



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2024년11월29일  
(11) 등록번호 10-2736498  
(24) 등록일자 2024년11월26일

- (51) 국제특허분류(Int. Cl.)  
A63F 13/352 (2014.01) A63F 13/358 (2014.01)  
A63F 13/795 (2014.01) A63F 13/847 (2014.01)  
G06F 9/50 (2018.01)
- (52) CPC특허분류  
A63F 13/352 (2015.01)  
A63F 13/358 (2015.01)
- (21) 출원번호 10-2023-7044078(분할)
- (22) 출원일자(국제) 2019년11월18일  
심사청구일자 2024년01월19일
- (85) 번역문제출일자 2023년12월20일
- (65) 공개번호 10-2024-0005136
- (43) 공개일자 2024년01월11일
- (62) 원출원 특허 10-2021-7022976  
원출원일자(국제) 2019년11월18일  
심사청구일자 2021년07월20일
- (86) 국제출원번호 PCT/US2019/062069
- (87) 국제공개번호 WO 2020/131274  
국제공개일자 2020년06월25일
- (30) 우선권주장  
16/228,401 2018년12월20일 미국(US)
- (56) 선행기술조사문헌  
JP2017037446 A

- (73) 특허권자  
소니 인터랙티브 엔터테인먼트 엘엘씨  
미국 캘리포니아 (우편번호 94404) 산 마테오 브릿지포인트 파크웨이 2207
- (72) 발명자  
용, 켈빈  
미국 캘리포니아 94404 산 마테오 브릿지포인트 파크웨이 2207
- (74) 대리인  
윤앤리특허법인(유한)

전체 청구항 수 : 총 18 항

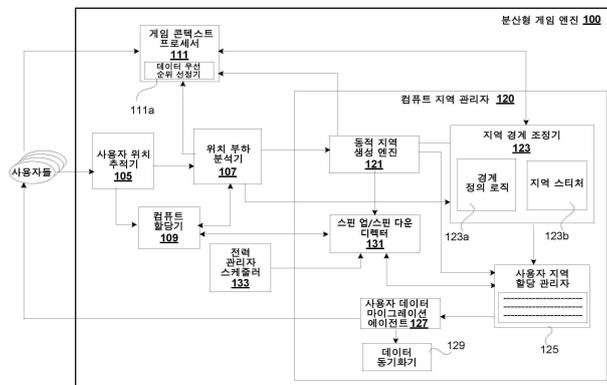
심사관 : 김대홍

(54) 발명의 명칭 대규모 멀티 플레이어 컴퓨터

(57) 요약

온라인 게임의 처리 리소스들을 관리하기 위한 방법들 및 시스템들은 한 명 이상의 사용자들을 위해 온라인 게임을 실행하는 것을 포함한다. 각 사용자에게는 온라인 게임의 게임 세계의 가상 위치들에 대한 액세스가 제공된다. 컴퓨터는 게임 세계의 한 명 이상의 사용자들의 게임 상호 작용을 처리하기 위해 할당된다. 가상 위치 (뒷면에 계속)

대표도



내의 지역에 액세스하는 사용자들의 수가 결정된다. 사용자들의 수가 지역의 사용자들에 할당된 컴퓨터에 대한 미리 정의된 한도를 초과할 때, 지역은 복수의 서브 지역들로 분할되며 추가 컴퓨터가 지역에 할당되어 컴퓨터와 추가 컴퓨터가 게임 세계에 정의된 서브 지역들 각각의 사용자들에 대한 게임 상호 작용을 처리할 수 있도록 한다.

(52) CPC특허분류

*A63F 13/795* (2015.01)

*A63F 13/847* (2015.01)

*G06F 9/5077* (2013.01)

*A63F 2300/535* (2013.01)

*A63F 2300/5533* (2013.01)

## 명세서

### 청구범위

#### 청구항 1

방법으로서, 상기 방법은:

복수의 사용자들-여기서 상기 복수의 사용자들 중 각각의 사용자에게는 온라인 게임에 의해 제공되는 게임 세계의 가상 위치에 대한 액세스가 제공됨-을 위한 상기 온라인 게임을 실행하는 단계;

상기 복수의 사용자들 중 하나 이상의 사용자들이 존재하는 상기 게임 세계의 하나 이상의 가상 위치들을 식별하는 단계;

상기 하나 이상의 사용자들에 의해 상기 게임 세계에 제공되는 게임 상호 작용을 처리하기 위해, 상기 게임 세계 내에서 식별되는 각각의 가상 위치에 하나 이상의 컴퓨터 리소스들-여기서 각각의 가상 위치에 할당된 컴퓨터 리소스들의 수는 상기 게임 상호 작용을 처리하는데 필요한 처리 부하에 기초함-을 할당하는 단계;

상기 게임 세계 내에서 식별된 상기 각각의 가상 위치에 할당된 상기 하나 이상의 컴퓨터 리소스들 각각에 대한 상기 처리 부하를 모니터링하는 단계; 및

상기 각각의 가상 위치에서 상기 처리 부하를 핸들링하기 위해, 상기 각각의 가상 위치에 할당된 컴퓨터 리소스들의 수를 동적으로 조정하는 단계

를 포함하고,

상기 온라인 게임의 게임 상태를 이용하여, 인공 지능(AI) 모델이 생성되고 학습되며, 상기 AI 모델에 제공된 정보는 상기 온라인 게임의 상기 각각의 가상 위치에 할당된 상기 컴퓨터 리소스들의 수를 식별하고 동적으로 조정하는데 사용되는 것을 특징으로 하는 방법.

#### 청구항 2

제1항에 있어서, 모니터링하는 단계는 상기 게임 세계 내에서 식별된 가상 위치에 할당된 각각의 컴퓨터 리소스에 대한 상기 처리 부하가 상기 컴퓨터 리소스에 대해 설정된 미리 정의된 한도 내에 있는지 확인하는 단계를 포함하는 것을 특징으로 하는 방법,

#### 청구항 3

제1항에 있어서, 동적으로 조정하는 단계는, 상기 가상 위치에서 감지된 상기 처리 부하에 기초하여, 하나 이상의 추가 컴퓨터 리소스들을 상기 가상 위치에 할당하거나 또는 상기 가상 위치에 할당된 하나 이상의 컴퓨터 리소스들의 할당을 해제하는 단계를 더 포함하고, 상기 처리 부하는 상기 가상 위치에 할당된 상기 컴퓨터 리소스들의 수 사이에 분산되는 것을 특징으로 하는 방법,

#### 청구항 4

제1항에 있어서, 모니터링하는 단계는,

다수의 사용자들이 게임 상호작용을 제공하고 있는 상기 게임 세계 내에서 식별된 가상 위치 내의 지역-여기서 상기 지역의 상기 다수의 사용자들의 수는 상기 지역의 상기 다수의 사용자들의 수에 의해 제공되는 게임 상호 작용을 처리하기 위한 하나 이상의 컴퓨터 리소스들과 연관되어 있음-을 식별하는 단계; 및

상기 가상 위치의 상기 지역에 액세스하는 상기 다수의 사용자들의 수가 상기 할당된 하나 이상의 컴퓨터 리소스들에 대해 미리 정의된 한도 미만인지를 확인하는 단계;

를 더 포함하고,

상기 다수의 사용자들 각각에 의해 제공되는 게임 상호 작용은 상기 가상 위치의 상기 지역 내에서 상기 처리 부하에 영향을 미치는 것을 특징으로 하는 방법,

**청구항 5**

제4항에 있어서, 동적으로 조정하는 단계는,

상기 미리 정의된 한도를 초과하는 상기 다수의 사용자들의 수의 감지에 응답하여, 상기 지역을 복수의 서브 지역들-여기서 상기 복수의 서브 지역들의 각각의 서브 지역은 상기 지역의 적어도 하나의 사용자를 포함함-로 분할하는 단계; 및

상기 하나 이상의 컴퓨터 리소스들 및 상기 지역에 대해 할당된 하나 이상의 추가 컴퓨터 리소스들이 상기 지역의 각각의 서브 지역에 있는 상기 다수의 사용자들에 의해 제공되는 게임 상호작용을 처리하게 구성되도록, 상기 지역에 하나 이상의 추가 컴퓨터 리소스들을 할당하는 단계

를 더 포함하는 것을 특징으로 하는 방법.

**청구항 6**

제5항에 있어서, 상기 지역 내에 생성된 상기 복수의 서브 지역들의 수는 상기 지역 내에 존재하는 상기 다수의 사용자들의 수에 기초하는 것을 특징으로 하는 방법.

**청구항 7**

제5항에 있어서, 하나 이상의 추가 컴퓨터 리소스들을 할당하는 단계는 각각의 서브 지역과 연관된 게임 상호작용을 나타내는 사용자 입력들을 처리하기 위한 별도의 컴퓨터를 할당하는 단계를 포함하는 것을 특징으로 하는 방법.

**청구항 8**

제5항에 있어서, 상기 지역에 액세스하는 사용자들의 수를 식별하는 단계는, 상기 하나 이상의 사용자들 각각이 상기 복수의 서브 지역들의 각각의 서브 지역으로 또는 상기 각각의 서브 지역 밖으로 이동하는 것과 상기 지역에 할당된 상기 컴퓨터 리소스들을 동적으로 조정하는데 사용되는 상기 하나 이상의 사용자들 각각의 이동을 추적하는 단계를 포함하는 것을 특징으로 하는 방법.

**청구항 9**

제8항에 있어서, 상기 컴퓨터 리소스들의 수를 동적으로 조정하는 단계는, 상기 하나 이상의 사용자들의 이동에 기초하여, 상기 지역 내의 상기 다수의 사용자들의 수가 상기 미리 정의된 한도보다 낮을 경우, 상기 지역 내에 할당된 상기 하나 이상의 추가 컴퓨터 리소스들의 할당을 해제하는 단계를 포함하는 것을 특징으로 하는 방법.

**청구항 10**

제9항에 있어서, 하나 이상의 추가 컴퓨터 리소스들의 할당을 해제하는 단계는 상기 지역에 할당된 상기 하나 이상의 컴퓨터 리소스들 및 상기 추가 컴퓨터 리소스들과 상기 할당 해제되는 상기 하나 이상의 추가 컴퓨터 리소스들 간의 상기 게임 세계의 게임 관련 데이터가 성공적으로 동기화된 후에 수행되며, 상기 게임 관련 데이터는 상기 지역 내에서 상기 하나 이상의 사용자들의 상기 게임 상호 작용으로부터 생성되는 것을 특징으로 하는 방법.

**청구항 11**

제8항에 있어서, 동적으로 조정하는 단계는 상기 하나 이상의 컴퓨터들의 리소스 용량, 상기 지역에 할당된 상기 하나 이상의 추가 컴퓨터들, 및 상기 지역의 리소스 요구사항들에 기초하여 수행되는 것을 특징으로 하는 방법.

**청구항 12**

제8항에 있어서, 상기 복수의 서브 지역들의 각각의 서브 지역은 경계에 의해 정의되고, 상기 서브 지역 내에 존재하는 사용자들로부터 발생하는 게임 상호작용을 처리하기 위한 하나 이상의 별개의 컴퓨터 리소스들이 할당되며;

상기 하나 이상의 사용자들 각각에 대한 이동을 추적하는 단계는,

제1 서버 지역으로부터 상기 지역 내에 정의된 상기 복수의 서버 지역들의 상기 제1 서버 지역과 연속적인 제2 서버 지역 사이에 정의된 경계로 이동하는 사용자를 감지하는 단계; 및

상기 사용자의 게임 상호작용 처리를 가능하게 하기 위해, 상기 경계 상에 있는 사용자를 감지하는 것에 응답하여, 상기 제1 서버 지역 및 상기 제2 서버 지역 각각에 할당된 컴퓨터 리소스들 간의 통신을 관리하는 단계

를 포함하고,

상기 게임 상호 작용의 처리는 상기 제1 서버 지역과 상기 제2 서버 지역에 할당된 컴퓨터 리소스들 간에 동기화되는 것을 특징으로 하는 방법.

**청구항 13**

제8항에 있어서, 상기 하나 이상의 사용자들 각각의 이동을 추적하는 단계는,

상기 지역 내의 상기 하나 이상의 사용자들 각각의 이동을 추적하는 단계; 및

상기 지역에 액세스하는 사용자들의 수에 기초하여 상기 지역 내의 상기 서버 지역들의 수를 조정하는 단계

를 포함하고,

상기 서버 지역들의 수를 조정하는 단계는 상기 지역 내의 2개 이상의 서버 지역들을 병합하거나 또는 상기 지역 내에 추가 서버 지역들을 생성하는 단계를 포함하며,

상기 서버 지역을 조정하는 것은 상기 지역에 할당된 컴퓨터 리소스들을 조정하는 것을 포함하는 것을 특징으로 하는 방법.

**청구항 14**

제4항에 있어서, 상기 지역 내에서의 상기 게임 상호작용은 상기 온라인 게임을 플레이하는 동안 상기 하나 이상의 사용자들에 의해 제공되는 사용자 입력들에 대응하며, 상기 하나 이상의 사용자들은 플레이어들 또는 관중들 중 하나를 포함하는 것을 특징으로 하는 방법.

**청구항 15**

방법으로서, 상기 방법은:

복수의 사용자들-여기서 상기 복수의 사용자들 중 각각의 사용자에게는 온라인 게임에 의해 제공되는 게임 세계의 가상 위치에 대한 액세스가 제공됨-을 위한 상기 온라인 게임을 실행하는 단계;

상기 복수의 사용자들 중 하나 이상의 사용자들이 존재하는 상기 게임 세계의 하나 이상의 가상 위치들을 식별하는 단계;

상기 하나 이상의 사용자들에 의해 상기 게임 세계에 제공되는 게임 상호 작용을 처리하기 위해, 상기 게임 세계 내에서 식별되는 각각의 가상 위치에 하나 이상의 컴퓨터 리소스들-여기서 각각의 가상 위치에 할당된 컴퓨터 리소스들의 수는 상기 가상 위치에서 발생한 상기 게임 상호 작용을 처리하는데 필요한 처리 부하에 기초함-을 할당하는 단계;

각각의 가상 위치 내에서 상기 하나 이상의 사용자들의 이동을 모니터링-여기서 상기 모니터링은 각각의 가상 위치로 이동하거나 또는 상기 가상 위치 밖으로 이동하는 하나 이상의 사용자들을 식별하는 것을 포함함-하는 단계; 및

상기 각각의 가상 위치에서 상기 처리 부하를 핸들링하기 위해, 상기 모니터링된 이동에 기초하여, 상기 각각의 가상 위치에 할당된 컴퓨터 리소스들의 수를 동적으로 조정하는 단계

를 포함하는 것을 특징으로 하는 방법.

**청구항 16**

제15항에 있어서, 상기 모니터링하는 단계는 상기 게임 세계 내에서 식별된 가상 위치에 할당된 각각의 컴퓨터 리소스에 대한 상기 처리 부하가 상기 컴퓨터 리소스에 대해 설정된 미리 정의된 한도 내에 있는지를 확인하는

단계를 포함하는 것을 특징으로 하는 방법.

**청구항 17**

제15항에 있어서, 각각의 가상 위치는 경계에 의해 정의되며, 상기 각각의 가상 위치에 상기 가상 위치 내에 존재하는 사용자들로부터 발생하는 게임 상호 작용을 처리하기 위한 하나 이상의 고유한 컴퓨터 리소스들이 할당되고,

이동을 모니터링하는 단계는,

사용자가 제1 가상 위치로부터 상기 제1 가상 위치와 연속적인 제2 가상 위치 사이에 정의된 경계로 이동하는 것을 감지하는 단계; 및

상기 사용자의 게임 상호작용 처리 및 상기 제1 가상 위치와 상기 제2 가상 위치에 할당된 상기 컴퓨터 리소스들 간에 동기화된 게임 상호 작용의 처리를 가능하게 하기 위해, 상기 사용자가 상기 경계로 이동하는 것을 감지하는 것에 응답하여, 상기 제1 가상 위치 및 상기 제2 가상 위치 각각에 할당된 컴퓨터 리소스들 간의 통신을 관리하는 단계

를 더 포함하는 것을 특징으로 하는 방법.

**청구항 18**

제15항에 있어서, 상기 하나 이상의 사용자들 각각의 이동을 모니터링하는 단계는,

상기 가상 위치 내에서 상기 하나 이상의 사용자들 각각의 이동을 추적하는 단계; 및

상기 가상 위치 내에서 미리 정의된 한도를 초과하는 상기 사용자들의 수를 감지하는 것에 응답하여, 상기 가상 위치를 복수의 지역들로 분할하는 단계

를 포함하고,

각각의 지역은 상기 가상 위치의 적어도 하나의 사용자를 포함하며, 상기 지역으로부터 발생하는 게임 상호 작용을 처리하기 위해 하나 이상의 고유한 컴퓨터 리소스들과 연관되는 것을 특징으로 하는 방법.

**청구항 19**

삭제

**청구항 20**

삭제

**청구항 21**

삭제

**청구항 22**

삭제

**청구항 23**

삭제

**청구항 24**

삭제

**청구항 25**

삭제

**청구항 26**

삭제

**발명의 설명**

**기술 분야**

[0001] 본 개시는 온라인 게임을 위한 리소스들을 할당하는 것에 관한 것으로, 보다 구체적으로는 다수의 플레이어들이 특정 가상 게임 영역에 진입할 때 컴퓨터들을 할당하는 결정론적 방법에 관한 것이다.

**배경 기술**

[0002] 온라인 게임은 게임계 내에서 인기를 끌고 있다. 전통적으로, 게임은 게임 플레이를 위해 게임을 실행하는 클라이언트-서버 설정 또는 피어 투 피어 설정을 사용했다. 전형적인 클라이언트-서버 설정에서, 서버는 게임의 게임 플레이 동안 생성되는 게임 데이터의 물리, 오디오, 전송 등을 처리하기에 충분한 컴퓨트 능력을 제공해야 한다. 대규모 멀티 플레이어 온라인 게임(MMO)의 인기가 높아짐에 따라, 기존의 클라이언트-서버 설정은 게임 플레이 동안 다수의 사용자들(예를 들어, 1000 명의 플레이어들)에 의해 제공된 입력들로 인한 처리 부하(예를 들어, 게임 데이터)를 핸들링할 수 없다. MMO 게임에서, 복수의 플레이어들 각각은 게임 플레이 동안 게임 세계의 그들 자신의 관점을 제공받아야 한다. 예를 들어, 제1 플레이어는 게임 세계의 오른쪽으로부터 특정 도전 과제에 접근할 수 있고, 제2 플레이어는 게임 세계의 왼쪽으로부터 동일한 도전 과제에 접근할 수 있으며, 제3 플레이어는 게임 세계의 전면으로부터 도전 과제에 접근할 수 있다. 이러한 플레이어들 각각은 특정 도전 과제를 극복하기 위한 입력들을 제공할 수 있다. 서버는 먼저 처리되어야 하는 플레이어의 입력들을 결정하고 게임의 전체 게임 상태를 생성해야 한다. 서버는 또한 게임 세계의 그들 자신의 관점을 제공하기 위해 플레이어들 각각에 대해 서로 다른 게임 데이터를 생성할 수 있어야 한다. 서버에는 특정 수의 플레이어들(예를 들어, 사용자들)만 지원할 수 있는 리소스들이 제한되어 있으며, 플레이어들의 수가 이 특정 수를 초과하면, 서버의 리소스들이 너무 얇아져 게임의 게임 플레이의 품질이 최적화되지 않을 수 있다. 피어 투 피어 설정에서, 처리 부담은 게임 데이터를 교환하는 데 사용되는 네트워크에 더 많으며, 처리는 피어들 사이에 분산된다. 추가로, 피어들 중 하나는 "서버" 역할을 하고, 추가 처리 부하를 수행하며, 이는 해당 피어의 리소스들에 스트레스를 더할 수 있다.

[0003] 전형적인 MMO 게임에서, 주어진 시간에 온라인 게임에 액세스할 수 있는 수백 또는 수천 명의 플레이어들이 있을 수 있다. 플레이어들이 최적의 게임 플레이 경험을 갖도록 하기 위해, 복수의 서버들이 수백 또는 수천 명의 플레이어들의 게임 플레이를 지원하는 데 사용될 수 있으며, 플레이어들은 각 논리적 그룹이 서버 영역에 할당되는 논리적 그룹으로 구성될 수 있다. 그러나, 상기 클라이언트-서버 설정에서와 같이, 서버 영역에는 사용자들(즉, 게임 플레이 캐릭터들)의 수에 대한 한도를 갖는다. 새로운 사용자가 특정 서버 부분에서 게임 플레이에 액세스하기 위한 요청을 시작하면, 서버 영역의 사용자들의 수가 서버의 한도에 도달하여 새로운 사용자가 게임 플레이를 시작하는 데 불필요한 지연이 발생하는 경우 서버 영역에 대한 대기열에 새로운 사용자가 배치될 수 있다. 추가로, 새로운 사용자가 다른 시간대에 있는 특정 서버 영역에 가입하도록 요청하면, 사용자의 현재 지리적 위치를 기반으로, 특정 서버 영역이 최적의 옵션이 아닐 수 있더라도 새로운 사용자가 특정 서버 영역에 할당될 수 있다. 이는 특히 새로운 사용자가 친구들과 온라인 게임의 게임 플레이에 참여하고 싶고 특정 레벨의 게임 플레이를 시작하고 싶을 때 문제가 된다. 특정 서버 영역에 새로운 사용자를 할당하면 새로운 사용자에게 지연이 발생할 것이며, 이는 새로운 사용자에게 만족스럽지 않은 경험을 유발할 수 있다.

[0004] 이러한 맥락에서 본 개시의 실시예들이 발생한다.

**발명의 내용**

**과제의 해결 수단**

[0005] 본 개시의 실시예들은 온라인 게임을 위한 리소스들을 프로비저닝하기 위한 시스템들 및 방법들에 관한 것이다. 온라인 게임은 분산형 게임 엔진을 사용하여 실행되어 온라인 게임을 플레이하거나 시청하는 사용자들의 클라이언트 장치들에서 렌더링하기 위해 게임의 게임 세계의 뷰를 포함하는 게임 데이터의 비디오 프레임들을 제공한다. 분산형 게임 엔진은 온라인 게임의 컨텍스트(즉, 게임 상태), 온라인 게임 내 각 사용자의 위치를 추적하고, 온라인 게임의 각 사용자에게 대한 게임 데이터를 생성하기 위해 사용자들에 의해 제공된 사용자 입력들을 처리하는 것을 돕기 위해 필요한 컴퓨트/컴퓨팅 리소스들을 사전에 프로비저닝하기 위한 복수의 컴포넌트들을 포함한다. 분산형 게임 엔진의 컴포넌트들은 온라인 게임 내의 복수의 가상 위치들을 식별하고, 온라인 게

임의 각 가상 위치에 존재하는 사용자들에게 할당된 하나 이상의 컴퓨터들(즉, 컴퓨팅 장치)에 대한 처리 부하를 식별하고, 각 가상 위치에서 처리 부하를 핸들링하기 위해 각 가상 위치에 할당된 컴퓨터들을 사전에 확장 또는 축소하도록 구성된다.

