도 14는 이동시킨 오브젝트가 설정 거리 이하인 것을 유저에게 나타내기 위한 UI의 일례를 도시한 도면.
- <16> 도 15는 오브젝트 신규 생성 시 실행되는 링크 작성 처리의 상세를 도시하는 플로우차트.
- <17> 도 16a는 링크 설정 처리의 상세한 처리의 흐름을 도시하는 플로우차트.
- <18> 도 16b는 링크 설정 처리의 상세한 처리의 흐름을 도시하는 플로우차트.
- <19> 도 17은 오브젝트를 생성 또는 이동하였을 때에 다른 오브젝트와 중첩된 경우에 표시되는 다이얼로그 윈도우의 일례를 도시한 도면.
- <20> 도 18은 링크 작성의 대상으로 되는 영역을 도시한 도면.
- <21> 도 19는 오브젝트들 간에 링크가 작성된 것을 도시한 도면.
- <22> 도 20은 오브젝트에 중첩되는 위치에 다른 오브젝트를 생성한 예를 도시하는 도면.
- <23> 도 21은 오브젝트 이동 시 실행되는 링크 작성 처리의 상세를 도시하는 플로우차트.
- <24> 도 22a는 링크 설정 처리의 상세한 처리의 흐름을 도시하는 플로우차트.
- <25> 도 22b는 링크 설정 처리의 상세한 처리의 흐름을 도시하는 플로우차트.
- <26> 도 23은 링크가 설정되어 있는 오브젝트가 선택된 모습을 도시한 도면.
- <27> 도 24는 자동 링크를 행하지 않는 속성을 부여한 오브젝트를 이동시킨 경우의 결과예를 도시하는 도면.
- <28> 도 25는 기존 링크 유지를 선택한 경우 실행되는 오브젝트 이동 시의 결과예를 도시하는 도면.
- <29> 도 26은 기존 링크 파기를 선택한 경우 실행되는 오브젝트 이동 시의 결과예를 도시하는 도면.
- <30> 도 27은 오브젝트에 자동 링크를 행할지의 여부를 설정시키기 위한 다이얼로그 윈도우를 도시한 도면.
- <31> 도 28은 가변 링크에 의해 컨테이너를 배치하였을 때 제공되는 유저 인터페이스에서의 표시예를 도시하는 도면.
- <32> 도 29는 실시 형태에 따른 링크의 설정 다이얼로그 윈도우를 도시한 도면.
- <33> 도 30은 고정 링크에 의한 레이아웃 결과예를 도시하는 도면.
- <34> 도 31은 가변 링크에 의한 레이아웃 결과예를 도시하는 도면.
- <35> <도면의 주요 부분에 대한 부호의 설명>
- <36> 101 : 호스트 컴퓨터
- <37> 103 : 유저 인터페이스
- <38> 105 : 레이아웃 엔진
- <39> 119 : 데이터베이스
- <40> 121 : 레이아웃 편집 어플리케이션 프로그램
- <41> 132 : 키보드
- <42> 133 : 마우스
- <43> 143 : I/O 인터페이스
- <44> 144 : 비디오 디스플레이

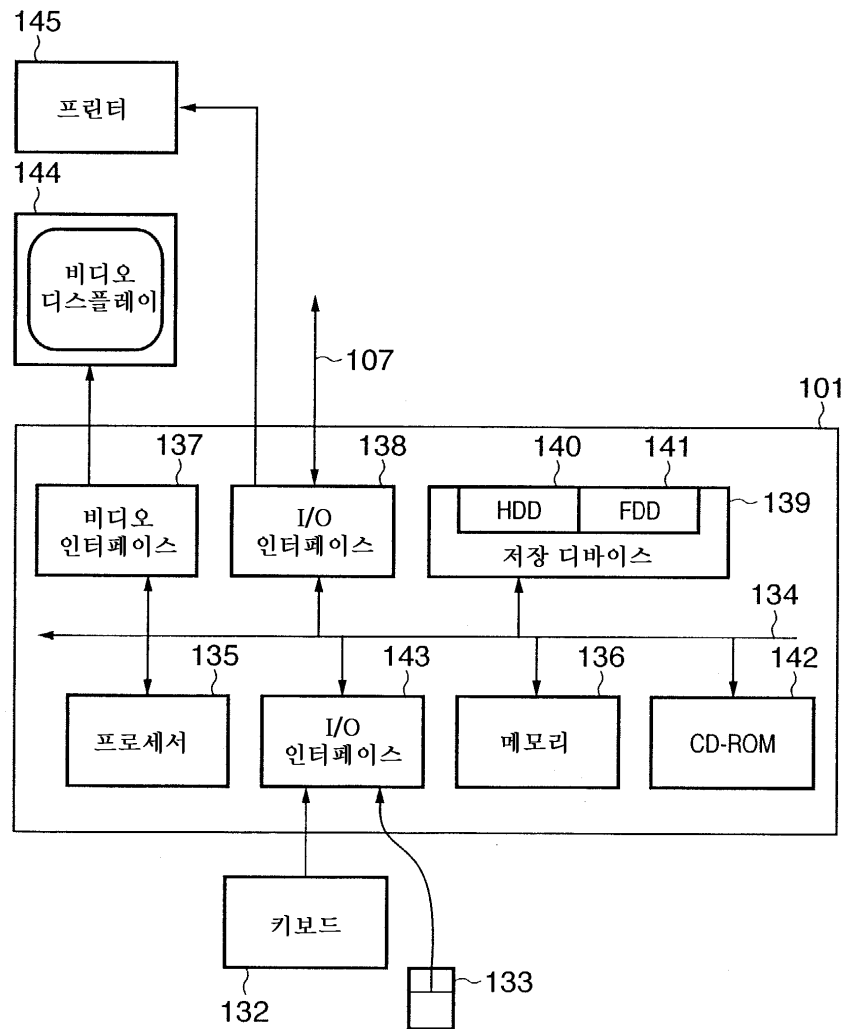
- <45> 301 : 어플리케이션 윈도우
- <46> 303 : 툴바
- <47> 313 : 커서/포인터
- <48> 406 : 링크 툴 버튼
- <49> 407, 408 : 오브젝트
- <50> 409 : 앵커 아이콘
- <51> 412 : 링크

