



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2019년06월24일  
(11) 등록번호 10-1956411  
(24) 등록일자 2019년03월04일

(51) 국제특허분류(Int. Cl.)  
G06F 15/16 (2018.01)  
(21) 출원번호 10-2014-7009722  
(22) 출원일자(국제) 2012년10월15일  
심사청구일자 2017년09월15일  
(85) 번역문제출일자 2014년04월11일  
(65) 공개번호 10-2014-0079402  
(43) 공개일자 2014년06월26일  
(86) 국제출원번호 PCT/US2012/060169  
(87) 국제공개번호 WO 2013/056204  
국제공개일자 2013년04월18일  
(30) 우선권주장  
13/274,041 2011년10월14일 미국(US)  
(56) 선행기술조사문헌  
US20110185068 A1\*  
US20100306381 A1\*  
KR1020100123847 A  
US20100013839 A1  
\*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
마이크로소프트 테크놀로지 라이선싱, 엘엘씨  
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이  
(72) 발명자  
차크라보티 파라그  
미국 워싱턴주 98052-6399 레드몬드 원 마이크로  
소프트 웨이 엘씨에이 - 인터내셔널 페이턴즈 마  
이크로소프트 코포레이션  
포스트 브래들리  
미국 워싱턴주 98052-6399 레드몬드 원 마이크로  
소프트 웨이 엘씨에이 - 인터내셔널 페이턴즈 마  
이크로소프트 코포레이션  
(74) 대리인  
제일특허법인(유)

전체 청구항 수 : 총 16 항

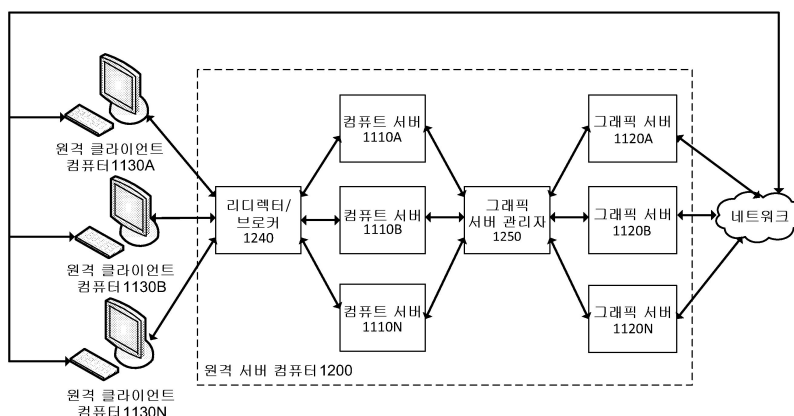
심사관 : 윤혜숙

(54) 발명의 명칭 복수의 서버로부터 클라이언트로의 단일 최종 사용자 경험 전달 기법

(57) 요약

본 명세서에는 원격 데스크톱 환경에서 멀티-포트 클라이언트 서버 접속을 생성하기 위한 방법, 시스템 및 컴퓨터 판독가능한 매체가 개시되었다. 실시예에서, 멀티-클라이언트 서버 접속은 클라이언트-서버 접속 및 서버로부터 클라이언트로의 그래픽 전달로부터의 클라이언트-서버 입력을 연결해제할 수 있다. 이러한 실시예는 보다 높은 서버 성능을 제공하며 동적 서버 리소스 관리를 가능하게 한다. 멀티-클라이언트 서버 접속은 가상 환경에서 또는 물리적 머신들의 일 세트가 클라이언트로부터 그래픽 요청을 수신하고 물리적 머신들의 다른 세트가 디스플레이 데이터를 생성하여 이를 클라이언트에게 전송하는 물리적 머신들 사이에서 구현될 수 있다.

대표도



## 명세서

### 청구범위

#### 청구항 1

데이터를 프로세싱하기 위한 시스템으로서,

원격 클라이언트 디바이스(1130)와 원격 데스크탑 세션을 실행하도록 구성된 컴퓨터 서버(1110)를 포함하되, 상기 컴퓨터 서버(1110)는,

상기 원격 클라이언트 디바이스(1130)와 제1 접속(1105)을 확립하고 그래픽 서버(1120)와의 제2 접속(1115)을 확립하고 - 상기 제1 접속은 상기 클라이언트 디바이스를 상기 클라이언트 디바이스에 할당된 원격 세션으로 접속함 -,

상기 원격 클라이언트 디바이스가 상기 그래픽 서버로의 제3 접속(1125)을 확립할 수 있도록 하기 위해 상기 그래픽 서버를 나타내는 정보를 상기 원격 클라이언트 디바이스에 제공하고 - 상기 제3 접속은 상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 수신된 데이터에 응답하여 상기 원격 클라이언트 디바이스로 인코딩된 그래픽 데이터를 전송하도록 구성됨 -,

상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 데이터를 수신하고,

상기 원격 세션의 런타임 환경 내에서 실행되는 애플리케이션을 이용해 상기 원격 클라이언트 디바이스로부터 수신된 데이터를 프로세싱하고,

상기 수신된 데이터에 응답하여 상기 애플리케이션에 의해 발행된 디스플레이 호출 데이터(display calls data)를 상기 제2 접속을 통해 상기 그래픽 서버로 전송하도록 구성되며,

상기 디스플레이 호출 데이터는 상기 그래픽 서버에 의해 상기 제3 접속을 통해 상기 원격 클라이언트 디바이스로 전송되는 그래픽 데이터를 생성하기 위해 사용되고, 상기 그래픽 데이터는 사용자 세션의 적어도 일부분의 그래픽 표현을 포함하는

시스템.

#### 청구항 2

제 1 항에 있어서,

상기 컴퓨터 서버는 적어도 하나의 가상 머신(virtual machine) 내에서 인스턴스화되는(instantiated)

시스템.

#### 청구항 3

제 1 항에 있어서,

상기 그래픽 서버는 적어도 하나의 가상화된 머신 상의 적어도 하나의 파티션(partition) 내에서 인스턴스화되는

시스템.

#### 청구항 4

제 2 항에 있어서,

상기 그래픽 서버는 상기 컴퓨터 서버와 동일한 물리적 컴퓨팅 디바이스 상의 적어도 하나의 파티션 내에서 인

스턴스화되고, 상기 컴퓨터 서버와 상기 그래픽 서버 사이의 상기 제2 접속은 적어도 하나의 인트라-파티션 통신 채널인

시스템.

#### 청구항 5

제 1 항에 있어서,

상기 원격 클라이언트 디바이스 및 상기 그래픽 서버와 상기 제3 접속을 확립하는 것은, 상기 원격 클라이언트 디바이스의 어드레스를 획득하고 상기 그래픽 서버의 적어도 하나의 어드레스를 획득하는 것과, 획득된 상기 원격 클라이언트 디바이스의 어드레스를 상기 그래픽 서버에 제공하고 획득된 상기 그래픽 서버의 어드레스를 상기 원격 클라이언트 디바이스에 제공하는 것과, 상기 원격 클라이언트 디바이스 및 상기 그래픽 서버로 하여금 제공된 어드레스들을 이용해 접속을 확립하도록 지시하는 것을 포함하는

시스템.

#### 청구항 6

제 1 항에 있어서,

상기 원격 클라이언트 디바이스 및 상기 그래픽 서버와 상기 제3 접속을 확립하는 것은, 상기 원격 클라이언트 디바이스와 상기 그래픽 서버에 부하 분산(load balancing)하는 것을 포함하는

시스템.

#### 청구항 7

제 1 항에 있어서,

상기 그래픽 데이터는 원격 데스크탑의 적어도 일부분의 그래픽 표현을 포함하는

시스템.

#### 청구항 8

제 1 항에 있어서,

상기 컴퓨터 서버는 상기 원격 클라이언트 디바이스와의 적어도 하나의 원격 데스크탑 세션에서 실행되는

시스템.

#### 청구항 9

제 1 항에 있어서,

상기 그래픽 데이터를 생성하는 것은 상기 디스플레이 호출 데이터를 렌더링하고, 캡처하고, 압축하는 것을 포함하는

시스템.

#### 청구항 10

제 1 항에 있어서,

상기 그래픽 서버는 복수의 클라이언트 디바이스에 대응하는 그래픽 데이터를 단일의 클라이언트 디바이스에 전송하도록 구성되는

시스템.

#### 청구항 11

컴퓨터 서버에 의해 수행되는, 그래픽 데이터를 프로세싱하는 방법으로서,

원격 클라이언트 디바이스와 제1 접속을 확립하고 그래픽 서버와 제2 접속을 확립하는 단계 - 상기 제1 접속은 상기 원격 클라이언트 디바이스를 상기 원격 클라이언트 디바이스에 할당된 원격 세션으로 접속함 - 와,

상기 원격 클라이언트 디바이스가 상기 그래픽 서버로의 제3 접속을 확립할 수 있도록 하기 위해 상기 그래픽 서버를 나타내는 정보를 상기 원격 클라이언트 디바이스에 제공하는 단계 - 상기 제3 접속은 상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 수신된 데이터에 응답하여 상기 원격 클라이언트 디바이스로 인코딩된 그래픽 데이터를 전송하도록 구성됨 - 와,

상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 데이터를 수신하는 단계와,

상기 원격 세션의 런타임 환경 내에서 실행되는 애플리케이션을 이용해 상기 원격 클라이언트 디바이스로부터 수신된 데이터를 프로세싱하는 단계와,

상기 수신된 데이터에 응답하여 상기 애플리케이션에 의해 발행된 디스플레이 호출 데이터를 상기 제2 접속을 통해 상기 그래픽 서버로 전송하는 단계

를 포함하고,

상기 디스플레이 호출 데이터는 상기 그래픽 서버에 의해 상기 제3 접속을 통해 상기 원격 클라이언트 디바이스로 전송되는 그래픽 데이터를 생성하기 위해 사용되고, 상기 그래픽 데이터는 사용자 세션의 적어도 일부분의 그래픽 표현을 포함하는

방법.

#### 청구항 12

제 11 항에 있어서,

상기 컴퓨터 서버는 적어도 하나의 가상 머신 내에서 인스턴스화되고, 상기 그래픽 서버는 적어도 하나의 가상화된 머신 상의 적어도 하나의 파티션 내에서 인스턴스화되는

방법.

#### 청구항 13

제 12 항에 있어서,

상기 그래픽 서버는 상기 컴퓨터 서버와 동일한 물리적 컴퓨팅 디바이스 상의 적어도 하나의 파티션 내에서 인스턴스화되고, 상기 컴퓨터 서버와 상기 그래픽 서버 사이의 상기 제2 접속은 적어도 하나의 인트라-파티션 통신 채널인

방법.

#### 청구항 14

제 11 항에 있어서,

상기 원격 클라이언트 디바이스 및 상기 그래픽 서버와 상기 제3 접속을 확립하는 것은, 상기 원격 클라이언트

디바이스의 어드레스를 획득하고 상기 그래픽 서버의 적어도 하나의 어드레스를 획득하는 것과, 획득된 상기 원격 클라이언트 디바이스의 어드레스를 상기 그래픽 서버에 제공하고 획득된 상기 그래픽 서버의 어드레스를 상기 원격 클라이언트 디바이스에 제공하는 것과, 상기 원격 클라이언트 디바이스 및 상기 그래픽 서버로 하여금 제공된 어드레스들을 이용해 접속을 확립하도록 지시하는 것을 포함하는 방법.

## 청구항 15

제 11 항에 있어서,

상기 그래픽 서버는 복수의 클라이언트 디바이스에 대응하는 그래픽 데이터를 단일의 클라이언트 디바이스에 전송하도록 구성되는

방법.

## 청구항 16

명령어가 저장된 컴퓨터 판독가능한 저장 매체로서,

상기 명령어는 하나 이상의 프로세서에 의해서 실행될 때 상기 하나 이상의 프로세서로 하여금,

원격 클라이언트 디바이스와 제1 접속을 확립하고 그래픽 서버와 제2 접속을 확립하고 - 상기 제1 접속은 상기 원격 클라이언트 디바이스를 상기 원격 클라이언트 디바이스에 할당된 원격 세션으로 접속함 -,

상기 원격 클라이언트 디바이스가 상기 그래픽 서버로의 제3 접속을 확립할 수 있도록 하기 위해 상기 그래픽 서버를 나타내는 정보를 상기 원격 클라이언트 디바이스에 제공하고 - 상기 제3 접속은 상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 수신된 데이터에 응답하여 상기 원격 클라이언트 디바이스로 인코딩된 그래픽 데이터를 전송하도록 구성됨 -,

상기 제1 접속을 통해 상기 원격 클라이언트 디바이스로부터 데이터를 수신하고,

상기 원격 세션의 런타임 환경 내에서 실행되는 애플리케이션을 이용해 상기 원격 클라이언트 디바이스로부터 수신된 데이터를 프로세싱하고,

상기 수신된 데이터에 응답하여 상기 애플리케이션에 의해 발행된 디스플레이 호출 데이터를 상기 제2 접속을 통해 상기 그래픽 서버로 전송하도록 하고,

상기 디스플레이 호출 데이터는 상기 그래픽 서버에 의해 상기 제3 접속을 통해 상기 원격 클라이언트 디바이스로 전송되는 그래픽 데이터를 생성하기 위해 사용되고, 상기 그래픽 데이터는 사용자 세션의 적어도 일부분의 그래픽 표현을 포함하는

컴퓨터 판독가능한 저장 매체.

## 발명의 설명

## 기술 분야

## 배경 기술

[0001]

원격 컴퓨팅 시스템은 사용자가 원격 컴퓨팅 시스템에 의해 호스팅되는 리소스에 액세스할 수 있게 한다. 원격 컴퓨팅 시스템상의 서버는 프로그램을 실행할 수 있고, TCP/IP, UDP 또는 그외의 프로토콜과 같은 통신 프로토콜을 준수하는 네트워크상에서 신호를 전송함으로써 접속할 수 있는 클라이언트에 사용자 인터페이스를 나타내는 신호를 전송할 수 있다. 각각의 접속하는 클라이언트에는 가상 데스크톱(virtual desktop) 또는 세션, 즉 리소스들의 세트를 포함하는 실행 환경이 제공될 수 있다. 각각의 클라이언트는 사용자 입력을 나타내는 신호를 서버에 전송할 수 있으며, 서버는 사용자 입력을 적절한 세션에 적용시킬 수 있다. 클라이언트는 서버 리소

스에 접속하기 위해서 원격 데스크톱 프로토콜(RDP; Remote Desktop Protocol)과 같은 프로토콜을 이용할 수 있다.

[0002] 네트워크 대역폭의 유용성 증가와 풍부한 2D 및 3D 클라이언트 그래픽 애플리케이션에 대한 수요의 증가로 인해, 원격 컴퓨팅 시스템 아키텍처의 변화가 있어왔다. 온전히 로컬 컴퓨팅 능력에 의존하는 대신, 서버는 그래픽 프로세싱 기능을 데이터 센터 내에 배치된 호스팅된 가상 데스크톱 인프라구조(VDI)로 이동시키는 그래픽 가상화 플랫폼을 통합한다. 클라이언트는 VDI 상에 설치된 프로세서 및 공유 그래픽 프로세싱 장치(GPU)의 그래픽 처리 능력을 활용하는, 매우 충실한 가상 데스크톱을 경험한다. 그래픽 가상화 플랫폼의 예시는, 호스팅된 VDI 데스크톱에 대해 설계된 새로운 페이로드를 전달하도록 RDP를 통합하고 Hyper-V® VDI 아키텍처 상에서 구축하는 Microsoft® RemoteFX®이다.

[0003] 전형적인 VDI 아키텍처는 호스트 파티션(host partition) 및 다수의 게스트 파티션(guest partition) 또는 가상 머신을 포함할 수 있다. 호스트 파티션은 GPU, 중앙 처리 장치(CPU) 및 메모리 공간과 같은 VDI의 언더라이징 물리적 리소스에 대한 액세스를 가지며, 이들 리소스에 대한 가상 머신의 액세스를 할당 및 관리할 수 있다. 각각의 가상 머신은 할당된 물리적 리소스를 가상화한 가상 리소스 세트를 구비한다.

[0004] 이렇게 원격 컴퓨팅 시스템에서, 클라이언트는 가상 머신 또는 가상 머신에서 실행중인 가상 데스크톱 세션에 접속할 수 있으며, 여기서 클라이언트의 인증이 관리된다. 클라이언트로부터 가상 머신으로 전송될 사용자 입력 데이터 또는 그래픽 데이터와 같은 데이터는 처음에 호스트 파티션 상의 네트워크 인터페이스 카드(NIC)로 전송된 다음 가상 머신으로 재라우팅(re-routed)된다. 가상 머신은 데이터를 프로세싱하기 위해 자신의 가상 리소스를 이용할 수 있다. 가상 머신은 언더라이징 물리적 리소스 상에서 추가로 프로세싱하기 위해 호스트 파티션에 프로세싱된 데이터를 전송한다. 호스트 파티션은 데이터를 추가로 프로세싱하여 클라이언트와의 인증을 위해 가상 머신에 다시 전송한다. 가상 머신은 데이터를 패키징하여 호스트 파티션 NIC 상에서 클라이언트로의 전송을 위해 호스트 파티션에 데이터를 재라우팅한다. 호스트 파티션과 가상 머신 사이에서의 데이터의 반복적인 이동은 상당한 양의 메모리와 CPU 리소스를 소비할 수 있고 클라이언트에 대한 데이터 전달 지연을 증가시킬 수 있는 강도높은 동작들을 필요로 한다.

## 발명의 내용

### 해결하려는 과제

### 과제의 해결 수단

[0005] 본 명세서에는 원격 컴퓨팅 시스템 환경에서 클라이언트로부터 수신된 그래픽 데이터와 같은 데이터의 프로세싱을 최적화하기 위한 시스템, 방법 및 컴퓨터 판독가능한 매체가 개시되었다. 현재의 아키텍처와 비교하여, 이러한 최적화는 메모리 및 CPU 리소스의 이용 감소와 클라이언트에 대한 데이터 전달 대기의 감소를 포함한다.

[0006] 실시예에서, 클라이언트는 가상 데스크톱 또는 가상 데스크톱의 세션에서 클라이언트 작업부하(workload)를 실행하기 위해서 컴퓨트 서버(compute server)와 같은 다른 컴퓨팅 디바이스와의 제 1 접속을 개시할 수 있다. 컴퓨트 서버는 제 1 접속을 인증할 수 있고 클라이언트 어드레스를 획득할 수 있다. 컴퓨트 서버는 그래픽 서버와의 접속을 협상할 수 있으며, 클라이언트 어드레스로부터 발생하는 접속을 예상하도록 그래픽 서버에 지시할 수 있다. 또한 컴퓨트 서버는 그래픽 서버로부터 그래픽 서버 어드레스를 획득할 수 있다. 컴퓨트 서버는 그래픽 서버 어드레스를 클라이언트에 제공할 수 있다. 결국, 클라이언트는 그래픽 서버 어드레스를 이용하여 그래픽 서버와의 제 3 접속을 확립할 수 있다. 접속이 확립되면, 클라이언트는 제 1 접속 상에서 키보드 입력, 마우스 입력 등과 같은 사용자의 스크린 입력을 컴퓨트 서버에 제공할 수 있다. 컴퓨트 서버는 제공된 입력을 프로세싱할 수 있으며, 디스플레이 커맨드 및 호출과 같이 프로세싱된 데이터를 출력할 수 있다. 컴퓨트 서버는 프로세싱된 데이터를 프로세싱하도록 그래픽 서버에 전송할 수 있다. 그래픽 서버는 수신된 데이터를 프로세싱할 수 있으며 그래픽 출력 데이터를 클라이언트에 전송할 수 있다. 이렇게, 그래픽 서버가 그래픽 출력 데이터를 다시 컴퓨트 서버로 전달할 필요가 없다. 유사하게, 컴퓨트 서버가 그래픽 출력 데이터를 클라이언트에 전달할 필요가 없다.