- [0006] 우선, 분산형 게임 엔진은 먼저 온라인 게임에서 식별된 각 가상 위치에 있는 사용자들에게 컴퓨터를 할당하고 온라인 게임과 관련된 할당된 컴퓨터(들)에 대한 처리 부하를 모니터링할 수 있다. 처리 부하는 온라인 게임의 가상 위치 내에 정의된 지역으로 또는 그 밖으로 이동하는 사용자들에 의해 영향을 받을 수 있다. 모니터링에 기초하여, 분산형 게임 엔진은 할당된 컴퓨터가 각각의 가상 위치에서 처리 부하를 핸들링하도록 구성되도록 온라인 게임의 각 가상 위치에 할당된 컴퓨터들을 조정할 수 있다. 컴퓨터들에 대한 조정은 특정 가상 위치에서 처리 부하의 증가를 핸들링하기 위해 추가 컴퓨터(들)을 할당하는 것, 또는 처리 부하의 감소에 기초하여 특정 가상 위치에 할당된 하나 이상의 컴퓨터들을 할당 해제하는 것을 포함할 수 있다. 추가 컴퓨터들을 추가하기 위해, 분산형 게임 엔진은 특정 가상 위치의 지역 내에서 사용자들의 수를 모니터링하고, 사용자들의 수가 특정 가상 위치에 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는지를 결정한다. 지역의 사용자들의 수가 할당된 컴퓨터에 대해 미리 정의된 수를 초과하는 경우, 분산형 게임 엔진은 특정 가상 위치 내의 지역을 복수의 서브 지역들로 세분화하고, 지역의 사용자들에 의해 제공된 게임 상호 작용을 정의하는 사용자 입력들을 처리하기 위해 추가 컴퓨터를 할당할 수 있다. 일부 구현들에서, 각 서브 지역에서 사용자들의 사용자 입력들을 처리하기 위해 별개의 컴퓨터가 할당될 수 있다. 일부 구현들에서, 지역 또는 특정 가상 위치에 할당된 컴퓨터는 지역 내에 정의된 서브 지역들 중 하나에 할당될 수 있다. 다른 구현에서, 특정 가상 위치 또는 지역에 할당된 컴퓨터는 특정 가상 위치의 지역 내에서 생성된 다양한 서브 지역들에 할당된 별개의 컴퓨터들에 의해 핸들링되는 게임 상호 작용의 처리를 조정하는 데 사용될 수 있다.
- [0007] 대안으로, 분산형 게임 엔진은 온라인 게임에 할당된 컴퓨터들의 일부를 할당 해제함으로써 온라인 게임에 할당된 컴퓨터들의 양을 최적화할 수 있다. 컴퓨터들 중 일부를 할당 해제하기 위해, 분산형 게임 엔진은 제1 가상 위치에서 생성된 게임 상호 작용을 처리하는 데 사용된 것을 먼저 할당하여 제1 및 제2 가상 위치들 둘 다의 게임 상호 작용(즉, 처리 부하)을 핸들링하고 제2 가상 위치에서 게임 상호 작용을 처리하기 위해 할당된 제2 컴퓨터를 할당 해제할 수도 있다. 이 경우, 제2 가상 위치에 할당된 컴퓨터는 게임 상호 작용 처리로 인한 게임 데이터가 제1 및 제2 컴퓨터들 사이에 동기화된 후 할당 해제된다. 분산형 게임 엔진은 가상 위치들 각각의 각 지역에 있는 사용자들의 수를 계속 모니터링하고, 온라인 게임에 할당된 컴퓨터들을 사전에 관리한다.
- [0008] 일 구현에서, 게임 세계를 나타내는 온라인 게임을 위한 처리 리소스들을 관리하는 방법이 개시된다. 방법은 한 명 이상의 사용자들을 위한 온라인 게임을 실행하는 단계를 포함한다. 한 명 이상의 사용자들 각각에는 게임 세계의 하나 이상의 가상 위치들에 대한 액세스가 제공된다. 게임 세계에서 게임 상호 작용을 처리하기 위해 한 명 이상의 사용자들에 컴퓨터가 할당된다. 온라인 게임에서 식별된 가상 위치 내의 지역에 액세스하는 사용자들의 수가 사용자들에게 할당된 상기 컴퓨터에 대해 미리 정의된 한도를 초과하는 경우, 지역은 복수의 서브 지역들로 분할된다. 컴퓨터 및 추가 컴퓨터가 서브 지역들 각각의 사용자들을 위한 게임 세계에서 게임 상호 작용을 처리하도록 구성되도록, 추가 컴퓨터가 지역에 할당된다.
- [0009] 일부 구현들에서, 지역에 생성된 서브 지역들의 수는 지역에 존재하는 사용자들의 수에 기초한다. 지역에 생성된 각 서브 지역은 지역의 적어도 한 명의 사용자를 포함한다.
- [0010] 일부 구현들에서, 지역에서 수신된 게임 상호 작용은 지역의 한 명 이상의 사용자들에 의해 제공된 사용자 입력들에 대응한다. 한 명 이상의 사용자들은 플레이어들 또는 관중들이다.
- [0011] 일부 구현들에서, 지역 내에서 생성된 각 서브 지역에는 상기 서브 지역에 대한 게임 상호 작용을 나타내는 사용자 입력들을 처리하기 위해 별개의 컴퓨터가 할당된다.
- [0012] 일부 구현들에서, 게임 상호 작용의 처리는 각 사용자에 의해 제공되는 게임 상호 작용의 상대적 콘텍스트에 기초하여 각 서브 지역에서 게임 상호 작용의 최적 처리 시퀀스를 결정하는 것을 포함한다. 게임 상호 작용은 최적의 처리 시퀀스에 따라 각 서브 지역에서 처리되고, 최적의 처리 시퀀스는 렌더링을 위해 클라이언트 장치에서 각 사용자에게 제공되는 게임 비디오 데이터의 시퀀스를 결정한다.
- [0013] 일부 구현들에서, 각 사용자에 대한 게임 비디오 데이터에 대한 최적의 처리 시퀀스는 우선 순위, 또는 중요도, 또는 임계도, 또는 필요성, 또는 이들의 둘 이상의 임의의 조합들에 기초하여 결정된다.
- [0014] 일부 구현들에서, 최적의 처리 시퀀스는 게임 플레이 동안 발생하는 게임 상호 작용의 타이밍에 기초한다.
- [0015] 일부 구현들에서, 지역의 각 서브 지역 내에서 사용자들의 이동이 추적되어 지역에 액세스하는 사용자들의 수를

결정한다. 사용자들의 이동은 지역에 할당된 컴퓨터를 조정하는 데 사용된다.

- [0016] 일부 구현들에서, 컴퓨터를 조정하는 단계는 지역 내 사용자들의 수가 한 명 이상의 사용자들의 움직임에 기초하여 컴퓨터에 대한 미리 정의된 한도 미만으로 떨어질 때, 지역에 할당된 추가 컴퓨터를 할당 해제하는 단계를 포함한다.
- [0017] 일부 구현들에서, 지역에 할당된 컴퓨터는 지역 밖으로 이동하는 한 명 이상의 사용자들을 검출하고, 이에 응답하여, 컴퓨터 또는 지역에 할당된 추가 컴퓨터를 할당 해제함으로써 조정된다. 할당 해제에는 컴퓨터와 추가 컴퓨터 간에 게임 세계의 게임 관련 데이터를 성공적으로 동기화한 후에 수행된다. 게임 관련 데이터는 지역의 사용자들의 게임 상호 작용의 처리로부터 생성된다.
- [0018] 일부 구현들에서, 조정은 컴퓨터와 추가 컴퓨터의 리소스 능력들 및 지역의 리소스 요구사항들에 기초하여 수행된다.
- [0019] 일부 구현들에서, 지역의 각 서브 지역은 경계에 의해 정의되고 해당 서브 지역의 사용자들에 대한 게임 상호 작용을 처리하기 위해 별개의 컴퓨터가 할당된다. 지역 내 사용자들의 움직임이 추적되고, 상기 지역의 서브 지역으로부터 사용자가 두 개의 연속적인 서브 지역들 사이의 경계로 이동한 것으로 결정되며, 두 개의 연속적인 서브 지역들 각각에 할당된 컴퓨터들 간의 통신은 사용자의 게임 상호 작용의 처리를 가능하게 하도록 관리된다. 게임 상호 작용의 처리는 컴퓨터들 사이에서 동기화된다.
- [0020] 일부 구현들에서, 가상 위치 내의 지역에서 사용자들의 이동이 추적되며, 상기 지역 내의 서브 지역들은 상기 지역에 액세스하는 사용자들 수에 기초하여 조정된다. 조정은 지역 내의 하나 이상의 서브 지역들을 병합하거나 지역 내의 추가 서브 지역들을 생성하는 것을 포함한다. 서브 지역들의 조정은 상기 지역에 할당된 컴퓨터에 대한 조정을 야기한다.
- [0021] 대안적인 구현에서, 온라인 게임의 리소스들을 관리하는 방법이 개시된다. 방법은 복수의 사용자들로부터 수신된 사용자 입력들에 응답하여, 온라인 게임을 실행하는 단계를 포함한다. 온라인의 게임의 실행은 온라인 게임의 게임 세계 내의 가상 위치들에 대한 액세스를 제공한다. 컴퓨터는 각 가상 위치에 있는 한 명 이상의 사용자들에게 할당된다. 컴퓨터는 사용자들 각각에 대한 게임 비디오 데이터를 생성하기 위해 대응하는 가상 위치 내에 존재하는 한 명 이상의 사용자들의 사용자 입력들을 처리하도록 구성된다. 각 사용자의 움직임은 게임 세계 내에서 추적되어 다수의 사용자들이 가상 위치들 중 한 위치 내의 지역의 액세스하는 시기를 결정한다. 지역은 사용자들의 수가 지역의 사용자들에 할당된 컴퓨터에 대한 미리 정의된 한도를 초과할 때, 복수의 서브 지역들로 분할된다. 상기 지역에 대해 추가 컴퓨터가 할당되어, 컴퓨터 및 추가 컴퓨터가 사용자들 각각에 대해 해당 비디오 게임 데이터를 생성하기 위해 지역의 사용자들 각각에 의해 제공된 사용자 입력들을 처리하도록 한다.
- [0022] 본 개시의 다른 양태들은 본 개시의 원리들을 예로서 예시하는 첨부 도면들과 함께 취해진 다음의 상세한 설명으로부터 명백해질 것이다.

**도면의 간단한 설명**

- [0023] 본 개시는 첨부 도면들과 관련하여 연계된 다음의 설명을 참조하여 가장 잘 이해될 수 있다.
- 도 1은 본 개시의 일 구현에 따른, 온라인 게임을 위한 리소스들을 프로비저닝하는데 사용되는 게임 클라우드 시스템에서 실행되는 분산형 게임 엔진의 단순화된 블록도를 예시한다.
- 도 2는 본 개시의 일 구현에 따른, 온라인 게임을 위한 리소스들을 프로비저닝하는 데 사용되는 컴포넌트들 중 일부를 식별하는 분산형 게임 엔진의 단순화된 블록도를 예시한다.
- 도 3a는 본 개시의 일 구현에 따른, 온라인 게임을 위한 리소스들을 프로비저닝하는 데 사용되는 분산형 게임 엔진의 컴퓨터 지역 관리자 모듈의 컴포넌트들 중 일부의 세부사항들을 식별하는 단순화된 블록도를 예시한다.
- 도 3b는 본 개시의 일 구현에 따른, 온라인 게임을 위한 리소스들을 프로비저닝하는 데 사용되는 분산형 게임 엔진 내의 컴퓨터 지역 관리자 모듈의 컴포넌트들 중 일부의 세부사항들을 식별하는 단순화된 블록도를 예시한다.
- 도 4a 내지 4c는 본 개시의 일 구현에 따른, 온라인 게임의 게임 세계 내의 가상 위치에 대해 서브 지역들이 생성될 수 있는 다양한 방식을 예시한다.

도 5는 본 개시의 일 구현에 따른, 클라우드 게임 네트워크를 통해 게임 데이터의 프레임들을 하나 이상의 클라이언트 장치들에 제공하기 위해 온라인 게임을 실행하는 데 사용되는 클라우드 게임 시스템의 시스템 도면을 예시한다.

도 6은 본 개시의 일 구현에 따른, 리소스 할당 모델을 사용하여, 온라인 게임에 대한 리소스들을 프로비저닝하기 위한 방법의 흐름 동작들을 예시한다.

도 7은 본 개시의 일 구현에 따른, 모델을 사용하여 리소스 할당 모델(인공 지능(AI) 모델)을 구축 및 트레이닝하는 데 사용되는 예시적인 컴퓨팅 장치의 컴포넌트들을 예시한다.

**발명을 실시하기 위한 구체적인 내용**

[0024] 다음의 상세한 설명은 예시의 목적으로 많은 특정 세부 사항들을 포함하지만, 당업자라면 다음 세부 사항들에 대한 많은 변형들 및 변경들이 본 개시의 범위 내에 있음을 이해할 것이다. 따라서, 이하에 설명되는 본 개시의 양태들은 이 설명을 따르는 청구 범위에 대한 일반성을 잃지 않고, 그리고 이에 대한 제한을 부여하지 않고 명시된다.

[0025] 일반적으로 말해서, 본 개시의 다양한 실시예들은 온라인 게임에 대한 리소스 요구 사항들, 온라인 게임에 할당된 각 컴퓨터의 리소스 능력들을 이해하고, 온라인 게임에 최적의 리소스 할당을 제공하기 위해 게임 플레이 중에 온라인 게임에 할당된 컴퓨터를 동적으로 조정하는 방법들을 제공하는 시스템들 및 방법들을 설명한다. 온라인 게임은 다수의 사용자들에 의해 액세스되는 대규모 멀티 플레이어 온라인(MMO) 게임일 수 있다. 복수의 사용자들은 게임 플레이를 위해 온라인 게임에 액세스하는 플레이어들이거나, 게임 플레이를 보거나 온라인 게임을 플레이하는 특정한 한 명 이상의 플레이어들을 팔로우하기 위해 온라인 게임에 액세스하는 관중들일 수 있거나, 또는 플레이어들과 관중들 둘 다의 조합일 수 있다. MMO 게임은 게임 상호 작용을 통해 복수의 사용자들로부터 수신된 사용자 입력들에 기초한 게임 데이터를 제공한다. 플레이어 및 플레이어를 팔로우하는 관중들을 위해 생성된 게임 데이터는 플레이어와 관련된 온라인 게임의 게임 상태를 포함한다. 복수의 플레이어들에 대해 생성된 게임 데이터는 온라인 게임의 게임 세계관을 포함한다.

[0026] 복수의 가상 위치들은 게임 세계에서 식별된다. 각 가상 위치는 온라인 게임의 한 명 이상의 사용자들(플레이어들 및/또는 관중들)이 존재하는 경계들이 정의된 게임 세계의 한 지역이다. 각 가상 위치에서 생성되는 게임 상호 작용의 양을 결정하기 위해 각 가상 위치에서 사용자 입력 형식의 게임 상호 작용이 모니터링된다. 컴퓨터는 게임 세계에서 생성된 게임 상호 작용의 처리를 핸들링하기 위해 온라인 게임 사용자들에게 할당된다. 온라인 게임에 할당된 컴퓨터는 게임 상호 작용에 의해 정의된 처리 부하를 처리하는 데 사용할 수 있는 처리 리소스들에 기초하여 식별된다. 각 사용자가 온라인 게임을 진행함에 따라, 게임 세계에서 사용자의 위치는 제1 가상 위치에서 제2 가상 위치로 변경될 수 있다. 분산형 게임 엔진은 게임 세계에서 각 사용자의 움직임을 모니터링하여 다수의 사용자들이 특정 가상 위치 내의 지역에 액세스하는 시기를 결정한다. 지역에 있는 사용자들의 수는 특정 가상 위치 내의 지역에서 게임 상호 작용을 증가시킬 수 있다. 각 컴퓨터에는 게임 상호 작용을 처리할 수 있는 사용자들의 수에 대한 미리 정의된 한도가 있다. 지역에 있는 사용자들의 수가 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는 경우, 특정 가상 위치의 지역은 각 서버 지역이 상기 지역으로부터의 적어도 한 명의 사용자를 갖는 복수의 서버 지역들로 동적으로 분할된다. 사용자들을 위한 게임 세계의 게임 상호 작용으로 인한 처리 부하를 핸들링하고 서버 지역들 각각의 사용자들 각각에 대한 게임 비디오 데이터를 생성하기 위해 상기 지역에 추가 컴퓨터가 할당된다. 일부 구현들에서, 각 서버 지역은 해당 서버 지역에 있는 사용자들의 게임 상호 작용을 핸들링하기 위해 별개의 컴퓨터가 할당된다. 서버 지역들 각각에 할당된 컴퓨터들은 서버 지역에서 게임 상호 작용으로부터 생성된 처리 부하를 핸들링하는 데 필요한 리소스들에 기초하여 식별된다.

[0027] 컴퓨터를 할당한 후, 사용자들의 이동은 한 명 이상의 사용자들이 한 지역에서 다른 지역으로 이동하는 시기를 결정하기 위해 모니터링 된다. 모니터링에 응답하여, 온라인 게임에 할당된 컴퓨터들은 온라인 게임 내에서 다른 가상 환경의 사용자들의 게임 상호 작용을 핸들링하기 위해 컴퓨터들 중 일부가 할당 해제되거나 재할당될 수 있는지를 결정하기 위해 평가될 수 있다. 컴퓨터의 할당 해제 또는 재할당은 사용자들의 게임 상호 작용의 처리에서 생성된 게임 데이터를 할당 해제되고 있는 컴퓨터로부터 온라인 게임의 게임 상호 작용을 핸들링하도록 할당되는 컴퓨터로 성공적으로 마이그레이션한 후 수행된다.

[0028] 다양한 실시예들의 상기 일반적인 이해와 함께, 이제 실시예들의 예시적인 세부 사항들이 다양한 도면들을 참조하여 설명될 것이다.

[0029] 도 1은 하나의 구현에서, 상이한 지리적 위치들로부터 복수의 사용자들에 의해 액세스되는 MMO 게임(즉, 온라인

게임)을 실행하기 위해 게임 클라우드 시스템에서 사용되는 분산형 게임 엔진(100)의 단순화된 블록도를 예시한다. 분산형 게임 엔진(100)은 온라인 게임의 특정 특징들을 처리하는데 사용되는 기능적 컴포넌트들을 포함한다. 온라인 게임의 게임 로직과 함께 분산형 게임 엔진(100)의 인스턴스들은 게임 클라우드 시스템의 하나 이상의 서버들(즉, 컴퓨터들(102a 내지 102n))에서 실행될 수 있으며, 인스턴스들 중 하나 이상은 온라인 게임의 특정 특징을 처리하는 데 사용될 수 있다. 기능적 컴포넌트들은 온라인 게임의 특정 특징들을 처리하기 위한 엔진들/모듈들을 포함한다. 분산형 게임 엔진의 다양한 기능적 컴포넌트들에 의해 처리될 수 있는 온라인 게임의 특징들 중 일부는 몇 가지 예를 들면, 물리현상, 오디오, 셰이딩, 텍스처링, 어셈블링/인코딩, 스크립팅, 인공 지능(AI), 메모리 관리, 스크립팅, 애니메이션, 장면 그래프를 포함할 수 있다.

[0030] 게임 클라우드 시스템은 본 개시의 일 구현에 따른, 하나 이상의 온라인 게임 애플리케이션들의 인스턴스들을 실행하는 백엔드 서버들(예를 들어, 컴퓨터들 1-n)의 네트워크를 포함한다. 하나 이상의 클라이언트 장치들(도시되지 않음)은 인터넷과 같은 네트워크(200)를 통해 게임 클라우드 시스템의 게임 서버에서 실행되는 온라인 게임에 액세스하는 데 사용된다. 클라이언트 장치들은 네트워크(200)를 통해 백엔드 서버들과 상호 작용하기 위해 적어도 메모리, 프로세서 및 네트워크 인터페이스를 포함하는 모바일 폰, 태블릿 컴퓨터, 개인용 컴퓨터 등과 같은 임의의 컴퓨팅 장치일 수 있거나, 계산 기능들을 제공하는 백엔드 서버와 상호 작용하는 인터페이스들을 제공하는 쉰 클라이언트(thin client)일 수 있다. 백엔드 서버는 하나 이상의 게임들의 하나 이상의 인스턴스들을 실행하도록 구성되는 게임 서버이다. 게임 서버는 게임 클라우드 시스템에서 사용 가능한 모든 유형의 서버 컴퓨팅 장치일 수 있다. 한 예시적인 구현에서, 게임 서버는 호스트의 리소스들을 이용하여, 사용자를 위해 온라인 게임의 인스턴스를 실행하는 게임 프로세서를 지원하는 가상 기계일 수 있다. 다른 예시적인 구현에서, 게임 서버는 독립형 물리적 서버 또는 단일 데이터 센터의 일부 또는 복수의 데이터 센터들의 일부인 물리적 서버일 수 있다. 한 구현에서, 분산형 게임 엔진(100) 및 게임 로직의 인스턴스를 실행하는 중앙 서버(즉, 중앙 컴퓨터)(102)는 네트워크(200)를 통해 다른 서버들(즉, 컴퓨터들)(102a-102n)에 대한 입력들 및 출력들을 조정할 수 있으며, 다른 서버들로부터의 출력들을 어셈블링하고 네트워크(200)를 통해 하나 이상의 클라이언트 장치들로의 지속적인 전송을 위해 온라인 게임의 출력들을 인코딩한다. 네트워크(200)는 인터넷, 광역 네트워크, 근거리 통신망, 사설 네트워크, 게임 네트워크 등을 포함하는 임의 유형의 네트워크일 수 있다.