도면

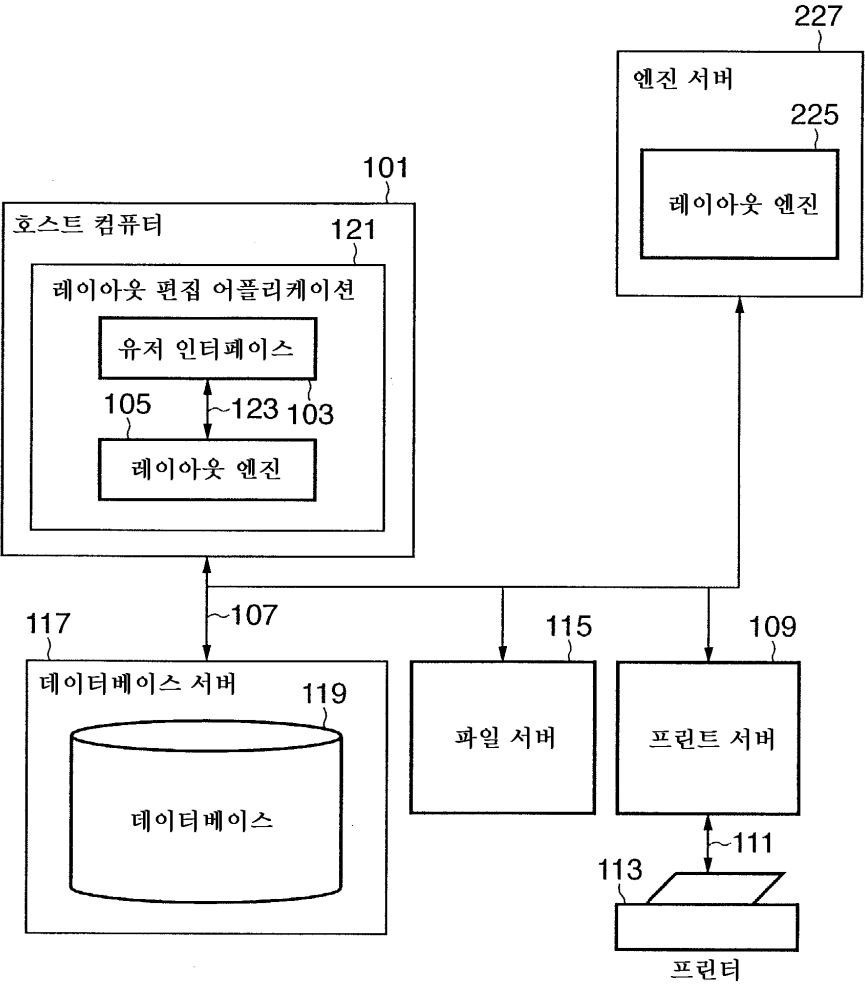
도면1a



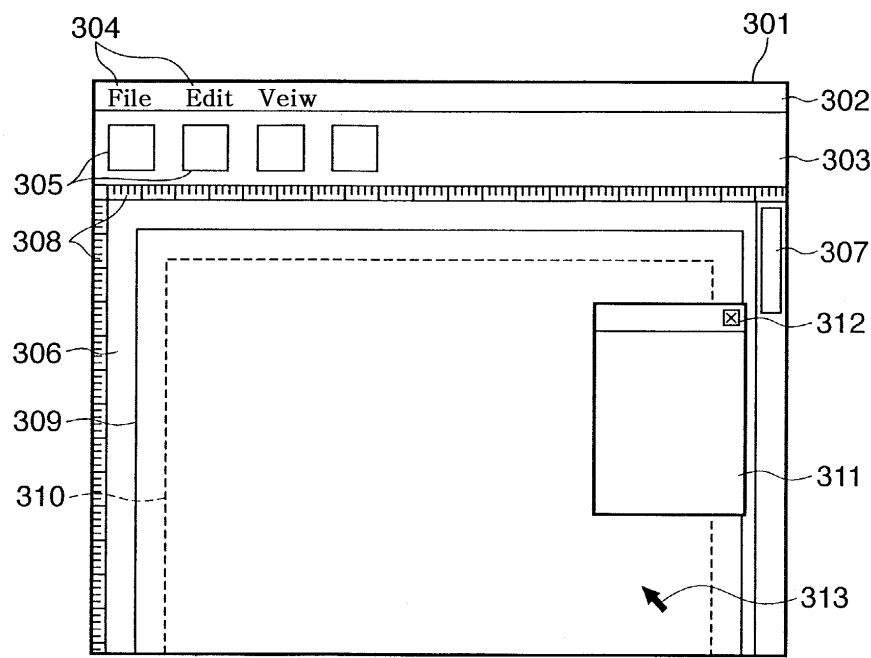
도면1b



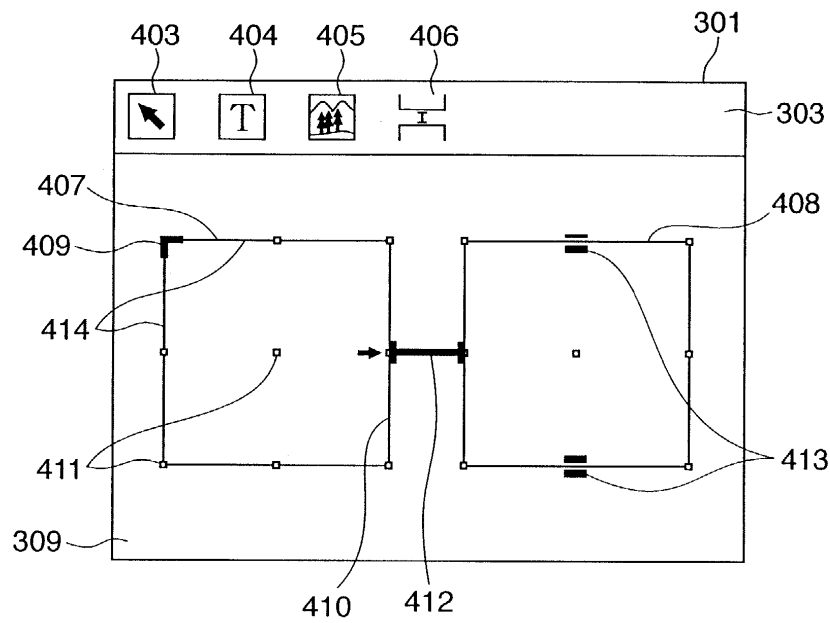
도면2



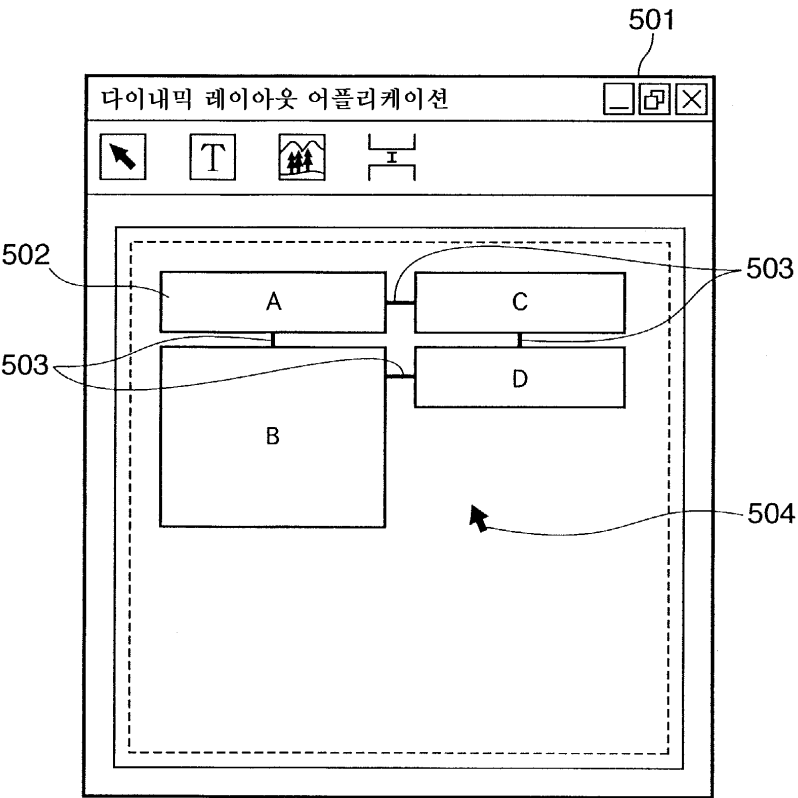
도면3



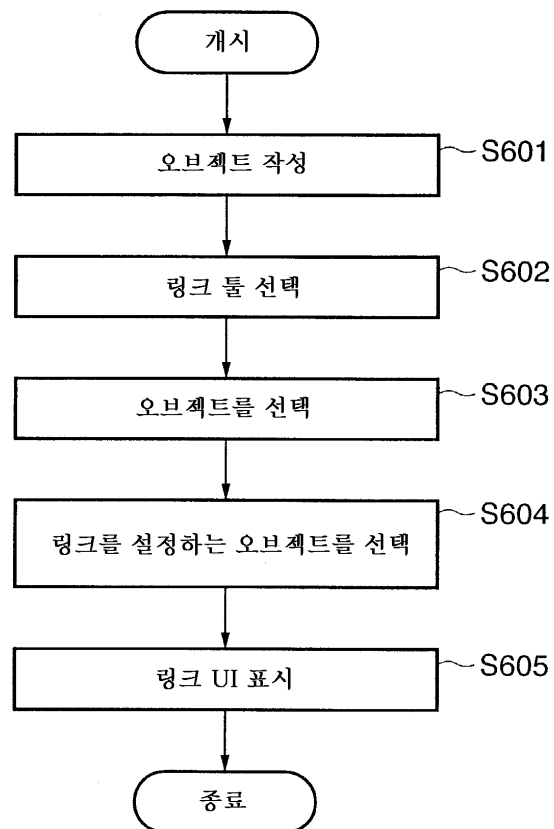