[0007] 컴퓨트 서버는 가상화된 컴퓨팅 시스템 상에서 호스팅된 게스트 파티션 또는 가상 머신일 수 있는 반면, 그래픽

서버는 가상화된 컴퓨팅 시스템 상의 호스트 파티션일 수 있다. 제 1 접속은, 예를 들어 TCP/IP, UDP, 또는 임의의 다른 네트워크 기반의 통신일 수 있으며, 원격 데스크톱 세션 접속을 포함할 수 있다. 제 2 접속은 예를 들어 VMBus와 같은 인트라-파티션 통신 채널, (가상 머신 모니터로도 알려진) 하이퍼바이저 상의 통신, TCP/IP, UDP, 또는 임의의 다른 네트워크 기반 접속을 포함할 수 있다. 제 3 접속은, 예를 들어 TCP/IP, UDP, FCOE, 100GB 이더넷, 또는 임의의 다른 네트워크 기반의 접속일 수 있다.

[0008] 실시예에서, 복수의 클라이언트는 복수의 컴퓨터 서버 및 복수의 그래픽 서버에 동시에 접속될 수 있다. 복수의 클라이언트의 클라이언트와의 접속을 확립하고 이를 인증하는 시작 단계에서, 리디렉터(redirector) 및/또는 브로커(broker)가 복수의 컴퓨터 서버의 제 1 컴퓨터 서버에 클라이언트를 접속하도록 사용될 수 있다. 리디렉터 및/또는 브로커는 복수의 컴퓨터 서버의 유용성을 결정할 수 있으며, 따라서 클라이언트와 제 1 컴퓨터 서버 사이의 접속을 할당할 수 있다. 이러한 컴퓨터 리소스의 부하 분산(load balancing)의 경우에, 가상 머신 라이브 이동(virtual machine live migration)과 같은 기술이 제 1 컴퓨터 서버로부터 제 2 컴퓨터 서버로 작업부하를 매끄럽게 전달할 수 있다. 제 2 컴퓨터 서버로의 클라이언트 접속이 재확립될 수 있는 한편, 클라이언트와 복수의 그래픽 서버의 그래픽 서버 사이의 접속은 동일하게 유지될 수 있다. 유사하게, 그래픽 서버 관리자는 복수의 그래픽 서버의 제 1 그래픽 서버에 클라이언트 접속된 컴퓨터 서버를 접속시키도록 사용될 수 있다. 그래픽 서버 관리자는 복수의 그래픽 서버의 유용성을 결정할 수 있으며, 따라서 클라이언트 접속된 컴퓨터 서버와 제 1 그래픽 서버 사이의 접속을 할당할 수 있다. 이렇게 그래픽 서버들의 부하 분산의 경우에서, 클라이언트 접속된 컴퓨터 서버는 제 2 그래픽 서버와의 새로운 접속을 생성할 수 있으며, 클라이언트가 제 2 그래픽 서버와의 새로운 접속을 확립할 것을 요청할 수 있다. 클라이언트는 제 2 그래픽 서버로 끊임없이 이전할 수 있다.

[0009] 복수의 클라이언트가 복수의 컴퓨터 서버 및 복수의 그래픽 서버에 동시에 접속된 실시예에서, 적어도 하나의 클라이언트는 복수의 그래픽 서버로부터 렌더링, 캡처 및 압축된 데이터를 수신하도록 구성될 수 있다. 이렇게, 적어도 하나의 클라이언트를 통한 사용자 인터페이스는 하나 이상의 클라이언트로부터 발생한 렌더링, 캡처 및 압축된 데이터를 볼 수 있다. 유사하게, 적어도 하나의 그래픽 서버는 복수의 클라이언트로부터 발생한 렌더링, 캡처 및 압축된 프로세싱 데이터를 하나의 클라이언트에 전송하도록 구성될 수 있다.

[0010] 본 요약부는 본 발명의 측면들의 개요를 제공하기 위한 것이다. 본 요약부는 본 발명의 임의의 필요한 단계 또는 구성요소를 식별하기 위한 것이 아니다. 전술된 것에 더하여, 다른 측면들이 본 명세서의 일부를 형성하는 특허청구범위, 도면 및 설명에서 기술되었다. 당업자는 본 발명의 하나 이상의 다양한 측면이 본 명세서에서 인용된 본 발명의 측면들에 영향을 미치기 위한 회로 및/또는 프로그래밍으로 한정되지 않으며, 회로 및/또는 프로그래밍은 시스템 설계자의 설계 선택에 의존하여 본 명세서에서 인용된 측면들에 영향을 미치도록 구성된 하드웨어, 소프트웨어, 및/또는 펌웨어의 가상의 임의의 조합일 수 있음을 이해할 수 있을 것이다.

## 도면의 간단한 설명

[0011] 본 발명에 따라 원격 컴퓨팅 환경에서 수신된 그래픽 데이터와 같은 데이터의 프로세싱을 최적화하는 시스템, 방법 및 컴퓨터 매체가 첨부된 도면을 참조로 하여 추가로 기술되었다.

도 1은 본 발명의 측면들이 구현될 수 있는 예시적인 컴퓨팅 환경을 도시한 도면.

도 2는 본 발명의 측면들을 실시하는 원격 컴퓨팅 환경을 도시한 도면.

도 3은 본 발명의 측면들을 실시하기 위한 복수의 원격 서버에 복수의 클라이언트가 접속할 수 있는 원격 환경을 도시한 도면.

도 4는 복수의 가상 머신을 갖는 예시적인 가상 머신 환경을 도시한 도면.

도 5는 복수의 가상 머신을 갖는 다른 예시적인 가상 머신 환경을 도시한 도면.

도 6은 본 발명의 측면들을 실시하는 복수의 가상 데스크톱 세션을 호스팅하는 원격 서버를 도시한 도면.

도 7은 본 발명의 측면들을 실시하는 예시적인 컴퓨터 서버 및 그래픽 서버를 도시한 도면.

도 8은 본 발명의 측면들을 실시하는 다른 예시적인 컴퓨터 서버 및 그래픽 서버 아키텍처를 도시한 도면.

도 9는 본 발명의 측면들을 실시하는 예시적인 컴퓨터 서버 및 복수의 그래픽 서버를 도시한 도면.



도 10은 본 발명의 측면들을 실시하는 가상 환경에서 발효된 예시적인 컴퓨트 서버 및 그래픽 서버를 도시한 도면.

도 11은 본 발명의 측면들을 실시하는 클라이언트, 컴퓨트 서버 및 그래픽 서버를 갖는 컴퓨팅 환경을 도시한 도면.

도 12는 본 발명의 측면들을 실시하는 복수의 클라이언트, 컴퓨트 서버 및 그래픽 서버를 갖는 컴퓨팅 환경을 도시한 도면.

도 13은 본 발명의 측면들을 실시하는 예시적인 방법을 도시한 순서도.

도 14는 본 발명의 측면들을 실시하는 예시적인 시스템 및 컴퓨터 판독가능한 저장 매체를 도시한 도면.

### 발명을 실시하기 위한 구체적인 내용

- [0012] 소정의 특정한 세부사항이 본 발명의 다양한 실시예에 대한 철저한 이해를 제공하기 위해 아래의 설명과 도면에서 개시되었다. 본 발명의 다양한 실시예를 불필요하게 흐리는 것을 방지하기 위해 종종 컴퓨팅 및 소프트웨어 기술과 연관된 어떤 잘 알려진 세부사항들은 개시되지 않았다. 또한, 당업자는 아래에 기술된 하나 이상의 세부사항 없이 본 발명의 다른 실시예를 실시할 수 있음을 이해할 것이다. 마지막으로, 다양한 방법들이 아래의 설명의 단계 및 시퀀스를 참조로 하여 기술되었지만, 이러한 설명은 본 발명의 실시예의 명확한 구현을 위한 것이며, 단계 및 단계의 순서가 본 발명을 실시하기 위해 요구되는 것으로 해석되어서는 안된다.
- [0013] 본 명세서에 기술된 다양한 기술들이 하드웨어 또는 소프트웨어, 또는 적당하다면 이들의 조합과의 접속에서 구현될 수 있음을 이해해야 한다. 따라서, 본 발명의 방법 및 장치, 또는 본 발명의 소정의 측면들 또는 그 일부가 플로피 디스켓, CD-ROM, 하드 드라이브, 또는 임의의 다른 머신 판독가능한 저장 매체로 구현되는 프로그램 코드(즉, 명령어)의 형태를 취할 수 있으며, 프로그램 코드가 컴퓨터와 같은 머신으로 로드되어 머신에 의해 실행될 때, 머신은 발명을 실시하기 위한 장치가 된다. 프로그램가능한 컴퓨터상의 프로그램 코드 실행의 경우에, 컴퓨팅 디바이스는 일반적으로 프로세서, (휘발성 및 비휘발성 메모리 및/또는 저장 요소를 포함하는) 프로세서에 의해 판독가능한 저장 매체, 적어도 하나의 입력 디바이스 및 적어도 하나의 출력 매체를 포함한다. 프로세스를 구현 또는 활용할 수 있는 하나 이상의 프로그램은 예를 들어 애플리케이션 프로그래밍 인터페이스(API)의 이용, 재사용가능한 제어 등을 통한 본 발명과 관련하여 기술되었다. 이러한 프로그램은 컴퓨터 시스템과의 통신을 위한 하이 레벨 절차적 또는 객체 지향적 프로그래밍 언어에서 바람직하게 구현된다. 그러나, 프로그램(들)은 원한다면 어셈블리 또는 머신 언어에서 구현될 수 있다. 임의의 경우에서, 언어는 컴파일 또는 번역된 언어일 수 있으며, 하드웨어 구현과 결합될 수 있다.
- [0014] 본 명세서 전반에서 사용되는 "회로(circuitry)"라는 용어는 하드웨어 인터럽트 컨트롤러와 같은 하드웨어 구성요소, 하드웨어 드라이브, 네트워크 어댑터, 그래픽 프로세서, 하드웨어 기반 비디오/오디오 코덱 및 이러한 하드웨어를 동작시키기 위해 사용되는 펌웨어/소프트웨어를 포함할 수 있다. "회로"라는 용어는 또한 멀티-코어 일반적인 프로세싱 장치의 하나 이상의 코어와 같은 하나 이상의 논리적 프로세서 또는 소정의 방식으로 설정된 스위치 또는 펌웨어에 의해 기능(들)을 수행하도록 구성된 멀티프로세서도 포함할 수 있다. 이러한 예시에서 논리적 프로세서(들)는 예를 들어 RAM, ROM, 펌웨어, 및/또는 가상 메모리와 같이 메모리로부터 로드된 기능(들)을 수행하도록 동작가능한 로직을 구현하는 소프트웨어 명령에 의해서 구성될 수 있다. 회로가 하드웨어와 소프트웨어의 조합을 포함하는 예시적인 실시예에서, 구현자는 논리적 프로세서에 의해 실행될 수 있는 머신 판독가능한 코드로 후속하여 컴파일되는 로직을 구현하는 소스 코드를 기록할 수 있다. 당업자는 당업계가 하드웨어, 소프트웨어, 또는 하드웨어/소프트웨어의 조합 사이의 차이가 거의 존재하지 않는다는 포인트와 진화되어왔음을 이해할 수 있기 때문에, 기능을 발효하기 위해서 하드웨어 또는 소프트웨어를 선택하는 것은 단지 설계 선택이다. 따라서, 당업자는 소프트웨어 프로세스가 동등한 하드웨어 구조로 변형될 수 있고, 하드웨어 구조가 동등한 소프트웨어 프로세스로 변형될 수 있음을 이해할 수 있기 때문에, 하드웨어 구현 또는 소프트웨어 구현의 선택은 구현자에게 남겨진다.
- [0015] 본 발명의 실시예는 하나 이상의 컴퓨터 상에서 실행할 수 있다. 도 1 및 아래의 개시 논의는 본 발명이 구현될 수 있는 적절한 컴퓨팅 환경에 대한 간략한 일반적인 설명을 제공하기 위한 것이다. 당업자는 컴퓨터 시스템이 아래에서 기술되는 일부 또는 모든 구성요소들을 구비할 수 있음을 이해할 수 있을 것이다.
- [0016] 도 1은 본 발명의 측면들을 이용하여 작동하도록 구성된 컴퓨팅 시스템의 예시를 도시한다. 컴퓨팅 시스템은 논리적 프로세싱 장치(102), 시스템 메모리(22) 및 시스템 메모리를 포함하는 다양한 시스템 구성요소들을 논리



적 프로세싱 머신(102)에 연결하는 시스템 버스(23)를 포함하는 컴퓨터(100) 등을 포함할 수 있다. 시스템 버스(23)는 임의의 다양한 버스 아키텍처를 이용하는 메모리 버스 또는 메모리 컨트롤러, 주변장치 버스 및 로컬 버스를 포함하는 다수의 타입의 임의의 버스 구조일 수 있다. 시스템 메모리는 판독 전용 메모리(ROM)(24) 및 랜덤 액세스 메모리(RAM)(104)를 포함한다. 예로서 스타트업(start up) 동안에 컴퓨터(100) 내의 요소들 사이에서 정보를 전달하는 것을 돕는 베이직 루틴을 포함하는 베이직 입력/출력 시스템(26)(BIOS)이 ROM(24)에 저장된다. 컴퓨터(100)는 도시되지 않은 하드 디스크에 기록하거나 그로부터 판독하기 위한 하드 디스크 드라이브(27), 제거가능한 자기 디스크(118)에 기록하거나 그로부터 판독하기 위한 자기 디스크 드라이브(28) 및 CD-ROM 또는 다른 광학 매체와 같은 제거가능한 광학 디스크(31)에 기록하거나 그로부터 판독하기 위한 광학 디스크 드라이브(30)를 더 포함할 수 있다. 일부 예시적인 실시예에서, 본 발명의 측면들을 포함하는 컴퓨터 실행가능한 명령어가 ROM(24), 하드 디스크(도시되지 않음), RAM(104), 제거가능한 자기 디스크(118), 광학 디스크(31), 및/또는 논리적 프로세싱 장치(102)의 캐시에 저장될 수 있다. 하드 디스크 드라이브(27), 자기 디스크 드라이브(28) 및 광학 디스크 드라이브(30)는 각각 하드 디스크 드라이브 인터페이스(32), 자기 디스크 드라이브 인터페이스(33) 및 광학 드라이브 인터페이스(34)에 의해 시스템 버스(23)에 접속된다. 드라이브 및 그와 연관된 컴퓨터 판독가능한 매체는 컴퓨터(100)에 대해 컴퓨터 판독가능한 명령어, 데이터 구조, 프로그램 모듈 및 다른 데이터의 비휘발성 저장을 제공한다. 본 명세서에 기술된 환경이 하드 디스크, 제거가능한 자기 디스크(118) 및 제거가능한 광학 디스크(31)를 이용하지만, 당업자는 자기 카세트, 플래시 메모리 카드, 디지털 비디오 디스크, 베르누리 카트리지(Bernoulli cartridges), 랜덤 액세스 메모리(RAM), 판독 전용 메모리(ROM) 등과 같이 컴퓨터에 의해 액세스 가능하며 데이터를 저장할 수 있는 다른 타입의 컴퓨터 판독가능한 매체 또한 운영 환경에서 사용될 수 있음을 이해해야만 한다.

[0017] 운영 시스템(35), 하나 이상의 애플리케이션 프로그램(36), 다른 프로그램 모듈(37) 및 프로그램 데이터(38)를 포함하는 다수의 프로그램 모듈이 하드 디스크, 자기 디스크(118), 광학 디스크(31), ROM(24) 또는 RAM(104) 상에 저장될 수 있다. 사용자는 키보드(40) 및 포인팅 디바이스(42)와 같은 입력 디바이스를 통해서 컴퓨터(100)에 커맨드 및 정보를 입력할 수 있다. (도시되지 않은) 다른 입력 디바이스는 마이크로폰, 조이스틱, 게임패드, 위성 접시, 스캐너 등을 포함할 수 있다. 이러한 입력 디바이스들과 그 외의 입력 디바이스들은 종종 시스템 버스에 접속된 직렬 포트 인터페이스(46)를 통해 논리적 프로세싱 장치(102)에 접속되지만, 병렬 포트, 게임 포트 또는 USB와 같은 다른 인터페이스에 의해서도 접속될 수 있다. 디스플레이(47) 또는 다른 타입의 디스플레이 디바이스는 또한 GPU/비디오 어댑터(112)와 같은 인터페이스를 통해서 시스템 버스(23)에 접속될 수도 있다. 디스플레이(47)에 더하여, 컴퓨터는 전형적으로 스피커 및 프린터와 같은 (도시되지 않은) 다른 주변 출력 디바이스를 포함한다. 도 1의 시스템은 또한 호스트 어댑터(55), SCSI(Small Computer System Interface) 버스(56) 및 SCSI 버스(56)에 접속된 외부 저장 디바이스(62)도 포함한다.

[0018] 컴퓨터(100)는 원격 컴퓨터(49)와 같은 하나 이상의 원격 컴퓨터에 대한 논리적 접속을 이용하여 네트워킹된 환경에서 동작할 수 있다. 원격 컴퓨터(49)는 다른 컴퓨터, 서버, 라우터, 네트워크 PC, 피어 디바이스 또는 다른 공용 네트워크 노드, 가상 머신일 수 있으며, 도 1에는 오직 메모리 저장 디바이스(50)만이 도시되었지만, 전형적으로 컴퓨터(100)에 대해 전송된 요소들을 다수 또는 전부 포함할 수 있다. 도 1에 도시된 논리적 접속은 로컬 영역 네트워크(LAN)(51)와, 일 예시로서 광역 네트워크(WAN)인 네트워크(52)를 포함할 수 있다. 이러한 네트워킹 환경은 사무실, 전사적 컴퓨터 네트워크, 인트라넷 및 인터넷에서 일반적이다.

[0019] LAN 네트워킹 환경에서 사용될 때, 컴퓨터(100)는 네트워크 인터페이스 컨트롤러(NIC)(114) 또는 어댑터를 통해서 LAN(51)에 접속될 수 있다. WAN 네트워킹 환경에서 사용될 때, 컴퓨터(100)는 전형적으로 모뎀(54) 또는 인터넷과 같이 네트워크(52) 상에서의 통신을 확립하기 위한 다른 수단을 포함할 수 있다. 내부 기기 또는 외부 기기일 수 있는 모뎀(54)은 직렬 포트 인터페이스(46)를 통해 시스템 버스(23)에 접속될 수 있다. 네트워킹된 환경에서, 컴퓨터(100)와 관련하여 도시된 프로그램 모듈 또는 그의 일부가 원격 메모리 저장 디바이스에 저장될 수 있다. 도시된 네트워크 접속이 예시적인 것이며, 컴퓨터들 사이의 통신 링크를 확립하는 다른 수단이 사용될 수 있음을 이해할 것이다. 또한, 본 발명의 다수의 실시예가 특히 컴퓨터 시스템에 잘 맞추어지도록 고안되었지만, 이러한 실시예로 본 발명이 제한되는 것은 아니다.