[0031] 온라인 게임은 게임 클라우드 시스템의 복수의 서버들(즉, 컴퓨터들(102a-102n)) 및 네트워크(200)를 통해 사용자들(플레이어들 또는 관중들)의 클라이언트 장치들로 스트리밍되는 게임 데이터에 의해 원격으로 실행될 수 있다. 플레이어는 온라인 게임의 게임 상태에 영향을 미치는 게임 상호 작용(사용자 입력들)을 제공하여 게임의 게임 플레이에 참여하는 사용자이다. 관중은 다수의 플레이어들의 게임 상호 작용(사용자 입력들)을 포함하는 온라인 게임의 일반적인 게임 플레이를 팔로우하거나 특정 플레이어의 게임 플레이를 팔로우하는 사용자이다. 온라인 게임의 게임 플레이는 클라이언트 장치들에서 제공되는 제어 옵션들을 사용하거나 클라이언트 장치들에 통신 가능하게 연결된 컨트롤러들을 사용하는 플레이어들에 의해 제어된다. 다른 구현들에서, 온라인 게임은 클라이언트 장치들에서 로컬로 실행될 수 있으며, 실행중인 온라인 게임으로부터의 메타 데이터는 게임 상태에 영향을 주기 위해 네트워크(200)를 통해 게임 클라우드 시스템의 백엔드 서버들로 전송된다.

[0032] 도 2는 일 구현에서, 온라인 게임의 게임 플레이 동안 생성된 게임 상호 작용을 핸들링하기 위해, 컴퓨터들 및 컴퓨팅 리소스들을 프로비저닝하는데 사용되는 분산형 게임 엔진(100)의 블록도를 예시한다. 온라인 게임은 복수의 사용자들에 의해 플레이되거나 액세스되는 MMO 게임일 수 있다. 복수의 사용자들은 하나의 지리적 위치로부터 또는 다양한 지리적 위치들로부터 온라인 게임에 액세스될 수 있다. 분산형 게임 엔진은 온라인 게임의 게임 세계 내에서 사용자들의 움직임을 지속적으로 추적하고 온라인 게임의 게임 세계에서 생성된 게임 상호 작용을 핸들링하기 위해 온라인 게임에 다른 컴퓨터들을 할당하는 알고리즘을 포함한다. 알고리즘은 사용자들에게 게임 세계의 다양한 가상 위치들에 대한 액세스를 제공하고, 가상 위치들 각각에서 생성된 게임 상호 작용의 처리를 관리한다. 온라인 게임의 사용자들의 게임 상호 작용을 빠르고 효율적인 방식으로 처리하기 위해, 분산형 게임 엔진의 인스턴스들과 온라인 게임의 게임 로직은 게임 세계에서 식별된 각 가상 위치에서 생성된 게임 상호 작용을 처리하는 데 사용되는 컴퓨터들을 갖는 서로 다른 컴퓨터들(즉, 서버들)에서 실행된다. 또한, 게임 엔진의 인스턴스를 실행하는 컴퓨터(예를 들어, 중앙 서버) 중 하나는 온라인 게임에 할당된 컴퓨터들 각각에 대한 입력들 및 출력들을 관리하고, 모든 컴퓨터들로부터의 출력들을 어셈블링하고, 온라인 게임을 플레이하거나 팔로우하는 사용자들 각각에 대해, 게임 비디오의 게임 상태들을 포함하는 게임 비디오 데이터를 생성하는 데 사용될 수 있다. 온라인 게임에 할당된 컴퓨터들은 단일 데이터 센터 또는 다수의 데이터 센터들의 일부일 수 있다. 분산형 게임 엔진은 병렬 처리를 사용하여 서로 다른 사용자들에 의해 제공되는 게임 상호 작용(사용자 입력들)으로 인해 지속적으로 변경되는 게임 비디오 데이터를 처리하고, 서로 다른 사용자들과 관련된 각각

의 클라이언트 장치들에 게임 비디오 데이터의 프레임들을 제공한다.

[0033] 분산형 게임 엔진(100)은 온라인 게임의 게임 세계에서 식별된 상이한 가상 위치들에서 생성된 게임 상호 작용을 처리하기 위해 할당된 컴퓨터들 및 컴퓨팅 리소스들을 관리하는 데 사용되는 복수의 컴포넌트들을 포함한다. 분산형 게임 엔진(100)의 컴포넌트들 중 일부는 사용자 위치 추적기(105), 위치 부하 분석기(107), 컴퓨터 할당기(109), 게임 콘텍스트 프로세서(111) 및 동적 컴퓨터 지역 관리자(120)를 포함한다. 사용자 위치 추적기(105)는 게임 세계에서 각 사용자의 움직임을 모니터링하고 각 사용자가 게임 세계 내에 존재하는 가상 위치를 식별하는 데 사용된다. 추적의 일부로서, 사용자 위치 추적기(105)는 온라인 게임의 다수의 사용자들이 특정 가상 위치 내의 지역에 액세스하는 시기를 결정할 수 있다. 지역 내의 다수의 사용자들은 온라인 게임에 액세스되는 사용자들의 서브셋일 수 있다. 점점 더 많은 사용자들이 지역에 액세스함에 따라, 지역에 있는 사용자들에 의해 게임 상호 작용의 증가를 경험할 수 있다. 게임 상호 작용의 증가는 지역의 각 사용자는 온라인 게임에 대한 그들 자신의 관점을 제공받아야 하기 때문에 상기 지역에서 생성되는 게임 비디오 데이터(본원에서는 게임 데이터라고도 함)의 증가를 초래한다. 사용자는 플레이어 또는 관중일 수 있다. 플레이어에 대한 게임 데이터는 해당 플레이어에 대한 온라인 게임의 게임 상태를 포함한다. 관중을 위해 생성된 게임 데이터는 관중이 온라인 게임의 지역에서 팔로우하고 있는 플레이어의 온라인 게임의 게임 상태를 포함한다. 일부 구현들에서, 컴퓨터는 핸들링할 수 있는 플레이어들의 수와 관중들의 수에 대해 별도의 사전 정의된 한도를 정의할 수 있다. 미리 정의된 한도는 게임 세계 내의 지역 또는 가상 위치에 액세스하고 있는 플레이어 유형(예를 들어, 전문가 레벨 플레이어, 중간 레벨 플레이어 또는 초보 레벨 플레이어)에 대해 특정될 수 있다. 이러한 구현들에서, 사용자 위치 추적기(105)는 게임 세계의 특정 가상 위치에 있는 지역에 액세스하고 있는 각 유형의 사용자들의 유형 및 수를 결정할 수 있으며, 온라인 게임에 할당된 컴퓨터를 평가하고 온라인 게임에 할당된 컴퓨터가 상기 지역에 있는 사용자들의 수를 핸들링하도록 구성되는지 여부를 결정하기 위해 위치 부하 분석기(107)에 신호를 생성할 수 있다.

[0034] 일부 구현들에서, 온라인 게임은 대응하는 가상 위치에서 사용자들에 의해 생성된 게임 상호 작용을 처리하기 위해 온라인 게임의 게임 세계에서 식별된 각 가상 위치의 사용자들에게 별개의 컴퓨터가 할당되도록 복수의 컴퓨터들이 할당될 수 있다. 온라인 게임에 할당된 각 컴퓨터는 컴퓨터가 핸들링할 수 있는 사용자들의 수에 대한 미리 정의된 한도를 가질 수 있다. 상기에 언급된 바와 같이, 각 컴퓨터에는 온라인 게임에 액세스하는 각 유형의 사용자에 대해 미리 정의된 한도가 있을 수 있다. 위치 부하 분석기(107)는, 사용자 위치 추적기(105)로부터의 신호에 응답하여, 상기 지역 내 사용자들의 유형 및 사용자들의 수를 평가한다. 각 유형의 사용자들의 수가 대응하는 미리 정의된 한도를 초과하는 경우, 위치 부하 분석기(107)는 컴퓨터 할당기(109)와 상호 작용하여 상기 지역에 대한 컴퓨터를 조정한다.

[0035] 위치 부하 분석기(107)에 의해 교환된 정보에 기초한 컴퓨터 할당기(109)는 게임 세계에서 각 가상 위치에 할당된 컴퓨터(들)의 능력을 결정하고, 특정 가상 위치의 지역에 할당하기 위한 추가 컴퓨터를 식별한다. 컴퓨터 할당기(109)는 식별된 추가 컴퓨터를 동적으로 프로비저닝하여, 컴퓨터 및 추가 컴퓨터가 특정 가상 위치의 지역에서 게임 상호 작용을 처리할 수 있도록 한다. 컴퓨터 할당기(109)는 기존 컴퓨터(들) 등에 대한 추가 CPU 코어들과 같은 마이크로 레벨에서 또는 하나 이상의 추가 서버들(즉, 컴퓨터들)과 같은 매크로 레벨에서 추가 컴퓨터를 특정 가상 위치에 프로비저닝할 수 있다. 추가 컴퓨터의 할당은 추가 컴퓨터들이 상기 지역에서 생성된 게임 상호 작용의 처리를 핸들링하기 위해 특정 가상 위치의 지역에 있는 사용자들 중 특정 사용자들의 게임 상호 작용을 추가 컴퓨터로의 마이그레이션을 포함한다. 특정 가상 위치에 있는 사용자들에게 할당된 컴퓨터 및 추가 컴퓨터는 함께 작동하여 상기 지역 및 특정 가상 위치에 있는 사용자들의 게임 상호 작용을 처리하고 특정 가상 위치에 있는 사용자들 각각에 게임 데이터를 생성한다.

[0036] 온라인 게임에서 사용자들의 게임 상호 작용으로 인한 처리 부하를 핸들링하기 위해 온라인 게임에 추가 컴퓨터들을 할당한 후, 분산형 게임 엔진은 동적 컴퓨터 지역 관리자(120)를 사용하여 게임 세계 내에서 사용자들의 움직임을 지속적으로 모니터링하고 온라인 게임에 할당된 컴퓨터들을 관리한다. 모니터링으로 인해 온라인 게임에 할당된 컴퓨터가 조정될 수 있다. 예를 들어, 게임 플레이 동안, 가상 위치에서 게임 상호 작용의 양은 가상 위치 안팎으로의 사용자들의 움직임에 의해, 또는 가상 위치 내의 임의의 두 사용자들 간의 또는 두 개의 서로 다른 가상 위치들 내의 증가된 상호 작용 등에 의해 영향을 받을 수 있다. 컴퓨터 지역 관리자(120)는 게임 세계에서 식별된 서로 다른 가상 위치들 내에서 사용자들의 움직임 및 게임 상호 작용을 추적하고, 서로 다른 가상 위치들에 있는 사용자들의 수, 사용자들의 게임 상호 작용들을 평가하고, 온라인 게임의 서로 다른 각 가상 위치에 있는 사용자들의 게임 상호 작용들을 처리하기 위해 할당된 컴퓨터들을 확장하거나 축소함으로써 온라인 게임에 할당된 컴퓨터(들)을 동적으로 조정한다.

[0037] 다수의 사용자들이 주어진 시간에 가상 위치 내의 특정 지역에 액세스하는 경우, 사용자들의 수는 가상 위치에서 사용자들의 게임 상호 작용을 처리하기 위해 할당된 컴퓨터에 대해 정의된 미리 정의된 한도에 대해 검증된다. 특정 지역에 있는 사용자들의 수가 가상 위치에 할당된 하나 이상의 컴퓨터들의 용량을 초과하는 것으로 결정되면, 컴퓨터 지역 관리자(120)는 일부 구현에서 특정 지역을 복수의 서버 지역들로 분할한다. 각 서버 지역은 서버 경계에 의해 정의된다. 생성되는 서버 지역들의 수는 특정 지역에 있는 각 유형의 사용자들의 유형과 수를 기반으로 한다. 일부 구현들에서, 사용자들(예를 들어, 플레이어들 또는 관중들)의 유형과 각 유형의 사용자들의 수를 식별하는 세분화 기준은 특정 지역 내에서 생성할 서버 지역들의 수를 결정하는 데 사용될 수 있다. 지역 내에 생성된 서버 지역들은 같은 크기이거나 다른 크기일 수 있다. 서버 지역들 각각은 지역으로부터 적어도 한 명의 사용자를 포함한다.

[0038] 도 4a 내지 4c는 서버 지역들이 게임 세계에서 식별된 가상 위치의 지역 내에서 정의될 수 있는 방법에 대한 다양한 시나리오들을 예시한다. 상기에 언급된 바와 같이, 가상 위치는 지역들을 포함할 수 있으며, 각 지역은 복수의 서버 지역들로 세분화될 수 있고, 서버 지역들의 수는 가상 위치의 특정 지역에 있는 사용자들의 밀도에 기초한다. 사용자들이 게임 플레이를 위한 온라인 게임을 선택하면, 사용자들에게 온라인 게임에 정의된 게임 세계의 가상 위치들에 대한 액세스가 제공된다. 사용자들이 게임 세계에서 이동함에 따라, 다수의 사용자들이 주어진 시간에 특정 가상 위치 내의 특정 지역에 진입할 수 있다. 특정 지역은 특정 가상 위치의 일부로 정의된다. 특정 지역의 사용자들의 수가 온라인 게임에 할당된 컴퓨터에 대해 정의된 미리 정의된 한도를 초과하는 경우, 지역은 특정 가상 위치의 특정 지역의 처리 부하를 분산시키기 위해 각 지역에 별개의 컴퓨터가 할당된 복수의 서버 지역들로 분할될 수 있다. 특정 가상 위치에 할당된 컴퓨터의 처리 부하는 특정 가상 위치에서 사용자들에 의해 제공되는 게임 상호 작용의 양에 해당한다. 서버 지역들의 수는 지역에 있는 사용자들의 밀도를 기반으로 한다. 일부 구현들에서, 서버 지역들은 지역 내의 사용자들의 분포에 기초하여 지역의 전체 영역을 커버할 수 있다. 대안적인 구현들에서, 서버 지역들은 지역에 있는 사용자들의 분포에 기초하여 지역의 일부만 커버할 수 있다.

[0039] 도 4a는 특정 가상 위치인, 위치 1의 일부를 나타내는 지역인, 지역 A가 주어진 시간에 다수의 사용자들에 의해 액세스되는 하나의 표현을 예시한다. 지역 A의 사용자들의 수는 위치 1에 있는 사용자들의 게임 상호 작용을 처리하기 위해 할당된 컴퓨터의 미리 정의된 한도를 초과하는 것으로 결정된다. 그 결과, 지역 A는 4개의 서로 다른 서버 지역들 - 서버 지역들 A1, A2, A3 및 A4로 세분된다. 도 4a의 예시적인 예시에서, 서버 지역들의 수는 가상 위치 1의 사용자들의 수를 기반으로 한다. 서버 지역들은 같은 크기이고 서버 지역들 각각은 지역 A로부터의 한 명 이상의 사용자들을 포함한다. 서버 지역들 각각에 있는 사용자들의 수는 같을 수도 있고 같지 않을 수도 있다. 서버 지역 각각에 있는 사용자들은 온라인 게임의 플레이어들 및/또는 관중들을 포함한다. 사용자들이 온라인 게임의 게임 상태에 영향을 미치기 위해 게임 상호 작용들(즉, 사용자 입력들)을 제공하지 않는 관중들인 경우에도, 플레이어를 팔로우하는 관중들 각각에게 온라인 게임의 해당 플레이어의 관점이 제시되어야 한다는 점에 유의해야 한다. 대안으로, 관중들 각각은 일반적으로 온라인 게임의 전반적인 흐름을 따라갈 수 있으며, 게임 세계에 대한 시점을 제공받을 수 있다.

[0040] 각 생성된 서버 지역에는 해당 서버 지역의 각 사용자에게 게임 데이터를 제공하기 위해 생성된 게임 상호 작용을 처리하도록 별개의 컴퓨터(들)이 할당된다. 예를 들어, 서버 지역 A1에는 CPU 1이 할당되고, 서버 지역 A2에는 CPU 2가 할당되고, 서버 지역 A3에는 CPU 3이 할당되고, 서버 지역 A4에는 CPU 4가 할당되며, 서버 지역들 A1, A2, A3 및 A4에 할당된 컴퓨터들(CPU 1, CPU 2, CPU 3 및 CPU 4) 각각은 각각의 서버 지역에서 생성된 게임 상호 작용으로 인한 처리 부하를 핸들링하는 데 사용할 수 있는 리소스들을 기반으로 각각 식별된다. 도 4a에 예시된 예에서, 각 서버 지역에는 하나의 컴퓨터가 할당된다. 다른 예들에서, 각 서버 지역에는 그 안에서 생성된 처리 부하를 핸들링하기 위해 복수의 컴퓨터들이 할당될 수 있다.

[0041] 도 4b는 가상 위치 1이 복수의 지역들인, 지역들 A 내지 C를 포함하며, 각 지역은 각 지역 내에서 생성된 게임 상호 작용을 핸들링하기 위해 별개의 컴퓨터(각각 CPU들 1 내지 3)이 할당되는 구현을 예시한다. 가상 위치 1 내의 다른 지역들 또는 게임 세계의 다른 가상 위치들에서의 다수의 사용자들이 주어진 시간에 가상 위치 1의 지역 A에 액세스함에 따라, 지역 A는 지역 A에 액세스하는 사용자들의 양에 기초하여 복수의 서버 지역들로 분할된다. 각 서버 지역은 지역 A의 한 명 이상의 사용자들을 포함하며, 각각의 서버 지역 내에서 처리 부하를 핸들링하기 위해 별개의 컴퓨터가 할당된다. 가상 위치 1 내에서 생성된 서버 지역들의 수는 단지 예시일 뿐이며, 더 적거나 많은 수의 서버 지역들이 가상 위치 1 내에서 생성될 수 있다. 도 4ba, 4bb 및 4bc는 지역 A가 분할될 수 있는 다양한 방법들을 보여준다. 예를 들어, 도 4ba에 도시된 바와 같이, 지역 A는 수직 축을 따라 지역 A를 분할하여 3개의 추가 서버 지역들 A-1, A-2 및 A-3로 분할된다. 도 4bb에 도시된 바와 같이, 지역 A는 수평

측에 따라 지역 A를 분할하여 3개의 추가 서버 지역들 A'-1, A'-2 및 A'-3로 분할된다. 도 4bc는 지역 A가 6개의 서버 지역들(A''-1 내지 A''-6)로 세분화되는 예를 도시한다. 지역 A의 분할에서 생성되는 서버 지역들의 수는 지역 A 내의 사용자들의 양에 기초하여 달라질 수 있다.

[0042] 도 4ba, 4bb 및 4bc에 예시된 예에서, 지역 A를 분할하여 생성된 서버 지역들은 크기가 같고, 서버 지역들 각각은 지역 A의 한 명 이상의 사용자들을 포함한다. 도 4ba 내지 4bc는 가상 위치 내의 지역이 서버 지역들로 분할되는 경우의 예들을 도시한다. 동일한 예들이 확장되어 가상 위치 또는 가상 위치의 지역 내에 정의된 서버 지역을 분할할 수 있다. 서버 지역들의 수와 지역을 분할하는 방향은 단지 예들로서 제공될 뿐이며, 각 지역은 수직, 수평, 대각선 또는 임의의 다른 방식으로 더 적거나 많은 수의 서버 지역들로 세분화될 수 있으며, 서버 지역들의 수는 사용자들의 밀도와 지역에 할당된 컴퓨터의 미리 정의된 한도를 기반으로 한다. 새로 생성된 서버 지역들 각각은 지역 A에 할당된 컴퓨터와는 별도의 개별 컴퓨터가 할당된다. 예를 들어, 도 4ba에서, 지역 A 내에 생성된 서버 지역들 A-1 내지 A-3은 각각 CPU 4 내지 6에 각각 할당되며, CPU 1 내지 3은 가상 위치 1에서 생성된 지역들 A 내지 C에 할당된다(도 4b 참조). 마찬가지로, 도 4bb에서, 서버 지역들 A'-1 내지 A'-3은 CPU 4 내지 6에 각각 할당되며, 도 4bc에서, 서버 지역들 A''-1 내지 A''-6은 CPU 4 내지 9에 각각 할당된다.

[0043] 도 4ba, 4bb 및 4bc에 예시된 예시적인 시나리오에서, 지역 A에 할당된 컴퓨터는 지역 A(도 4ba의 서버 지역들 A-1 내지 A-3, 도 4bb의 서버 지역들 A'-1 내지 A'-3, 또는 도 4bc의 서버 지역들 A''-1 내지 A''-6) 내에서 생성된 모든 서버 지역들의 게임 상호 작용 처리를 조정할 수 있다. 대안적인 구현에서, 지역 A 내의 각 서버 지역에는 한 명 이상의 사용자들에 의해 해당 서버 지역에서 생성된 게임 상호 작용을 처리하기 위한 별개의 컴퓨터가 할당되며, 지역 A 내의 서버 지역들 중 하나에 할당된 컴퓨터는 지역 A의 서버 지역들 내의 게임 상호 작용의 처리를 조정할 수 있다. 조정은 지역 A 내에서 서로 다른 컴퓨터들의 입력들 및 출력들을 관리하고, 지역 A에 할당된 컴퓨터에 조정 컴퓨터의 출력들을 제공하는 것을 포함한다. 앞서 언급된 바와 같이, 가상 위치 내에서 및 가상 위치의 서버 지역 내에서 생성된 지역들의 수와 각 생성된 서버 지역에 할당된 컴퓨터들의 수는 예시들로서 제공되며 제한적인 것으로 간주해서는 안된다.