도면4



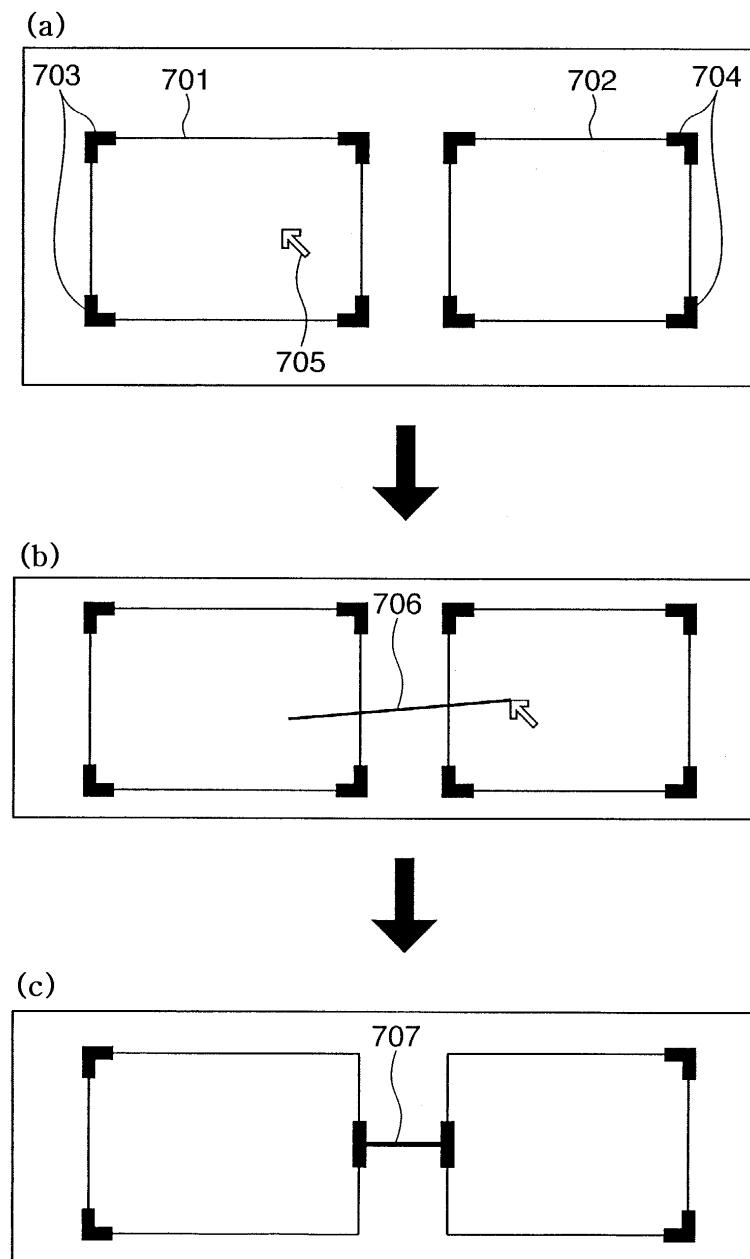
도면5



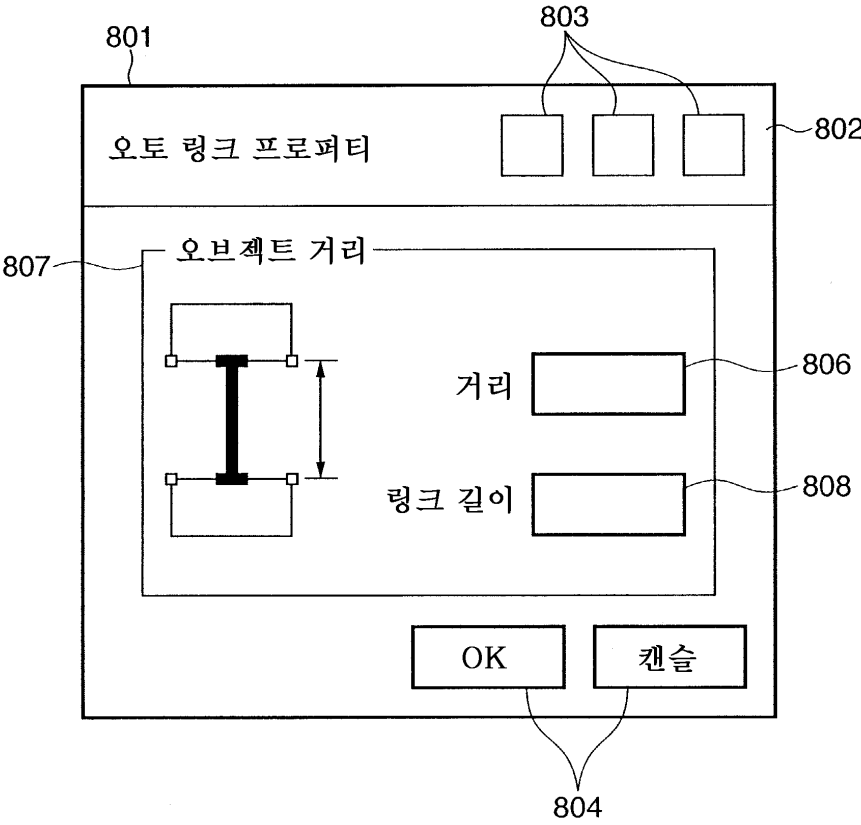
도면6



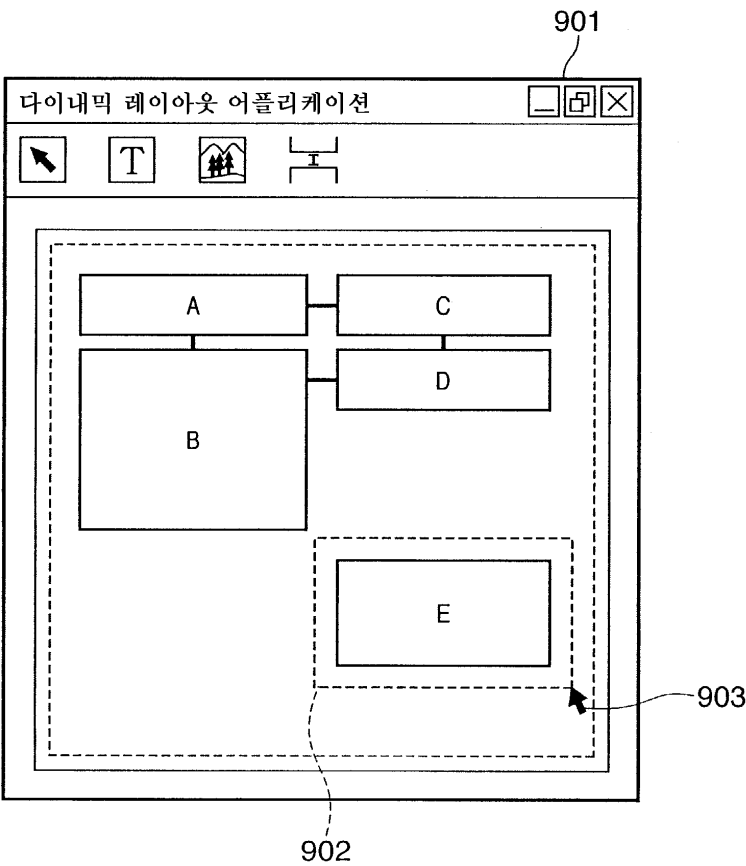
도면7



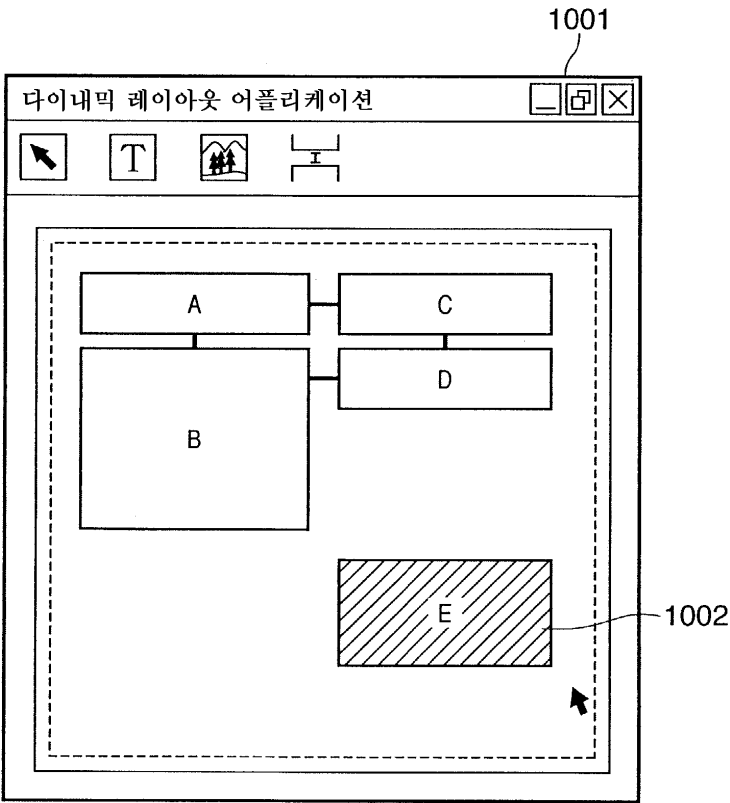
도면8