[0020] 일부 경우에서, 사용자는 컴퓨팅 애플리케이션을 원격으로 액세스하길 원할 수 있으며, 즉 별개의 컴퓨팅 디바이스 상에서 구동하는 애플리케이션에 액세스하길 원할 수 있다. 일 구현은 사용자에게 가상 데스크톱과 같은 원격 데스크톱을 통한 이러한 액세스를 제공한다. 원격 데스크톱 시스템의 실시예는 하나 이상의 컴퓨터를 실행할 수 있거나 또는 도 1의 컴퓨터(100)와 관련하여 기술된 구성요소들의 일부 또는 전부를 구비할 수 있다. 원격 데스크톱 시스템은 클라이언트 컴퓨터 시스템에 의해 원격으로 실행될 수 있고 클라이언트 컴퓨터 시스템

상에서 디스플레이될 수 있는 애플리케이션을 보유하는 컴퓨터 시스템이다. 도 2는 원격 데스크톱 시스템(200)의 예시적인 아키텍처를 도시한다. 원격 데스크톱 시스템(200)은 원격 클라이언트 컴퓨터(210) 및 원격 서버 컴퓨터(220)를 포함할 수 있다. 원격 클라이언트 컴퓨터(210) 및 원격 서버 컴퓨터(220)는 서로와의 가상 데스크톱 세션과 같은 원격 세션을 수행하도록 구성된다.

[0021] 도시된 바와 같이, 원격 서버 컴퓨터(220)는 원격 클라이언트 컴퓨터(210)로의 원격 세션을 수행하며, 이때 원격 서버 컴퓨터(220)는 사용자의 원격 클라이언트 세션(222) 실행으로부터 클라이언트 그래픽 출력을 전송한다. 원격 사용자 입력은 원격 클라이언트 컴퓨터(210)에서 입력된다. 입력 관리자(212)는 원격 사용자 입력을 프로세싱하여 (예컨대, 원격 데스크톱 프로토콜(RDP)과 같이 프로토콜들의 국제 텔레커뮤니케이션 연합(ITU) T.120 패밀리에 기초한 프로토콜을 이용해) 네트워크상에서 원격 서버 컴퓨터(220) 상의 원격 사용자 애플리케이션(224)에 전송할 수 있다. 네트워크는 로컬 영역 네트워크, 광역 네트워크, 케이블 네트워크, 인터넷, 월드 와이드 웹(WWW), 또는 기업 내 네트워크와 같은 임의의 타입의 통신 네트워크일 수 있다. 원격 사용자 애플리케이션(224)은 원격 서버 컴퓨터(220) 상에서 호스팅되는 원격 클라이언트 세션(222)에서 실행될 수 있다. 원격 사용자 애플리케이션(224)은 만약 원격 서버 컴퓨터(220)에서 입력이 되면 해당 입력을 프로세싱한다. 원격 사용자 애플리케이션(224)은 수신된 입력에 응답하여 원격 서버 출력을 생성하고, 이 출력은 네트워크 상에서 원격 클라이언트 컴퓨터(210)로 전송된다. 원격 클라이언트 컴퓨터(210)는 원격 사용자에게 출력 데이터를 제시한다. 따라서, 원격 클라이언트 컴퓨터(210)에서 입력이 수신되고 출력이 제시되지만, 프로세싱은 실제로 원격 서버 컴퓨터(220)에서 발생한다.

[0022] 원격 사용자 애플리케이션(224)에 추가로, 원격 클라이언트 세션(222)은 데스크톱과 같은 사용자 인터페이스 및 셸(shell), 데스크톱 내에서 마우스 움직임을 추적하는 서브시스템, 아이콘 상에서의 마우스 클릭을 프로그램의 인스턴스를 발효시키는 커맨드로 해석하는 서브시스템, 그외의 애플리케이션 등을 포함할 수 있다. 전송된 논의는 단지 예시적인 것이며, 기술된 청구사항은 다양한 클라이언트/서버 환경에서 구현될 수 있고 특정한 원격 프레젠테이션 제품으로 제한되지 않음을 이해해야만 한다.

[0023] 모든 원격 데스크톱 환경에서, 또는 대부분의 원격 데스크톱 환경에서, (원격 클라이언트 컴퓨터(210)에서 입력된) 원격 사용자 입력 데이터는 전형적으로 애플리케이션에 대한 커맨드를 나타내는 마우스 및 키보드 데이터를 포함한다. (원격 서버 컴퓨터(220)에서 원격 사용자 애플리케이션에 의해 생성되는) 출력 데이터는 전형적으로 원격 클라이언트 컴퓨터(210)에서의 디스플레이를 위한 그래픽 데이터를 포함한다. 다수의 원격 데스크톱 환경은 또한 다른 타입의 데이터를 전송하도록 확장하는 기능을 포함한다. 예시적인 실시예에서, 사용자 애플리케이션(224)으로부터의 그래픽 데이터 출력은 원격 서버 컴퓨터(220) 상에서 호스팅되는 그래픽 관리자(226)로 전송될 수 있다. 그래픽 관리자(226)는 그래픽 데이터를 렌더링, 캡처, 압축하고, 네트워크상에서 원격 클라이언트 컴퓨터(210) 상의 원격 사용자 디스플레이(214)로 그래픽 데이터를 전달할 수 있다. 원격 사용자 디스플레이(214)는 원격 사용자에게 그래픽 출력 데이터를 디스플레이할 수 있다.

[0024] 원격 데스크톱 환경의 실시예에서, 원격 서버 컴퓨터는 복수의 원격 클라이언트 컴퓨터에 대한 복수의 원격 세션(또는 가상 데스크톱)을 실행할 수 있다. 이렇게, 복수의 원격 클라이언트 컴퓨터에 대한 세션 할당을 제어하기 위해 브로커(broker)가 사용될 수 있다. 또한 원격 데스크톱 환경에서, 특정한 원격 클라이언트 컴퓨터로서의 역할을 수행할 수 있는 복수의 원격 서버 컴퓨터가 존재할 수 있다. 이렇게, 특정한 원격 클라이언트 컴퓨터로서의 역할을 하는 원격 서버 컴퓨터들의 할당을 제어하기 위해 리디렉터(redirector)가 사용될 수 있다. 도 3은 이러한 원격 데스크톱 시스템(300)의 예시적인 실시예를 도시한다.

[0025] 복수의 원격 클라이언트 컴퓨터(310)(A-N)는 도 2의 원격 클라이언트 컴퓨터(210)와 같이, 네트워크상에서 원격 서버 시스템(350)과 통신할 수 있는 임의의 컴퓨팅 디바이스일 수 있다. 원격 서버 시스템(350)은 리디렉터(330), 브로커(340) 및 복수의 원격 서버 컴퓨터(320)(A-N)를 포함할 수 있다. 리디렉터(330) 및 브로커(340)는 아래에서 기술되는 바와 같이 이들 디바이스들 각각의 기능을 구현하도록 구성된 프로세서 및 메모리를 포함하는 컴퓨팅 디바이스일 수 있다. 원격 서버 컴퓨터(320)(A-N)는 도 1의 컴퓨터(100) 및 도 2의 원격 서버 컴퓨터(220)와 관련하여 기술된 구성요소의 일부 또는 전부를 구비할 수 있다. 원격 서버 컴퓨터(320)(A-N)는 또한 가상 머신으로서 구현될 수도 있다. 가상 머신은 단일 하드웨어 인프라구조 상에서 또는 개별적인 하드웨어 인프라구조 상에서 실행될 수 있다. 브로커(340)는 (도시되지 않은) 게이트웨이를 이용하여 리디렉터(330)에 접속되는 독립형 디바이스일 수 있거나, 리디렉터(330) 내에 배치될 수 있거나, 또는 원격 서버 컴퓨터(320)(A-N) 내에 배치될 수 있다. 리디렉터(330) 또한 원격 서버 컴퓨터(320)(A-N) 내에 배치될 수 있다.

[0026] 브로커(340)는 브로커(340) 내에 저장된 세션 상태 정보에 기초하여 원격 클라이언트 컴퓨터에 세션을

할당한다. 세션 상태 정보는, 예를 들어 세션 ID, 사용자 이름, 세션이 존재하는 원격 서버 컴퓨터의 명칭, 각 원격 서버 컴퓨터 내에서 활동중인 세션의 개수 등을 포함할 수 있다. 본 명세서에 사용되는 바와 같이, 세션은 (가상 머신 세션으로도 알려진) 가상 데스크톱 세션일 수 있다.

[0027] 원격 클라이언트 컴퓨터(310)는 먼저 원격 클라이언트 컴퓨터(310)(A-N)의 부하 분산을 제공할 수 있는 리디렉터(330)에 접속한다. 이러한 경우에, 전형적으로 리디렉터(330)는 먼저 접속에 대한 요청을 수신한다. 그 다음 리디렉터(330)는 접속 요청을 수락하고, 원격 클라이언트 컴퓨터(310)가 어디로 리다이렉트될지를 결정하도록 브로커(340)에게 질의한다. 브로커(340)는 해당 특정 환경의 세션 상태 정보를 분석하여, 원격 클라이언트 컴퓨터(310)가 리다이렉트될 수 있는 원격 서버 컴퓨터(320)를 식별한다. 식별된 원격 서버 컴퓨터(320)는 원격 클라이언트 컴퓨터(310)에 의해 이전에 액세스되었지만 접속이 끊긴 후에 원격 클라이언트 컴퓨터(310)가 다시 재접속될 수 있는 세션을 소유할 수 있다. 실시예에서, 식별된 원격 서버 컴퓨터(320)는 원격 클라이언트 컴퓨터(310)가 접속될 수 있는 새로운 세션을 제공할 수 있으며, 원격 클라이언트 컴퓨터(310)는 어떠한 현존하는 다른 세션도 소유하지 않는다.

[0028] 브로커(340)는 요청된 원격 서버 컴퓨터(320)에 정보를 전송하여, 원격 클라이언트 컴퓨터(320)가 식별된 원격 서버 컴퓨터(310)와의 접속을 확립하게 할 수 있다. 예를 들어, 정보는 머신 ID, 세션 ID 및 식별된 원격 서버 컴퓨터(320)의 위치를 포함할 수 있다. 원격 클라이언트 컴퓨터(310)가 식별된 원격 서버 컴퓨터(320)와의 접속을 확립하면, 원격 클라이언트 컴퓨터(310)는 식별된 원격 서버 컴퓨터(320) 내에 존재하는 애플리케이션에 액세스할 수 있다. 이러한 애플리케이션은 원격 서버 시스템(350)에서 원격 서버 컴퓨터(320)를 식별하는데에 사용된 브로커(340)의 로직과 호환가능할 수 있다.

[0029] 실시예에서, 전송된 시스템은 예를 들어 원격 클라이언트 컴퓨터(310)를 원격 서버 컴퓨터 상에서 실행중인 복수의 가상 데스크톱 또는 세션 중 하나에 접속시키도록 사용될 수 있다. 원격 클라이언트 컴퓨터는 원격 데스크톱 프로토콜(RDP) 패킷 내의 리디렉터 토큰을 검사한다. 원격 클라이언트 컴퓨터는 리디렉터 토큰에 포함된 정보에 기초하여 다수의 가상 데스크톱 중 하나에 접속한다.

[0030] 다른 실시예에서, 원격 클라이언트 컴퓨터(310)는 브로커(340) 및 풀 관리자(pool manager)(도시되지 않음)를 이용하여 가상 데스크톱들 중 하나에 접속될 수 있다. 풀 관리자는 브로커(340) 내에 배치될 수 있다. 원격 클라이언트 컴퓨터(310)가 가상 머신(VM) 상에서 호스팅된 가상 데스크톱에 접속되었을 때 브로커(340)는 가상 데스크톱을 원격 클라이언트 컴퓨터(310)에 할당하며, 풀 관리자는 어느 가상 데스크톱이 할당에 이용가능한지를 나타낸다.

[0031] 다른 실시예에서, 원격 클라이언트 컴퓨터(310)는 가상 데스크톱에 접속될 수 있다. 원격 클라이언트 컴퓨터(310)는 인터넷 프로토콜(IP) 어드레스를 생성하고 원격 클라이언트 컴퓨터(310)와 가상 데스크톱 사이에서의 접속을 확립하기 위해서 브로커(340)에 의해 사용되는 네트워크 이름을 나타낸다. 원격 클라이언트 컴퓨터(310)(A-N)로부터 개별적인 가상 데스크톱 IP 어드레스를 숨김으로써, 원격 클라이언트 컴퓨터(310)(A-N)에 외부적으로 노출될 브로커(340)의 단일 네트워크 이름만이 초기에 필요하다.

[0032] 도 4는 복수의 가상 머신을 갖는 예시적인 가상 머신 환경을 도시한다. 가상화된 컴퓨터 시스템(400)은 도 2의 원격 서버 컴퓨터(220) 및 도 3의 원격 서버 컴퓨터(320)(A-N)를 구현하도록 사용될 수 있다.

[0033] 도 4에 도시된 바와 같이, 컴퓨터 시스템(400)은 도 1에 기술된 요소들과 가상 머신을 발효하도록 동작가능한 구성요소들을 포함할 수 있다. 이러한 하나의 구성요소는 당업계에서 가상 머신 모니터로서도 지칭될 수 있는 하이퍼바이저 마이크로커널(hypervisor microkernel)(402)이다. 하이퍼바이저 마이크로커널(402)은 컴퓨터 시스템(400)의 하드웨어로의 액세스를 제어하고 중재하도록 구성될 수 있다. 하이퍼바이저 마이크로커널(402)은 게스트 파티션 1 내지 게스트 파티션 N(N은 1보다 큰 정수이다)과 같이 파티션(partition)으로 불리는 실행 환경을 생성할 수 있다. 이때, 게스트 파티션은 하이퍼바이저 마이크로커널(402)에 의해 지원되는 기초적인 분리 단위이다. 또한 게스트 파티션은 자식 파티션(child partition)으로도 알려져 있을 수 있다. 하이퍼바이저 마이크로커널(402)은 하나의 파티션에서의 프로세스를 다른 파티션의 리소스에 액세스하는 것으로부터 분리할 수 있다. 각각의 게스트 파티션은 하이퍼바이저 마이크로커널(402)의 제어하에 있는 메모리, 디바이스, 프로세서 사이클 등과 같은 하드웨어 리소스들의 세트에 맵핑될 수 있다. 실시예에서, 하이퍼바이저 마이크로커널(402)은 독립형 소프트웨어 제품, 운영 시스템의 일부, 마더보드의 펌웨어 내의 내장물, 전용화된 집적 회로, 또는 이들의 조합일 수 있다.

[0034] 하이퍼바이저 마이크로커널(402)은 물리적 컴퓨터 시스템 내의 메모리의 게스트 운영 시스템의 뷰를 제한함으로써



써 파티션을 강요할 수 있다. 하이퍼바이저 마이크로커널(402)이 가상 머신을 인스턴스화할 때, 이는 예를 들어 시스템 물리적 메모리(SPM)의 시작 및 종료 어드레스를 갖는 메모리의 고정된 길이의 블록들과 같은 페이지를 게스트 물리적 메모리(GPM)로서 가상 머신에 할당할 수 있다. 여기에서, 시스템 메모리의 게스트의 제한된 뷰는 하이퍼바이저 마이크로커널(402)에 의해 제어된다. "게스트 물리적 메모리"라는 용어는 가상 머신의 관점으로부터 메모리의 페이지를 기술하는 약칭 방식이며, "시스템 물리적 메모리"라는 용어는 물리적 시스템의 관점으로부터 메모리의 페이지를 기술하는 약칭 방식이다. 따라서, 가상 머신에 할당된 메모리의 페이지는 게스트 물리적 어드레스(가상 머신에 의해 사용되는 어드레스) 및 시스템 물리적 어드레스(페이지의 실제 어드레스)를 가질 것이다.

[0035] 게스트 운영 시스템은 게스트 물리적 메모리를 가상화할 수 있다. 가상 메모리는 운영 시스템이 메모리를 오버커밋(over commit)하게 하고 애플리케이션에게 인접한 동작중인 메모리에 대한 단독 액세스를 부여하는 관리 기술이다. 가상화된 환경에서, 게스트 운영 시스템은 가상 게스트 어드레스로 알려진 가상 어드레스를 게스트 물리적 어드레스로 번역하기 위해서 하나 이상의 페이지 테이블을 이용할 수 있다. 이러한 예시에서, 메모리 어드레스는 게스트 가상 어드레스, 게스트 물리적 어드레스 및 시스템 물리적 어드레스를 구비할 수 있다.

[0036] 도시된 예시에서, 컴퓨터 시스템(400)은 Xen의 오픈 소스 하이퍼바이저의 도메인 0과 유사한 것으로 생각될 수도 있는 호스트 파티션을 포함한다. 호스트 파티션은 또한 부모 파티션(parent partition) 또는 루트 파티션(root partition)으로 지칭될 수도 있다. 도시된 바와 같이, 호스트 파티션은 호스트(404)를 포함할 수 있다. 호스트(404)는 컴퓨터 시스템(400)의 언더라이징 물리적 하드웨어와 애플리케이션의 상호작용을 가능케 하는 디바이스 드라이버(424)를 포함할 수 있다. 이렇게, 호스트(404)는 논리적 프로세싱 장치(102), GPU(112) 및 NIC(114)와 같은 컴퓨터 시스템(400)의 물리적 하드웨어로의 액세스를 가질 수 있다. 호스트(404)는 운영 시스템(또는 구성 유틸리티들의 세트)일 수 있다.

[0037] 호스트(404)는 가상화 서비스 제공자(VSP)(428)를 이용함으로써 게스트 파티션(1-N)에서 실행하는 게스트 운영 시스템에 리소스를 제공하도록 구성될 수 있다. 전형적으로 오픈 소스 커뮤니티에서 백-엔드(back-end) 드라이버로 지칭되는 VSP(428)는, (전형적으로 오픈 소스 커뮤니티 또는 초가상화된(paravirtualized) 디바이스에서 프론트-엔드(front-end) 드라이버로 지칭되는) 가상화 서비스 클라이언트(VSC)의 방식으로 하드웨어 리소스에 대한 인터페이스를 멀티플렉싱하도록 사용될 수 있다. 도 4에 도시된 바와 같이, 가상화 서비스 클라이언트는 게스트 운영 시스템의 맥락에서 실행한다. 그러나, 이러한 드라이버에는 게스트가 아닌 하이퍼바이저가 공급될 수 있다는 점에서 게스트 내의 나머지 드라이버들과 다르다. 예시적인 실시예에서, VSC(416, 418)과 통신하기 위해 가상화 서비스 제공자(428)에 의해 사용되는 경로는 가상화 경로로서 간주될 수 있다.