[0044] 대안적인 구현들에서, 서버 지역들이 가상 위치의 지역의 특정 부분 내에 생성될 수 있거나, 추가 서버 지역들이 게임 플레이 동안 지역 또는 서버 지역의 특정 부분에 액세스하는 사용자들의 수에 기초하여 서버 지역의 특정 부분 내에서 생성될 수 있다. 도 4c는 가상 위치 1 내에서 식별된 지역 A의 일부가 지역 A의 일부 내 사용자들의 밀도에 기초하여 3개의 추가 서버 지역들인, A'''-1 내지 A'''-3으로 각각 더 분할된다. 이 구현에서, 지역 A의 나머지 부분은 제4 서버 지역 A'''-4로 간주된다. 3 개의 서버 지역들인, A'''-1 내지 A'''-3은 같은 크기이며, 제4 서버 지역인, A'''-4는 지역 A의 일부 내에서 생성된 3개의 서버 지역들과 크기가 다르다. 도 4c에 예시된 예에서, 서버 지역들 A'''-1 내지 A'''-3 각각은 A 지역의 한 명 이상의 사용자들을 포함하는 반면, 서버 지역 A'''-4는 지역 A의 어느 사용자를 포함하거나 포함하지 않을 수 있다. 이는 지역 A의 사용자들이 지역 A의 나머지 부분에 사용자들이 없는 동안 서버 지역들이 생성되는 지역 A의 부분에 모두 액세스하는 경우일 수 있다. 게임이 진행됨에 따라, 한 명 이상의 사용자들이 서버 지역들 A'''-1 내지 A'''-3 중 하나 이상의 밖으로 또는 특정 가상 위치의 다른 지역들로부터 또는 다른 가상 위치들로부터 서버 지역 A'''-4로 이동할 수 있다. 그 결과, 서버 지역들을 생성 시, 서버 지역들 A'''-1 내지 A'''-4 각각은, 한 구현에서, 별개의 컴퓨터(CPU 4 내지 7, 각각)가 할당된다. 다른 구현에서, 서버 지역들 A'''-1 내지 A'''-3 각각에는 별개의 컴퓨터들(CPU 4 내지 6)이 할당되는 반면 서버 지역 A'''-4에는 지역 A의 컴퓨터(CPU 3)가 할당된다. 도 4c에 예시된 예는 하나의 컴퓨터가 서버 지역들 A'''-1 내지 A'''-4 각각에 할당되는 것을 도시하고 있지만, 각 서버 지역에서 검출된 처리 부하에 따라 서버 지역들 각각에 할당된 다수의 컴퓨터들이 있을 수 있다.

[0045] 하나 이상의 컴퓨터들이 가상 위치 1의 지역 A의 서버 지역 A 내에서 생성된 서버 지역들 각각에 할당 시, 컴퓨터 지역 관리자(120)는 지역 A 내의 사용자들 각각의 관련 게임 상호 작용을 새로 생성된 서버 지역들에 할당된 컴퓨터들의 각각의 컴퓨터로 마이그레이션한다. 지역들, 서버 지역들을 생성하고 컴퓨터들을 다른 가상 위치들, 지역들 및/또는 서버 지역들에 할당함으로써 가상 위치들을 관리하는 프로세스는 사용자들이 하나 이상의 지역들 또는 가상 위치들에 계속 액세스하고 온라인 게임을 위한 게임 상호 작용에 참여하는 한 지속하여, 게임 상호 작용을 핸들링하기 위해 온라인 게임에 충분한 리소스들이 프로비저닝되도록 한다.

[0046] 게임 콘텍스트 프로세서(111)는 위치 부하 분석기(107)로부터 서로 다른 가상 위치들에 할당된 컴퓨터들의 부하 정보(즉, 게임 상호 작용의 양) 및 컴퓨터 지역 관리자(120)로부터의 가상 위치들 중 하나 이상 내에서 생성된 특정 지역 또는 서버 지역에 액세스하는 사용자들과 관련된 정보를 포함하는 게임 세계에서의 사용자의 움직임을 처리하는 입력들로 수신하고, 상기 정보를 사용하여 사용자를 위한 게임 데이터를 생성하도록 각 사용자의 게임 상호 작용의 처리를 최적화한다. 게임 데이터의 최적화는 각 사용자에 대한 데이터의 중요도에 기초하여

각 사용자의 게임 상호 작용의 다른 부분에 대해 그리고 서로 다른 사용자에게 의해 제공되는 게임 상호 작용의 처리 순서를 결정하고, 그에 따라 각 사용자의 게임 상호 작용을 나타내는 사용자 입력들을 처리하는 것을 포함할 수 있다. 처리 순서는 각 사용자에게 대한 사용자 입력들의 일부 부분들이 먼저 처리되고 나머지 부분들은 나중에 처리되도록 한다. 각 사용자에게 대한 사용자 입력들의 일부 부분들만 즉시 처리되어야 하므로, 게임 상호 작용 처리를 최적화하는 것은 컴퓨터들에 대한 스트레스를 줄이는 데 도움이 된다. 게임 상호 작용 및 사용자 입력들은 온라인 게임의 게임 플레이 동안 사용자들에 의해 제공된 입력들을 참조하기 위해 이 애플리케이션에서 상호 교환적으로 사용된다는 점에 유의해야 한다. 플레이어에 의해 제공되는 입력들은 플레이어에 대한 게임의 게임 상태를 제어하기 위한 것일 수 있으며, 관중에 의해 제공되는 입력들은 관중이 팔로우하기를 원하는 플레이어의 게임 플레이를 제어하기 위한 것일 수 있다.

[0047] 예를 들어, 가상 위치에서 3개의 지역들이 식별되는 가상 위치를 고려할 때, 제1 지역의 제1 사용자 세트는 게임 세계의 오른쪽에서 마주하거나 접근할 수 있으며, 오른쪽에 배치되고 제1 지역에 바로 인접한 제2 지역의 제2 사용자 또는 게임 오브젝트 세트를 보거나/상호 작용할 수 있다. 마찬가지로, 제2 지역의 제2 사용자 세트는 게임 세계의 좌측에서 마주하거나 접근할 수 있으며, 제2 지역의 좌측에 배치된 제1 지역의 제1 사용자 또는 게임 오브젝트 세트를 보거나/상호 작용할 수 있다. 이 시나리오에서, 제1 지역에서 제1 사용자 세트의 게임 상호 작용을 처리하기 위한 처리 시퀀스는 제1 사용자 세트의 오른쪽을 향한 상호 작용들(즉, 제1 서버 지역의 사용자들이 마주하고 있는 방향과 관련된 게임 데이터)의 일부가 먼저 처리되고 제1 사용자 세트의 왼쪽을 향한 상호 작용(즉, 제1 서버 지역의 사용자들이 마주하고 있는 방향과 반대인 방향과 관련된 게임 데이터)이 두 번째 또는 나중에 처리되도록 정의된다. 이는 제1 서버 지역에 있는 제1 사용자 세트가 그들 왼쪽(즉, 그들 뒤에 있는 방향 - 그들이 마주하고 있는 방향과 반대 방향)보다 그들 오른쪽(즉, 그들이 마주하고 있는 방향)에서 일어나는 것에 더 관심이 있기 때문이다. 각 사용자에게 대한 상호 작용 처리의 시퀀스는 해당 사용자의 뷰의 시야에 기초할 수 있다.

[0048] 마찬가지로, 제2 지역에서 제2 사용자 세트에 대한 상호 작용을 처리하기 위한 처리 시퀀스는 제2 사용자 세트의 왼쪽을 향하는 상호 작용들(즉, 사용자들이 마주하는 방향과 관련된 게임 데이터)가 먼저 처리되고 제2 사용자 세트의 오른쪽을 향하는 상호 작용들(즉, 사용자들이 마주하는 방향과 반대인 방향과 관련된 게임 데이터)은 나중에 처리되도록 정의된다. 대안으로, 처리 시퀀스는 상호 작용들과 관련된 타이밍에 기초할 수 있다. 예를 들어, 지역 A의 제1 사용자가 지역 B의 제2 사용자를 공격하고 그 결과 제2 사용자가 부상을 입는다. 이에 응답하여, 지역 B의 제3 사용자가 지역 A의 제1 또는 제4 사용자를 공격하고 그 결과 제1 또는 제4 사용자가 부상을 입는다. 게임 콘텍스트 프로세서(111)는 각 사용자의 활동 타이밍을 이용하고 이에 따라 온라인 게임의 게임 상태에 도달하기 위해 서로 다른 사용자들의 상호 작용을 처리함으로써 어떤 상호 작용이 먼저 처리되어야 하는지를 결정하여, 사용자들(제1, 제2, 제3 및 제4 사용자들) 각각에게 사용자에게 적합한 게임 데이터가 제시될 수 있도록 한다. 컴퓨터 지역 관리자는 서로 다른 가상 위치들 및 서로 다른 지역들, 하나 이상의 가상 위치들 내의 서버 지역들에 위치한 사용자들의 입력들(상호 작용들)을 관리하고, 서로 다른 위치들, 하나 이상의 가상 위치들 내의 지역들 및 서버 지역들 간에 게임 상태들을 공유한다. 컴퓨터 지역 관리자(120)로부터의 정보는 사용자들을 가상 위치 내의 다른 서버 지역들 및/또는 다른 가상 위치들에 할당하는 데 사용된다. 컴퓨터 지역 관리자에 의해 생성된 각 사용자에게 대한 게임 상태를 갖는 게임 데이터는 렌더링을 위해 사용자들의 각각의 클라이언트 장치들로 포워딩된다.

[0049] 도 3a는 한 구현에서, 온라인 게임에 대한 리소스들을 프로비저닝하는 데 사용되는 컴포넌트들 중 일부를 예시한다. 분산형 게임 엔진(100)은 복수의 사용자들의 사용자 입력들을 처리하여 각 사용자에게 대한 게임 데이터를 생성하고, 게임 세계 내에서 사용자들의 가상 위치를 식별하고, 사용자들이 주어진 시간에 특정 가상 위치 내의 지역에 액세스하는 시기를 검출하고, 지역에 있는 사용자들의 게임 상호 작용이 적시에 처리될 수 있도록 지역에 컴퓨터 리소스들을 응답식으로 프로비저닝하는 데 사용되는 복수의 컴포넌트들을 포함한다. 복수의 사용자들이 게임 플레이를 위해 게임 클라우드 시스템에서 실행되는 MMO 게임과 같은 온라인 게임에 액세스하는 경우, 분산형 게임 엔진(100)은 사용자 위치 추적기(105)를 이용하여 사용자들이 온라인 게임을 진행함에 따라 사용자들을 팔로우한다. 게임 상호 작용을 통해 제공되는 사용자의 입력들은 각 사용자에게 대한 게임 데이터를 생성하는 데 사용된다. 온라인 게임의 가상 위치는 특정 레벨의 온라인 게임에 대한 게임 오브젝트들 및 도전 과제들을 포함하는 게임 세계 내의 지역으로 정의될 수 있다. 예를 들어, 은하계간 우주 게임에서, 가상 위치는 행성, 별, 소행성, 우주선 등과 같은 천문학적 오브젝트에 의해 덮인 영역 또는 사용자들에 액세스 가능한 임의의 두 개의 천문학적 오브젝트들 사이의 공간과 같은 은하계 내의 영역으로 정의될 수 있다. 마찬가지로, 세계를 망라하는 국가 간 전쟁 게임에서, 가상 위치는 국가 또는 대륙으로 정의될 수 있으며, 지역은 국가 내의 주 또는 시

티 또는 타운 또는 산 또는 강 또는 댐 등으로 정의될 수 있다.

[0050] 사용자들이 게임 플레이를 위한 온라인 게임을 선택하면, 사용자들에게 온라인 게임의 게임 세계 내에서 하나 이상의 가상 위치들에 대한 액세스가 제공된다. 사용자 위치 추적기(105)는 게임 플레이에서 각 사용자의 현재 가상 위치를 식별하는 데 사용된다. 사용자 위치 추적기(105)에 의해 제공된 정보는 다수의 사용자들이 특정 가상 위치 내의 지역에 액세스하는 시기를 결정하는 데 사용될 수 있다. 복수의 사용자들은 게임 플레이의 도전 과제를 극복하려고 시도하면서 주어진 시간에 특정 가상 위치의 지역에 모일 수 있다. 지역에 있는 사용자들의 수는 상기 지역의 사용자 입력들(즉, 게임 상호 작용들)의 증가를 야기할 수 있으며, 그 결과, 상기 지역에서 생성되는 게임 데이터의 증가를 야기할 수 있다.

[0051] 게임 세계 내의 각 가상 위치에 있는 사용자들에게는 게임 상호 작용들을 처리하고 각 사용자에 대한 게임 데이터를 생성하기 위해 하나 이상의 컴퓨터들이 할당된다. 일부 구현들에서, 각 컴퓨터는 핸들링할 수 있는 사용자들의 수에 대한 미리 정의된 한도를 가질 수 있다. 위치 부하 분석기(107)는 가상 위치의 지역에 액세스하는 사용자들의 수가 가상 위치에 할당된 컴퓨터에 의해 핸들링될 수 있는지 또는 컴퓨터에 정의된 미리 정의된 한도를 초과하는지를 결정하는 데 사용된다. 사용자들의 수가 컴퓨터의 용량을 초과할 경우, 위치 부하 분석기(107)는 가상 위치에서 게임 상호 작용을 처리하기 위해 기존 컴퓨터를 지원하는 추가 컴퓨터를 프로비저닝하도록 컴퓨터 할당기(109)에 신호를 보낸다. 컴퓨터 할당기(109)는 기존 컴퓨터의 컴퓨팅 리소스들을 증가시키는 것으로 게임 상호 작용의 증가가 핸들링될 수 있는지 또는 새로운 컴퓨터들이 가상 위치에 할당되어야 하는지를 결정한다. 결정에 기초하여, 컴퓨터 할당기는 기존 컴퓨터에 추가 리소스를 프로비저닝하거나 가상 위치에 추가 컴퓨터들을 제공하도록 동적 지역 생성 엔진(121)에 신호를 보낸다.

[0052] 동적 지역 생성 엔진(121)은 플레이되는 게임의 유형(예를 들어, 그래픽 인텐스 또는 컴퓨터 인텐스), 게임 세계의 특정 가상 위치 내의 지역을 액세스하는 사용자들의 유형(예를 들어, 하이 레벨 플레이어들, 중간 레벨 플레이어, 초보 레벨 플레이어들 또는 관중들)을 분석하고, 온라인 게임에 할당된 기존 컴퓨터가 가상 위치의 지역에 액세스하는 사용자들의 수를 핸들링하기에 충분한지를 결정한다. 온라인 게임에 할당된 각 컴퓨터는 핸들링할 수 있는 사용자들의 수에 대한 미리 정의된 한도가 있을 수 있으며, 이 미리 정의된 한도는 컴퓨터에서 이용 가능한 리소스들의 유형과 양에 기초할 수 있다. 온라인 게임을 플레이하고 있는 복수의 사용자들 각각에게 게임 데이터를 제공하는 것을 돕기 위해, 한 명 이상의 사용자들이 존재하는 게임 세계의 각 가상 위치에 하나 이상의 컴퓨터들이 제공될 수 있다. 분석에 기초하여, 특정 가상 위치 내의 지역 내의 사용자들의 수가 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는 경우, 동적 지역 생성 엔진(121)은 특정 가상 위치의 지역을 복수의 서브 지역들로 분할할 수 있다. 각 서브 지역은 상기 지역의 적어도 한 명의 사용자가 존재하는 경계에 의해 정의된다. 그런 다음, 동적 지역 생성 엔진(121)은 컴퓨터 할당기(109)와 상호 작용하여 지역의 게임 상호 작용의 처리를 돕기 위해 지역에 추가 컴퓨터를 할당한다. 일부 구현들에서, 각 서브 지역에는 서브 지역에서 사용자(들)의 게임 상호 작용을 처리하고 해당 서브 지역에서 각 사용자에 대한 게임 데이터를 생성하기 위해 별개의 하나 이상의 컴퓨터들이 할당된다. 각 사용자에 대한 게임 데이터는 사용자에게 의해 제공된 상호 작용들에 기초한, 사용자를 위한 온라인 게임의 게임 상태를 포함한다. 서브 지역들에 할당된 컴퓨터들에 의해 생성된 게임 데이터는 서브 지역들의 컴퓨터들 중 하나 또는 지역이나 특정 가상 위치에 할당된 컴퓨터에 의해 어셈블링된다. 어셈블링된 게임 데이터는 온라인 게임의 게임 상태를 생성하는 데 사용된다. 게임 상태 조정기(121a)는 동적 지역 생성 엔진(121)에 의해 다른 컴퓨터들의 게임 데이터를 조정하고 온라인 게임을 위한 게임 상태를 생성하는 데 사용된다.

[0053] 게임 상태 정보 및 사용자 위치 정보는 동적 지역 생성 엔진(121)에 의해 사용자 지역 할당 관리자(125)로 포워딩된다. 사용자 지역 할당 관리자(125)는 온라인 게임을 위한 사용자들의 현재 위치를 로컬에 저장하고 사용자 데이터 마이그레이션 에이전트(127)와 위치 정보를 공유한다. 사용자 데이터 마이그레이션 에이전트(127)는 사용자들 각각의 위치 정보를 사용하여 각 사용자에게 제공되어야 할 게임 데이터의 시퀀스를 결정하고 이에 따라 게임 데이터의 프레임들을 각각의 사용자의 클라이언트 장치들로 포워딩한다. 게임 데이터의 시퀀스는 게임 콘텍스트 프로세서(111)에 의해 결정된다.

[0054] 각 가상 위치 내의 사용자의 입력들은 온라인 게임의 현재 콘텍스트 및 각 사용자에 대한 게임 데이터의 처리 시퀀스를 결정함으로써 게임 콘텍스트 프로세서(111)에 의해 처리된다. 예를 들어, 각 사용자에 대한 처리 시퀀스는 각 사용자 뷰의 시야, 게임 데이터의 서로 다른 부분들의 중요도, 게임 데이터와 관련된 타이밍 등에 기초하여 데이터 우선 순위 선정기(111a)에 의해 결정될 수 있다. 데이터 우선 순위 선정기(111a)는 게임 플레이 중 각 사용자가 마주하는 방향 및/또는 온라인 게임 내에서 사용자가 진행하는 방향을 검토하여 각 사용자 뷰의 시야를 결정할 수 있다. 사용자가 게임 세계의 왼쪽에서 오른쪽으로 진행하는 경우, 상기 뷰의 시야는 사용자가

게임 세계에서 마주하는 쪽(즉, 오른쪽)이 될 것이다. 그 결과, 사용자의 오른쪽에 있는 게임 세계와 관련된 게임 데이터는 사용자가 오른쪽을 마주하고 있는 시간 동안 사용자 뒤에 있는 게임 세계(즉, 게임 세계의 왼쪽)보다 더 높은 우선 순위가 부여될 것이다. 오른쪽의 도전 과제를 극복한 후, 사용자가 게임 세계의 왼쪽의 도전 과제에 도전하기 위해 뛰어다니면, 사용자 뷰의 시야에 있는 게임 세계의 왼쪽에는 사용자가 왼쪽을 마주하고 있는 시간 동안 더 높은 우선순위가 부여되고, 동시에 사용자 뒤에 있는 게임 세계의 오른쪽에는 낮은 우선 순위가 부여된다. 게임 콘텍스트 프로세서(111)는 데이터 우선순위 선정기(111a)에 의해 제공된 처리 시퀀스 정보를 사용하여 각 사용자의 게임 상호 작용을 처리한다. 온라인 게임이 임의의 주어진 시간에 복수의 플레이어들(예를 들어, 수백 또는 수천 명의 플레이어들)로부터 입력들을 수신하므로, 각 사용자의 게임 상호 작용에 우선 순위를 부여하고 게임 상호 작용의 일부를 처리하는 것은 가상 위치에 할당된 각 컴퓨터에 의해 생성되는 게임 데이터의 양을 감소시킬 것이며, 이에 따라 할당된 처리 리소스들을 효율적으로 사용하는 동시에 사용자들에게 적절한 게임 데이터가 적절한 방식으로 제시되어 게임 플레이 경험을 풍부하게 한다.

[0055] 각 사용자의 처리된 게임 데이터는 사용자 데이터 마이그레이션 에이전트(127)로 포워딩되어 각 사용자에게 대한 게임 데이터를 어셈블링하고, 게임 데이터를 인코딩하며, 게임 데이터의 프레임들을 사용자들의 각각의 클라이언트 장치로 스트리밍한다. 각 사용자에게 대한 게임 데이터는 사용자 마이그레이션 에이전트(127)에서 수신되고, 각 사용자에게 대해 결정된 처리 시퀀스에 따라 어셈블링 및 전송된다. 처리된 게임 데이터는 또한 온라인 게임의 사용자 메트릭들 및 게임 데이터 메트릭들을 결정하기 위해 데이터 저장소(260)에 저장되며, 이는 사용자를 분류하고 온라인 게임에 할당되어야 하는 컴퓨터들의 유형을 결정하는 데 사용될 수 있다. 사용자들이 온라인 게임과 계속 상호 작용함에 따라, 더 많은 사용자들이 특정 가상 위치 내의 지역으로 이동하거나 기존 사용자들이 특정 가상 위치의 지역 밖으로 이동할 수 있다. 동적 지역 생성 엔진(121) 및 게임 콘텍스트 프로세서(111)는 각 가상 위치 및 주어진 가상 위치 내의 각 서브 지역에서 사용자들의 게임 상호 작용을 처리하도록 할당된 컴퓨터들이 온라인 게임의 처리 부하를 핸들링하기에 충분한 리소스들을 갖도록 함께 작동한다.

[0056] 도 3b는 온라인 게임의 게임 세계 내에서 사용자들의 가상 위치들을 추적하고, 게임 세계 내의 가상 위치들에 할당된 컴퓨터들을 관리하고, 대안 구현에서, 사용자들의 게임 데이터를 다른 컴퓨터들로 마이그레이션하는 데 사용되는 분산형 게임 엔진(100)의 컴퓨터 지역 관리자(120)의 다양한 컴포넌트들을 예시한다. 이 구현에서, 동적 지역 생성 엔진(121)은 컴퓨터 지역 관리자(120) 내의 추가 컴포넌트들과 상호 작용하여 서브 지역들을 생성함으로써 그리고 온라인 게임의 서로 다른 가상 위치들에 대해 컴퓨터들/컴퓨팅 리소스들을 스핀 업 또는 스핀 다운하는 신호들을 생성함으로써 온라인 게임 내에서 사용자들의 게임 상호 작용의 처리를 조정할 수 있다. 컴퓨터 지역 관리자(120) 내의 컴포넌트들 중 일부는 몇 가지를 예를 들면, 동적 지역 생성 엔진(121), 지역 경계 조정기(123), 사용자 지역 할당 관리자(125), 사용자 데이터 마이그레이션 에이전트(127), 데이터 동기화기(129) 및 스핀 업/스핀 다운 디렉터(131)를 포함한다.