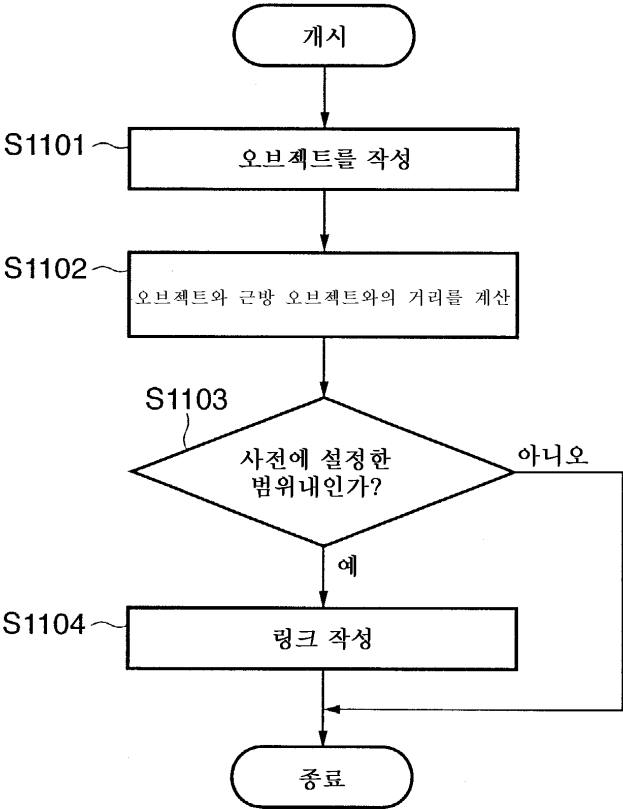
도면9



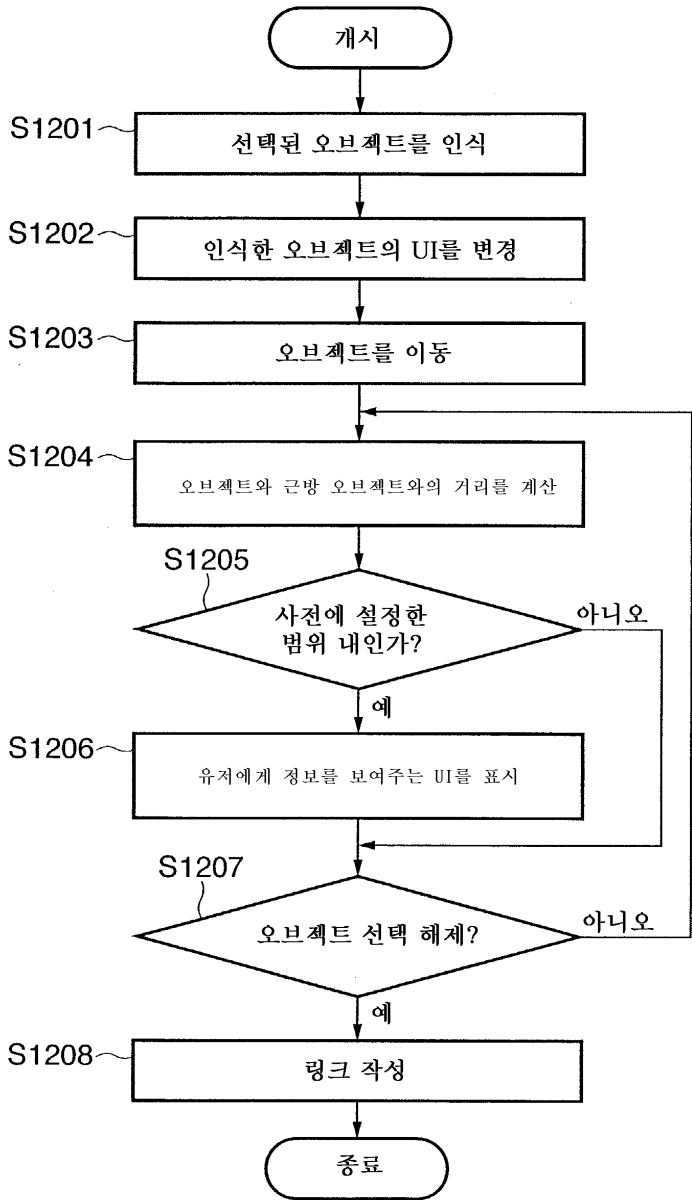
도면10



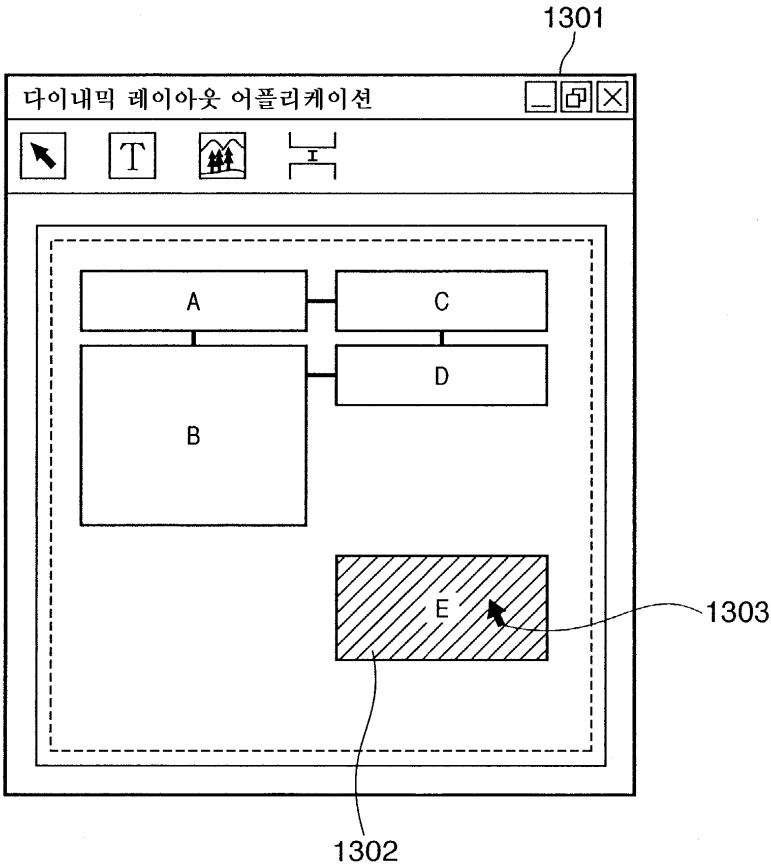
도면11



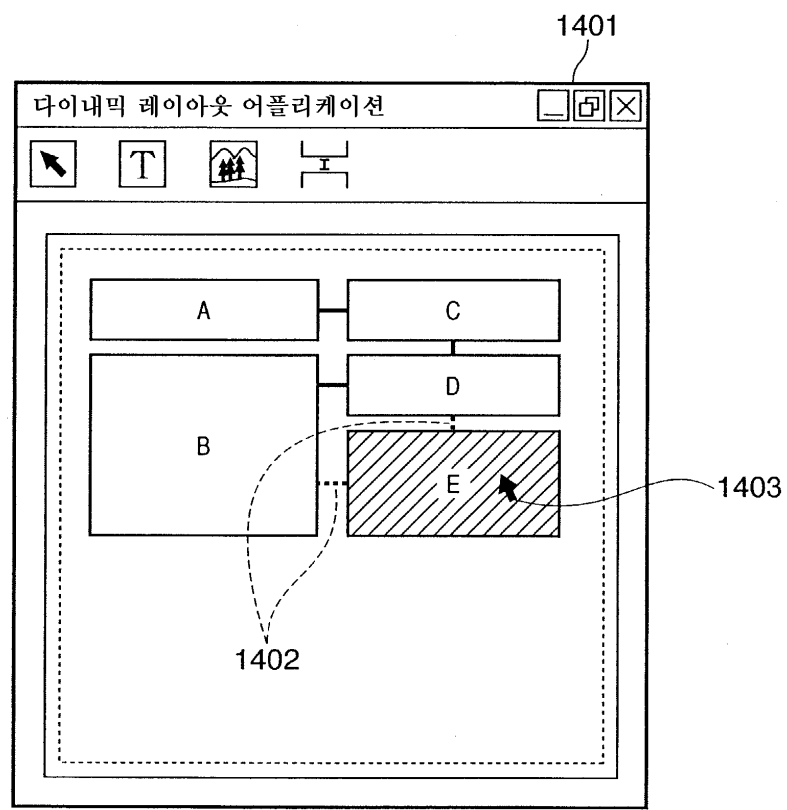
도면12



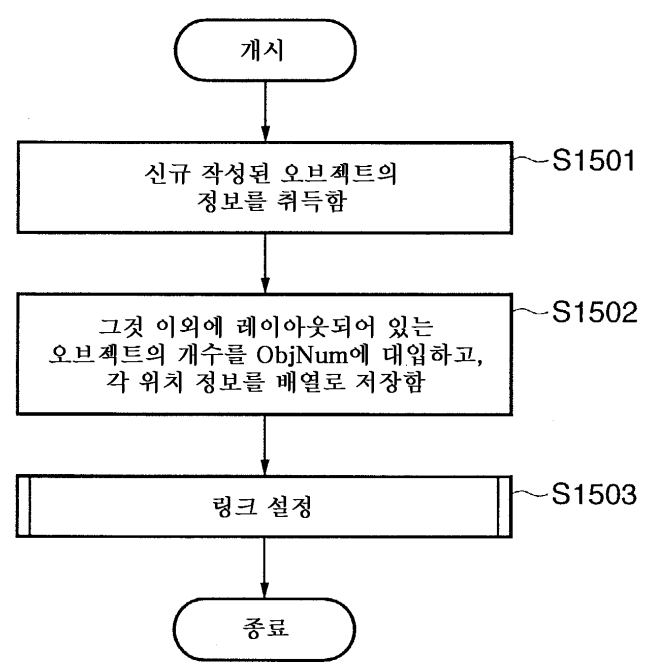