[0038] 호스트 파티션(VSP)(428) 및 게스트 파티션(VSC)(416, 418)은 가상 머신 버스(VMBus)에 접속할 수 있다. VMBus는 상호-파티션 통신을 가능하게 하는 논리적 채널이다. 게스트 파티션은 가상 프로세서(430, 432)와 같은 가상 리소스에 대한 게스트 파티션 요청은 요청을 관리할 수 있는 호스트 파티션 내의 디바이스에 VMBus를 통해서 리다이렉트될 수 있다. 호스트 파티션으로부터 게스트 파티션으로의 응답 또한 VMBus를 통해서 리다이렉트될 수 있다. 이러한 전체 프로세스는 게스트 운영 시스템(220, 222)에 대해 투명할 수 있다. 다른 실시예에서, 호스트 파티션 VSP(228) 및 게스트 파티션 VSC(216, 218)는 메시지 패킷을 전송 및 수신함으로써 TCP/IP 네트워크와 같은 네트워크 상에서 통신할 수 있다.

[0039] 도 4에 도시된 바와 같이, 예를 들어 가상화된 IDE 디바이스, 가상화된 비디오 어댑터, 가상화된 NIC 등과 같은 에뮬레이터(434)는 호스트(404) 내에서 구동하도록 구성될 수 있으며, 게스트 운영 시스템(420, 444)에 이용가능한 리소스에 부착된다. 예를 들어, 게스트 OS가 디바이스의 레지스터가 존재할 수 있는 곳으로 맵핑된 메모리 위치 또는 메모리 맵핑된 디바이스를 터치할 때, 하이퍼바이저 마이크로커널(402)은 요청을 가로채서 게스트가 입력하고자 시도했던 값을 연관된 에뮬레이터에 전달할 수 있다. 이때, 이러한 예시에서의 리소스는 가상 디바이스가 위치된 장소로서 생각될 수 있다. 이러한 방식으로 에뮬레이터를 이용하는 것은 에뮬레이션 경로로 고려될 수 있다. 에뮬레이션 경로는 디바이스를 에뮬레이션하기 위해서 VSP와 VSC 사이에서 메시지를 전달하기 위해 요구하는 것보다 더 많은 CPU 리소스를 요구하기 때문에, 가상화된 경로에 비교하였을 때 비효율적이다. 예를 들어, 에뮬레이션 경로를 통해서 디스크에 값을 기록하기 위해 필요한, 레지스터로 맵핑되는 메모리 상에서의 수백 가지 동작들이 가상화 경로에서는 VSC로부터 VSP로 전달되는 단일 메시지로 감소될 수 있다.

[0040] 각각의 게스트 파티션은 게스트 운영 시스템(420, 422)이 스레드를 실행하기 위해서 관리 및 스케줄링할 수 있는 하나 이상의 가상 프로세서(430, 432)를 포함할 수 있다. 일반적으로, 물리적 프로세서의 표현에 특정한 아키텍처를 제공하는 가상 프로세서는 실행가능한 명령어 및 연관된 상태 정보이다. 예를 들어, 하나의 가상 머

신은 Intel x86 프로세서의 특징을 갖는 가상 프로세서를 구비할 수 있는 반면, 다른 가상 프로세서는 PowerPC 프로세서의 특징을 구비할 수 있다. 이러한 예시에서 가상 프로세서는 컴퓨터 시스템의 프로세서에 맵핑될 수 있으며, 그 결과 가상 프로세서를 발표하는 명령어가 프로세서에 의해 되돌려질 것이다. 따라서, 복수의 프로세서를 포함하는 실시예에서 가상 프로세서는 프로세서들에 의해 동시에 실행될 수 있는 반면, 예를 들어 다른 프로세서는 하이퍼바이저 명령어를 실행한다. 파티션 내의 메모리와 가상 프로세서의 조합은 가상 머신에서 고려될 수 있다.

[0041] 게스트 운영 시스템(420, 422)은 예를 들어 Microsoft®, Apple®, 오픈 소스 커뮤니티 등으로부터의 운영 시스템과 같은 임의의 운영 시스템일 수 있다. 게스트 운영 시스템은 사용자/운영의 커널 모드를 포함할 수 있으며, 스케줄러, 메모리 관리자 등을 포함할 수 있는 커널을 구비할 수 있다. 일반적으로, 커널 모드는 적어도 특권을 가진 프로세서 명령어에 액세스를 승인하는 프로세서 내의 실행 모드를 포함할 수 있다. 각각의 게스트 운영 시스템은 원격 서비스 또는 가상 데스크톱 세션, 터미널 서버, 전자상거래 서버, 이메일 서버 등과 같이 게스트 운영 시스템 상에 저장된 애플리케이션들과 게스트 운영 시스템 자체를 구비할 수 있는 연관된 파일 시스템을 구비할 수 있다. 게스트 운영 시스템은 가상 프로세서 상에서 실행하기 위한 스레드를 스케줄링할 수 있으며 이러한 애플리케이션의 인스턴스가 발효될 수 있다.

[0042] 도 5는 도 4와 유사한 구성요소들을 도시한다. 그러나, 이러한 예시적인 실시예에서, 하이퍼바이저(542)는 마이크로커널 구성요소 및 가상화 서비스 제공자(428) 및 디바이스 드라이버(424)와 같이 도 4의 호스트(404) 내의 것과 유사한 구성요소들을 포함할 수 있는 반면, 관리 운영 시스템(540)은 예를 들어 하이퍼바이저(542)를 구성하도록 사용되는 구성 유틸리티들을 포함할 수 있다. 이러한 아키텍처에서, 하이퍼바이저(542)는 도 4의 하이퍼바이저 마이크로커널(402) 및 호스트(404)와 동일하거나 유사한 기능을 수행할 수 있다. 하이퍼바이저(542)는 독립형 소프트웨어 제품, 마더보드의 펌웨어 내에 내장된 운영 시스템의 일부일 수 있고, 및/또는 하이퍼바이저(542)의 일부는 전용 집적 회로에 의해 달성될 수 있다.

[0043] 다양한 실시예에서, 도 2의 원격 서버 컴퓨터(220)와 같은 원격 서버 컴퓨터는 복수의 원격 클라이언트 세션 또는 가상 데스크톱을 실행할 수 있다. 원격 클라이언트 세션(222)과 같은 각각의 원격 클라이언트 세션은 클라이언트를 접속시키기 위한 애플리케이션 환경을 나타낼 수 있다. 원격 서버 컴퓨터는 도 6에 도시된 바와 같은 각각의 원격 클라이언트 컴퓨터를 접속시키기 위한 적어도 하나의 원격 클라이언트 세션을 생성할 수 있다. 또한, 전송된 바와 같이, 원격 서버 컴퓨터(220)는 차례로 복수의 원격 클라이언트 세션을 실행하는 도 4 및 5의 컴퓨터 시스템(400)의 일부 또는 전체 구성요소를 실행하는 가상 머신일 수 있다.

[0044] 도 6에는 원격 서버 컴퓨터를 발효시키도록 구성된 회로를 포함할 수 있는 컴퓨터 시스템(600)이 도시되었으며, 다른 실시예에서 컴퓨터 시스템(600)은 원격 데스크톱 접속을 지원하도록 구성된 회로를 포함할 수 있다. 도시된 예시에서, 컴퓨터 시스템(600)은 세션 1 내지 N(N은 2보다 큰 정수)과 같은 클라이언트를 접속하기 위한 하나 이상의 원격 클라이언트 세션을 생성하도록 구성될 수 있다. 간략하게, 본 발명의 예시적인 실시예에서의 세션은 일반적으로 컴퓨터 시스템(600)의 커널(614)과 상호작용하도록 구성된, 예로서 소프트웨어 코드와 같은 복수의 서브시스템에 의해 발효되는 운영 환경을 포함할 수 있다. 예를 들어, 세션은 데스크톱 윈도우와 같은 사용자 인터페이스를 인스턴스화하는 프로세스, 윈도우 내에서 마우스의 움직임을 추적하는 서브시스템, 아이콘 상의 마우스 클릭을 프로그램의 인스턴스를 발효시키는 커맨드로 번역하는 서브시스템 등을 포함할 수 있다. 세션은 예를 들어 컴퓨터 시스템(600)이 도 2의 원격 클라이언트 컴퓨터(210)와 같은 클라이언트로부터 네트워크 접속 상에서의 접속 요청을 수신할 때, 컴퓨터 시스템(600)에 의한 사용자에게 기초하여 컴퓨터 시스템(600)에 의해 생성될 수 있다. 일반적으로, 접속 요청은 먼저 예를 들어 컴퓨터 시스템(600)의 회로에 의해 발효될 수 있는 수송 로직(610)에 의해 다루어질 수 있다. 수송 로직(610)은 접속 메시지를 수신하고 그것을 엔진(612)으로 포워딩하도록 구성될 수 있는 네트워크 어댑터, 펌웨어 및 소프트웨어를 포함한다. 도 6에 도시된 바와 같이, 일부 실시예에서 수송 로직(610)은 각 세션에 대한 프로토콜 스택 인스턴스를 포함할 수 있다. 일반적으로, 각각의 프로토콜 스택 인스턴스는 사용자 인터페이스 출력을 클라이언트로 라우팅하고, 수신된 사용자 입력을 클라이언트로부터 자신의 세션과 연관된 세션 코어(644)로 라우팅하도록 구성될 수 있다.

[0045] 도 6의 전반적인 설명에서 계속하여, 본 발명의 일부 예시적인 실시예에서 엔진(612)은 세션에 대한 요청을 프로세싱하고, 각 세션에 대한 기능을 결정하고, 세션에 대한 물리적 리소스들의 세트를 할당함으로써 세션을 생성하며, 세션에 대한 프로토콜 스택 인스턴스를 인스턴스화하도록 구성될 수 있다. 일부 실시예에서, 엔진(612)은 전송된 동작 절차의 일부를 구현할 수 있는 전용 회로 구성요소들에 의해 발효될 수 있다. 예를 들어, 일부 예시적인 실시예에서의 회로는 엔진(612)을 발효시키는 코드를 실행하도록 구성된 프로세서 및 메모리를 포함할 수 있다. 도 6에 의해 도시된 바와 같이, 일부 경우에서 엔진(612)은 접속 요청을 수신할 수 있으며,

예를 들어 라이선스가 이용가능하고 세션이 요청에 대해 생성될 수 있다고 결정할 수 있다. 컴퓨터 시스템(600)이 원격 데스크톱 성능을 포함하는 원격 컴퓨터인 경우에, 엔진(612)은 라이선스를 확인하지 않고 접속 요청에 응답하여 세션을 생성하도록 구성될 수 있다. 도 6에 도시된 바와 같이, 세션 관리자(616)는 엔진(612)으로부터 메시지를 수신하도록 구성될 수 있으며 메시지에 응답하여 세션 관리자(616)는 테이블에 세션 식별자를 추가할 수 있고, 세션 식별자에 메모리를 할당할 수 있으며, 세션 식별자에 할당된 메모리 내에 서브시스템 프로세스의 인스턴스 및 시스템 환경 변수를 생성할 수 있다.

[0046] 도 6에 의해 도시된 바와 같이, 세션 관리자(616)는 세션 코어(644)와 같은 커널 모드 부분을 포함할 수 있는 런타임 서브시스템(640)과 같은 환경 서브시스템을 인스턴스화할 수 있다. 예를 들어, 실시예에서의 환경 서브시스템은 서비스의 일부 서브셋을 애플리케이션 프로그램에 노출시키고 컴퓨터 운영 시스템(602)의 커널에 액세스 포인트를 제공하도록 구성된다. 예시적인 실시예에서, 런타임 서브시스템(640)은 프로세스 및 스레드의 실행을 제어할 수 있고, 세션 코어(644)는 스레드와 이것이 실행되기 위한 스케줄 시간에 대한 메모리를 할당하라는 요청을 커널(614)의 이식큐티브(executive)에 전송할 수 있다. 실시예에서, 세션 코어(644)가 그래픽 디스플레이 인터페이스(646)(GDI), 보안 서브시스템(650) 및 입력 서브시스템(652)을 포함할 수 있다. 이러한 실시예에서, 입력 서브시스템(652)은 세션과 연관된 프로토콜 스택 인스턴스를 통해서 클라이언트로부터 사용자 입력을 수신하고, 적절한 세션을 위해 입력을 세션 코어(644)로 전송하도록 구성될 수 있다. 일부 실시예에서, 사용자 입력은 절대적 및/또는 상대적 마우스 움직임 커맨드를 나타내는 신호, 마우스 조작, 마우스 클릭, 키보드 신호, 조이스틱 움직임 신호 등을 포함할 수 있다. 예를 들어 아이콘 상의 마우스 더블 클릭과 같은 사용자 입력은 세션 코어(644)에 의해 수신될 수 있으며, 입력 서브시스템(652)은 아이콘이 더블 클릭과 연관된 좌표에 위치되었음을 결정하도록 구성될 수 있다. 입력 서브시스템(652)은 아이콘과 연관된 애플리케이션에 대한 프로세스를 실행할 수 있는 런타임 서브시스템(640)으로 통지를 전송하도록 구성될 수 있다.

[0047] 클라이언트로부터의 입력을 수신한 것에 더하여, 애플리케이션 및/또는 데스크톱으로부터 드로우 커맨드(draw command)가 수신될 수 있으며 GDI(646)에 의해서 프로세싱될 수 있다. GDI(646)는 그래픽 객체 드로우 커맨드를 생성할 수 있는 프로세스를 포함할 수 있다. 이러한 예시적인 실시예에서의 GDI(646)는, 세션에 부착된 디스플레이 드라이버에 대해 커맨드가 포맷화되는 원격 디스플레이 서브시스템(654)에 GDI(646) 출력을 전달되도록 구성될 수 있다. 어떤 예시적인 실시예에서는, 하나 이상의 물리적 디스플레이가 예를 들어 원격 데스크톱 조건에서 컴퓨터 시스템(600)에 부착될 수 있다. 이러한 예시적인 실시예에서, 원격 디스플레이 서브시스템(654)은 원격 컴퓨터 시스템의 디스플레이 드라이버(들)에 의해 렌더링되는 드로우 커맨드를 미러링(mirror)하고 세션과 연관된 스택 인스턴스를 통해 미러링된 정보를 클라이언트에게 전송하도록 구성될 수 있다. 다른 예시적인 실시예에서, 원격 디스플레이 서브시스템(654)은 예를 들어 컴퓨터 시스템(600)이 헤드없이(headless) 구동할 수 있는 컴퓨터 시스템(600)에 물리적으로 부착된 디스플레이와 연관되지 않을 수 있는 시각적 디스플레이 드라이버(들)를 포함하도록 구성될 수 있다. 이러한 실시예에서 원격 디스플레이 서브시스템(654)은 하나 이상의 시각적 디스플레이에 대한 드로우 커맨드를 수신하고 그것을 세션과 연관된 스택 인스턴스를 통해 클라이언트에 전송하도록 구성될 수 있다. 본 발명의 실시예에서, 원격 디스플레이 서브시스템(654)은 각각의 디스플레이 드라이버에 대해 디스플레이 해상도를 결정하도록, 예컨대 가상 디스플레이와 연관된 가상 디스플레이 드라이버(들)의 디스플레이 해상도 또는 물리적 디스플레이와 연관된 디스플레이 드라이버들의 디스플레이 해상도를 결정하도록 구성될 수 있으며, 연관된 프로토콜 스택 인스턴스를 통해 클라이언트로 패킷을 라우팅하도록 구성될 수 있다.

[0048] 일부 예시적인 실시예에서, 세션 관리자(616)는 추가적으로 세션에 대한 로그-온 및 로그-오프를 조절하도록 구성될 수 있는 세션의 세션 식별자와 연관된 로그온 프로세스의 인스턴스를 인스턴스화할 수 있다. 이러한 예시적인 실시예에서, 로그온 프로세스와 연관된 그래픽 사용자 인터페이스를 나타내는 드로잉 커맨드는 클라이언트에게 전송될 수 있으며, 클라이언트의 사용자는 예를 들어 사용자이름/패스워드 조합, 스마트카드 식별자, 및/또는 생체학적 정보와 같은 계정 식별자를 로그온 스크린에 입력할 수 있다. 정보는 컴퓨터 시스템(600)으로 전송될 수 있고 엔진(612) 및 세션 코어(644)의 보안 서브시스템(650)에 라우팅될 수 있다. 예를 들어, 어떤 예시적인 실시예에서는 사용자 계정이 라이선스와 연관되었는지 여부를 엔진(612)이 결정하도록 구성될 수 있으며, 보안 서브시스템(650)은 세션에 대한 보안 토큰을 생성하도록 구성될 수 있다.

[0049] 도 6에 의해 설명된 바와 같이, 원격 서버 컴퓨터는 접속하는 원격 클라이언트 컴퓨터에 복수의 원격 데스크톱 세션을 제공할 수 있다. 원격 데스크톱 세션은 원격 클라이언트 컴퓨터에 의해서 요청된 하나 이상의 애플리케이션과 연관될 수 있다. 추가로, 그리고 도 6에 도시된 바와 같이, 원격 서버 컴퓨터는 사용자 인터페이스 스크린, 사용자 입력 커맨드 등과 같은 클라이언트 데스크톱을 대표하는 그래픽 데이터를 프로세싱할 수 있다.



또한, 원격 서버 컴퓨터는 그래픽 데이터를 렌더링, 캡처, 압축하여 클라이언트 원격 컴퓨터에 전송할 수 있다. 렌더링(rendering)은 원격 데스크톱 세션 내에서 구동하는 애플리케이션에 의해서 이루어지는 회전, 플립(flip) 및 드로우(draw)와 같은 원본 디스플레이 호출을 해석하는 프로세스를 지칭한다. 캡처(capture)는 온-스크린 비트맵 또는 프레임 변화와 같은 렌더링된 애플리케이션 콘텐츠를 취하고 애플리케이션 콘텐츠의 앞선 렌더링에 대한 변화를 지능적으로 캡처하는 프로세스를 지칭한다. 인코딩으로도 지칭되는 압축(compressing)은 각각의 접속된 원격 클라이언트 컴퓨터로 그래픽 리소스를 최적의 방식으로 공정하게 전달하는 프로세스를 지칭한다. 네트워크 품질 조건 및 타겟 원격 클라이언트 컴퓨터는 최적의 방식으로 캡처된 콘텐츠를 전달하기 위해 사용되는 압축/인코딩의 타입을 결정한다.

[0050] 아래에서 기술되는 바와 같이, 다양한 실시예에서 원격 서버 컴퓨터가 컴퓨터 서버 및 그래픽 서버를 포함할 수 있다. 컴퓨터 서버는 원격 클라이언트 컴퓨터로부터 그래픽 데이터를 수신하고, 그래픽 데이터를 프로세싱하며, 프로세싱된 그래픽 데이터를 그래픽 서버에 전송하도록 구성될 수 있다. 그래픽 서버는 컴퓨터 서버로부터 수신된 데이터를 출력 데이터로 렌더링, 캡처 및 압축하도록 구성될 수 있다. 그래픽 출력 데이터를 컴퓨터 서버로 전송하여 그래픽 출력 데이터를 원격 클라이언트 컴퓨터로 전송하도록 컴퓨터 서버를 이용하지 않고, 그래픽 서버가 그래픽 출력 데이터를 원격 클라이언트 컴퓨터에 직접 전송되도록 구성될 수도 있다.

[0051] 도 7은 컴퓨터 서버(710) 및 그래픽 서버(720)를 포함하는 원격 서버 컴퓨터(700)의 예시적인 실시예를 도시한다. 원격 서버 컴퓨터(700)의 실시예는 도 1의 컴퓨터(100), 도 2의 원격 서버 컴퓨터(220), 도 3의 원격 서버 컴퓨터(320), 도 4 및 5의 컴퓨터 시스템(400) 및 도 6의 컴퓨터 시스템(600)과 관련하여 기술된 구성요소들의 일부 또는 전부를 실행할 수 있다.