[0057] 분산형 게임 엔진은 복수의 사용자들로부터 수신된 게임 상호 작용을 처리하는 MMO 게임(즉, 온라인 게임)의 게임 로직을 실행하고 온라인 게임에 대한 게임 데이터를 생성한다. 온라인 게임은 다양한 지리적 위치들에 걸쳐 분산된 복수의 사용자들에 의해 액세스되므로, 분산형 게임 엔진 및 게임 로직의 인스턴스들은 게임 클라우드 시스템 내의 복수의 서버들에서 실행되고, 인터넷과 같은 네트워크(200)를 통해 사용자들에 의해 액세스된다. 분산형 게임 엔진 내의 사용자 위치 추적기(105)는 온라인 게임의 게임 세계의 가상 위치들로 사용자들의 움직임을 모니터링하는 데 사용된다. 사용자들은 게임 오브젝트 또는 아바타를 사용하여 게임 세계의 가상 위치들 내에서 식별될 수 있다. 서로 다른 레벨의 게임 플레이에 대해 서로 다른 가상 위치들이 정의될 수 있으며, 게임 플레이의 레벨과 온라인 게임 내 진행 상황에 기초하여 서로 다른 가상 위치들에서 사용자들이 식별될 수 있다. 사용자 위치 추적기(105)는 게임 플레이 동안 생성된 게임 데이터를 분석하여 온라인 게임의 사용자들 각각의 가상 위치를 결정하고 다수의 사용자들이 게임 세계의 하나의 가상 위치 내의 지역에 액세스하는 시기를 검출할 수 있다.

[0058] 특정 가상 위치의 지역에서 사용자들의 수를 검출 시, 분산형 게임 엔진(100)의 위치 부하 분석기(107)는 상기 지역의 사용자들의 수가 게임 세계에서 식별된 가상 위치 내에서 생성된 상호 작용들을 처리하기 위해 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는지를 결정하는 데 사용된다. 각 사용자가 게임을 플레이 동안 게임 세계 내에서 이동함에 따라, 특정 가상 위치의 지역에 액세스하는 사용자들의 수는 달라질 수 있다. 그 결과, 위치 부하 분석기(107)는 사용자들의 수가 특정 가상 위치에 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는지를 결정하기 위해 특정 가상 위치의 지역에 존재하는 사용자들의 수를 지속적으로 모니터링한다.

[0059] 위치 부하 분석기(107) 및 사용자 위치 추적기(105)로부터의 정보는 특정 가상 위치에서 게임 데이터 처리 부하를 핸들링하는 데 필요한 컴퓨터 량을 결정하기 위해 컴퓨터 할당기(109)에 의해 사용된다. 컴퓨터 할당기(10

9)는 또한 게임 세계에서 식별된 상이한 가상 위치들에서 생성된 게임 데이터를 처리하기 위해 온라인 게임에 이미 할당된 컴퓨터들의 유형 및 양을 결정할 수 있다. 이미 할당된 컴퓨터의 양과 온라인 게임에 필요한 컴퓨터의 양, 및 게임 플레이 동안 주어진 시간에 지역에 액세스하는 사용자들의 수와 관련된 정보를 사용하여, 컴퓨터 할당기는 온라인 게임에 할당된 컴퓨터들이 가상 위치의 지역 및 이에 따른 온라인 게임에 대한 상호 작용들을 핸들링하기에 충분한지, 또는 온라인 게임에 할당된 컴퓨터들에 대한 임의의 조정들이 이루어져야 하는지를 결정할 수 있다. 예를 들어, 현재 할당된 컴퓨터들은 특정 가상 위치의 지역에 모인 다수의 사용자들의 상호 작용들을 핸들링하기에 충분하지 않을 수 있으며, 특정 가상 위치에 있는 다수의 사용자들의 상호 작용들을 핸들링하기 위해 추가 컴퓨터들이 할당되어야 할 수 있다. 이런 상황들에서, 위치 부하 분석기(107)는 특정 가상 위치에 추가 컴퓨터들을 할당하기 위해 컴퓨터 지역 관리자(120) 내의 동적 지역 생성 엔진(121)에 신호를 전송할 수 있다.

[0060] 대안으로, 온라인 게임의 특정 가상 위치에 현재 할당된 컴퓨터들은 지역에 모인 사용자들의 수를 핸들링하는데 필요한 수를 초과할 수 있다. 그 결과, 가상 위치에 할당된 컴퓨터들 중 일부는 할당 해제되어야 할 수 있으며, 사용자들의 상호 작용은 온라인 게임에 할당되는 다른 컴퓨터들로 마이그레이션된다. 위치 부하 분석기(107)는 특정 가상 위치에 추가 컴퓨터들을 할당하기 위해 컴퓨터 지역 관리자(120) 내의 동적 지역 생성 엔진(121)에 신호를 전송할 수 있다. 사용자 위치 추적기(105)로부터의 사용자 위치 정보, 컴퓨터 할당기(109)에 의해 제공된 컴퓨터 정보 및 위치 부하 분석기(107)에 의해 제공된 위치 부하 정보는 동적 지역 생성 엔진(121)과 공유된다.

[0061] 동적 지역 생성 엔진(121)은 위치 부하 분석기(107)로부터의 신호를 사용하여 온라인 게임의 특정 가상 위치에 할당된 컴퓨터들을 조정한다. 온라인 게임은 복수의 플레이어들(즉, 사용자들)이 온라인 게임에 액세스하여 플레이하는 MMO 게임이므로, 플레이어들 각각은 자신의 속도로 온라인 게임을 진행할 수 있다. 글 결과, 플레이어들 각각은 온라인 게임의 서로 다른 가상 위치들에 있을 수 있다. 결과적으로, 서로 다른 컴퓨터들이 온라인 게임에 할당되어 서로 다른 컴퓨터들이 해당 가상 위치에서 사용자들의 상호 작용들을 처리하도록 한다. 다수의 사용자들이 특정 가상 위치 내의 특정 지역에 액세스하면, 할당된 컴퓨터가 핸들링할 수 있거나 핸들링할 수 없는 지역의 상호 작용들의 증가가 있을 것이다. 현재 할당된 컴퓨터가 특정 가상 위치의 지역에 있는 다수의 사용자들을 핸들링할 수 없는 경우, 동적 지역 생성 엔진(121)은 상기 지역을 각 서버 지역이 상기 지역의 적어도 한 명의 사용자를 갖는 복수의 서버 지역들로 분할한다. 각 서버 지역은 서버 지역의 사용자들이 존재하는 경계를 갖도록 정의된다. 그런 다음, 동적 지역 생성 엔진은 스핀 업/스핀 다운 디렉터(131)와 상호 작용하여 지역에 추가 컴퓨터를 할당한다. 일부 구현들에서, 스핀 업/스핀 다운 디렉터(131)는 마이크로 레벨 또는 매크로 레벨에서 추가 컴퓨터를 할당할 수 있다.

[0062] 사용자들은 플레이어들 또는 관중들일 수 있다. 플레이어들은 초보, 중간 레벨 또는 전문가 카테고리 더 분류될 수 있다. 마찬가지로, 관중들은 온라인 게임의 일반적인 게임 플레이를 팔로우하는 관중들 또는 플레이어들(전문가, 중간 레벨 또는 초보)의 유형들 중 하나를 팔로우하는 관중들로 분류될 수 있다. 동적 지역 생성 엔진(121)은 특정 가상 위치 내의 지역에 모인 사용자들의 수가 특정 가상 위치에 할당된 컴퓨터에 대해 미리 정의된 한도 내에 있는지를 결정할 수 있다. 사용자들의 수가 임계 퍼센트(예를 들어, 5% 또는 10%) 내에는 미리 정의된 한도를 초과하는 경우, 동적 지역 생성 엔진(121)은 스핀 업/스핀 다운 디렉터(131)에 신호를 보내 마이크로 레벨에서 컴퓨터를 조정한다. 그러나, 특정 가상 위치의 지역에 있는 사용자들의 수가 임계 퍼센트보다 크게 컴퓨터에 대해 미리 정의된 한도를 초과하는 경우, 동적 지역 생성 엔진(121)은 스핀 업/스핀 다운 디렉터(131)에 신호를 보내 매크로 레벨에서 컴퓨터들을 조정할 수 있다. 동적 지역 생성 엔진(121)은 온라인 게임의 특정 가상 위치에 할당된 컴퓨터를 조정할 때 지역에 모인 사용자들의 유형 및 각 유형의 사용자들의 수를 결정할 수 있음에 유의해야 한다.

[0063] 하나의 구현에서, 마이크로 레벨 조정은 추가 CPU 코어들, GPU가 공유되는 경우 GPU 대역폭들, 메모리 또는 기타 컴퓨팅 리소스 컴포넌트들을 기존 컴퓨터들에 추가하여 추가 컴퓨팅 리소스들이 특정 가상 위치의 게임 데이터를 핸들링하는 것을 돕도록 하는 것을 포함할 수 있다. 상술한 마이크로 레벨 조정은 일 예이며, 다른 형태의 마이크로 레벨 조정이 수행될 수도 있다. 일 구현에서, 매크로 레벨 조정은 그 안에서 생성된 게임 데이터의 처리를 돕기 위해 특정 가상 위치에 추가 서버 또는 서버 클러스터를 추가하는 것을 포함할 수 있다. 추가 서버는 물리적 서버, 호스트의 컴퓨팅 리소스들을 사용하여 실행하는 가상 서버, 컨테이너 서버 또는 이들의 임의의 조합들일 수 있다. 상술한 매크로 레벨 조정은 일 예이며, 다른 형태의 매크로 레벨 조정이 수행될 수도 있다.

[0064] 서버 지역들이 생성되면, 서버 지역들을 정의하는 데 사용되는 가상 서버 경계들과 같은 서버 지역들과 관련된

정보는 지역 경계 조정기(123)와 공유된다. 지역 경계 조정기(123)는 특정 가상 위치의 지역에서 생성된 각각의 서버 지역들 내의 사용자들을 식별하고 관리하기 위해 경계 정의 로직(123a) 및 지역 스티치(123b)를 체결한다. 경계 정의 로직(123a)은 지역 내에서 생성된 서버 지역들을 정의하는 경계들, 가상 위치들의 지역들의 경계들 및 게임 세계 내에서 식별된 가상 위치들의 경계들을 사용하여 한 명 이상의 사용자들이 지역 스티치(123b)와 상호작용함으로써 두 개의 연속적인 서버 지역들 또는 두 개의 연속적인 지역들 또는 두 개의 연속적인 가상 위치들로 구분되는 경계에 존재하는지를 결정한다. 분리된 경계에 존재하는지 여부를 결정한다. 지역 스티치(123b)는 서버 지역들 내의 각 사용자의 게임 상태들, 지역들 및 가상 위치들을 함께 조정함으로써 각 지역 내의 서버 지역들, 각 가상 위치 내의 지역들, 게임 세계 내의 가상 위치들을 스티칭하여 온라인 게임의 게임 세계의 원활한 가상 세계관을 제공한다. 게임 세계의 완벽한 가상 세계관으로부터의 정보는 지역 경계 조정기(123)에 의해 임의의 용자가 특정 가상 위치 내에서 생성된 임의의 두 서버 지역들 사이의 경계에 위치되는지 또는 가상 위치 내의 두 개의 연속적으로 정의된 지역들 사이의 경계에 위치되는지를 결정하는 데 사용된다. 예를 들어, 서버 지역들을 생성하는 동안, 동적 지역 생성 엔진(121)은 지역 내의 대부분의 사용자들이 각 서버 지역의 경계 내에 포함되도록 특정 가상 위치 내의 지역을 세분화하려고 시도한다. 그러나, 사용자들 중 한 명 이상은 연속적으로 정의된 두 개의 서버 지역들을 구분하는 경계에 속할 수 있다. 결과적으로, 지역 경계 조정기(123)는 임의의 두 개의 연속적인 서버 지역들 사이의 경계에 속하는 사용자들을 식별하고, 각각의 서버 지역들에 할당되는 컴퓨터들 사이에서 해당 사용자들에 대한 게임 상태 동기화를 조정하여, 경계에 있는 사용자들에게 온라인 게임의 적절한 게임 상태가 제공되도록 한다.

[0065] 지역 경계 조정기(123)에 의해 제공된 가상 세계관으로부터의 정보는 사용자 지역 할당 관리자(125)에 의해 사용되어 사용자들 각각이 속한 서버 지역들 및/또는 가상 위치들을 식별하고 게임 플레이 동안 사용을 위한 이 정보를 로컬로 저장한다. 사용자들이 게임 세계를 돌아다닐 때, 사용자들의 위치는 지속적으로 변경되며 각 사용자와 관련된 변경 사항들은 변경이 검출될 때 지역 할당 관리자(125)에 의해 업데이트된다. 사용자의 현재 위치는 또한 사용자 데이터 마이그레이션 에이전트(127) 및 스핀 업/스핀 다운 디렉터(131)와 사용자 지역 할당 관리자(125)에 의해 공유된다. 스핀 업/스핀 다운 디렉터(131)는 게임 세계 내의 각 사용자의 현재 위치를 획득하고, 컴퓨터 할당기(109)에 쿼리하여 해당 가상 위치/서버 지역에 할당된 컴퓨터들을 식별하고, 컴퓨터 할당기(109)에 신호를 보내 각 서버 지역에 대한 추가 컴퓨터를 스핀 업한다. 특정 가상 위치들에 프로비저닝된 추가 컴퓨터는 이전에 프로비저닝된 컴퓨터와 함께 사용되어 각 서버 지역 내의 사용자들에 의해 생성된 게임 상호작용들을 처리한다.

[0066] 다른 구현들에서, 서버 지역들 및/또는 가상 위치들 중 일부는 이들을 함께 병합함으로써 재설정되어 정제된 서버 지역/가상 위치를 형성할 수 있다. 예를 들어, 게임 세계의 특정 가상 위치 내의 지역은 서버 지역에 있는 사용자들에 의해 생성된 게임 상호 작용을 처리하기 위해 별개의 컴퓨터가 할당된 각 서버 지역과 함께 복수의 서버 지역들로 세분화되었을 수 있다. 게임 플레이 동안, 제1 서버 지역 내에 있던 한 명 이상의 사용자들이 제1 서버 지역에서 제2 서버 지역으로 이동하거나 다른 지역이나 다른 가상 위치로 이동할 때, 제1 서버 지역에 할당된 컴퓨터는 활용도가 낮거나 유휴 상태일 수 있다. 온라인 게임에 할당된 컴퓨팅 리소스들을 최적화하기 위해, 하나 이상의 서버 지역들에 할당된 컴퓨터들이 특정 가상 위치의 하나 이상의 서버 지역들을 재설정함으로써 재할당될 수 있다. 하나 이상의 서버 지역들의 재설정은 결합된 서버 지역을 생성하기 위해 제1 서버 지역을 제2 서버 지역과 병합하는 것을 포함할 수 있으며, 제1 서버 지역에 할당된 컴퓨터는 결합된 서버 지역에 할당되고, 제2 서버 지역에 할당된 컴퓨터는 일부 다른 서버 지역, 가상 위치 또는 다른 온라인 게임에 재할당된다. 마찬가지로, 특정 가상 위치 내의 지역이 게임 플레이 동안 서버 지역들 중 일부만이 사용자들을 갖는 복수의 서버 지역들로 세분화되면, 사용자들을 포함하는 서버 지역들 중 하나에 할당된 컴퓨터가 어떠한 사용자들도 포함하지 않는 서버 지역들에도 할당되도록 컴퓨터들이 할당된다. 예를 들어, 도 4c에 도시된 예에서, 한 명 이상의 사용자들을 포함하는 처음 세 개의 서버 지역들 각각에는 별개의 컴퓨터가 할당되고 어떠한 사용자들도 포함하지 않는 네 번째 서버 지역에는 가상 위치 또는 서버 지역들 중 하나의 컴퓨터가 할당된다.

[0067] 제1 서버 지역의 사용자들 중 한 명 이상이 제1 서버 지역에서 제2 서버 지역으로 이동할 때, 제2 서버 지역에 할당된 컴퓨터는 제2 서버 지역으로 이동하는 한 명 이상의 사용자들로 인해 발생하는 초과 처리 부하를 핸들링하도록 할당될 수 있다. 제1 서버 지역으로부터의 사용자들의 이동으로 인해, 제1 컴퓨터가 유휴 상태가 되거나 활동도가 낮아질 수 있다. 그 결과, 제1 컴퓨터는 제1 서버 지역을 제2 서버 지역과 병합하고 제2 컴퓨터가 결합된 서버 지역에서 게임 상호 작용들로 인한 처리 부하를 핸들링하도록 함으로써 다른 가상 위치 또는 온라인 게임에 재할당될 수 있다. 스핀 업/스핀 다운 디렉터(131)는 각 컴퓨터가 핸들링할 수 있는 처리 부하, 컴퓨터의 컴퓨팅 리소스들 각각이 확장될 수 있는 정도, 및 활성 상태로 사용해야 하는 컴퓨터/컴퓨팅 리소스 및 감소되거나 비활성 상태로 만들어야 하는 컴퓨터/컴퓨팅 리소스를 결정하는 데 사용된다. 일부 구현들에서, 스핀 업

/스핀 다운 디렉터(131)는 비용 또는 수요에 기초하여 온라인 게임에 할당된 컴퓨터들을 추가하거나 감소시키도록 신호를 보낼 수 있다. 예를 들어, 스핀 업/스핀 다운 디렉터(131)는 전력 관리자 스케줄러(133)와 상호 작용하여 어떤 컴퓨터들을 활성 상태로 유지할 것인지, 그리고 어떤 컴퓨터를 비활성화 상태로 만들 것인지를 결정할 수 있으며, 이러한 결정은 전력 비용에 기초하여 이루어질 수 있다. 예를 들어, 특정 시간에 전력 비용이 높으면, 전력 관리자 스케줄러(133)는 스핀 업/스핀 다운 디렉터(131)에 신호를 보내 가상 위치 또는 서버 지역에 할당된 컴퓨터(들)를 유지하고 가상 위치 또는 서버 지역의 처리 부하를 핸들링하기 위해 추가 컴퓨터들을 할당하는 대신 할당된 컴퓨터(들)에 더 많은 스트레스를 줄 수 있다. 마찬가지로, 다른 시간에 전력 비용이 낮다면, 전력 관리자 스케줄러(133)는 스핀 업/스핀 다운 디렉터(131)에 신호를 보내 추가 컴퓨터들을 추가하여 가상 위치 또는 서버 지역의 처리 부하를 핸들링하기 위해 더 많은 처리 능력을 제공할 수 있다. 따라서, 전력 관리자 스케줄러(133)는 온라인 게임의 게임 세계의 다른 가상 위치에 제공되는 컴퓨터들이 최적으로 프로비저닝되고 활용되도록 하는 데 사용되어 할당된 리소스들이 처리 부하를 효율적으로 핸들링할 수 있는 동시에 비용 효율적이 되도록 한다.

[0068] 컴퓨터들이 서로 다른 서버 지역들 및/또는 가상 위치에 할당되면, 사용자 데이터 마이그레이션 에이전트(127)는 서로 다른 사용자들과 관련된 관련 게임 상호 작용들을 각각의 서버 지역들/가상 위치들에 할당된 컴퓨터들 중 대응하는 컴퓨터들로 마이그레이션하는 데 사용된다. 다른 사용자들의 게임 상호 작용들을 마이그레이션하는 것은 사용자의 온라인 게임의 게임 상태를 사용자가 존재하는 서버 지역 또는 가상 위치에 할당된 해당 컴퓨터로 마이그레이션하는 것을 포함한다. 온라인 게임의 게임 상태를 마이그레이션하는 것은 물리 상태, 렌더링 상태, 게임 세계 내에 연결된 모든 손상 정보 및 한 명 이상의 사용자들의 온라인 게임에 대한 임의의 기타 게임 플레이 데이터를 한 컴퓨터에서 다른 컴퓨터로 마이그레이션하는 것을 포함한다. 예를 들어, 특정 가상 위치의 지역 내에 서버 지역들이 생성된 경우, 각각의 서버 지역들에 있는 사용자들의 게임 상태 및 게임 상호 작용들은 지역 또는 특정 가상 위치에 할당된 컴퓨터에서 다른 서버 지역들에 할당된 대응하는 컴퓨터들로 마이그레이션되어야 한다. 게임 상호 작용들은 각 부분에 부여된 우선 순위에 기초하여 상호 작용들의 일부를 마이그레이션함으로써 적절한 컴퓨터들로 마이그레이션될 수 있으며, 부분에 부여된 우선 순위는 상호 작용들과 관련된 각 사용자에게 대한 타이밍, 중요도, 인계도, 필요성, 시야 등에 기초할 수 있다. 예를 들어, 게임 세계가 사용자와 게임 오브젝트의 상호 작용으로 인한 게임 오브젝트의 큰 폭발을 포함하는 경우, 폭발과 관련된 사용자의 상호 작용들의 일부가 먼저 마이그레이션되어 사용자의 상호 작용이 먼저 처리되고 폭발과 관련된 게임 데이터가 사용자에게 제시되도록 한다. 사용자의 다른 상호 작용들은 다른 상호 작용들에 부여된 우선 순위에 따라 나중에 마이그레이션될 수 있다. 각 사용자의 서로 다른 게임 상호 작용들에 부여된 처리 우선 순위와 관련된 정보는 게임 콘텍스트 프로세서(111)에 의해 제공될 수 있다. 처리 우선 순위는 컴퓨팅 리소스들이 효율적으로 사용되는 동시에 각 사용자에게 적절한 게임 데이터를 적시에 제시되도록 컴퓨터들에 의한 데이터 처리를 최적화하는 데 사용된다.

[0069] 게임 콘텍스트 프로세서(111)는 지역 경계 조정기(123)로부터 온라인 게임의 게임 세계에서 정의된 사용자들 및 가상 위치들의 위치 정보를 수신하고 각 사용자의 상호 작용들을 처리한다. 일부 구현들에서, 게임 콘텍스트 프로세서(111)는 데이터 우선 순위 선정기(111a)를 사용하여 각 사용자의 상호 작용들의 처리에 우선 순위를 부여한다. 데이터 우선 순위 선정기(111a)는 사용자에게 제공되는 게임 데이터가 사용자와 관련된 순서대로 제시되는 게임 콘텐츠를 포함하도록 상대적 콘텍스트에 기초하여 게임 세계와 관련된 상호 작용들을 처리할 수 있다. 예를 들어, 상기 예의 게임 오브젝트 폭발과 같이 사용자 뷰의 시야 내에서 게임 세계와 관련된 게임 데이터의 일부가 먼저 제공될 수 있으며, 사용자 뷰의 시야 밖에 있는 다른 게임 콘텐츠는 나중에 제공된다. 게임 콘텍스트 프로세서(111)는 사용자 데이터 마이그레이션 에이전트(127)가 각 사용자의 각각의 클라이언트 장치에 대한 관련성 순서대로 게임 데이터의 다른 부분들을 인코딩하고 포워딩할 수 있도록 각 사용자와 관련된 게임 데이터의 서로 다른 부분을 관련성 순서대로 처리하고 이를 사용자 데이터 마이그레이션 에이전트(127)에 제공한다.