도면13



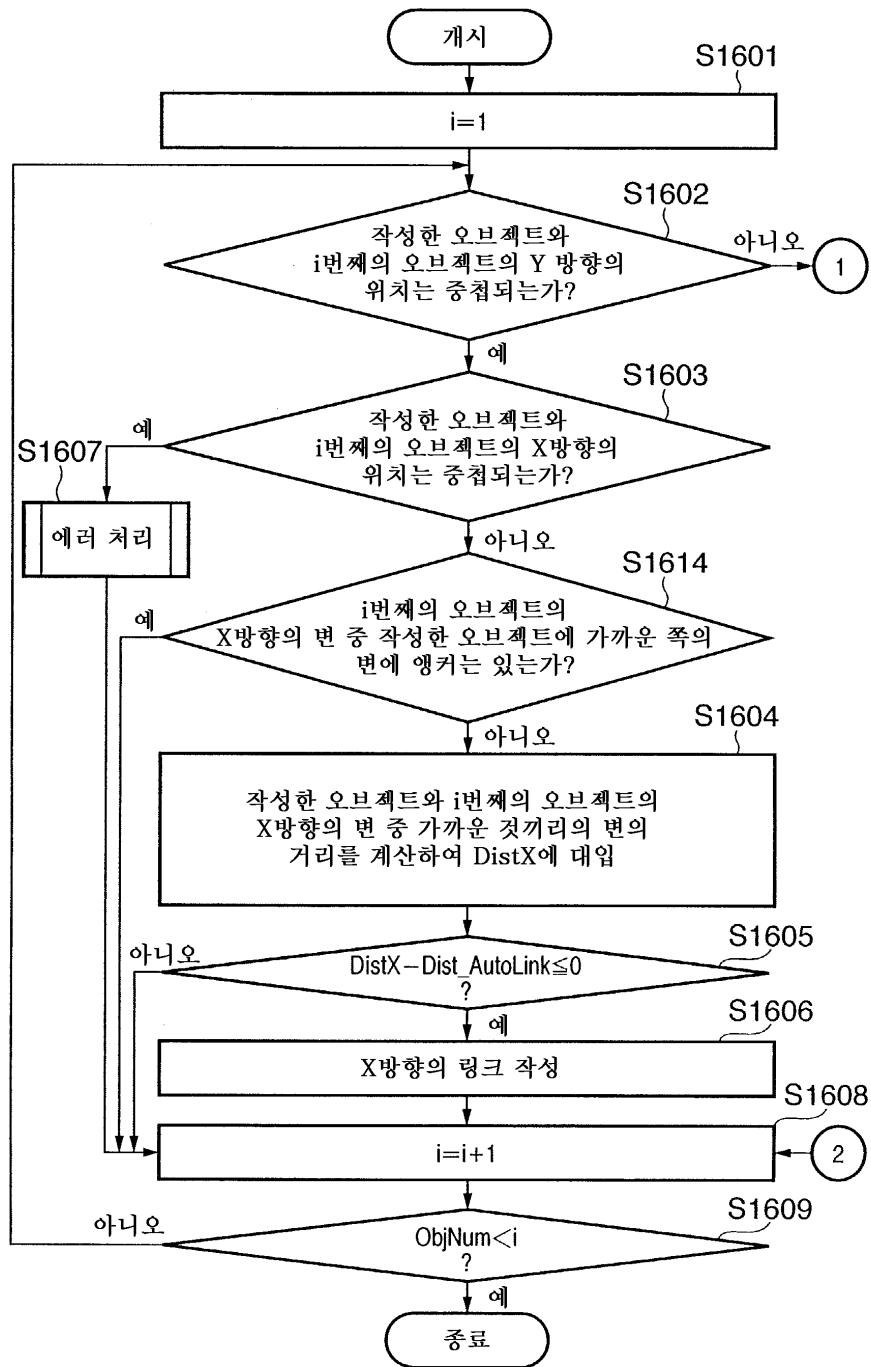
도면14



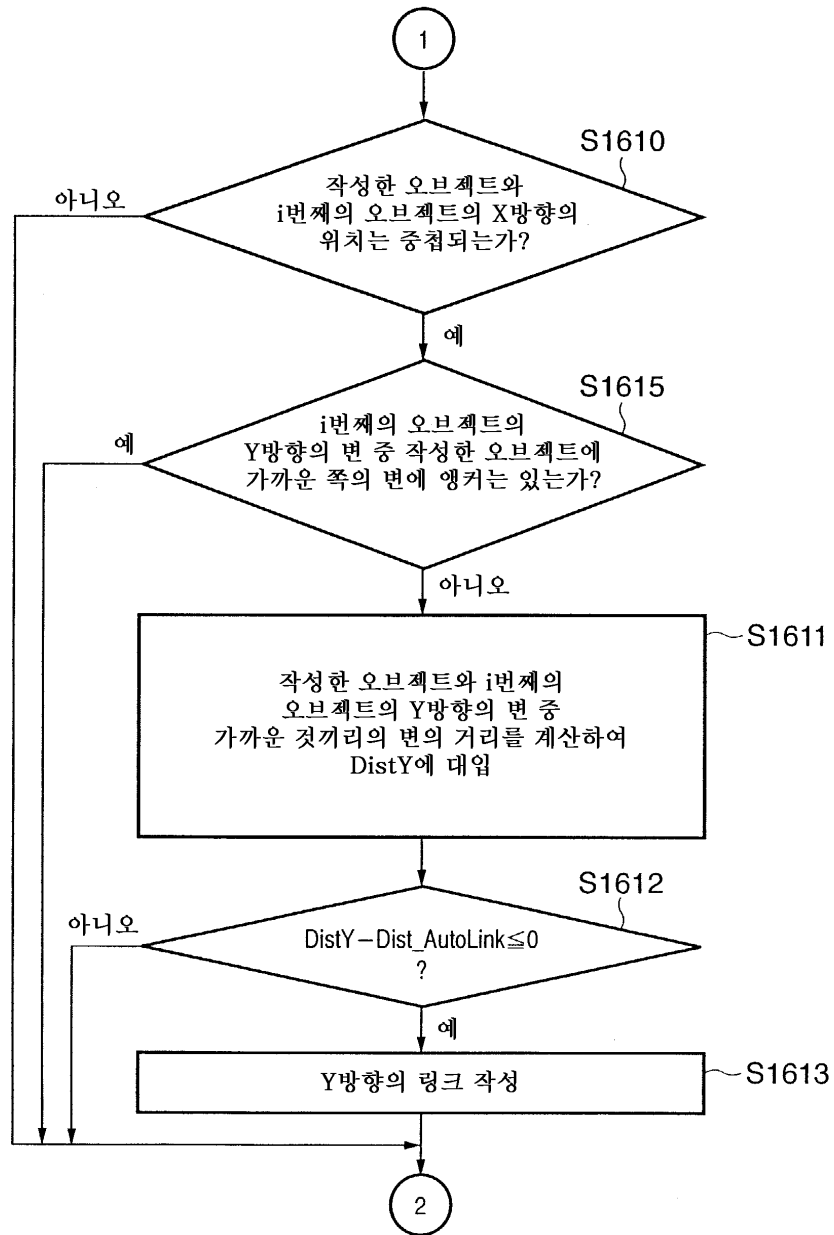
도면15



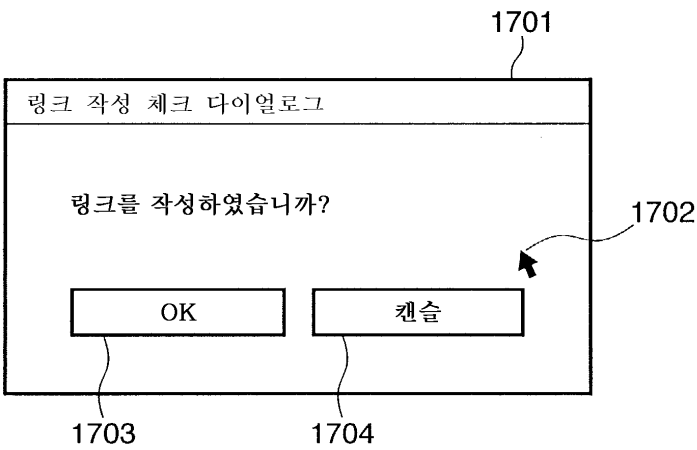
도면16a



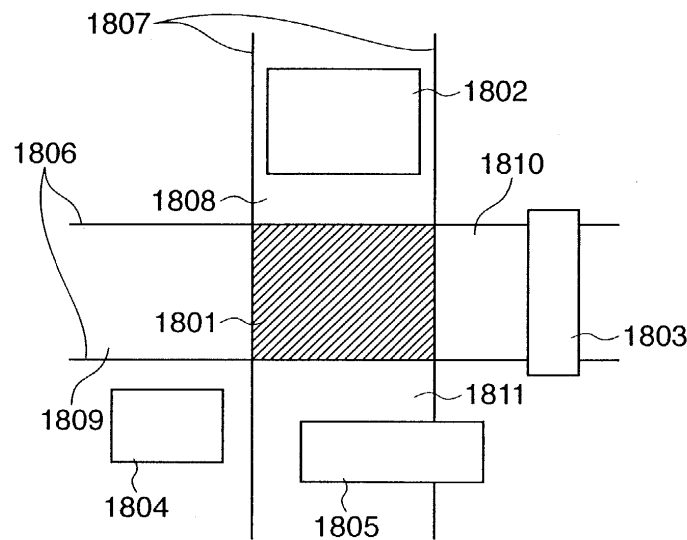
도면16b