[0052] 컴퓨터 서버(710)의 실시예는 도 1의 컴퓨터(100), 도 2의 원격 서버 컴퓨터(220), 도 4 및 5의 컴퓨터 시스템(400) 또는 가상 머신(440) 및 도 6의 컴퓨터 시스템(600)과 관련하여 기술된 구성요소들의 일부 또는 전부를 실행할 수 있다. 컴퓨터 서버(710)는 또한 불충분한 GPU를 가지고 있거나 GPU를 가지고 있지 않을 수도 있다. 다른 실시예에서, 컴퓨터 서버(710)는 아래에서 기술되는 컴퓨팅 리소스를 적절하게 제공하도록 구성된 표준 서버 컴퓨터일 수 있다. 다른 실시예에서, 컴퓨터 서버(710)는 특정한 기능을 위해 구성된 컴퓨팅 디바이스일 수 있다. 예를 들어, 컴퓨터 서버는 단일 타입의 프로세싱 장치 및 오직 적은 양의 캐시 메모리만을 구비할 수 있다.

[0053] 그래픽 서버(720)는 렌더링, 캡처 및 압축 동작과 같은 그래픽 동작들에 대해 리소스를 제공하도록 구성될 수 있다. 그래픽 서버는 또한 복수의 GPU 리소스로 구성될 수도 있다. 실시예에서, 그래픽 서버(720)는 도 1의 컴퓨터(100)와 관련하여 기술된 구성요소들의 일부 또는 전부를 실행할 수 있다. 다른 실시예에서, 그래픽 서버는 도 4의 호스트(404)와 같은 호스트 파티션 상에서 호스팅될 수 있다. 컴퓨터 서버(710) 및 그래픽 서버(720)는 네트워크(섬유 채널, LAN, 무선, 이더넷 등)를 통해 접속될 수 있다. 도 4의 환경과 같은 가상화된 환경에서, 그래픽 서버(720) 및 컴퓨터 서버(710)는 VMBus를 이용하여 접속될 수도 있다.

[0054] 컴퓨터 서버(710)는 하나 이상의 애플리케이션(712)을 구동할 수 있다. 일 측면에서, 애플리케이션은 그래픽 디바이스 드라이버(714)와 연관될 수 있다. 그래픽 디바이스(714), 애플리케이션(712), 및/또는 컴퓨터 서버(710)가 그래픽 서버(720) 상의 그래픽 서버 관리자(740)와 연관될 수 있다. 그래픽 디바이스 드라이버(714), 애플리케이션(712), 및/또는 컴퓨터 서버(710)는 명령 및 데이터를 그래픽 서버 관리자(740)로 또는 이로부터 전송 및 수신할 수 있다. 일 예시로서, 그래픽 디바이스 드라이버(714), 애플리케이션(712), 및/또는 컴퓨터 서버(710)는 GPU 리소스에 대한 요청을 나타내는 제 1 데이터를 그래픽 서버 관리자(740)로 전송할 수 있다. 그래픽 서버 관리자(740)는 그래픽 서버(720)로부터의 GPU 명령을 위한 라우팅을 나타내는 제 2 데이터를 그래픽 디바이스 드라이버(714), 애플리케이션(712), 및/또는 컴퓨터 서버(710)로 전송할 수 있다.

[0055] 그래픽 서버 관리자(740)는 그래픽 서버(720)를 관리할 수 있다. 그래픽 서버 관리자(740)는 명령 및 데이터를 그래픽 서버(720)의 구성요소들로 전송할 수 있으며, 명령 및 데이터를 그래픽 서버(720)의 구성요소들로부터의 응답으로서 수신할 수 있다. 그래픽 서버(720)는 GPU 호스팅 및 프로세싱에 대해 특정화될 수 있다. 그래픽 서버(720)는 그래픽 서버 관리자(740), 프록시 그래픽 애플리케이션(722), 커널(726) 및 GPU 하드웨어(730)를 포함할 수 있다. 프록시 그래픽 애플리케이션(722)은 제 1 그래픽 디바이스 드라이버(724)와 연관될 수 있으며, 커널(726)은 제 2 그래픽 디바이스 드라이버(728)와 연관될 수 있다. 그래픽 디바이스 드라이버(724, 728)는 그래픽 프로세싱 태스크와 연관된 데이터 및 정보를 해석, 수신 및 전송할 수 있다. 일 실시예에서, 그래픽 디바이스 드라이버(724, 728)는 특정한 GPU 하드웨어(730)와 그래픽 서버 관리자(740), 컴퓨터 서버(710), 및/또는 원격 클라이언트 컴퓨터 상의 애플리케이션, 하드웨어 및 운영 시스템 사이에서 번역을 위해 선택된다.



- [0056] 도 7의 실시예에서, 그래픽 프로세싱 태스크와 연관된 명령은 애플리케이션(712)으로부터 그래픽 서버 관리자(740), 프록시 그래픽 애플리케이션(722), 커널(726), 하드웨어(730)로 일련의 계층들을 통해 흐를 수 있다. 프로세싱된 정보는 동일한 경로에서 역으로 흐를 수 있다. 도 8은 정보 경로에 대한 다른 실시예를 도시한다. 그래픽 서버 관리자(850)는 컴퓨터 서버(710), 애플리케이션(712), 및/또는 그래픽 디바이스 드라이버(714)로부터 GPU 리소스에 대한 요청을 수신하며, 라우팅 명령, 상태 명령 등을 컴퓨터 서버(710) 및 그래픽 서버(720)로 전송한다. 그 후에, GPU 태스크, 프로세싱된 정보 및 명령이 그래픽 서버(720) 및 컴퓨터 서버(710) 사이에서 직접 전송될 수 있다. 그래픽 서버 관리자(850)는 상호작용을 모니터링할 수 있으며 GPU 하드웨어(730)와 같은 GPU 상의 리소스의 할당과 관련된 다른 태스크를 수행할 수 있다.
- [0057] 도 9는 GPU 하드웨어(730)의 단일 세트와 연관된 리소스들이 GPU 프로세싱 태스크를 수행하기에 불충분할 때 사용될 수 있는 복수의 그래픽 서버(720)(A-N)를 도시한다. 이러한 실시예에는 또한 그래픽 서버 관리자(850)가 GPU 프로세싱 태스크의 일부를 제 1 그래픽 서버(720A)로부터 제 2 그래픽 서버(720B)로 이동시킬 때 사용될 수 있다. 이러한 실시예에서, 그래픽 서버 관리자(850)는 제 1 그래픽 서버(720A)의 상태를 제 2 그래픽 서버(720B)로 카피하도록 동작할 수 있다.
- [0058] 도 10은 가상화된 환경 또는 가상 데스크톱 인프라구조(VDI)에서 구현되는 컴퓨터 서버(1010) 및 그래픽 서버(1020)를 갖는 원격 서버 컴퓨터(1000)의 예시적인 실시예를 도시한다. 컴퓨터 서버(1010) 및 그래픽 서버(1020)는 도 7 내지 9의 컴퓨터 서버(710) 및 그래픽 서버(720)를 발효하도록 구성될 수 있다. 원격 서버 컴퓨터(1000)의 실시예는 도 1의 컴퓨터(100), 도 2의 원격 서버 컴퓨터(220), 도 3의 원격 서버 시스템(350), 도 4 및 5의 컴퓨터 시스템(400) 및 도 6의 컴퓨터 시스템(600)과 관련하여 기술된 구성요소들의 일부 또는 전부를 실행할 수 있다.
- [0059] 도 10은 VDI에 대한 호스트-기반 그래픽 가상화를 도시한다. 이러한 아키텍처는, 예를 들어 Microsoft® RemoteFX® 플랫폼을 이용하여 구현될 수 있다. 가상 머신(1011)(A-N)은 가상 데스크톱으로도 지칭될 수 있다. 아키텍처는 호스팅된 멀티-사용자 환경에서 최적의 방식으로 GPU 리소스를 공유하기 위해서 게스트 운영 시스템(OS)(1014)(A-N)과 물리적 GPU(112) 사이의 관계를 추출하는 가상 그래픽 프로세싱 장치(vGPU)(1016)(A-N)를 사용한다.
- [0060] 하이퍼바이저 마이크로커널(1002)은 복수의 호스트 및 게스트 파티션을 발효하도록 구성될 수 있다. 다른 실시예에서, 하이퍼바이저 마이크로커널(1002)은 (도시되지 않은) 원격 데스크톱 세션 구성요소들을 통합할 수 있다.
- [0061] 각각의 호스트 파티션은 원격 컴퓨터 서버(1000)의 물리적 GPU 리소스로서의 액세스를 갖는 그래픽 서버(1020)로서 구성될 수 있다. 각각의 호스트 파티션은 또한 그래픽 렌더링, 캡처 및 인코딩을 위한 관리 구성요소들을 포함할 수 있다. 각각의 호스트 파티션은 또한 물리적 GPU(112) 및 ASICS와 같은 호스트 기반 인코더(도시되지 않음)에 인터페이스를 제공하는 디바이스 드라이버(1026)도 포함할 수 있다. 디바이스 드라이버(1026)는 GPU, CPU 및 인코더 특정 드라이버를 포함할 수 있다.
- [0062] 각각의 게스트 파티션은 vGPU 리소스에 대한 액세스를 갖는 컴퓨터 서버(1010)로서 구성될 수 있다. 각각의 게스트 파티션은 복수의 접속된 원격 클라이언트 컴퓨터(도시되지 않음)에 대한 세션 또는 하나 이상의 가상 데스크톱을 발효할 수 있다. 각각의 게스트 파티션과 원격 클라이언트 컴퓨터 사이의 접속은 도 2-3에서 도시된 것과 유사한 RDP 7.1과 같은 원격 데스크톱 세션을 포함할 수 있다.
- [0063] vGPU(1016)는 각각의 가상 머신(1011)에서 설치된 가상 그래픽 어댑터를 제공할 수 있다. vGPU(1016)는 하나 이상의 GPU(112)를 이용하는 복수의 가상 머신에 대한 그래픽 프로세싱을 추출할 수 있다. 가상 머신(1011)에서 구동하는 애플리케이션이 예로서 DirectX® 또는 GDI 동작과 같은 그래픽 동작을 적용할 때, vGPU(1016)는 GPU(112)로부터 리소스를 획득하기 위해서 게스트 파티션(1010)과 호스트 파티션(1016) 사이의 통신 채널을 이용할 수 있다. 통신 채널은 VMBus 또는 TCP/IP 채널을 포함할 수 있다. VMBus는 메모리 공유 및 vGPU에 대해 특정한 다른 기능을 위해 하이퍼바이저 마이크로커널(1002) 내에 구성될 수 있다. 이러한 VMBus 구성은 통합 메커니즘을 하이퍼바이저 마이크로커널(1002)로 직접 제공할 수 있으며, 이때 그래픽 관련 디바이스에 대한 모든 리소스 요청이 전달될 수 있다.
- [0064] vGPU(1016)는 또한 가상 머신(1011)에 서비스 품질 메커니즘을 제공할 수 있다. 서비스 품질 메커니즘은 GPU(112)를 가장 효율적으로 이용하게 하는 부하-균형 정책에 기초하여 가상 머신(1011)으로 공정하게 GPU(112) 리소스를 전달할 수 있다.

- [0065] 호스트 파티션(1020)은 원격 데스크톱 가상 그래픽 관리자(RDVGM) 기능을 가상 머신(1011)으로 제공하도록 구성될 수 있다. RDVGM은 리소스 할당을 관리할 수 있고 원격 컴퓨터 서버(1000)의 물리적 리소스들과 각각의 가상 머신 게스트 운영 시스템(OS)(1014)으로의 vGPU(1016) 리소스 할당 사이의 제어를 프로세싱한다. RDVGM 기능은, 렌더링, 캡처 및 압축(RCC) 프로세스의 관리, vGPU(1016)를 통한 가상 머신(1011)으로의 GPU(112) 리소스 할당, 가상 머신(1011)으로의 리소스 정책의 할당 및 복수의 가상 머신(1011)(A-N)에 걸친 부하 분산 GPU(112) 리소스를 포함할 수 있다. RDVGM은 또한 부팅시에 적절한 GPU(112) 리소스를 가상 머신(1011)(A-N)으로 할당할 수 있다.
- [0066] RDVGM은 그래픽 데이터의 렌더링, 캡처 및 압축을 조절하는 RCC 엔진(1022)을 통합할 수 있다. RCC는 각각의 가상 머신(1011)으로부터의 출력으로서 그래픽 요청을 수신할 수 있고, 예를 들어 호스트 파티션(1020) 상의 DirectX® 호환 커맨드로 이들 요청을 번역할 수 있다. VMBus는 가상 머신(1011)(A-N) 내에서 구동하는 호스팅된 애플리케이션(1012)(A-N)으로부터 물리적 GPU(112) 리소스로의 그래픽 요청에 대한 고속 통신 백플레인(backplane)을 제공할 수 있다. DirectX® 호환 커맨드에 있어서, 애플리케이션(1012)(A-N)은 DirectX® 9 또는 그 후 모델을 지원해야 하는 반면, GPU(112)는 DirectX® 10 또는 그 후 모델을 지원해야 한다.
- [0067] 앞서 기술된 바와 같이, 렌더링은 vGPU(1016)(A-N)를 통해서 애플리케이션(1012)(A-N)에 의해 행해지는 회전, 플립 및 드로우와 같은 원본 디스플레이 호출을 번역하고, GPU(112)으로 이러한 요청을 이행하며, 따라서 애플리케이션 콘텐츠를 렌더링하는 프로세스를 지칭한다. 렌더링은 표준 DirectX® 문법에 기초할 수 있다. 캡처는 온-스크린 비트맵 또는 프레임 변경과 같은 렌더링된 애플리케이션 콘텐츠를 취하고, 애플리케이션 콘텐츠의 앞선 렌더링에 걸쳐 변화를 지능적으로 캡처하는 것을 지칭한다. 캡처의 두 번째 기능은 캡처 품질 및 인코딩 레벨에 대한 서비스 품질 정책을 할당하는 것이다. 압축 또는 인코딩은 예를 들어 원격 데스크톱 세션 프로토콜을 포함하는 통신 채널 상에서 vGPU(1016)를 통해 원격 클라이언트 컴퓨터로 GPU(112) 리소스를 최적의 방식으로 공정하게 전달하는 프로세스를 지칭한다. 통신 채널의 품질 및 조건과 타겟화된 원격 클라이언트 컴퓨터의 타입은 캡처된 콘텐츠를 최적의 방식으로 전달하기 위해 사용되는 압축/인코딩의 타입을 결정할 수 있다.
- [0068] VM(1011) 내에서 구동하는 애플리케이션(1012)이 드로우, 리사이징 및 회전과 같은 디스플레이 호출을 발행할 때, vGPU(1016)는 모든 렌더링 요청을 브로커할 수 있다. 가상화 경로는 게스트 OS(1014)에 대해 투명할 수 있다. 앞서 설명된 바와 같이, 그래픽 프로세싱 커맨드는 호스트 파티션(1020)에 의해 인터셉트될 수 있다. 인터셉트는 소프트웨어 스택 내의 로우 레벨에서 이루어질 수 있다. 그 다음 그래픽은 GPU(112) 상에서 단일 프레임 버퍼로 렌더링될 수 있으며, 이것은 그래픽 업데이트에 대한 임시 홀딩 스테이션으로서의 역할을 한다. 프레임 버퍼는 최종 사용자의 가상화된 디스플레이를 나타낼 수 있으며, 최종 사용자는 원격 클라이언트 컴퓨터를 이용하여 가상 머신(1011)에 접속된 사용자이다. 풍부한 그래픽 애플리케이션, 3D 플러그-인 및 다른 그래픽 호출과 커맨드가 마치 애플리케이션이 GPU를 포함하는 전용 워크스테이션 상에서 실행되는 것처럼 정확하게 구동할 수 있다.
- [0069] 호스트 파티션(1020)은 렌더링된 콘텐츠를 빠르고 효율적으로 캡처할 수 있다. 콘텐츠 내의 각각의 프레임은 관리가능한 단위로 나누어질 수 있다. 프레임 내의 영역들을 변경하는 것은 RCC 엔진(1022)에 의해 제공되는 최적화 성능을 통해 프로세싱될 수 있다. 이러한 캡처 메커니즘에도 불구하고, 개별적인 프레임들은 디스플레이 변화에 대해 인터셉트된다. 오직 변경된 프레임 내의 영역들만이 인코딩을 위해 캡처된다.
- [0070] 호스트 파티션(1020)은 또한 캡처된 콘텐츠를 압축할 수 있다. 압축 프로세스는 관리 도구를 통해, 소정의 가상 머신에 보다 높거나 낮은 우선순위를 부여함으로써, 또는 캡처된 프레임 내의 변경 영역들의 크기에 의해 동적으로 제어될 수 있다. 또한, RDP 7.1과 같은 원격 데스크톱 세션을 이용하는 실시예에서, 원격 데스크톱 세션은 네트워크 용도 및 공정성(fairness)에 기초하여 프레임-비율 최적화를 제공할 수 있다. RCC 엔진(1022)은 원격 클라이언트 컴퓨터 디코딩 성능을 포함하는 원격 클라이언트 컴퓨터의 상태를 평가하기 위해서 원격 데스크톱 세션 리스너에 도달할 수 있다. 프레임 버퍼에 대한 변경은 변경을 소모하도록 네트워크 조건 및 원격 클라이언트 컴퓨터의 능력을 동적으로 적응시키는 프레임 속도로 원격 클라이언트 컴퓨터로 전송될 수 있다. 인코딩된 출력은 원격 데스크톱 세션 내에서 터널링될 수 있으며, 원격 클라이언트 컴퓨터로 전송될 수 있다.
- [0071] 하나 이상의 컴퓨터 서버 및 하나 이상의 그래픽 서버를 포함하는 원격 서버 컴퓨터를 구현하기 위한 도 7 내지 10에 도시된 모든 아키텍처 변형은 예시적인 구현이다. 본 명세서에 기술된 내용이 임의의 특정한 구현 측면으로 본 발명을 제한하는 것으로 해석되어서는 안된다.
- [0072] 원격 컴퓨팅 환경에서, 원격 클라이언트 컴퓨터는 전형적으로 오직 컴퓨터 서버와 통신하며 그래픽 서버와는 직접 접촉을 갖지 않는다. 컴퓨터 서버와의 통신은 도 3에 도시된 것과 유사한 브로커 및/또는 리더렉터를 이용

하는 것을 포함할 수 있다. 컴퓨터 서버는 전형적으로 원격 클라이언트 컴퓨터와의 접속을 관리한다. 예를 들어, 컴퓨터 서버의 게스트 운영 시스템은 클라이언트를 인증하도록 구성될 수 있다. 인증이 완료되면, 원격 클라이언트 컴퓨터와 컴퓨터 서버 사이의 데이터 전송이 시작될 수 있다. 이렇게, 원격 클라이언트 컴퓨터로부터의 그래픽 요청은 컴퓨터 서버에 의해서 수신될 수 있다. 컴퓨터 서버는 그래픽 요청을 그래픽 호출 및 커맨드로 프로세싱할 수 있으며 렌더링, 캡처 및 압축을 위해 이러한 호출 및 커맨드를 그래픽 서버로 전송할 수 있다. 그래픽 서버가 원격 클라이언트 컴퓨터와 직접 통신 경로를 갖지 않기 때문에, 그래픽 서버로부터의 그래픽 출력은 컴퓨터 서버로 전송될 수 있다. 컴퓨터 서버는 그래픽 출력을 패키징하여 최종 사용자에게 디스플레이하기 위해 원격 클라이언트 컴퓨터로 그래픽 출력을 전송할 수 있다.