[0070] 사용자가 두 개의 연속적인 서버 지역 사이의 경계에 또는 가상 위치 내의 두 개의 연속적인 지역들 사이의 경계에 있는 상황에서, 데이터 동기화기(129)는 사용자들에 대해 두 개의 연속적인 서버 지역들/지역들에 할당된 두 개 이상의 컴퓨터들 사이에서 데이터를 동기화하는 데 사용된다. 예를 들어, 제1 컴퓨터가 제1 서버 지역에 할당되고 제2 컴퓨터가 제2 서버 지역에 할당되면, 제1 및 제2 서버 지역들의 경계에 있는 사용자의 (게임 상태를 포함하는) 게임 데이터는 제1 컴퓨터와 제2 컴퓨터 사이에서 동기화된다. 데이터 동기화기(129)는 두 개의 연속적인 서버 지역들/지역들에 할당된 제1 및 제2 컴퓨터들을 식별하고 이들이 사용자와 관련된 게임 데이터를 동기화할 수 있도록 제1 및 제2 컴퓨터들에 동기화 신호를 생성한다. 사용자에게 대한 게임 데이터 동기화는 게임 데이터의 다른 부분들에 부여된 우선 순위를 따르며, 동기화된 데이터는 사용자 데이터 마이그레이션 에이전트

(127)에 의해 사용자의 클라이언트 장치로 포워딩된다.

- [0071] 사용자 위치 정보 및 컴퓨터 할당은 온라인 게임에 액세스하는 각 사용자의 사용자 프로파일에 저장된다. 새로운 사용자들이 온라인 게임에 액세스하고 온라인 게임의 게임 세계 내의 특정 가상 위치에 진입할 때, 컴퓨터 할당과 함께 사용자 프로파일 정보는 분산형 게임 엔진(100)에 의해 새로운 사용자들을 새로운 사용자들에 의해 생성된 게임 데이터를 핸들링하기 위한 적절한 컴퓨터들에 할당하는 데 사용된다.
- [0072] 일부 구현에서, 분산형 게임 엔진은 서로 다른 사용자들의 온라인 게임의 게임 상태, 서로 다른 가상 위치들에 있는 사용자들에 의해 제공된 입력들의 유형 및 양, 사용자들의 전문가 레벨 등에 기초하여, 서로 다른 사용자들을 위한 온라인 게임의 서로 다른 레벨들에 필요한 컴퓨터들의 유형 및 양을 결정하기 위해 인공 지능(AI) 모델을 트레이닝하고, 분산형 게임 엔진의 컴퓨터 지역 관리자(120)는 AI 모델에 포함된 정보를 사용하여 게임 세계의 서로 다른 가상 위치들에 필요한 컴퓨팅 리소스들을 식별하고 프로비저닝할 수 있으며, 여기서 컴퓨팅 리소스들은 마이크로 레벨 또는 매크로 레벨에서 프로비저닝될 수 있다.
- [0073] 게임 상태는 특정 시점에서 온라인 게임의 게임 플레이의 복잡성을 제공한다. 예를 들어, 게임 상태는 게임 캐릭터, 게임 오브젝트, 게임 오브젝트 속성, 게임 속성, 게임 오브젝트 상태, 그래픽 오버레이 등을 포함할 수 있다. 게임 상태는 또한 렌더링을 위해 게임 플레이 데이터를 생성하고 처리하는 데 사용되는, CPU, GPU, 메모리, 레지스터 값, 프로그램 카운터 값, 프로그래밍 가능한 DMA 상태, DMA를 위한 버퍼링된 데이터, 오디오 칩 상태, CD-ROM 상태, 사용 가능한 통신 대역폭 등과 같은, 온라인 게임의 인스턴스들의 실행에 사용되는 서버들의 네트워크의 모든 장치 또는 모듈 또는 컴포넌트의 상태를 포함한다. 게임 상태는 또한 현재 실행중인 온라인 게임의 실행 코드의 부분들 및 게임 플레이를 계속하기 위해 로딩되어야 하는 실행 가능 코드의 부분들을 식별한다. 게임 상태는 또한 플레이어의 사용자 프로필, 게임 오브젝트에 대한 사용자 맞춤 설정, 게임 유형, 플레이어에 의해 선택된 게임 레벨/난이도, 게임 툴의 양, 상금, 포인트, 현금, 상금 등을 포함하여, 각 사용자의 저장된 데이터를 포함할 수 있다. 게임 상태는 또한 온라인 게임의 게임 플레이와 관련된 게임 세부 사항들을 이해하는 데 유용한 랜덤하게 생성된 특징들 또는 AI 오버레이들 또는 게임 복잡도를 포함할 수 있다. 게임 상태 및 기타 관련 게임 관련 데이터는 온라인 게임의 처리 부하가 쉽게 추정될 수 있는 포괄적인 세부 사항들을 제공한다.
- [0074] 게임 클라우드 시스템에서 실행되는 온라인 게임은 대규모 멀티플레이어 온라인(MMO) 게임일 수 있으며, 온라인 게임에 할당된 게임 서버는 플레이어들(즉, 사용자들)의 활동들을 캡처하고 플레이어들의 활동들을 사용하여 각 플레이어에 대한 온라인 게임의 게임 데이터를 생성하기 위해 온라인 게임의 게임 로직과 통신하는 멀티플레이어 분산형 게임 엔진의 인스턴스를 포함할 수 있다. 게임 로직은 MMO 게임 세션 동안 플레이어들 각각의 게임 상태들을 관리하고 게임 상태들을 분산형 게임 엔진에 제공하도록 구성되며, 이는 MMO 게임 세션에 참여하는 각 플레이어 또는 관중에 대해 생성된 게임 데이터에 대해 게임 물리, 셰이딩, 스크립팅, 텍스처링, 오디오, 애니메이션, 메모리 관리, 장면 그래프, 어셈블링, 인코딩 등과 같은, 온라인 게임의 다양한 특징들을 처리하는 기능 컴포넌트들을 제공한다.
- [0075] 분산형 게임 엔진(100)은 온라인 게임의 게임 플레이 동안 제공된 사용자 입력들에 응답하여 생성된 게임 데이터의 다양한 특징들의 병렬 처리를 가능하게 하는 복수의 프로세스 엔진들 또는 컴포넌트들을 포함한다. 따라서, 게임 로직의 인스턴스들과 함께 분산형 게임 엔진(100)의 인스턴스들은 온라인 게임의 게임 플레이 세션 동안 게임 클라우드 시스템(GCS) 내의 복수의 서버들에서 실행되어 분산형 게임 엔진이 게임 데이터를 빠르고 효율적인 방식으로 처리할 수 있도록 한다. 사용자들이 온라인 게임을 함에 따라, 사용자 입력들이 수집되고 처리되어 게임의 게임 상태에 영향을 미치며 사용자들의 클라이언트 장치들로 전송할 게임 플레이 데이터를 생성한다. 온라인 게임의 게임 상태가 분석되어 사용자의 사용자 메트릭들 및 온라인 게임의 게임 상태 메트릭들을 식별한다. 사용자 메트릭들은 사용자의 사용자 데이터에 업데이트된다. 마찬가지로, 온라인 게임의 게임 상태 메트릭들은 온라인 게임의 게임 데이터에 업데이트된다. 사용자 메트릭들 및 게임 상태 메트릭들은 게임 플레이 동안 서로 다른 사용자들에 의해 생성된 게임 데이터의 우선 순위를 지정하는 데 사용되며 각각의 사용자들의 클라이언트 장치들로 포워딩된다.
- [0076] 분산형 게임 엔진(100)의 다양한 컴포넌트들은 온라인 게임을 위한 게임 세계의 각 가상 위치에서 생성된 게임 데이터를 처리하는 데 필요한 리소스들의 유형과 각 유형의 리소스의 양을 결정하는 데 사용된다. 결정은 각 사용자에 대한 온라인 게임의 게임 상태, 사용자 입력들 및 온라인 게임에 대해 정의된 처리 기준에 기초한다. 예를 들어, 처리 기준은 플레이어들의 게임 데이터의 특정 부분들이 게임 데이터의 다른 특정 부분들보다 더 높은 우선 순위가 부여되어야 함을 나타낼 수 있다. 상기 결정에 기초하여, 게임 콘텍스트 프로세서(111)는 게임 데

이터의 서로 다른 부분들을 처리하고, 스핀 업/스핀 다운 디렉터는 컴퓨팅 리소스들 또는 컴퓨터들을 온라인 게임에 대한 게임 세계에서 식별된 서로 다른 가상 위치들로 프로버저닝하여 온라인 게임의 게임이 각 플레이어 및 모든 플레이어에 대해 빠르고 효율적인 방식으로 처리되도록 한다.

[0077] 일반적으로 말해서, 게임 엔진은 MMO 게임과 같은 게임의 기반 역할을 하는 소프트웨어 계층이며, 온라인 게임 개발에 사용되는 프레임워크를 제공한다. 게임 엔진은 모든 온라인 게임에 필요한 공통 관련 태스크들(즉, 게임 엔진 태스크들)을 수행하는 세부 사항을 추상적으로 구성하는 반면, 해당 온라인 게임의 게임 로직은 각 온라인 게임이 플레이되어야 하는 방식의 세부 사항들을 제공한다. 게임 엔진 프레임워크는 온라인 게임에 생명을 불어넣는 온라인 게임의 핵심 특징들을 처리하기 위한 복수의 재사용 가능한 기능 컴포넌트들을 포함한다. 게임 엔진에 의해 처리되는 기본 핵심 특징들은 물리학(예를 들어, 충돌 감지, 충돌 응답, 궤적, 중력에 기반한 오브젝트의 움직임, 마찰 등), 그래픽, 오디오, 인공 지능, 스크립팅, 애니메이션, 네트워킹, 스트리밍, 최적화, 메모리 관리, 스테딩, 위치 지정 지원 등을 포함할 수 있다. 재사용 가능한 컴포넌트들은 온라인 게임에 대해 식별된 핵심 특징들을 처리하는 데 사용되는 프로세스 엔진들을 포함한다. 게임 실행 시간 동안, 게임 엔진은 온라인 게임의 게임 로직을 관리하고, 게임 엔진의 기능적 부분들의 할당 및 동기화를 관리하여 게임 데이터를 최적의 방식으로 처리하고, 렌더링을 위해 클라이언트 장치들로 다시 전송되는 게임 데이터의 프레임들을 생성한다. 현재 사용 가능한 다양한 게임 엔진들은 서로 다른 핵심 기능들을 제공하며, 온라인 게임을 실행하는 데 사용할 수 있는 기능들에 기초하여 적절한 게임 엔진이 선택될 수 있다.

[0078] 일부 구현들에서, 게임 엔진 및 게임 로직의 인스턴스들은 각 서버가 게임 엔진의 특정 기능 부분을 실행하는 복수의 서버들에 걸쳐 분산된다. 서로 다른 서버들은 서로 다른 인스턴스들을 동시에 실행하여 게임 데이터를 병렬로 처리할 수 있다. 이런 방식으로, 분산형 게임 엔진은 게임 데이터를 더욱 빠르고 효율적으로 처리하도록 한다. 게임 엔진은 온라인 게임에 생명을 불어넣는 데 필요한 복수의 기능적 부분들을 처리하도록 구성된다. MMO 게임 환경을 위해 개발된 각 온라인 게임은 온라인 게임이 기능하는 데 필요한 게임 엔진의 서로 다른 기능적 부분들을 식별할 수 있다.

[0079] 기능적 부분들은 온라인 게임의 개발자에 의해 식별될 수 있거나 온라인 게임의 게임 로직을 분석하여 결정될 수 있다. 온라인 게임의 사양 요구 사항에 기초하여, 서버에서 실행되는 각 게임 엔진 인스턴스(즉, 상호 교환적으로 "컴퓨터"라고도 함)는 분산형 게임 엔진의 특정 프로세스 엔진을 사용하여 온라인 게임의 게임 플레이로부터 생성된 게임 데이터의 특정 특징과 관련된 기능적 부분을 처리한다. 게임 엔진의 인스턴스들은 제1 서버의 출력(즉, 기능적 부분의 처리 결과)이 제2 서버에 대한 입력으로 작동하고, 제2 서버의 출력이 제3 서버(또는 제1 서버)에 대한 입력 등으로 작동하도록 하여 게임 데이터의 반복 처리를 제공한다. 일부 구현들에서, 게임 엔진의 특정 기능적 부분을 처리하는 데 사용되는 게임 엔진 및 게임 로직의 인스턴스를 실행하기 위해 단일 서버가 할당될 수 있다. 다른 구현들에서, 각각이 게임 엔진 및 게임 로직의 인스턴스를 각각 실행하는 복수의 서버들은 특정 기능적 부분을 처리하도록 할당될 수 있다. 복수의 서버들은 가상 위치에서 생성된 처리 데이터의 양에 기초하여 식별 및 할당될 수 있다. 게임 엔진과 게임 로직은 온라인 게임에 생명을 불어넣는 데 필요한 게임 엔진의 서로 다른 기능적 부분들을 처리하기 위해 복수의 서버들에서 인스턴스화되기 때문에, 게임 엔진은 "분산"된다고 말한다. 따라서, 이 애플리케이션에서 사용되는 분산형 게임 엔진에 대한 모든 참조는 게임 엔진과 게임 로직의 해당 부분이 다수의 서버들에서 인스턴스화되는 이 구성을 말한다.

[0080] 서로 다른 서버들에서 출력되는 게임 데이터는 렌더링을 위해 동기화, 어셈블링, 인코딩 및 각각의 클라이언트 장치들로 전송된다. 게임 데이터의 어셈블링을 돕기 위해, 게임 엔진은 게임 엔진의 다양한 기능적 부분들을 다른 서버들로의 분산을 동기화하고, 다른 서버들에 할당된 게임 엔진의 다른 기능적 부분들의 처리를 동기화하고, 처리된 결과들을 서버들 간, 서버와 사용자 마이그레이션 에이전트 간에 라우팅하는 동기화 엔진을 포함할 수 있다. 분산형 게임 엔진의 사용자 데이터 마이그레이션 에이전트는 특정 사용자의 게임 데이터를 처리하는 데 사용되는 서버들(즉, 컴퓨터들)의 추적을 유지하고, 다른 서버들/컴퓨터들로부터의 출력 게임 데이터를 수집하고, 각 사용자에게 대한 출력 게임 데이터를 어셈블링하고, 인코딩하고 렌더링을 위해 인코딩된 게임 데이터를 다른 클라이언트 장치들로 포워딩하는 데 사용된다.

[0081] 분산형 게임 엔진이 AI 모델을 사용하여 게임 데이터를 처리하는 데 필요한 컴퓨터들/컴퓨팅 리소스들의 유형을 결정할 때, 분산형 게임 엔진은 기계 학습 알고리즘을 사용하여 이전 게임 플레이 세션 동안 생성된 온라인 게임의 데이터로부터의 정보를 사용하여 AI 모델 노드들의 신경망을 채울 수 있다. 기계 학습 알고리즘은 AI 모델을 생성/세분화하기 위해 출력을 제공하는 데 사용되는 분류기들을 포함할 수 있다. 각 분류기는 게임 세계에서 식별된 특정 가상 위치 또는 서브 지역에서 생성된 게임 데이터의 특정 특징을 식별 및 처리하고, 처리 부하 및 유형, 특정 특징을 처리하는 데 사용되는 리소스들의 양과 관련된 정보로 AI 모델의 노드를 업데이트하도록 구

성되는 엔진/모듈을 가질 수 있다. 처리된 정보는 게임 데이터 다운스트림과 함께 추가 처리를 위한 다른 분류기들로 포워딩된다. AI 모델의 출력은 컴퓨터들/컴퓨팅 리소스들을 온라인 게임에 프로비저닝하거나, 기존 컴퓨팅 리소스들 또는 서버 컴퓨터들이 게임 데이터를 처리하기에 충분한지를 결정하거나, 또는 온라인 게임의 게임 데이터의 최적의 처리를 위해 서버들/서버 리소스들에 대한 조정이 이루어져야 하는지를 결정하는 데 사용된다.

[0082] 스핀 업/스핀 다운 디렉터(131)는 온라인 게임의 게임 세계의 각 가상 위치에 할당된 컴퓨터들/컴퓨팅 리소스들을 동적으로 조정하고, 다른 사용자들과 관련된 게임 상호 작용들을 적절한 컴퓨터들/컴퓨팅 리소스들에 할당하는 데 사용되어 각 사용자의 상호 작용들이 효율적으로 처리되고 생성된 게임 데이터가 적시에 렌더링을 위해 사용자들의 클라이언트 장치들로 전송될 수 있도록 한다. 각 사용자에게 제공되는 게임 데이터는 게임 세계에 대한 각각의 사용자의 관점을 포함한다. 관중들이 사용자들의 경우, 관중에게 제공되는 게임 데이터는 온라인 게임에 대한 자신의 관점 또는 관중이 팔로우하는 플레이어의 관점을 포함할 수 있다. 새로운 사용자들이 게임 세션에 참여하면, 분산형 게임 엔진 시스템 구성은 새로운 사용자들을 라우팅하는 데 사용되어, 온라인 게임의 사용자들(즉, 기존 사용자들 및 새로운 사용자들)이 만족스러운 게임 플레이 경험을 가질 수 있도록 한다.

[0083] 분산형 게임 엔진의 인스턴스들과 온라인 게임의 게임 로직을 실행하고, 일부 예시적인 구현들에서는, 온라인 게임을 위해 생성된 게임 데이터를 처리하기 위해 다양한 유형의 서버들이 제공될 수 있다. 서버는 물리적 서버일 수 있다. 물리적 서버(또는 간단히 "서버")는 독립형 서버일 수 있거나 게임 클라우드 시스템 내 데이터 센터의 일부일 수 있다. 데이터 센터의 서버는 데이터 센터에 있는 독립형 서버일 수 있거나 랙의 일부일 수 있으며, 각 랙에는 다수의 서버 컴퓨팅 장치들이 포함된다. 물리적 서버는 운영 체제, 메모리, 하나 이상의 애플리케이션들 및/또는 게임 엔진의 하나 이상의 기능적 부분들을 실행하기 위한 마이크로 프로세서와 같은 필수 리소스들을 포함한다. 게임 엔진의 애플리케이션들 또는 기능적 부분들은 메모리에 저장되고, 마이크로 프로세서를 사용하여 처리되며, 운영 체제에 의해 관리된다.

[0084] 대안으로, 서버는 게임 클라우드 시스템의 일부인 가상 기계일 수 있다. 게임 클라우드 시스템은, 하나 이상의 가상 기계들은 하이퍼바이저에 사용할 수 있는 하드웨어 리소스들을 사용하여 온라인 게임의 인스턴스를 실행하는 게임 프로세서를 지원하도록 구성된, 호스트 기계의 하이퍼바이저에서 실행되는 복수의 가상 기계들(VM들)을 포함할 수 있다. 복수의 가상 기계들과 관련된 복수의 게임 프로세서들이 온라인 게임의 인스턴스들을 실행할 때, 사용자의 입력들이 수집되고, 게임 클라우드 시스템의 하나 이상의 백엔드 서버들(즉, 물리적 서버들 또는 가상 서버들)에 의해 구성되며, 온라인 게임의 게임 상태에 영향을 미치도록 처리된다. 사용자 입력들에 응답하여 온라인 게임에 대해 생성된 게임 데이터는 게임 데이터의 스트리밍 프레임들로 어셈블링되고, 인코딩되고 렌더링을 위해 사용자의 클라이언트 장치로 포워딩된다. 컴퓨터들 및 컴퓨팅 리소스들을 프로비저닝하기 위해 분산형 게임 엔진에 의해 AI 모델이 사용되는 경우, 사용자의 입력들은 AI 모델을 트레이닝하기 위한 트레이닝 데이터를 생성하는데도 사용된다. 각각의 가상 기계들은 호스트 기계의 가상화 계층(즉, 하이퍼바이저)에 이용 가능한 하드웨어 리소스들을 사용하여 운영 체제에 의해 실행되는 하나 이상의 애플리케이션들(예를 들어, 하나 이상의 온라인 게임 애플리케이션들)을 포함할 수 있다.

[0085] 다른 대안적인 구현들에서, 서버는 온라인 게임 애플리케이션과 같은 애플리케이션, 및 온라인 게임 애플리케이션을 실행하는 데 필요한 모든 종속성, 라이브러리 및 구성 파일이 애플리케이션이 애플리케이션을 실행하기 위해 컨테이너 내에 할당된 리소스들을 사용하는 가상 컨테이너(또는 단순히 "컨테이너"라고도 함)로 함께 묶을 수 있도록 하는 클라우드 컨테이너의 일부일 수 있다. 컨테이너는 전체 운영 체제없이 동작하며, 이에 따라 운영 체제 레벨의 가상화를 제공한다. 컨테이너는 온라인 게임을 위한 게임 엔진 및 게임 로직의 인스턴스를 실행하는 것과 같은 태스크를 수행하기 위해 최소한의 리소스 양으로 동작할 수 있으며, 리소스들(즉, 물리적 하드웨어 리소스들)은 가상화된다. 클라우드 컨테이너는 다른 서버들에 완전히 배치될 수 있으므로 휴대 가능하다. 일 구현에서, 컨테이너는 커널을 컨테이너 호스트와 공유하여 프로세스 및 네임스페이스 격리 기술을 통해 애플리케이션 격리를 제공한다. 대안적인 구현에서, 컨테이너 호스트의 커널은 공유되지 않으며, 이에 따라 가상 기계로부터 컨테이너화된 애플리케이션을 실행함으로써 향상된 격리를 제공한다. 앞서 언급된 컴퓨터들(즉, 서버들)이 유형들은 단순한 예일 뿐이며, 제한적인 것으로 간주해서는 안된다. 게임 엔진 및 게임 로직을 인스턴스화하기 위한 리소스들을 제공하는 다른 형태의 서버들 또는 컴퓨터들(즉, 컴퓨팅 장치들)도 사용될 수 있다.