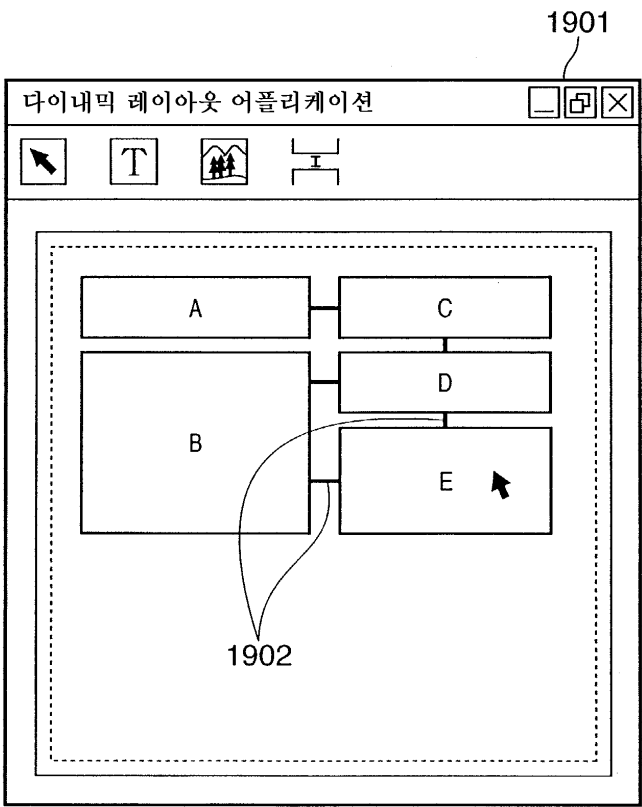
도면17



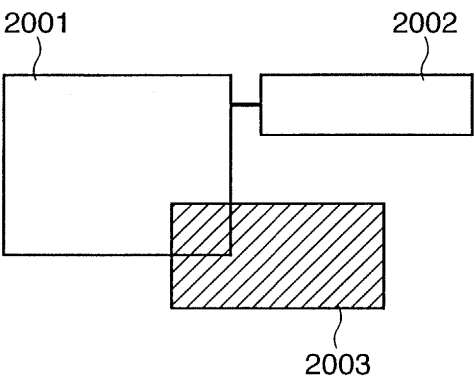
도면18



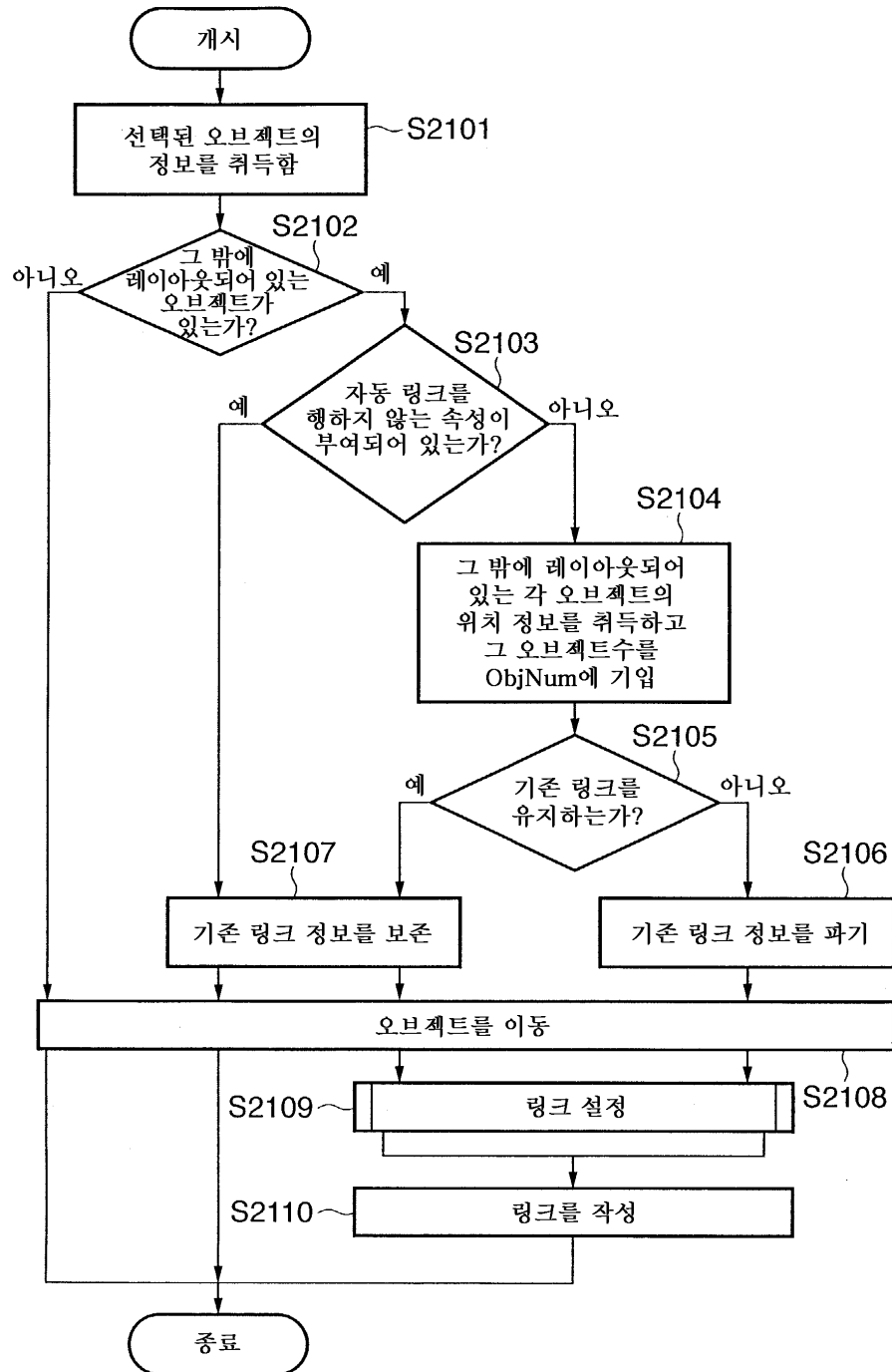
도면19



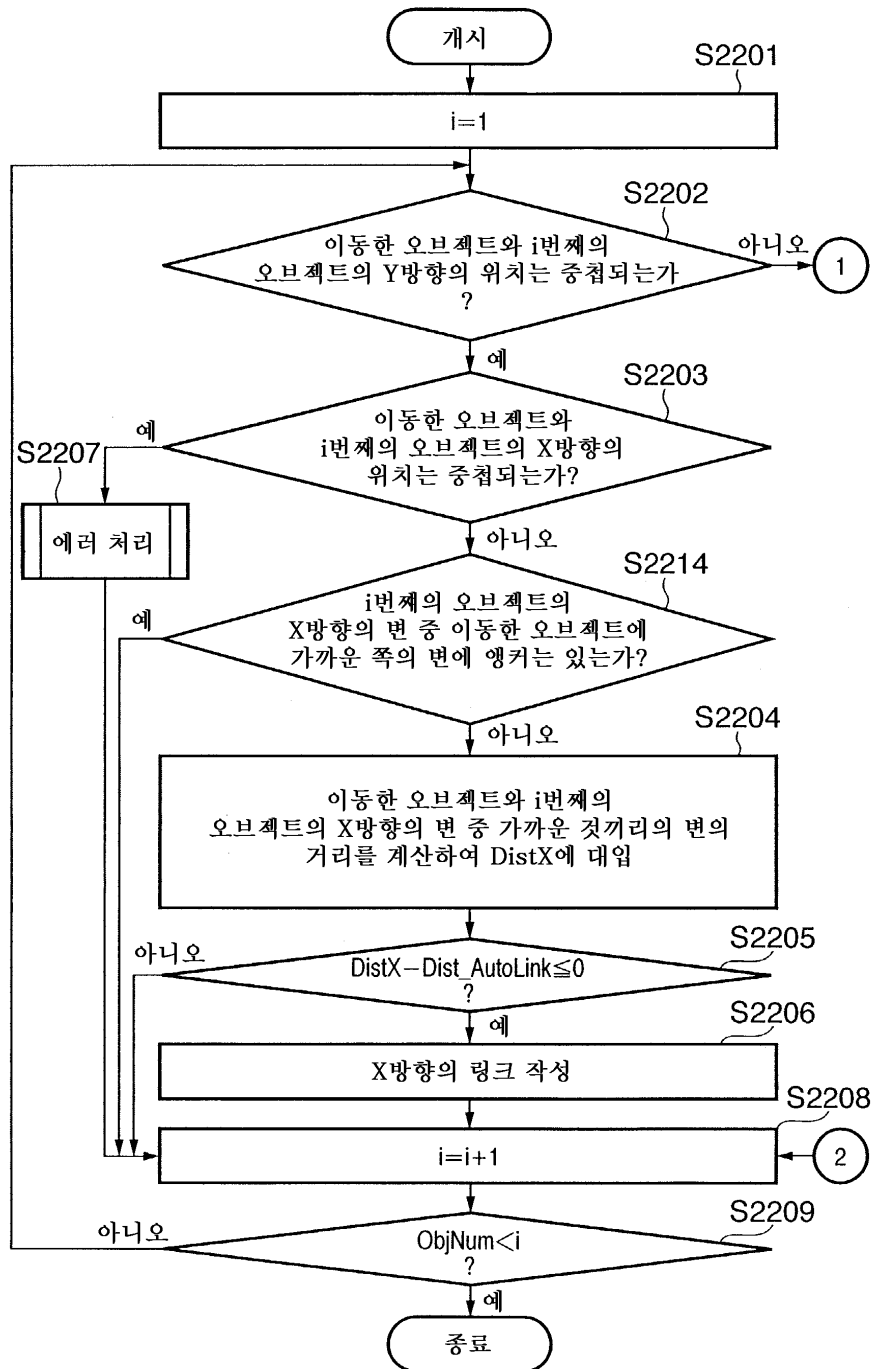
도면20



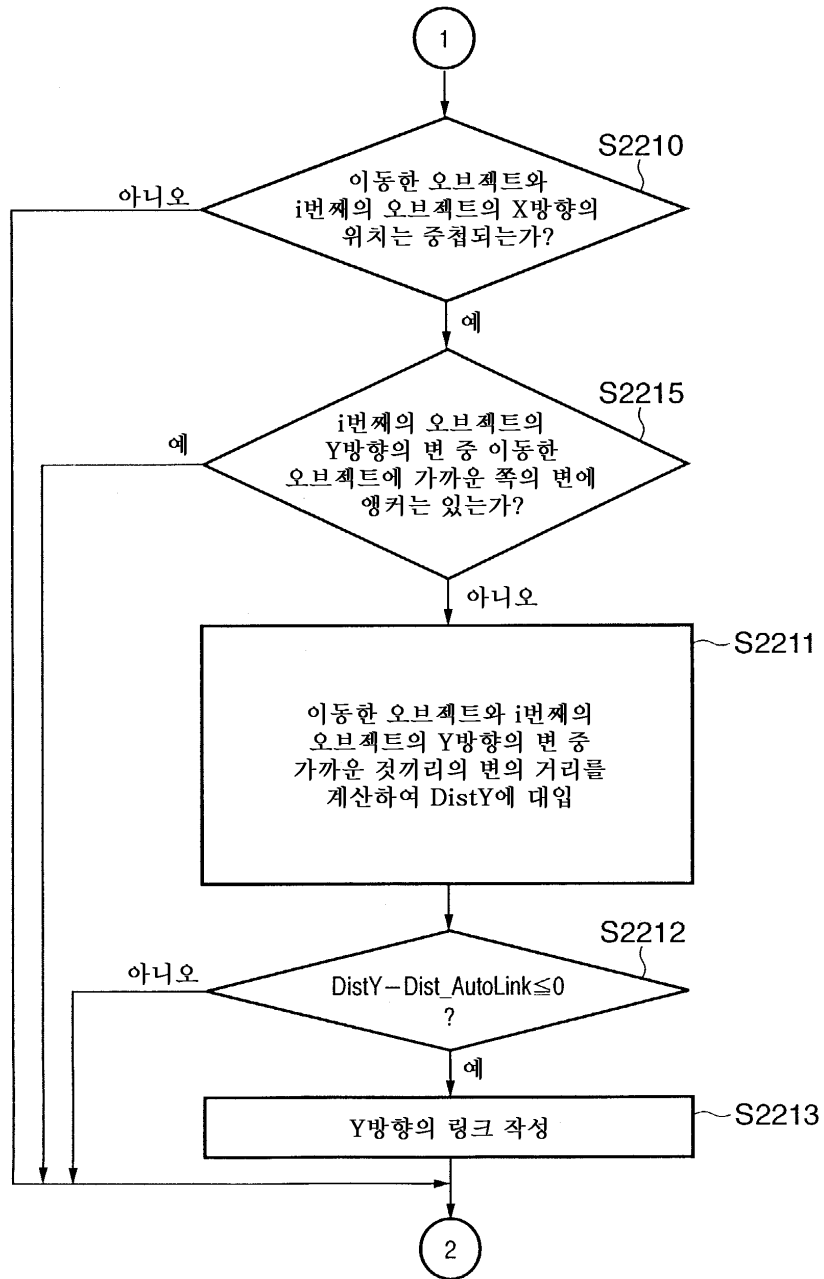