[0073] 도 10의 가상 환경에서 전송된 설명을 적용하여, 원격 클라이언트 컴퓨터로부터의 그래픽 요청이 게스트 파티션(1010)의 가상 NIC(도시되지 않음)를 통해 원격 서버 컴퓨터(1000)의 물리적 NIC(114) 상에서 애플리케이션(1012)으로 전송된다. 그 다음 그래픽 요청은 렌더링, 캡처 및 압축을 위해 호스트 파티션(1020)으로 프로세싱 및 라우팅될 수 있다. 렌더링, 캡처 및 압축되면, 출력 데이터는 vGPU(1016)를 이용하여 호스트 파티션(1020)으로부터 컴퓨터 서버(1010)로 다시 라우팅될 수 있다. 게스트 파티션(1010)으로부터 원격 클라이언트 컴퓨터로 출력 데이터를 전송하기 위해서, 게스트 파티션(1010)은 가상 NIC를 이용하여 데이터를 패키징하고 전송할 수 있다. 가상 NIC는 네트워크 상에서 원격 클라이언트 컴퓨터로 데이터를 전송하는 물리적 NIC(114)로 출력 데이터를 리다이렉트할 수 있다. 기술된 바와 같이, 게스트 파티션(1010), 호스트 파티션(1020) 및 원격 서버 컴퓨터(1000)의 언더라이크 리소스 사이에서의 반복적인 데이터 이동은 강도높은 동작을 필요로 할 수 있으며, 상당한 양의 메모리 및 CPU 리소스를 소비할 수 있고, 이는 원격 클라이언트 컴퓨터로의 데이터 전달 지연을 증가시킬 수 있다.

[0074] 도 11은 원격 서버 컴퓨터(1100)의 리소스들 사이에서의 반복적인 데이터 이동을 없애는 다른 아키텍처를 도시한다. 도 11은 컴퓨터 서버(1110) 내에서의 원격 클라이언트 컴퓨터(1130)와의 접속 관리를 유지하고 그래픽 서버(1120)가 원격 클라이언트 컴퓨터(1130)로 직접 그래픽 출력 데이터를 스트리밍할 수 있게 하는 아키텍처를 기술한다. 원격 서버 컴퓨터(1100)의 실시예는 도 7-9의 원격 서버 컴퓨터(700) 및 도 10의 원격 서버 컴퓨터(1000)와 관련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다. 원격 클라이언트 컴퓨터(1130)의 실시예는 도 1의 컴퓨터(100), 도 2의 원격 클라이언트 컴퓨터(210) 및 도 3의 원격 클라이언트 컴퓨터(310)와 관련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다.

[0075] 원격 클라이언트 컴퓨터(1130)는 (도시되지 않은) 네트워크 상에서 컴퓨터 서버(1110)와의 접속(1105)을 시작할 수 있다. 접속(1105)은 TCPI/IP 기반일 수 있으며 RDP 7.1과 같은 원격 데스크톱 세션을 포함할 수 있다. 컴퓨터 서버(1110)는 원격 클라이언트 컴퓨터(1130)와의 접속(1105)을 인증할 수 있다. 인증 방법은 원격 클라이언트 컴퓨터(1130)에 대한 가상 데스크톱 또는 세션이 컴퓨터 서버(1110) 내에서 셋업되기 이전에 인증될 것을 필요로 할 수 있다. 인증 방법은, 예를 들어 RDP 7.1에서 이용가능한 네트워크 레벨 인증일 수 있다. 또한, 컴퓨터 서버(1110)는 원격 클라이언트 컴퓨터(1130)의 IP 어드레스와 같은 어드레스를 획득할 수 있다.

[0076] 인증이 완료되어 원격 클라이언트 컴퓨터(1130)와 컴퓨터 서버(1110) 사이의 접속(1105)이 확립되면, 컴퓨터 서버(1110)는 원격 데스크톱 세션, 가상 머신, 가상 머신 내의 데스크톱 세션, 또는 원격 클라이언트 컴퓨터(1130)에 대한 컴퓨터 서버(1110) 내에서 구동하는 이들의 조합을 개시할 수 있다. 이러한 가상 머신 또는 데스크톱 세션은 도 2 내지 10의 기술을 구현할 수 있다. 또한, 컴퓨터 서버(1110)는 그래픽 서버(1120)를 초기화할 수 있다. 초기화는 예를 들어 만약 그래픽 서버가 비활동 상태에 있을 때 그래픽 서버(1120)를 복구 또는 웨이크-업(waking-up)하고 컴퓨터 서버(1110)와 그래픽 서버(1120) 사이의 접속(1115)을 확립하는 것을 포함할 수 있다. 접속(1115)은 TCP/IP 기반일 수 있거나 또는 가상 환경에서 VMBus를 이용할 수 있다. 컴퓨터 서버(1110)는 또한 그래픽 서버(1120)에 원격 클라이언트 컴퓨터(1130)를 제공할 수 있으며 원격 클라이언트 컴퓨터(1130) 어드레스로부터 발생한 접속 요청에 대한 준비를 그래픽 서버(1120)에 지시할 수 있다. 컴퓨터 서버는 또한 IP 어드레스와 같은 그래픽 서버(1120)의 어드레스를 획득할 수 있으며, 원격 클라이언트 컴퓨터(1130)에 그래픽 서버(1120) 어드레스를 제공할 수 있다.

[0077] 그래픽 서버(1120)가 초기화되고 원격 클라이언트 컴퓨터(1130)가 그래픽 서버(1120) 어드레스를 획득하면, 클라이언트 컴퓨터(1130)는 (도시되지 않은) 네트워크 상에서 그래픽 서버(1120)와의 접속(1125)을 시작할 수 있다. 접속(1125)은 TCP/IP 기반일 수 있다. 접속(1125)은 또한 RDP 7.1과 같은 원격 데스크톱 세션을 포함할 수 있다. 다른 실시예에서, 만약 접속(1105) 및 접속(1125)이 원격 데스크톱 세션을 포함한다면, 원격 데스크톱 세션은 접속(1105) 및 접속(1125)에 걸쳐 분리될 수 있거나 동일할 수 있다.



- [0078] 접속(1105, 1115, 1125)이 확립되면, 원격 클라이언트 컴퓨터(1130)는 키보드 또는 마우스 입력과 같은 최종 사용자 그래픽 입력을 컴퓨트 서버(1110)로 전송할 수 있다. 컴퓨터 서버는 도 2, 6-10의 기술을 이용하여 그래픽 입력을 프로세싱할 수 있다. 일 실시예에서, 컴퓨트 서버(1110)는 최종 사용자가 키보드를 치거나 아이콘 상에서 마우스를 클릭하는 것을 회전, 플립, 리사이징 및 드로우와 같은 디스플레이 호출 및 애플리케이션의 인스턴스를 발효시키는 디스플레이 커맨드로 번역할 수 있다. 컴퓨트 서버(1110)는 렌더링, 캡처 및 압축을 위해 디스플레이 커맨드 및 호출 데이터를 그래픽 서버(1120)로 전송할 수 있다. 그래픽 서버(1120)는 디스플레이 커맨드 및 호출을 렌더링, 캡처 및 압축할 수 있으며, 출력 그래픽 데이터로서 출력을 인코딩할 수 있다. 출력 그래픽 데이터를 컴퓨트 서버(1110)에 전송하는 대신, 그래픽 서버(1120)가 접속(1125) 상에서 출력 그래픽 데이터를 원격 클라이언트 컴퓨터(1130)로 직접 전송할 수 있다. 원격 클라이언트 컴퓨터(1130)는 최종 사용자에게 디스플레이하기 위해서 출력 그래픽 데이터를 디코딩할 수 있다.
- [0079] 실시예에서, 도 10의 원격 서버 컴퓨터(1000)와 같이 원격 서버 컴퓨터(1100)가 가상화되면, 컴퓨트 서버(1110)는 게스트 파티션 상에서 발효될 수 있으며 그래픽 서버(1120)는 호스트 파티션 상에서 발효될 수 있다. 각각의 파티션은 IP 어드레스와 같은 어드레스를 갖도록 구성될 수 있다. 게스트 파티션 또는 컴퓨트 서버(1110)의 어드레스는 가상 NIC와 연관될 수 있다. 호스트 파티션 또는 그래픽 서버(1120)의 어드레스는 가상 NIC 또는 원격 서버 컴퓨터(1100)의 언더라이징 하드웨어의 물리적 NIC(114)와 연관될 수 있다. 컴퓨트 서버(1110)는 원격 클라이언트 컴퓨터(1130)에 자신에서 실행중인 세션 또는 가상 머신을 제시할 수 있다. 컴퓨트 서버(1110)와 그래픽 서버(1120) 사이의 접속(1115)은 VMBus와 같은 인트라-파티션 통신 채널을 포함할 수 있다.
- [0080] 원격 클라이언트 컴퓨터(1130)에서의 최종 사용자 입력과 같은 그래픽 데이터는 접속(1105) 상에서 원격 클라이언트 컴퓨터(1130)로부터 컴퓨트 서버(1110) 가상 NIC와 연관된 컴퓨트 서버(1110) 어드레스로 전송될 수 있다.
- [0081] 컴퓨트 서버(1110)는 앞서 기술된 바와 같이 수신된 그래픽 데이터를 디스플레이 커맨드 및 호출 데이터로 프로세싱할 수 있다. 컴퓨트 서버(1110)는 렌더링, 캡처 및 압축을 위해 프로세싱된 그래픽 데이터를 그래픽 서버(1120)로 전송할 수 있다. 실시예에서, 프로세싱된 데이터를 전송하는 것은 컴퓨트 서버(1110)에 할당된 메모리 공간으로부터 그래픽 서버(1120)에 할당된 메모리 공간으로 프로세싱된 데이터를 전송하는 것을 포함할 수 있다. 다른 실시예에서, 원격 서버 컴퓨터(1100) 내의 파티션들은 메모리 공간을 공유할 수 있다. 이러한 실시예에서, 컴퓨트 서버(1110)로부터 그래픽 서버(1120)로의 데이터 전송은, 두 개의 메모리 공간들 사이에서 프로세싱된 데이터를 카피하지 않고 프로세싱된 데이터를 포함하는 메모리 공간의 어드레스를 라우팅하는 것을 포함할 수 있다. 예를 들어, 컴퓨트 서버(1110)와 그래픽 서버(1120)는 프로세싱된 데이터를 라우팅하기 위해서 VMBus 내의 공유된 메모리 공간의 장점을 취할 수 있다.
- [0082] 그래픽 서버(1120)는 전송된 바와 같이 프로세싱된 데이터를 렌더링, 캡처 및 압축할 수 있다. 그래픽 서버(1120)는 렌더링, 캡처 및 압축된 데이터를 출력 데이터로 인코딩할 수 있으며, 원격 클라이언트 컴퓨터(1130)와 연관된 어드레스로 그래픽 서버 NIC를 이용하여 출력 데이터를 전송할 수 있다. 원격 클라이언트 컴퓨터(1130)는 수신된 출력 데이터를 최종 사용자에게 디스플레이하기 위해 디코딩할 수 있다.
- [0083] 도 12는 복수의 원격 클라이언트 컴퓨터(1130)(A-N), 복수의 컴퓨트 서버(1110)(A-N) 및 복수의 그래픽 서버(1120)(A-N)를 포함하는 환경에서 도 11의 아키텍처의 실시예를 도시한다. 원격 클라이언트 컴퓨터, 컴퓨트 서버 및 그래픽 서버의 수가 동일할 필요가 없음을 인지하는 것이 중요하다. 다시 말하면, 원격 클라이언트 컴퓨터(1130)(A-N)와 컴퓨트 서버(1110)(A-N), 컴퓨트 서버(1110)(A-N)와 그래픽 서버(1120)(A-N), 그래픽 서버(1120)(A-N)와 원격 클라이언트 컴퓨터(1130)(A-N) 사이의 관계는 일대일, 일대다, 또는 이들의 조합일 수 있다.
- [0084] 원격 서버 컴퓨터(1200)의 실시예는 도 2의 원격 서버 컴퓨터(220), 도 3의 원격 서버 컴퓨터(320), 도 7-10의 원격 서버 컴퓨터(700), 도 10의 원격 서버 컴퓨터(1000) 및 도 11의 원격 서버 컴퓨터(1100)와 관련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다. 일 실시예에서, 원격 서버 컴퓨터(1200)는 복수의 물리적 컴퓨팅 시스템을 포함할 수 있다. 리더랙터 및 브로커(1240)는 도 3의 리더랙터(330) 및 브로커(340)와 관련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다. 그래픽 서버 관리자(1250)는 도 7의 그래픽 서버 관리자(740) 및 도 8-9의 그래픽 서버 관리자(850)와 관련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다. 또한, 리더랙터 및 브로커(1240)는 컴퓨트 서버(1110)(A-N)와 통합될 수도 있다. 유사하게, 그래픽 서버 관리자(1250)는 그래픽 서버(1120)(A-N)와 통합될 수 있다. 다른 실시예에서, 원격 서버 컴퓨터는 가상화될 수 있으며 도 4-5의 컴퓨터 시스템(400), 도 10의 원격 서버 컴퓨터(1000) 및 도 11의 원격 서버 컴퓨터(1100)와 관

련하여 기술된 일부 또는 모든 구성요소들을 실행할 수 있다. 이러한 실시예에서, 컴퓨터 서버(1110(A-N)는 하나 이상의 게스트 파티션 상에서 발효될 수 있고, 그래픽 서버(1120)(A-N)는 하나 이상의 호스트 파티션 상에서 발효될 수 있다. 리더랙터 및 브로커(1240) 및 그래픽 서버 관리자(1250)는 또한 전송된 기술들을 이용하여 하나 이상의 게스트 파티션 및/또는 하나 이상의 호스트 파티션에서 가상화될 수 있다. 다른 실시예에서, 원격 서버 컴퓨터(1200)는 가상 및 물리적 머신의 조합일 수 있다. 예를 들어, 컴퓨터 서버(1110)(A-N) 및 그래픽 서버(1120(A-N)가 가상화될 수 있는 반면, 리더랙터 및 브로커(1240) 및 그래픽 서버 관리자(1250)는 물리적 컴퓨팅 디바이스일 수 있다.

[0085] 리더랙터 및 브로커(1240)는 원격 클라이언트 컴퓨터(1130)와 컴퓨터 서버(1110) 사이의 제 1 접속을 확립하도록 사용될 수 있다. 예를 들어, 리더랙터 및 브로커(1240)는 복수의 컴퓨터 서버(1110)(A-N) 중의 컴퓨터 서버(1110) 및/또는 그에서의 세션을 원격 클라이언트 컴퓨터(1130)로 할당할 수 있다. 이렇게, 리더랙터 및 브로커(1240)는 컴퓨터 서버(1110)(A-N)의 이용가능성과 관련된 부하 분산 기술을 제공할 수 있다. 할당이 완료되면, 할당된 컴퓨터 서버(1110)는 원격 클라이언트 컴퓨터(1130)를 인증할 수 있다. 원격 클라이언트 컴퓨터(1130) 및 할당된 컴퓨터 서버(1110)는 전송된 바와 같이 제 1 접속을 확립할 수 있다. 예를 들어, 제 1 접속은 도 11의 접속(1105)을 포함할 수 있다. 할당된 컴퓨터 서버(1110)는 또한 원격 클라이언트 컴퓨터(1130)의 어드레스를 획득할 수 있다.

[0086] 그래픽 서버 관리자(1250)는 복수의 그래픽 서버(1120)(A-N) 중의 그래픽 서버(1120)와 할당된 컴퓨터 서버(1110) 사이의 제 2 접속을 확립하도록 사용될 수 있다. 그래픽 서버 관리자(1250)는 부하 분산 기술을 실행할 수 있으며 그래픽 서버(1120)(A-N)의 이용가능성을 결정할 수 있다. 따라서, 그래픽 서버 관리자(1250)는 그래픽 서버(1120)를 할당된 컴퓨터 서버(1110)와의 접속에 할당할 수 있다. 할당이 완료되면, 제 2 접속이 확립될 수 있다. 예를 들어, 제 2 접속은 도 11의 접속(1115)을 포함할 수 있다.

[0087] 할당된 컴퓨터 서버(1110)는 원격 클라이언트 컴퓨터(1130)의 어드레스를 그래픽 서버(1120)에 제공할 수 있고, 할당된 그래픽 서버(1120) 어드레스를 획득하여 이를 원격 클라이언트 컴퓨터(1130)에 제공할 수 있다. 원격 클라이언트 컴퓨터(1130) 및 할당된 그래픽 서버(1120)는 원격 클라이언트 컴퓨터(1130) 및 할당된 그래픽 서버(1120)의 획득된 어드레스를 이용하여 제 3 접속을 획득할 수 있다. 제 3 접속은 도 11의 접속(1125)을 포함할 수 있다.

[0088] 또한, 부하 분산 기술은 도 12의 아키텍처 상에서 실행될 수 있다. 예를 들어, 그래픽 서버(1120)(A-N)의 부하 분산의 경우에, 할당된 그래픽 서버(1120)가 더 이상 원격 클라이언트 컴퓨터(1130)의 역할을 적절하게 할 수 없을 때, 컴퓨터 서버(1110)는 이용가능한 그래픽 서버(1120)와의 새로운 접속을 확립할 수 있으며, 원격 클라이언트 컴퓨터(1130)는 이용가능한 그래픽 서버(1120)와의 상응하는 새로운 접속을 확립하도록 지시한다. 원격 클라이언트 컴퓨터(1130)는 이용가능한 그래픽 서버(1120)로의 매끄럽게 트랜지션할 수 있다. 또한, 컴퓨터 서버(1110)(A-N)의 부하 분산의 경우에서, 할당된 컴퓨터 서버(1110)가 더 이상 원격 클라이언트 컴퓨터(1130)로서의 역할을 할 수 없을 때, 원격 클라이언트 컴퓨터(1130)와 이용가능한 컴퓨터 서버(1110) 사이의 새로운 접속 및 이용가능한 컴퓨터 서버(1110)와 이미 할당된 그래픽 서버(1120) 사이의 새로운 접속이 확립될 수 있는 한편, 원격 클라이언트 컴퓨터(1130)와 이미 할당된 그래픽 서버(1120) 사이의 접속이 동일하게 남아있을 수 있다. 가상 머신 라이브 이동(virtual machine live migration)과 같은 기술이 하나의 컴퓨터 서버(1110)로부터 다른 컴퓨터 서버(1110)로 작업부하를 매끄럽게 전달할 수 있는 반면, 그래픽 서버(1120) 작업부하는 동일하게 남아있을 수 있다.