[0086] 일부 구현들에서, 사용자들은 전 세계 어디서나 클라우드 게임 시스템의 온라인 게임 애플리케이션에 연결할 수 있다. 온라인 게임 애플리케이션은 전 세계에 걸쳐 분산된 서로 다른 데이터 센터들의 서로 다른 서버들에서 다수의 게임 인스턴스들을 시작하여 이러한 연결을 가능하게 한다. 온라인 게임의 게임 플레이 동안 각각의 클라이언트 장치들에서 생성된 사용자 입력들은 서버 네트워크로부터 수집되고 게임 데이터가 생성된다. 게임 데이

터는 네트워크 모델들 중 어느 하나를 사용하여, 각각의 클라이언트 장치들에 게임 데이터 프레임들로 반환된다. 이 애플리케이션에 설명된 게임 클라우드 시스템의 서버들 간에 통신하는 데 사용될 수 있는 네트워크 모델의 몇 가지 예들은 TCP/UDP 네트워크 모델 또는 ISO(International Organization of Standardization) 기반 OSI(Open System Interconnection) 네트워크 아키텍처를 포함한다. 분산형 서버들 간의 데이터 처리 및 통신은 다수의 기존 프로토콜들 중 어느 하나를 따를 수 있다. ISO 모델은 애플리케이션 계층 세트(애플리케이션 계층, 프레젠테이션 계층, 세션 계층) 및 데이터 흐름 계층(전송 계층, 네트워크 계층, 데이터 링크 계층 및 물리 계층)을 포함한다. 애플리케이션 계층들은 송신 및 수신 엔티티들을 식별하고, 온라인 게임과 같은 애플리케이션의 데이터를 각각의 엔티티들에 의해 이해되는 방식으로 송신 및 수신 엔티티들 간에 전달하고, 최소한의 데이터 손실로 송신 및 수신 엔티티들 간 통신을 관리하고 동기화하도록 구성된다. 데이터 흐름 계층들은 데이터 전송이 순차적 또는 병렬 경로에서 수행되어야 하는지 여부를 결정하도록 구성되며, 이는 데이터를 패킷화하고 패킷화된 데이터를 프레임들로 어셈블링하고 선호하는 경로를 따라 데이터의 프레임들을 전송하기 위해 따라야 하는 프로토콜이다. 상기에 언급된 설명은 온라인 게임을 위한 서버 네트워크를 통해 데이터를 통신하는 데 사용되는 ISO 모델의 개요만을 제공하며, 다양한 계층들의 역할에 대한 더욱 자세한 세부 사항들은 당업자에 의해 이해되며, 따라서 더 자세히 설명되지 않는다는 점에 유의해야 한다.

[0087] 도 5는 본 개시의 일 구현에 따른, 게임 데이터를 업데이트하기 위한 사용자 입력들의 수집을 지원하고 업데이트된 게임 데이터를 사용하여 온라인 게임의 게임 플레이와 관련된 AI 모델을 구축 및/또는 트레이닝하는 예시적인 시스템(300A)의 개요를 예시한다. 특히, 시스템(300A)은 게임 클라우드 시스템(GCS)(300)에 저장된 비디오 게임들의 액세스 및 플레이를 가능하게 한다. 일반적으로 말해서, 게임 클라우드 시스템(GCS)(300)은 해당 게임 플레이를 통해 온라인 게임 애플리케이션을 플레이하는 복수의 플레이어들을 지원하기 위해 네트워크(200)를 통해 동작하는 클라우드 컴퓨팅 시스템일 수 있다. 이러한 게임 플레이와 관련된 데이터는 온라인 게임 애플리케이션의 게임 플레이와 관련된 AI 모델을 구축 및/또는 트레이닝하는 데 사용될 수 있다. 특히, 시스템(300A)은 GCS(300), 하나 이상의 소셜 미디어 공급자들(240) 및 복수의 클라이언트 장치들(101)을 포함하며, 이들 모두는 네트워크(200)(예를 들어, 인터넷)를 통해 연결된다. 복수의 사용자들과 연관된 복수의 클라이언트 장치들은 GCS(300) 및 소셜 미디어 공급자들(240)에 의해 제공되는 서비스들에 액세스하기 위해 네트워크(200)에 연결될 수 있다. 사용자들은 특정 플레이어 또는 온라인 게임의 게임 플레이를 팔로우하는 플레이어들 또는 관중들일 수 있다.

[0088] 일 구현에서, 게임 클라우드 시스템(300)은 게임 서버(205), 비디오 레코더(271), 태그 프로세서(273) 및 사용자 프로필 관리자를 포함하는 계정 관리자(274), 게임 선택 엔진(즉, 게임 선택기)(275), 게임 세션 관리자(285), 사용자 액세스 로직(280), 네트워크 인터페이스(290) 및 소셜 연결 관리자(295)를 포함한다. GCS(300)은 일반적으로 데이터 저장소(260)에 저장될 수 있는 게임 상태 저장소, 사용자 계정, 사용자 데이터 저장소 등과 같은 복수의 저장 시스템들을 더 포함할 수 있다. 다른 게임 저장 시스템들은 게임 코드 저장소(261), 레코딩된 게임 저장소(262), 태그 데이터 저장소(263), 비디오 게임 타이틀 저장소(264) 및 게임 네트워크 사용자 저장소(265)를 포함할 수 있다. 일 구현에서, GCS(300)은 온라인 게임 애플리케이션들, 서비스들, 게임 관련 디지털 콘텐츠, 및 시스템들, 애플리케이션들, 사용자들, 및 소셜 네트워크들 사이의 상호 연결성을 제공할 수 있는 시스템이다. GCS(300)은 네트워크 인터페이스(290)를 통해 소셜 연결 관리자(295)를 통해 사용자의 클라이언트 장치들(101) 및 소셜 미디어 공급자들(240)과 통신할 수 있다. 소셜 연결 관리자(295)는 하나 이상의 친구들을 관련시키도록 구성될 수 있다. 일 구현에서, 각 소셜 미디어 공급자(240)는 사용자 소셜 네트워크 연결들을 보여주는 적어도 하나의 소셜 그래프(245)를 포함한다.

[0089] 플레이어(즉, 사용자)는 게임 세션 관리자(285)를 통해 GCS(300)에 의해 제공되는 서비스들에 액세스할 수 있다. 예를 들어, 계정 관리자(274)는 플레이어를 위해 GCS(300) 상의 게임 타이틀들에 대한 인증 및 액세스를 가능하게 한다. 계정 관리자(274)는 사용자들(플레이어들, 관중들)에 대한 정보를 저장하고 각 사용자에게 대한 사용자 프로필을 관리한다. 사용자 정보는 인증을 위해 계정 관리자(274)에 의해 사용될 수 있다. 추가로, 사용자가 소유한 게임 타이틀은 계정 관리자(274)에 의해 관리될 수 있다. 예를 들어, 게임 타이틀 저장소(264)에 저장된 비디오 게임들은 해당 비디오 게임들을 소유하거나 그에 대한 액세스를 제공받는 사용자에게 이용 가능하게 된다.

[0090] 일 구현에서, 사용자는 네트워크(200)를 통한 연결을 통해 클라이언트 장치(101)에 의해 GCS(300) 및 소셜 미디어 공급자들(240)이 제공하는 서비스들에 액세스할 수 있다. 클라이언트 장치(101)는 유선 또는 무선, 휴대용 또는 휴대용이 아닌 프로세서 및 메모리를 갖는 모든 유형의 장치를 포함할 수 있다. 일 구현에서, 클라이언트 장치(101)는 스마트폰, 태블릿 컴퓨터 또는 휴대용 폼 팩터에서 터치 스크린 기능을 제공하는 하이브리드 형태

일 수 있다. 하나의 예시적인 클라이언트 장치는 운영 체제를 실행하고, 네트워크(200)를 통해 획득될 수 있고 로컬 휴대용 장치(예를 들어, 스마트폰, 태블릿, 랩톱, 데스크톱 등)에서 실행될 수 있는 다양한 애플리케이션들(앱들)에 대한 액세스가 제공되는 휴대용 전화 장치를 포함할 수 있다.

[0091] 클라이언트 장치(101)는 사용자가 입력 명령들(236)을 전송하고 GCS(300) 및 소셜 미디어 공급자들(240)로부터 수신된 데이터 및/또는 정보(135)를 디스플레이하기 한 인터페이스 역할을 하는 디스플레이(132)를 포함한다. 디스플레이(132)는 터치 스크린, 또는 일반적으로 평판 디스플레이, 음극선관(CRT), 또는 디스플레이를 렌더링 할 수 있는 기타 장치에 의해 제공되는 디스플레이로 구성될 수 있다. 대안으로, 클라이언트 장치(100)는 데스크톱 컴퓨터 또는 랩톱 컴퓨터와 유사한, 장치로부터 분리된 디스플레이(132)를 가질 수 있다.

[0092] 일 구현에서, 클라이언트 장치(101)는 사용자(플레이어)가 비디오 게임을 할 수 있도록 GCS(300)와 통신하도록 구성된다. 예를 들어, 플레이어는 게임 선택 엔진(275)을 통해 게임 타이틀 데이터 저장소(264)에서 이용 가능한 비디오 게임을 (예를 들어, 게임 타이틀 등에 의해) 선택할 수 있다. 선택된 비디오 게임이 활성화되고 비디오 게임의 인스턴스가 GCS(300) 상의 게임 서버(205)에 의해 실행되도록 로드된다. 일 구현에서, 게임 플레이는 주로 GCS(300)에서 실행되어, 클라이언트 장치(101)가 GCS(300)로부터 게임 비디오 프레임들(135)의 스트림을 수신하도록 할 것이며, 게임 플레이를 구동하기 위한 사용자 입력 명령들(236)이 GCS(300)로 다시 전송되도록 한다. 스트리밍 게임 플레이로부터 수신된 비디오 프레임들(135)은 클라이언트 장치(101)의 디스플레이(132)에 렌더링된다.

[0093] 일 구현에서, 플레이어가 플레이할 이용 가능한 게임 타이틀을 선택한 후, 선택된 게임 타이틀에 대한 게임 세션은 게임 세션 관리자(285)를 통해 플레이어에 의해 개시될 수 있다. 게임 세션 관리자(285)는 먼저 데이터 저장소(260)의 게임 상태 저장소에 액세스하여 (선택된 게임에 대한) 플레이어에 의해 플레이된 마지막 세션의 저장된 게임 상태를 검색하여, 있다면, 플레이어가 이전 게임 플레이 중지 지점으로부터 게임 플레이를 다시 시작할 수 있도록 한다. 재개 또는 시작 지점이 식별되면, 게임 세션 관리자 285)는 게임 코드 저장소(261)로부터 선택된 게임 타이틀의 게임 코드를 실행하도록 게임 서버(205)의 게임 실행 엔진에 알릴 수 있다. 게임 세션이 시작된 후, 게임 세션 관리자(285)는 네트워크 인터페이스(290)를 통해 게임 비디오 프레임들(135)(즉, 스트리밍 비디오 데이터)을 클라이언트 장치, 예를 들어, 클라이언트 장치(100)로 전달할 수 있다.

[0094] 게임 플레이 동안, 게임 세션 관리자(285)는 게임 서버(205), 비디오 레코더(271)의 레코딩 엔진 및 태그 프로세서(273)와 통신하여 게임 플레이 또는 게임 플레이 세션의 레코딩(예를 들어, 비디오)을 생성하거나 저장할 수 있다. 일 구현에서, 게임 플레이의 비디오 레코딩은 게임 플레이 동안 입력되거나 제공된 태그 콘텐츠 및 기타 게임 관련 메타데이터를 포함할 수 있다. 태그 콘텐츠는 또한 스냅샷을 통해 저장될 수 있다. 해당 게임 플레이에 대응하는 임의의 게임 메트릭들과 함께, 게임 플레이의 비디오 레코딩은 레코딩된 게임 저장소(262)에 저장될 수 있다. 임의의 태그 콘텐츠는 태그 데이터 저장소(263)에 저장될 수 있다.

[0095] 게임 플레이 동안, 게임 세션 관리자(285)는 비디오 게임의 대응하는 게임 플레이의 결과에 영향을 미치는 데 사용되는 사용자 입력 명령들(236)을 전달하고 획득하기 위해 게임 서버(205)와 통신할 수 있다. 플레이어에 의해 입력된 입력 명령들(236)은 클라이언트 장치(101)로부터 GCS(300)의 게임 세션 관리자(285)로 전송될 수 있다. 태그 콘텐츠(예를 들어, 텍스트, 이미지, 비디오 레코딩 클립 등)와 같은 게임 플레이 및 사용자 상호 작용 입력을 구동하는 데 사용되는 입력 명령들을 포함하는 입력 명령들(예를 들어, 컨트롤러 입력들 또는 사용자 입력들)(236). 게임 입력 명령들뿐만 아니라 모든 사용자 플레이 메트릭들(사용자가 게임을 하는 시간 등)은 게임 네트워크 사용자 저장소에 저장될 수 있다. 비디오 게임에 대한 게임 플레이와 관련된 선택 정보는 사용자에 사용될 수 있는 여러 특징들을 활성화시키는 데 사용될 수 있다.

[0096] 따라서, 본 개시는 온라인 게임의 게임 세계 내에서 가상 위치들을 식별하는 알고리즘을 구현하는 시스템들 및 방법들을 설명하며, 온라인 게임의 효율적인 처리를 보장하기 위해 각 가상 위치에서 처리 부하를 핸들링하기 위한 컴퓨터들 및 컴퓨팅 리소스들을 프로비저닝한다.

[0097] 도 6은 일 구현에 따른, 게임 세계를 나타내는 온라인 게임에 대한 처리 리소스들을 관리하기 위한 방법의 샘플 동작 흐름을 예시한다. 방법은 한 명 이상의 사용자들에 대해 온라인 게임이 실행되는 동작(610)에서 시작한다. 온라인 게임의 하나 이상의 인스턴스들은 게임 플레이를 위해 온라인 게임에 액세스하는 복수의 사용자들에 응답하여 실행될 수 있다. 한 명 이상의 사용자들 각각에는 온라인 게임의 게임 세계에 정의된 가상 위치들에 대한 액세스가 제공된다. 컴퓨터는 동작(620)에 예시된 바와 같이, 게임 세계의 한 명 이상의 사용자들의 게임 상호 작용을 처리하기 위해 할당된다. 별개의 컴퓨터가 게임 세계의 각 가상 위치에 있는 사용자들에 의해 생성된 게임 상호 작용을 처리하기 위해 할당될 수 있다. 각 가상 위치에 할당된 각 컴퓨터는 할당된 컴퓨터가 핸들링

할 수 있는 사용자들의 수에 대한 미리 정의된 한도를 포함할 수 있다. 미리 정의된 한도는 사용자들에 의해 생성된 게임 상호 작용의 양에 기초할 수 있으며, 여기서 게임 상호 작용은 게임 플레이 동안 플레이어들과 같은 사용자들에 의해 제공되는 사용자 입력들을 포함한다.

[0098] 온라인 게임에서 사용자들의 움직임이 모니터링되고 온라인 게임 내의 특정 가상 위치 내의 지역에 액세스하는 사용자의 수가 결정된다. 지역 내의 사용자들의 수가 특정 가상 위치에 할당된 컴퓨터에 대해 미리 정의된 한도를 초과하는 경우, 동작(630)에 예시된 바와 같이, 지역은 동작(640)에 예시된 바와 같이 복수의 서브 지역들로 동적으로 분할된다. 지역에 대해 생성되는 서브 지역들의 수는 지역의 사용자들의 밀도 및 분산형 게임 엔진에 의해 사용되는 분할 기준에 기초할 수 있다. 서브 지역들은 정의된 경계를 사용하여 생성되어, 서브 지역들 각각은 지역의 적어도 한 명의 사용자를 포함하도록 한다. 일부 구현들에서, 지역의 전체 부분은 서브 지역들로 분할된다. 다른 구현들에서, 사용자가 지역 내에서 모이는 위치에 따라, 특정 가상 위치 내의 지역의 일부는 서브 지역들로 세분화되는 반면, 지역의 나머지 부분은 별도의 서브 지역으로 처리된다. 이러한 구현들에서, 특정 가상 위치의 일부에서 생성된 서브 지역들은 각각 특정 가상 위치의 한 명 이상의 사용자들을 가질 수 있는 반면 나머지 부분은 어떠한 사용자도 포함하거나 포함하지 않을 수 있다.

[0099] 지역을 서브 지역들로 분할한 후, 할당된 컴퓨터 및 추가 컴퓨터가 동작(650)에 예시된 바와 같이, 서브 지역들 각각의 사용자들에 대한 게임 상호 작용을 처리하도록 추가 컴퓨터가 지역에 할당된다. 일부 구현들에서, 개별 컴퓨터는 각 개별 컴퓨터가 서브 지역에 있는 사용자들의 게임 상호 작용을 처리하도록 구성되는 각 서브 지역에 할당된다. 컴퓨터 처리 및 지역에 할당된 추가 컴퓨터는 각 사용자에게 대한 게임의 게임 상태를 생성하도록 조정된다. 각 사용자에게 대한 게임 데이터와 함께 게임의 게임 상태는 각 사용자의 각각의 클라이언트 장치로 포워딩된다. 각 사용자에게 대한 게임 데이터는 온라인 게임의 게임 세계에 대한 사용자의 관점을 포함하며, 이러한 관점은 온라인 게임의 게임 상태에 기초하여 결정된다.

[0100] 따라서, 분산형 게임 엔진은 각 가상 위치 내의 각 가상 위치/지역이 그 안에서 처리 부하를 처리하는 데 필요한 컴퓨터들/컴퓨팅 리소스들이 제공되는 것을 보장하기 위해 게임 세계 내에서 사용자들의 움직임을 추적하고 실시간으로 순방향으로 조치를 취하여 온라인 게임 플레이 동안 각 가상 위치 내 각 가상 위치/지역들의 처리 부하를 핸들링하도록 컴퓨터들/컴퓨팅 리소스들을 프로비저닝한다. 액션은 지역 내에서 서브 지역들 또는 서브 지역 내에서 서브 지역들을 생성하거나, 가상 위치들 내의 서브 지역들/지역들에 있는 사용자들의 수를 기반으로 두 개 이상의 서브 지역들을 재설정하고, 그에 따라 컴퓨터들/컴퓨팅 리소스들을 할당/할당 해제하는 것을 포함할 수 있다. 온라인 게임에 대한 컴퓨터들/컴퓨팅 리소스들의 동적 할당/할당 해제는 각 사용자에게 대해 빠르고 효율적인 방식으로 게임 데이터가 생성되는 동안 컴퓨터들/컴퓨팅 리소스들의 최적 사용을 보장한다.

[0101] 사전 추적 및 프로비저닝을 통해 한 시간대의 사용자가 다른 시간대의 친구 서클과 같은 다른 사용자들과 함께 온라인 게임의 게임 플레이에 참여하도록 할 수 있다. 온라인 게임의 게임 플레이는 게임 세계에 대한 각 사용자의 관점을 포함하는 게임 데이터를 각 사용자에게 제공하기 위해 게임 세계에서 식별된 서로 다른 가상 위치들에 할당된 서로 다른 컴퓨터들/컴퓨팅 리소스들에 의해 서비스된다. 가상 위치에 할당된 컴퓨터들/컴퓨팅 리소스들은 서로 다른 데이터 센터들로부터 프로비저닝될 수 있으며, 게임 데이터는 최소한의 레이턴시로 동기화, 처리, 어셈블링 및 전송된다. 물론, 서로 다른 시간대에 위치한 컴퓨터들 간 및 컴퓨터들과 클라이언트 장치들 간의 게임 데이터 처리 및 전송은 컴퓨터들, 클라이언트 장치들의 전송 거리로 인한 시간 지연을 고려해야 하며, 게임 데이터가 사용자들에게 실시간으로 제시되는 것처럼 보이게 하기 위해 사용 가능한 대역폭에 따라 게임 데이터를 전송하거나 우선 순위를 지정하기 위한 추가 대역폭을 제공해야 할 수 있다.

[0102] 사용자들의 위치와 관련된 정보는 사용자 지역 할당 관리자로부터 획득될 수 있으며, 스핀 업/스핀 다운 디렉터(131)에 의해 컴퓨터들/컴퓨팅 리소스들을 프로비저닝하고, 데이터 동기화기(129)에 의해 사용자들의 데이터를 동기화하는 데 사용될 수 있다. 데이터 동기화기(129)는 게임 플레이의 콘텍스트에 기초하고 데이터 우선 순위 선정기(111a)에 의해 제공된 정보에 기초하여 게임 데이터를 생성하기 위해 가상 위치에서 사용자의 상호 작용들의 처리를 우선 순위화하여 사용자들에게 온라인 게임의 그들 자신의 관점을 적시에 제공되도록 한다. 각 플레이어는 온라인 게임에 대한 자신의 관점을 필요로 하기 때문에, 각 가상 위치에서 생성된 처리 데이터를 핸들링하기 위해 개별 컴퓨터들/컴퓨팅 리소스들을 프로비저닝하여, 컴퓨터들/컴퓨팅 리소스들이 너무 많아지지 않도록 하는 것이 보다 효율적이다. 가상 위치의 지역 내에 생성된 서브 지역들은 사용자의 대부분이 각 서브 지역의 경계 내에 있도록 정의된다. 가상 위치들에 할당된 컴퓨터들은 끊임없이 변화하는 공간에서 사용자들에 대해 생성된 게임 데이터의 처리를 병렬화하도록 구성된다.

[0103] 본원에 설명된 다양한 구현들은 다수의 플레이어들이 주어진 시간에 온라인 게임의 게임 세계 내에서 동일한 가

상 위치에 들어갈 때 컴퓨터들을 분산시키는 결정론적 방법을 설명한다. 서로 다른 가상 위치에 할당된 컴퓨터들/컴퓨팅 리소스들은 대응하는 가상 위치들의 처리 요구 사항에 기초하여 탄력적으로 확장 또는 축소되며, 이에 따라 컴퓨터들/컴퓨팅 리소스들이 효과적으로 사용된다. 가상 위치들의 세분화 및 컴퓨터들/컴퓨팅 리소스들의 분포는 게임 세계의 각 가상 위치 내 사용자들(캐릭터들/게임 오브젝트들/아바타들로 표현)의 밀도에 기초한다.