도면21



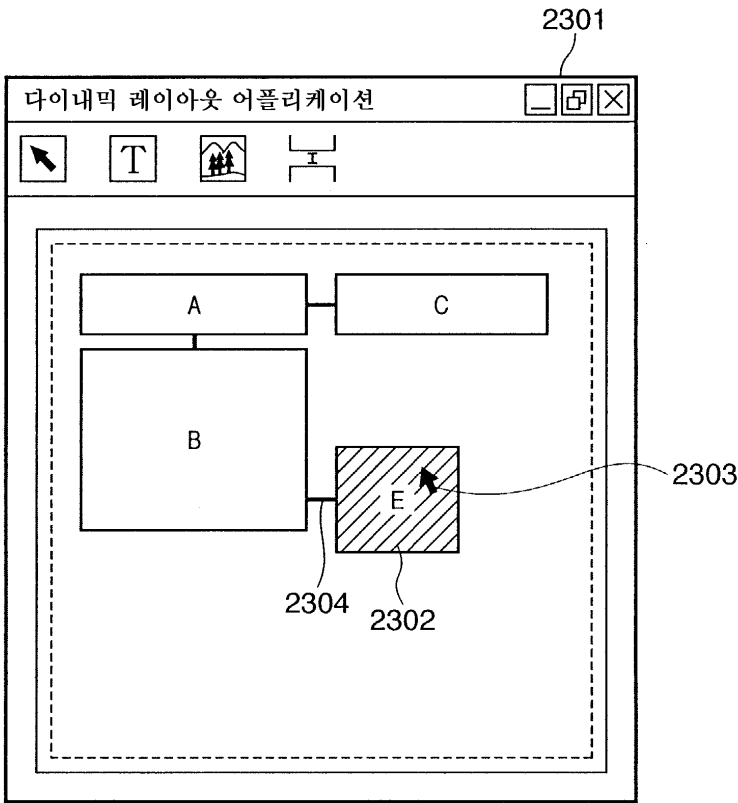
도면22a



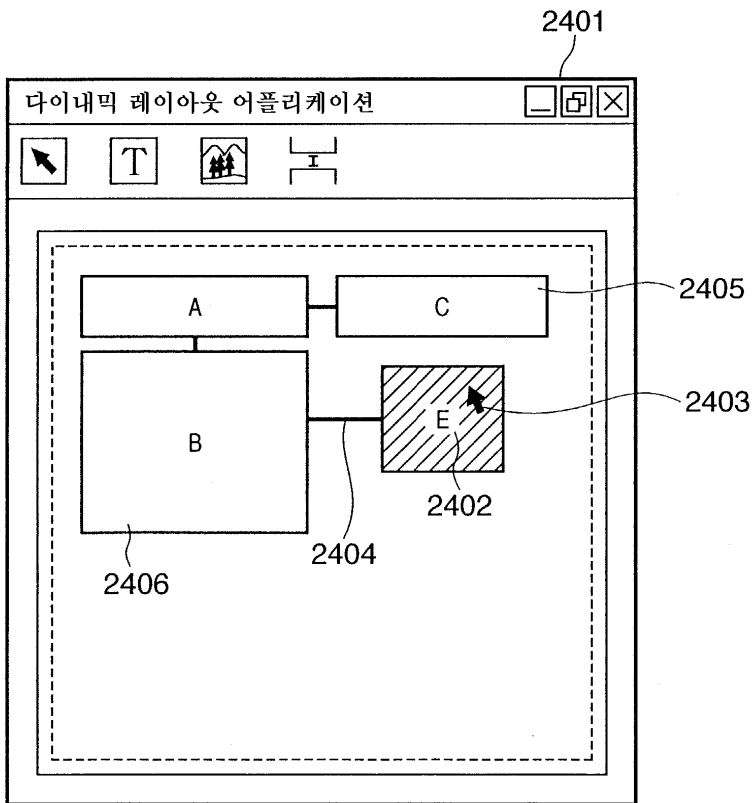
도면22b



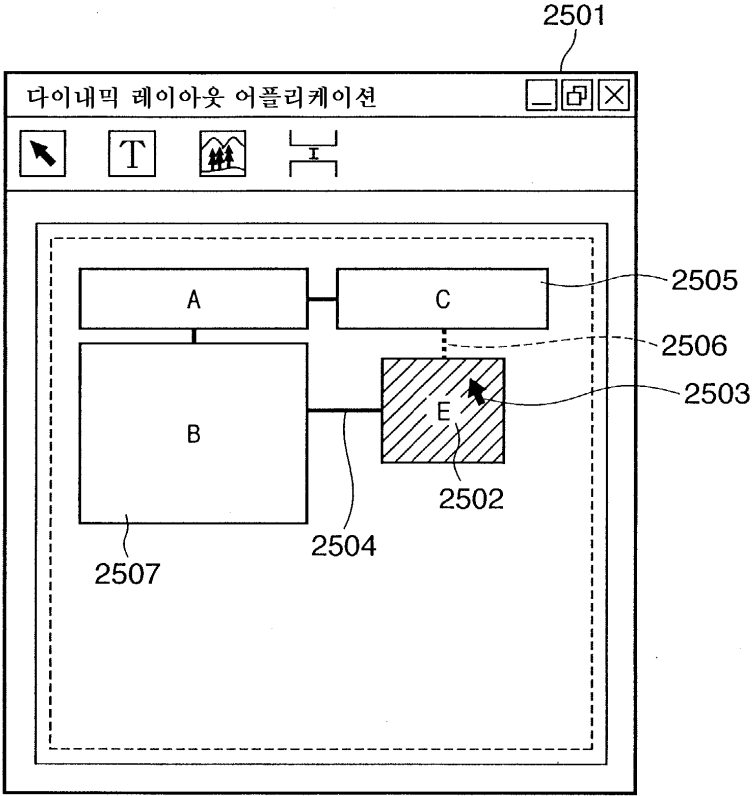
도면23



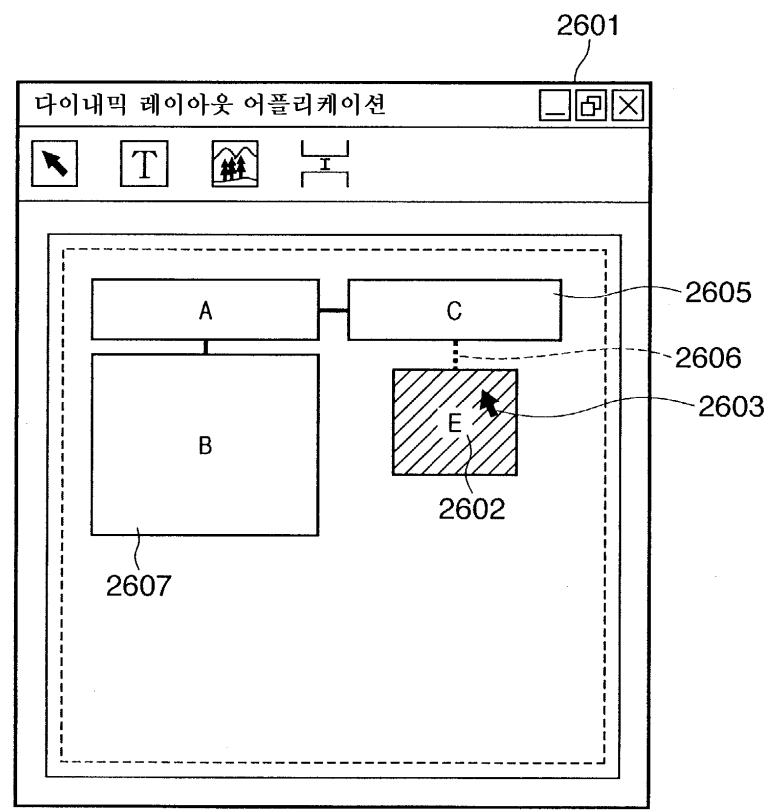
도면24