[0089] 실시예에서, 하나 이상의 원격 클라이언트 컴퓨터(1130)가 다양한 그래픽 서버(1120)로부터의 그래픽 출력 데이터 또는 디스플레이 스트림을 수용할 수 있게 하도록 사용될 수 있다. 예를 들어, 원격 클라이언트 컴퓨터(1130A)는 다른 원격 클라이언트 컴퓨터(1130)(B-N)와 연관된 최종 사용자 디스플레이를 보기 위해서 다른 원격 클라이언트 컴퓨터(1130)(B-N)로부터의 허가를 획득할 수 있다. 예를 들어, 허가는 각각의 다른 원격 클라이언트 컴퓨터(1130)(B-N)로 할당된 다양한 그래픽 서버(1120)의 어드레스를 포함할 수 있다. 또한, 허가는 다른 원격 클라이언트 컴퓨터(1130)(B-N)의 어드레스, 각각의 다른 원격 클라이언트 컴퓨터(1130)(B-N)에 할당된 컴퓨터 서버(1110)의 어드레스, 상응하는 세션 ID, 사용자 이름, 컴퓨터 서버(1110) 및 그래픽 서버(1120)의 이름, 각각의 컴퓨터 서버(1110) 내의 활동중인 세션의 수 등을 포함할 수 있다. 원격 클라이언트 컴퓨터(1130A)는 각각의 다른 원격 클라이언트 컴퓨터(1130)(B-N)에 할당된 그래픽 서버(1120)와의 접속을 확립하기 위한 허가를 이용할 수 있다. 접속은 도 11의 접속(1125)을 포함할 수 있다. 접속이 확립되면, 다양한 그래픽 서버(1120)가 다른 원격 클라이언트 컴퓨터(1130)(B-N)의 그래픽 데이터 출력을 원격 클라이언트 컴퓨터(1130A)로 전송할 수 있다. 원격 클라이언트 컴퓨터(1130A)는 다양한 그래픽 서버(1120)로부터 수신된 그래픽 출력

데이터를 디코딩할 수 있으며, 디코딩된 데이터를 최종 사용자에게 디스플레이할 수 있다. 예를 들어, 원격 클라이언트 컴퓨터(1130A)는 타일드(tiled) 방식으로 디코딩된 데이터를 디스플레이할 수 있거나 또는 최종 사용자가 다른 원격 클라이언트 컴퓨터(1130)(B-N)와 연관된 각각의 스크린을 통해 플립하게 할 수 있다. 또한, 원격 클라이언트 컴퓨터(1130A)는 사용자 이름, 상응하는 컴퓨터 서버(1110) 및 그래픽 서버(1120)의 이름 등과 같이 허가에서 이용가능한 다른 정보를 디스플레이할 수 있다. 다른 실시예에서, 원격 클라이언트 컴퓨터(1130)(B-N)에 상응하는 다양한 그래픽 출력 데이터를 전송하는 그래픽 서버(1120)는, 상응하는 원격 클라이언트 컴퓨터(1130)(B-N)에 대한 다른 정보 및 다양한 그래픽 출력 데이터를 다른 원격 클라이언트 컴퓨터(1130A)에 전송하도록 구성될 수 있다. 이렇게, 원격 클라이언트 컴퓨터(1130A)는 원격 클라이언트 컴퓨터(1130)(B-N)의 스크린 및 그에 대한 정보를 디스플레이할 수 있다.

[0090] 도 13은 동작(1300, 1310, 1320, 1330, 1340, 1350, 1360)을 포함하는 클라이언트로의 전송을 위한 그래픽 데이터 프로세싱을 위한 예시적인 동작 절차를 도시한다. 동작(1300)에서 동작 절차를 시작하며, 동작(1310)은 원격 클라이언트와 컴퓨터 서버 사이의 제 1 접속을 확립한다. 제 1 접속은 도 11의 접속(1105)을 포함할 수 있다. 동작(1320)은 컴퓨터 서버와 그래픽 서버 사이의 제 2 접속을 확립한다. 제 2 접속은 도 11의 접속(1115)을 포함할 수 있다. 동작(1330)은 원격 클라이언트와 그래픽 서버 사이의 제 3 접속을 확립한다. 도 3 접속은 도 11의 접속(1125)을 포함할 수 있다. 동작(1340)은 제 1 접속 상에서 원격 클라이언트로부터 컴퓨터 서버에 의해 그래픽 데이터를 수신한다. 수신된 그래픽 데이터는, 예를 들어 원격 클라이언트와 연관된 마우스 클릭 또는 키보드 치기와 같은 사용자 그래픽 입력을 포함할 수 있다. 동작(1350)은 수신된 그래픽을 프로세싱하여 제 2 접속 상에서 그래픽 서버로 프로세싱된 그래픽 데이터를 전송한다. 수신된 그래픽 데이터의 프로세싱은 수신된 그래픽 데이터를 디스플레이 커맨드 및 호출로 번역하는 컴퓨터 서버를 포함할 수 있다. 동작(1360)은 프로세싱된 그래픽 데이터를 렌더링, 캡처, 압축하고, 제 3 접속 상에서 프로세싱된 그래픽 데이터를 원격 클라이언트로 전송한다.

[0091] 도 14는 전송된 바와 같이 클라이언트 컴퓨터로의 전송을 위해 그래픽 데이터를 프로세싱하는 예시적인 시스템을 도시한다. 시스템(1400)은 프로세서(1410) 및 메모리(142)를 포함한다. 실시예에서, 프로세서(1410)는 도 1의 논리적 프로세싱 장치(102)로서 구현될 수 있는 반면, 메모리(1420)는 도 1의 시스템 메모리(22)의 구성요소들의 일부 또는 전부를 갖는 것으로서 구현될 수 있다. 메모리(1420)는 시스템으로 하여금 원격 클라이언트로의 전송을 위해 그래픽 데이터를 프로세싱하게 하도록 구성된 컴퓨터 명령을 더 포함한다. 블록(1422)은 원격 클라이언트와 컴퓨터 서버 사이의 제 1 접속을 확립하는 것을 도시한다. 블록(1424)은 컴퓨터 서버와 그래픽 서버 사이의 제 2 접속을 확립하는 것을 도시한다. 블록(1426)은 원격 클라이언트와 그래픽 서버 사이의 제 3 접속을 확립하는 것을 도시한다. 블록(1428)은 제 1 접속 상에서 원격 클라이언트로부터 컴퓨터 서버에 의해 그래픽 데이터를 수신하는 것을 도시한다. 블록(1430)은 수신된 그래픽 데이터를 프로세싱하고 프로세싱된 그래픽 데이터를 제 2 접속 상에서 그래픽 서버로 전송하는 것을 도시한다. 블록(1432)은 프로세싱된 그래픽 데이터를 렌더링, 캡처, 압축하고 이것을 제 3 접속 상에서 원격 클라이언트로 전송하는 것을 도시한다.

[0092] 임의의 전송된 측면들이 방법, 시스템, 컴퓨터 판독가능한 매체, 또는 임의의 타입의 제품으로 구현될 수 있다.

[0093] 전송된 상세한 설명은 예시 및/또는 동작 다이어그램을 통해 시스템 및/또는 프로세스의 다양한 실시예를 개시한다. 이러한 블록도에 있어서, 및/또는 예시가 하나 이상의 기능 및/또는 동작을 포함하는 한에 있어서, 넓은 범위의 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 가상 조합에 의해 개별적으로 및/또는 집합적으로 이러한 블록도 내의 각각의 기능 및/또는 동작, 또는 예시가 구현될 수 있음이 당업자에 의해 이해될 것이다.

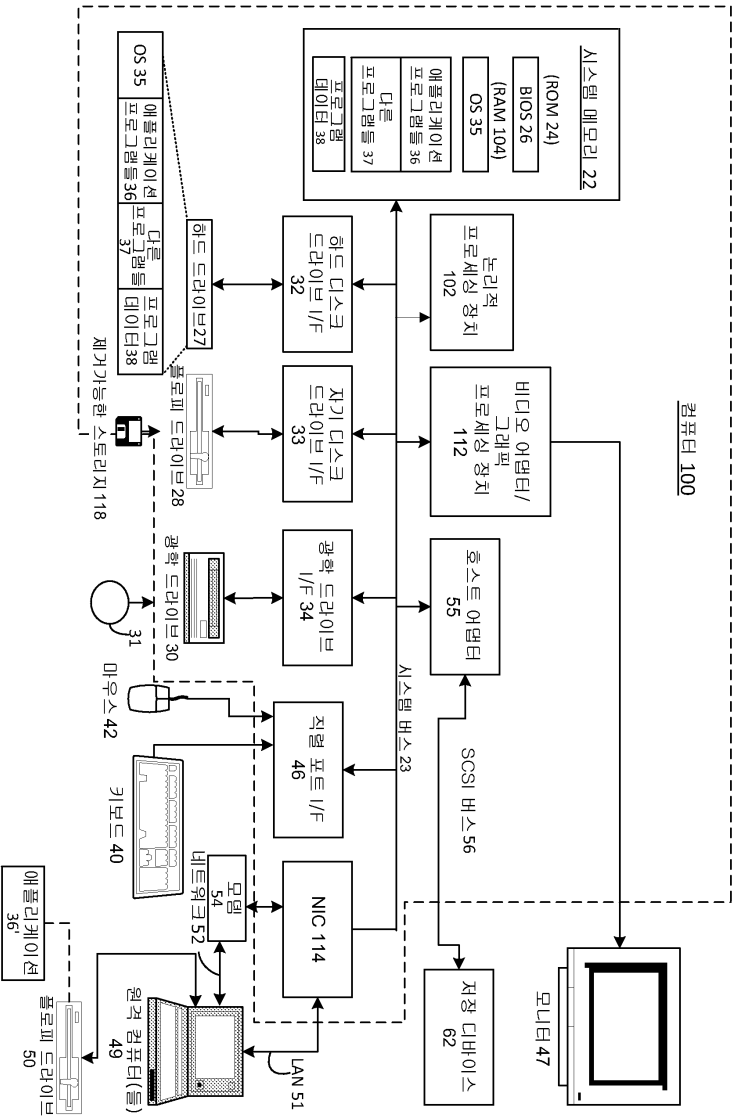
[0094] 본 명세서에 설명된 다양한 기술들이 하드웨어 또는 소프트웨어, 또는 적절하다면 둘의 조합과 관련하여 구현될 수 있다는 것을 이해해야만 한다. 따라서, 본 발명의 방법 및 장치, 소정의 측면들 또는 그 일부가 플로피 디스켓, CD-ROM, 하드 드라이브, 또는 임의의 다른 머신 판독가능한 저장 매체와 같은 실제적인 매체에서 구현되는 프로그램 코드(즉, 명령어)의 형태를 취할 수 있으며, 프로그램 코드가 컴퓨터와 같은 머신에 로딩되어 머신에 의해 실행되면, 머신은 본 발명을 실시하기 위한 장치가 된다. 프로그램가능한 컴퓨터 상에서의 프로그램 코드 실행의 경우에, 컴퓨팅 디바이스는 일반적으로 프로세서, (휘발성 및 비휘발성 메모리 및/또는 저장 요소를 포함하는) 프로세서에 의해 판독가능한 저장 매체, 적어도 하나의 입력 디바이스 및 적어도 하나의 출력 디바이스를 포함한다. 예를 들어 애플리케이션 프로그래밍 인터페이스(API), 재이용가능한 제어 등의 이용을 통해, 프로세스를 구현 또는 활용할 수 있는 하나 이상의 프로그램이 본 발명과 관련하여 기술되었다. 이러한 프로그램은 컴퓨터 시스템과 통신하기 위한 하이 레벨 절차적 또는 객체 지향적 프로그래밍 언어에서 바람직하게 구현된다. 그러나, 프로그램(들)은 원한다면 어셈블리 또는 머신 언어로 구현될 수도 있다. 임의의 경우에서,

언어는 컴파일 또는 해석된 언어일 수 있으며, 하드웨어 구현과 결합될 수 있다.

본 발명이 바람직한 실시예를 참조로 하여 특정하게 도시되고 기술되었지만, 당업자는 형태 및 세부사항에서의 다양한 변화가 아래의 특허청구범위에서 개시된 것과 같은 본 발명의 범주로부터 벗어나지 않고 이루어질 수 있다. 또한, 본 발명의 요소들이 단일 명사로 기술되거나 청구될 수 있지만, 명백하게 단일 명사인 것으로 제한되지 않는 한 복수형이 고려된다.