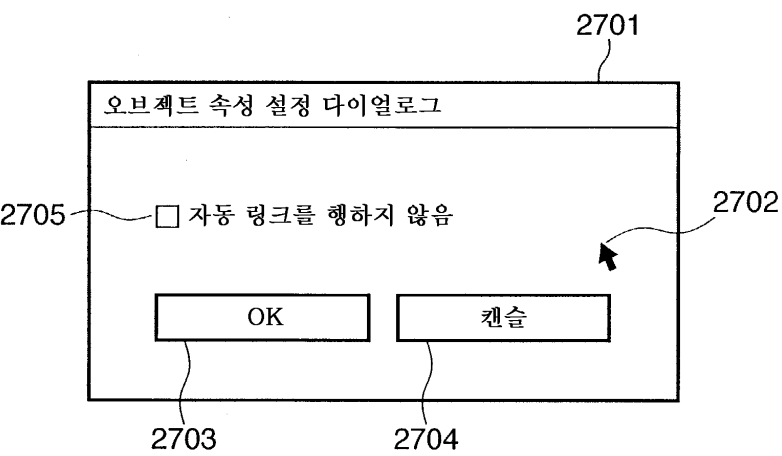
도면25



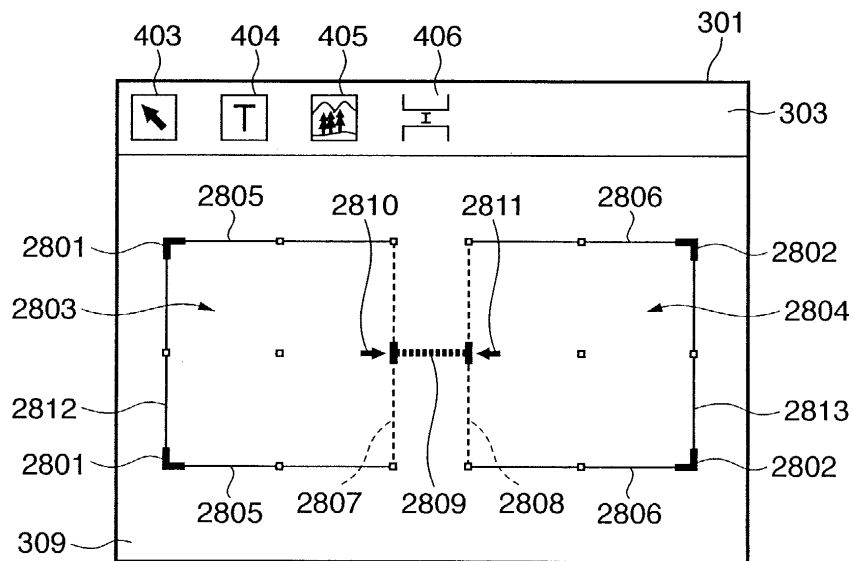
도면26



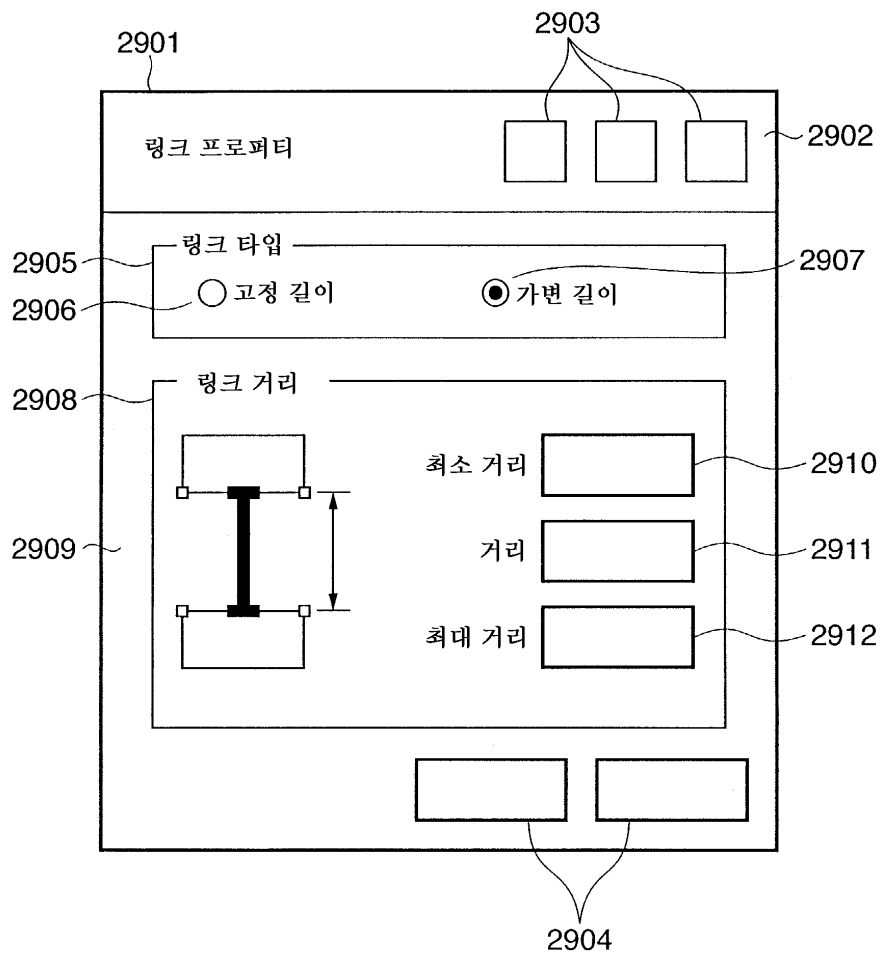
도면27



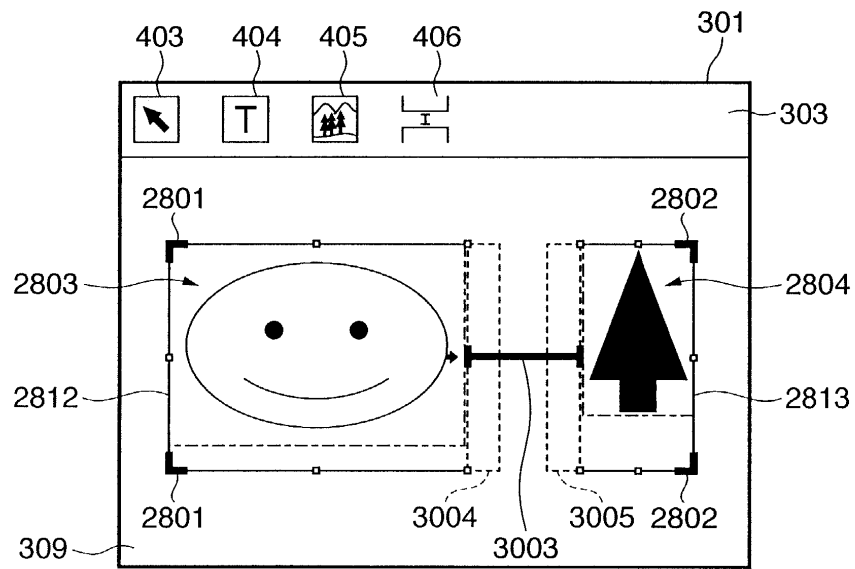
도면28



도면29



도면30



도면31