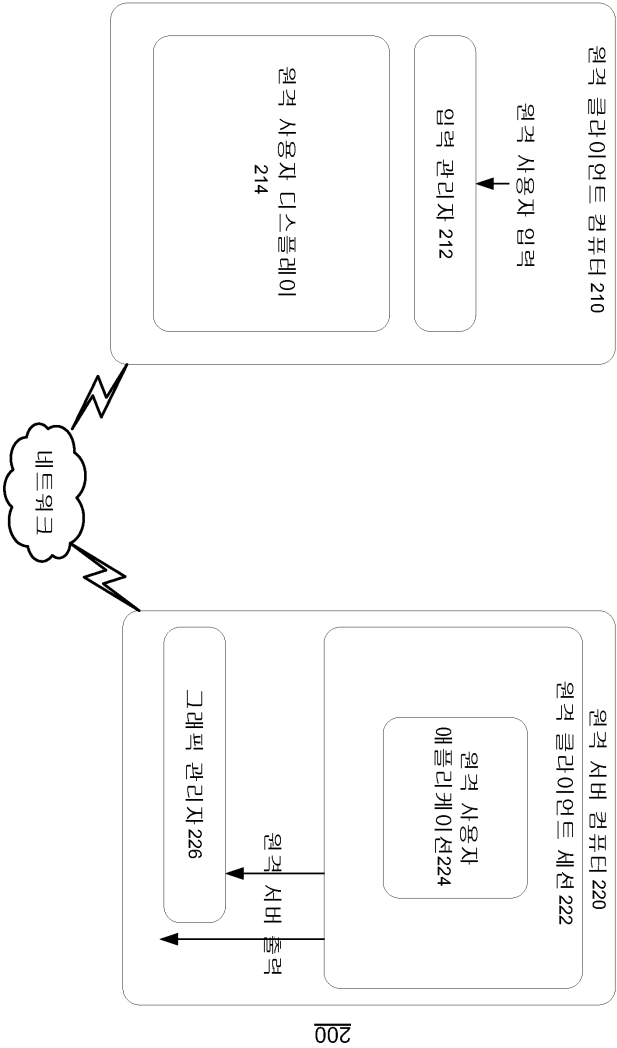
도면

도면1

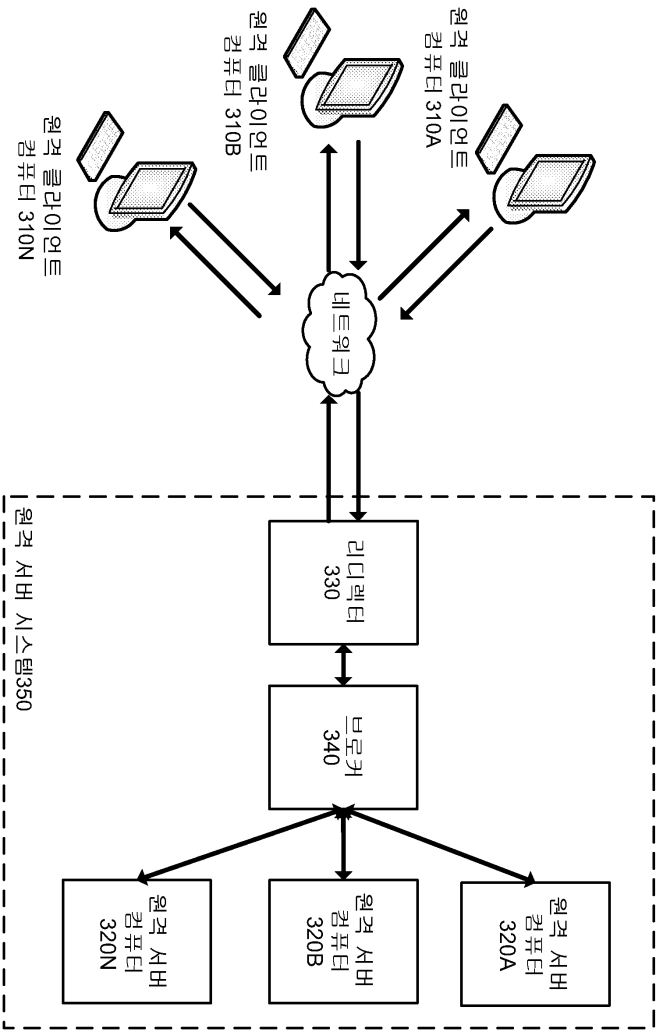




도면2

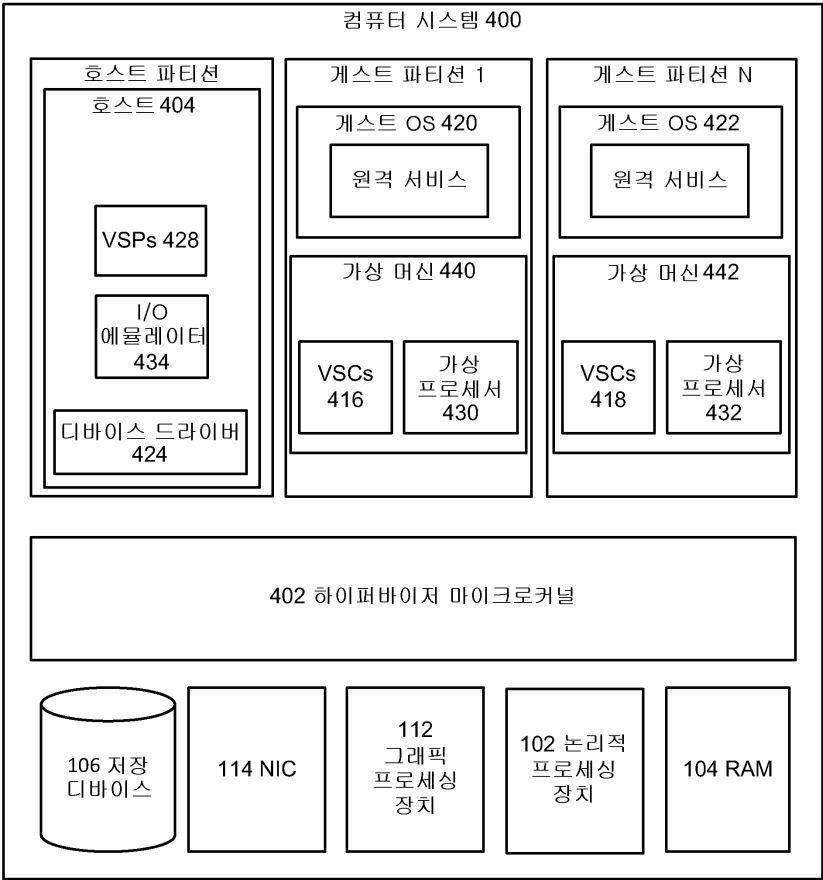


도면3

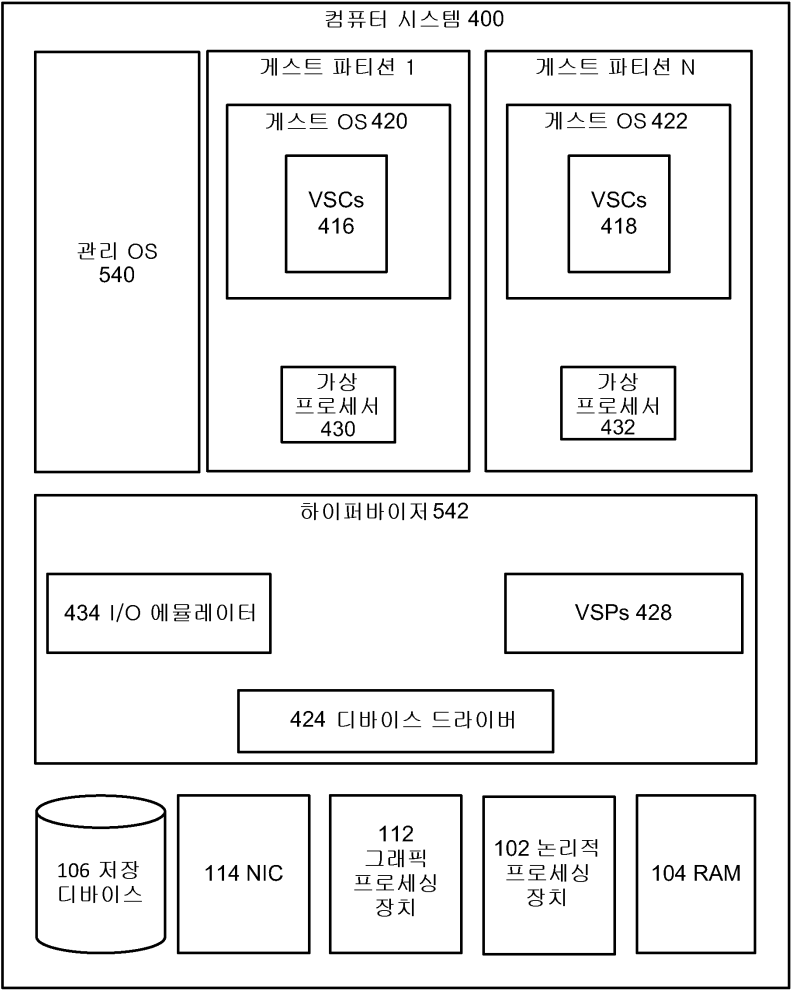


300

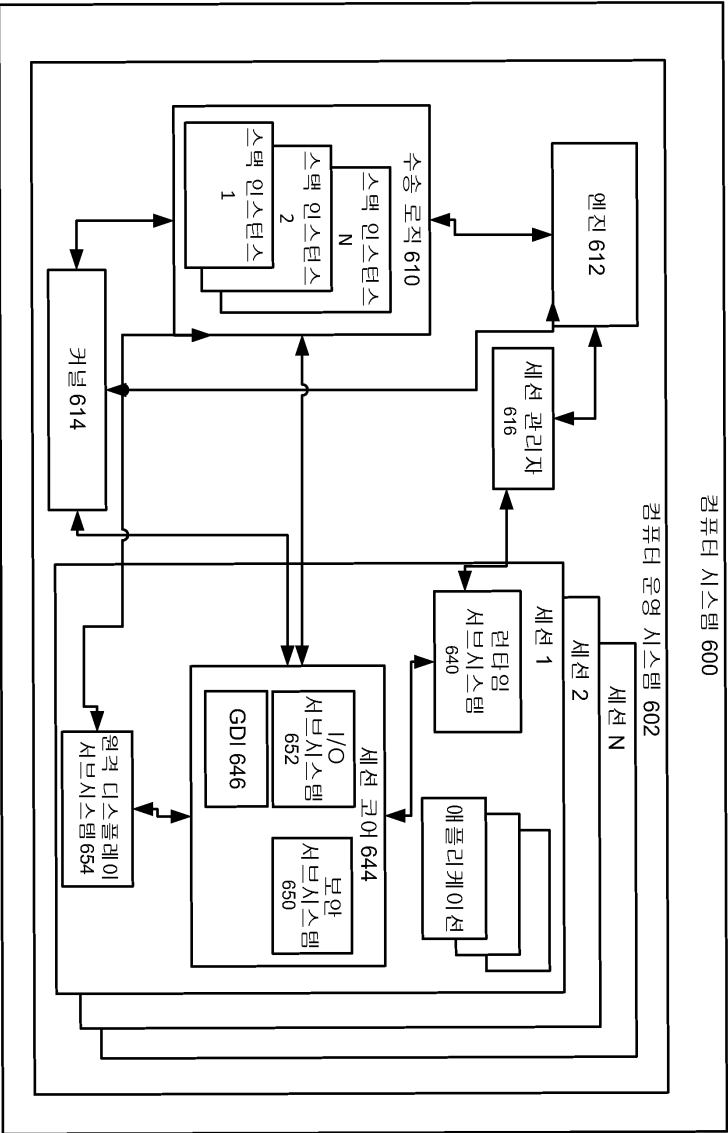
도면4



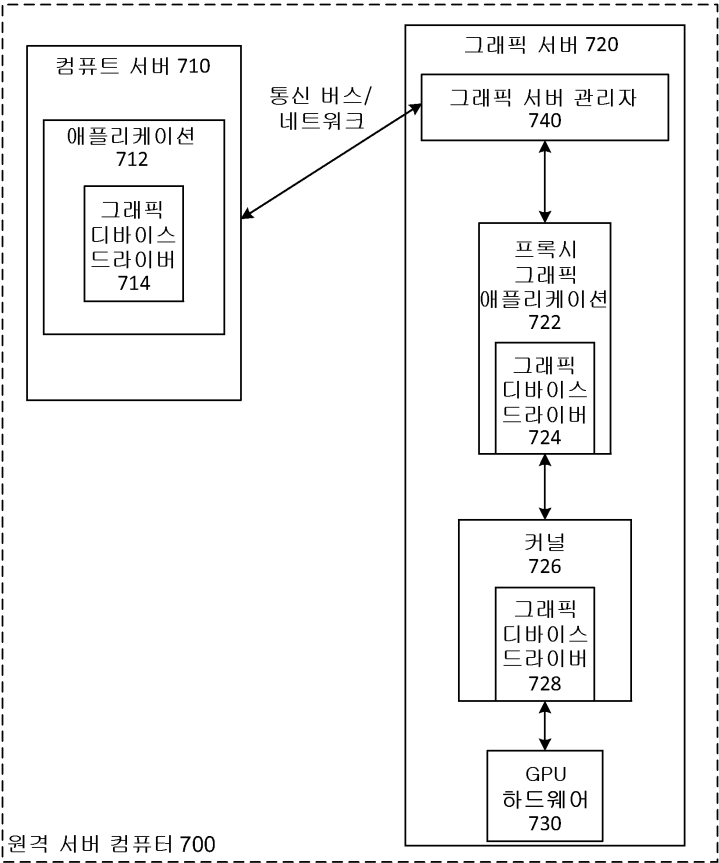
도면5



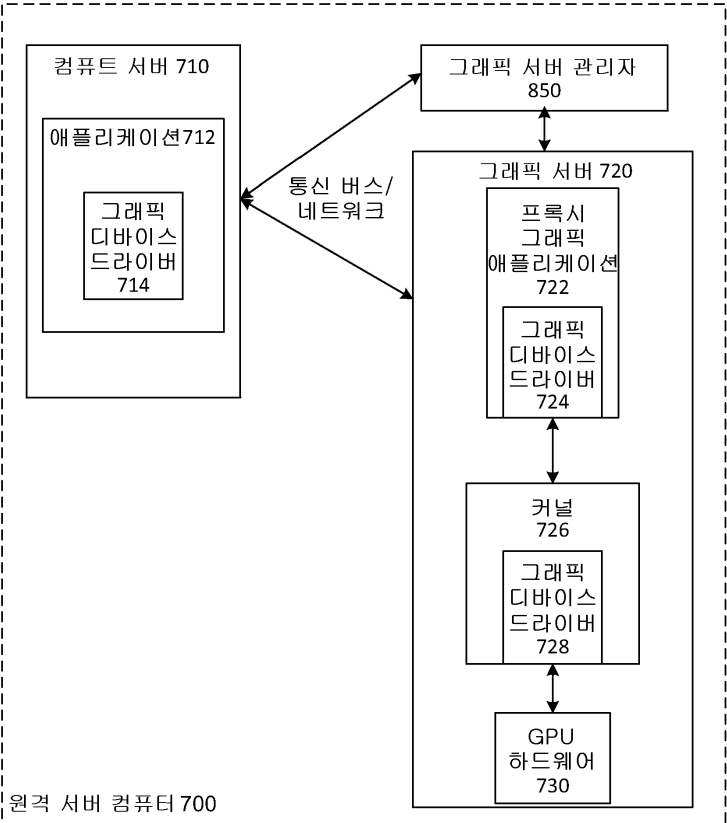
도면6



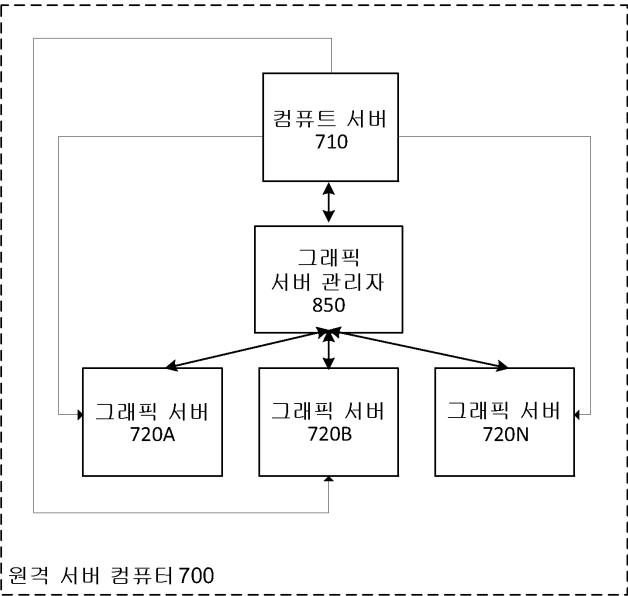
도면7



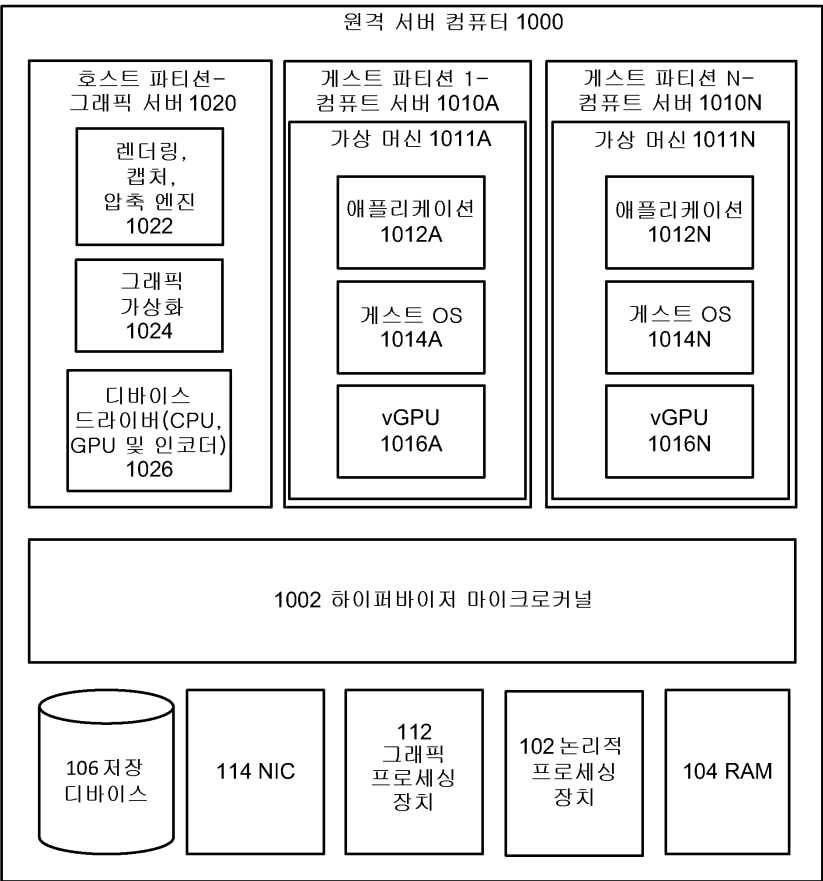
도면8



도면9

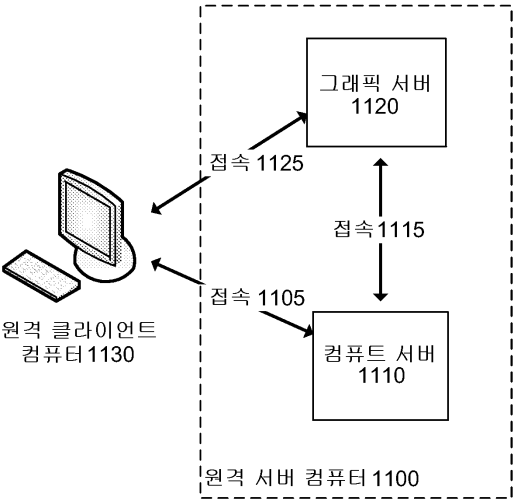


도면10

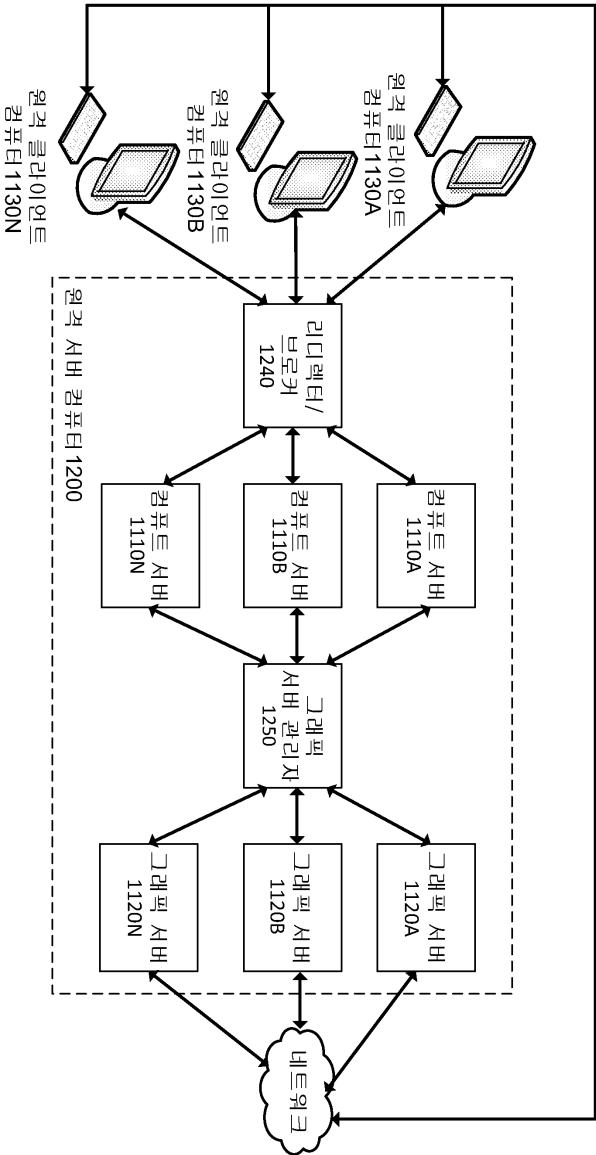




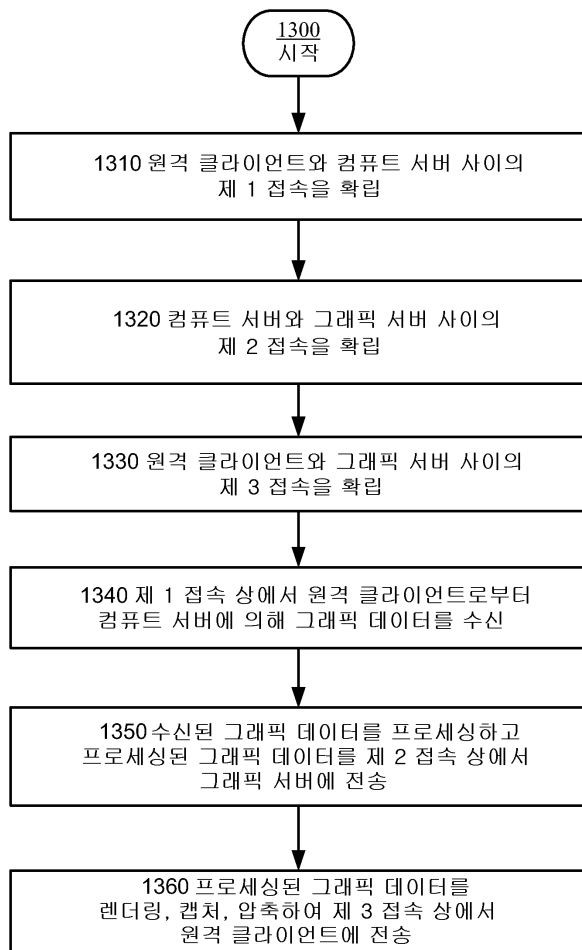
도면11



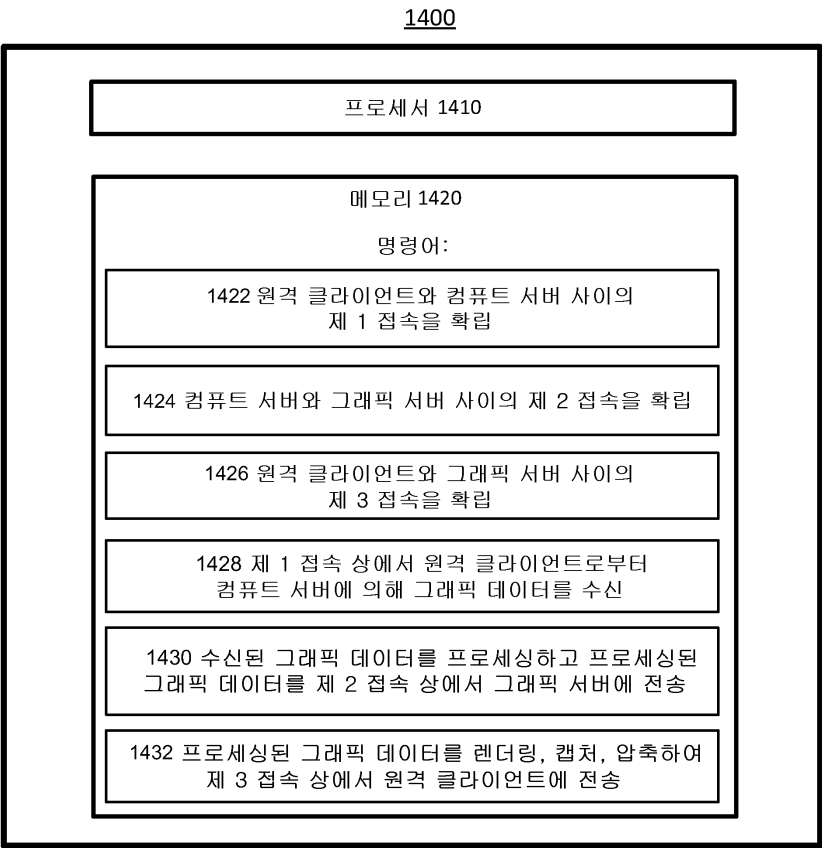
도면12



도면13



도면14



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 제6항

【변경전】

상기 원격 컴퓨팅 디바이스

【변경후】

상기 원격 클라이언트 디바이스