[0104] 도 7은 본 개시의 다양한 실시예들의 양태들을 수행하는 데 사용될 수 있는 예시적인 장치(700)의 컴포넌트들을 예시한다. 예를 들어, 도 7은 본 개시의 일 실시예에 따른, 온라인 게임 및/또는 온라인 게임의 게임 플레이어들과 관련하여 다양한 기능들을 수행할 수 있는 AI 모델을 트레이닝하기에 적합한 예시적인 하드웨어 시스템을 예시한다. 이 블록도는 개인용 컴퓨터, 서버 컴퓨터, 게임 콘솔, 모바일 장치 또는 기타 디지털 장치일 수 있거나 이를 통합할 수 있는 장치(700)를 예시하며, 이들 각각은 본 발명의 일 실시예를 실행하기에 적합하다. 대안으로, 장치의 기능들은 물리적 서버 또는 가상 기계 또는 컨테이너 서버에서 구현될 수 있다. 장치(700)는 소프트웨어 어플리케이션들 및 선택적으로는 운영 체제를 실행하기 위한 중앙 처리 장치(CPU)(702)를 포함한다. CPU(702)는 하나 이상의 동종 또는 이종 프로세싱 코어들로 구성될 수 있다.

[0105] 다양한 실시예들에 따르면, CPU(702)는 하나 이상의 프로세싱 코어들을 갖는 하나 이상의 범용 마이크로 프로세서들이다. 추가 실시예들은 딥 러닝, 콘텐츠 분류 및 사용자 분류를 위해 구성된 어플리케이션들의 미디어 및 대화형 엔터테인먼트 어플리케이션들과 같은, 고도로 병렬 및 계산 집약적인 어플리케이션들에 특히 적용된 마이크로프로세서 아키텍처들을 갖는 하나 이상의 CPU들을 사용하여 구현될 수 있다. 예를 들어, CPU(702)는 온라인 게임 및/또는 온라인 게임의 게임 플레이어들과 관련하여 다양한 기능들(예를 들어, 예측)을 제공하는 것과 관련하여 학습 동작들을 지원 및/또는 수행하도록 구성된 딥 러닝 (AI) 엔진(701)을 포함하도록 구성될 수 있다. 딥 러닝 엔진(701)은 온라인 게임 및/또는 온라인 게임의 게임 플레이어들과 관련된 다양한 기능들을 제공하도록 구성된 AI 모델을 구축 및/또는 트레이닝하도록 구성된 모델러(720)를 포함할 수 있다. 또한, CPU(702)는 AI 모델을 생성하고 트레이닝하도록 구성된 분석기(740)를 포함한다. 트레이닝된 AI 모델은 특정 사용자 입력 세트에 대한 응답하여 출력을 제공하며, 출력은 트레이닝된 AI 모델의 사전 정의된 기능에 따라 달라진다. 트레이닝된 AI 모델은 온라인 게임에 대해 정의된 성공 기준을 충족하기 위해 게임 플레이 동안 구현될 리소스 할당을 결정하는 데 사용될 수 있다. 분석기(740)는 주어진 입력(예를 들어, 컨트롤러 입력, 게임 상태 데이터, 성공 기준)에 대해 트레이닝된 AI 모델의 출력을 분석하는 것을 포함하여, 온라인 게임 및/또는 온라인 게임의 게임 플레이어들과 관련된 다양한 기능들을 수행하고, 응답을 제공하도록 구성된다.

[0106] 메모리(704)는 CPU(702)에 의해 사용하기 위한 어플리케이션들 및 데이터를 저장한다. 저장 장치(706)는 어플리케이션들 및 데이터를 위한 비휘발성 저장 장치 및 기타 컴퓨터 판독 가능 매체를 제공하며, 고정식 디스크 드라이브, 이동식 디스크 드라이브, 플래시 메모리 디바이스 및 CD-ROM, DVD-ROM, 블루-레이, HD-DVD 또는 기타 광 저장 장치, 뿐만 아니라 신호 전송 및 저장 매체를 포함할 수 있다. 사용자 입력 장치들(708)은 한 명 이상의 사용자들로부터 장치(700)로 사용자 입력들을 전달하며, 그 예들은 키보드, 마우스, 조이스틱, 터치 패드, 터치 스크린, 스틸 또는 비디오 레코더/카메라, 게임 컨트롤러 및/또는 마이크를 포함할 수 있다. 네트워크 인터페이스(714)는 장치(700)가 전자 통신 네트워크를 통해 다른 컴퓨터 시스템들과 통신할 수 있게 하고, 근거리 통신망 및 인터넷과 같은 광역 통신망을 통한 유선 또는 무선 통신을 포함할 수 있다. 오디오 프로세서(712)는 CPU(702), 메모리(704) 및/또는 저장 장치(706)에 의해 제공된 인스트럭션들 및/또는 데이터로부터 아날로그 또는 디지털 오디오 출력을 생성하도록 적용된다. CPU(702), 메모리(704), 데이터 저장 장치(706), 사용자 입력 장치들(708), 네트워크 인터페이스(710) 및 오디오 프로세서(712)를 포함하는 장치(700)의 컴포넌트들은 하나 이상의 데이터 버스들(722)을 통해 연결된다.

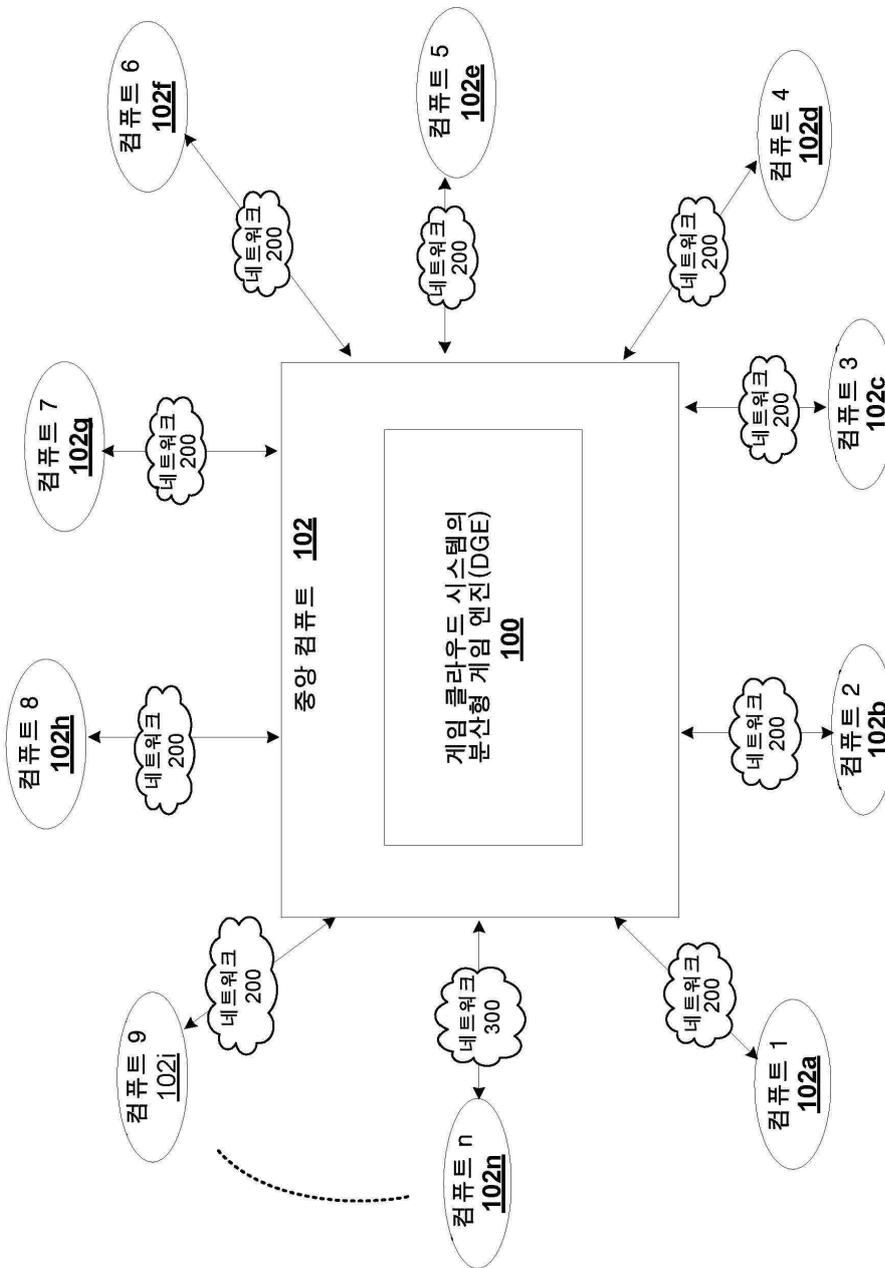
[0107] 그래픽 서브시스템(713)은 데이터 버스(722) 및 장치(700)의 컴포넌트들과 추가로 연결된다. 그래픽 서브시스템(713)은 그래픽 처리 장치(GPU)(716) 및 그래픽 메모리(718)를 포함한다. 그래픽 메모리(718)는 출력 이미지의 각 픽셀에 대한 픽셀 데이터를 저장하는 데 사용되는 디스플레이 메모리(예를 들어, 프레임 버퍼)를 포함한다. 그래픽 메모리(718)는 GPU(716)와 동일한 장치에 통합될 수 있고, GPU(716)와 별도의 장치로서 연결되고/되거나 메모리(704) 내에 구현될 수 있다. 픽셀 데이터는 CPU(702)로부터 직접 그래픽 메모리(718)로 제공될 수 있다. 대안으로, CPU(702)는 GPU(716)가 하나 이상의 출력 이미지들의 픽셀 데이터를 생성하는 원하는 출력 이미지들을 정의하는 데이터 및/또는 인스트럭션들을 GPU(716)에 제공한다. 원하는 출력 이미지들을 정의하는 데이터 및/또는 인스트럭션들은 메모리(704) 및/또는 그래픽 메모리(718)에 저장될 수 있다. 일 실시예에서, GPU(716)는 장면에 대한 지오메트리, 조명, 음영, 텍스처링, 모션 및/또는 카메라 파라미터들을 정의하는 인스트럭션들 및 데이터로부터 출력 이미지에 대한 픽셀 데이터를 생성하기 위한 3D 렌더링 능력들을 포함한다. GPU(716)는 세이

더 프로그램들을 실행할 수 있는 하나 이상의 프로그램 가능 실행 유닛들을 더 포함할 수 있다. 일 실시예에서, GPU(716)는 AI 또는 딥 러닝 기능과 같은 추가 처리 능력을 제공하기 위해 AI 엔진 내에서 구현될 수 있다.

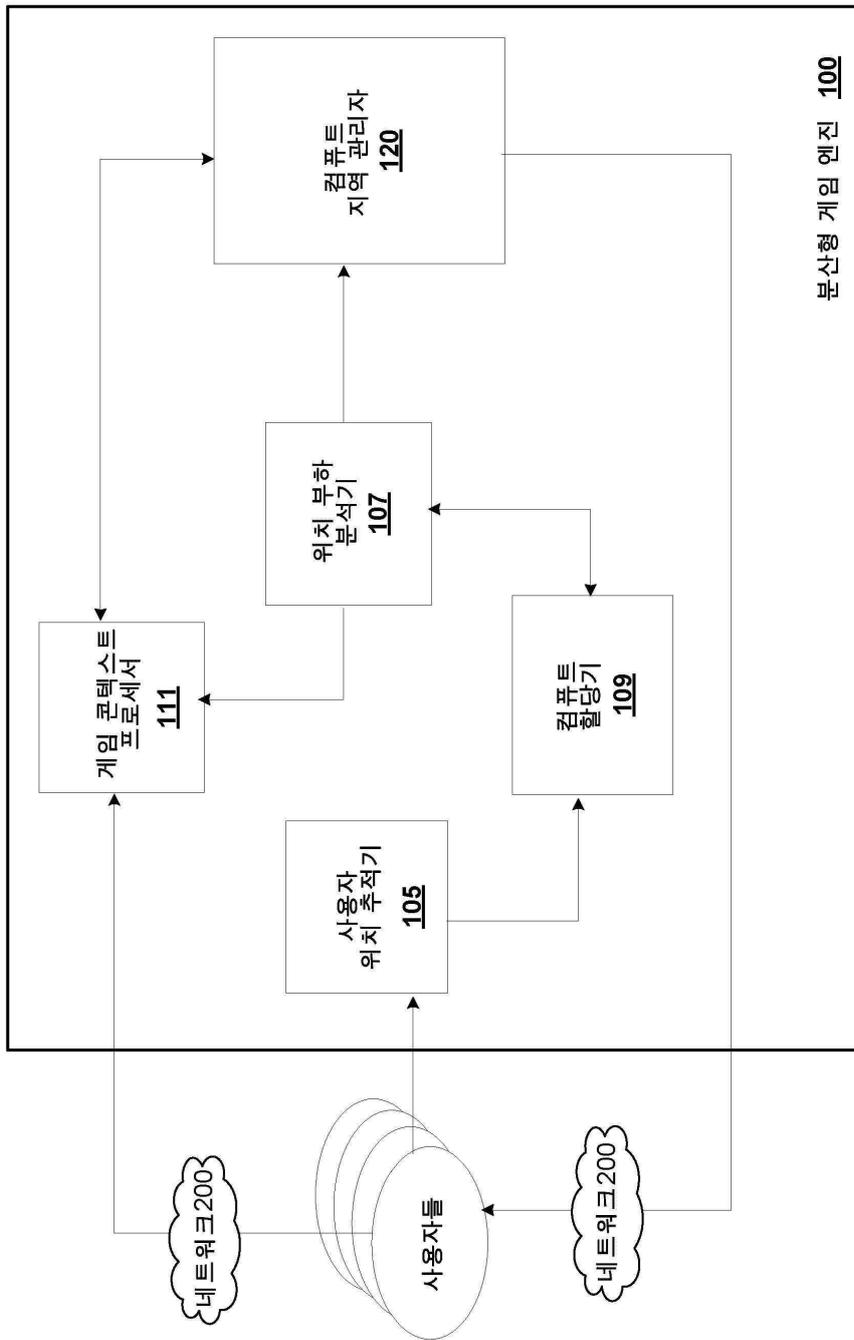
- [0108] 그래픽 서브시스템(713)은 디스플레이 장치(710) 상에 디스플레이될 또는 프로젝션 시스템(740)에 의해 투사될 그래픽 메모리(718)로부터의 이미지에 대한 픽셀 데이터를 주기적으로 출력한다. 디스플레이 장치(710)는 CRT, LCD, 플라즈마 및 OLED 디스플레이들을 포함하는 장치(700)로부터의 신호에 응답하여 시각 정보를 디스플레이할 수 있는 임의의 장치일 수 있다. 장치(700)는 예를 들어, 아날로그 또는 디지털 신호를 디스플레이 장치(710)에 제공할 수 있다.
- [0109] 본원에 정의된 다양한 실시예들은 본원에 개시된 다양한 특징들을 사용하여 특정 구현으로 결합되거나 어셈블링 될 수 있음을 이해해야 한다. 따라서, 제공된 예들은 더 많은 구현예들을 정의하기 위해 다양한 요소들을 결합하여 가능한 다양한 구현예들에 대한 제한 없이, 일부 가능한 예들일 뿐이다. 일부 예들에서, 일부 구현예들은 개시되거나 동등한 구현예들의 정신에서 벗어나지 않고, 더 적은 요소들을 포함할 수 있다.
- [0110] 본 개시의 구현예들은 핸드 헬드 장치, 마이크로프로세서 시스템, 마이크로 프로세서 기반 또는 프로그램 가능 소비자 전자 제품, 미니 컴퓨터, 메인프레임 컴퓨터 등을 포함하는 다양한 컴퓨터 시스템 구성들로 실시될 수 있다. 본 개시의 실시예들은 또한 태스크들이 유선 기반 또는 무선 네트워크를 통해 링크된 원격 처리 장치들에 의해 수행되는 분산형 컴퓨팅 환경들에서 실시될 수 있다.
- [0111] 상기 실시예들을 염두에 두고, 본 개시의 실시예들은 컴퓨터 시스템들에 저장된 데이터를 포함하는 다양한 컴퓨터 구현 동작들을 사용할 수 있다는 것이 이해되어야 한다. 이러한 동작들은 이러한 물리적 수량의 물리적 조작을 필요로 한다. 본 개시의 실시예들의 일부를 형성하는 본원에 설명된 임의의 동작들은 유용한 기계 동작들이다. 본 개시의 실시예들은 또한 이러한 동작들을 수행하는 기기 또는 장치에 관한 것이다. 장치는 요구된 목적들을 위해 특별히 구성될 수 있거나, 장치는 컴퓨터에 저장된 컴퓨터 프로그램에 의해 선택적으로 활성화되거나 구성된 범용 컴퓨터일 수 있다. 특히, 다양한 범용 기계들은 본원의 교시에 따른 작성된 컴퓨터 프로그램들과 함께 사용될 수 있거나, 요구된 동작들을 수행하기 위해 보다 특화된 장치를 구성하는 것이 보다 편리할 수 있다.
- [0112] 본 개시는 또한 컴퓨터 판독 가능 매체 상의 컴퓨터 판독 가능 코드로서 구현될 수 있다. 컴퓨터 판독 가능 매체는 데이터를 저장할 수 있는 임의의 데이터 저장 장치이며, 이는 이후 컴퓨터 시스템에 의해 판독될 수 있다. 컴퓨터 판독 가능 매체의 예들은 하드 드라이브, 네트워크 접속 기반 파일 서버(NAS), 읽기 전용 메모리, 랜덤 액세스 메모리, CD-ROM, CD-R, CD-RW, 자기 테이프 및 기타 광학 및 비광학 데이터 저장 장치들을 포함한다. 컴퓨터 판독 가능 매체는 컴퓨터 판독 가능 코드가 분산된 방식으로 저장되고 실행되도록 네트워크 결합 컴퓨터 시스템에 분산된 컴퓨터 판독 가능 유형 매체를 포함할 수 있다.
- [0113] 방법 동작들이 특정 순서로 설명되었지만, 다른 하우스키핑 동작들이 동작들 사이에 수행될 수 있거나, 또는 동작들은 약간 다른 시간대에서 발생하도록 조정될 수 있거나, 오버레이 동작들의 처리가 원하는 방식으로 수행되는 한, 처리와 관련된 다양한 간격들로 처리 동작들의 발생을 허용하는 시스템에 분산될 수 있음이 이해되어야 한다.
- [0114] 전술한 개시는 이해의 명확성을 위해 일부 상세하게 설명되었지만, 특정 변경들 및 수정들이 첨부된 청구항들의 범위 내에서 실시될 수 있음이 명백할 것이다. 따라서, 본 실시예들은 제한적인 것이 아니라 예시적인 것으로 간주되어야 하며, 본 개시의 실시예들은 본원에 제공된 세부 사항들로 제한되지 않지만, 첨부된 청구항들의 범위 및 등가물 내에서 수정될 수 있다.

도면

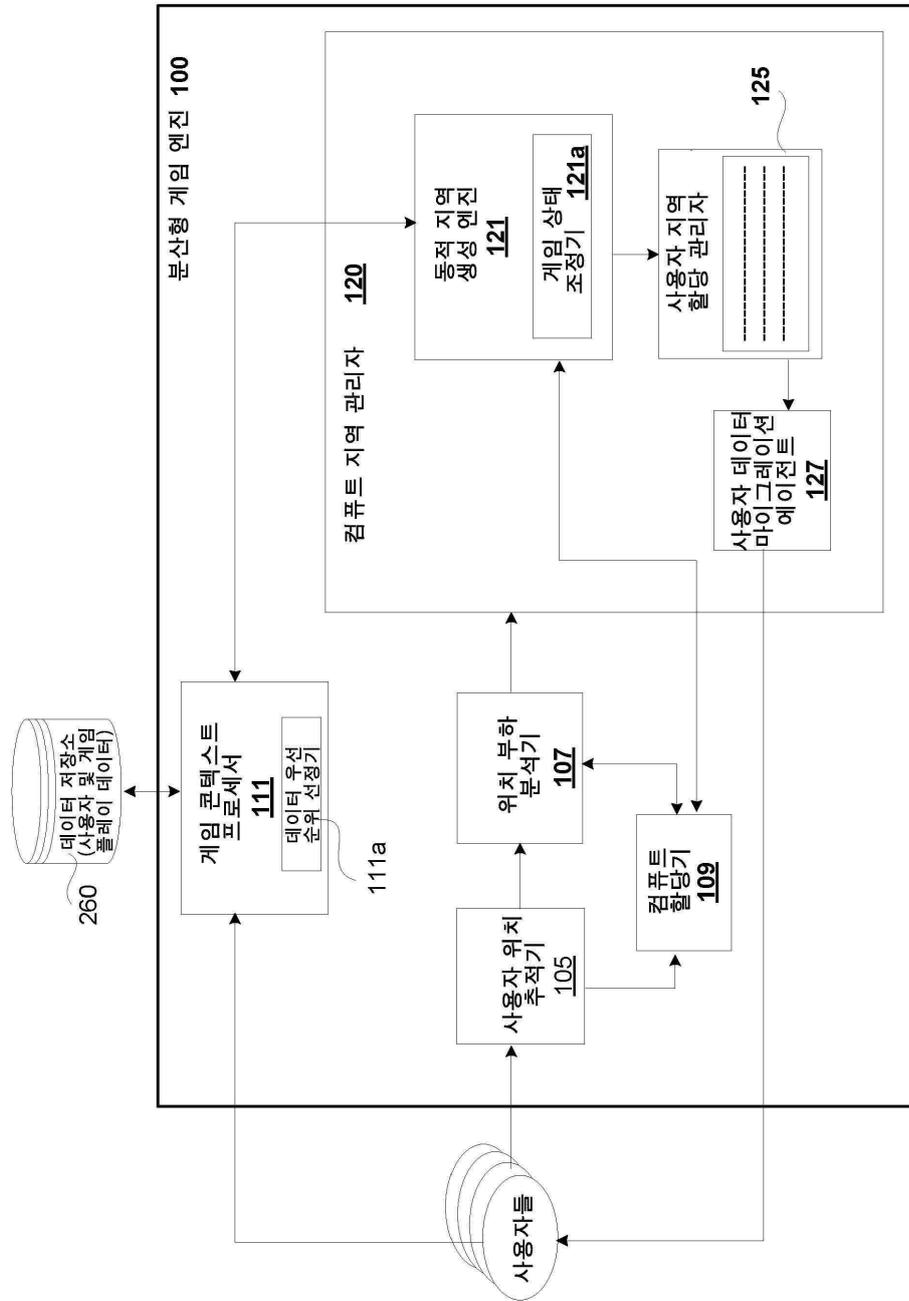
도면1



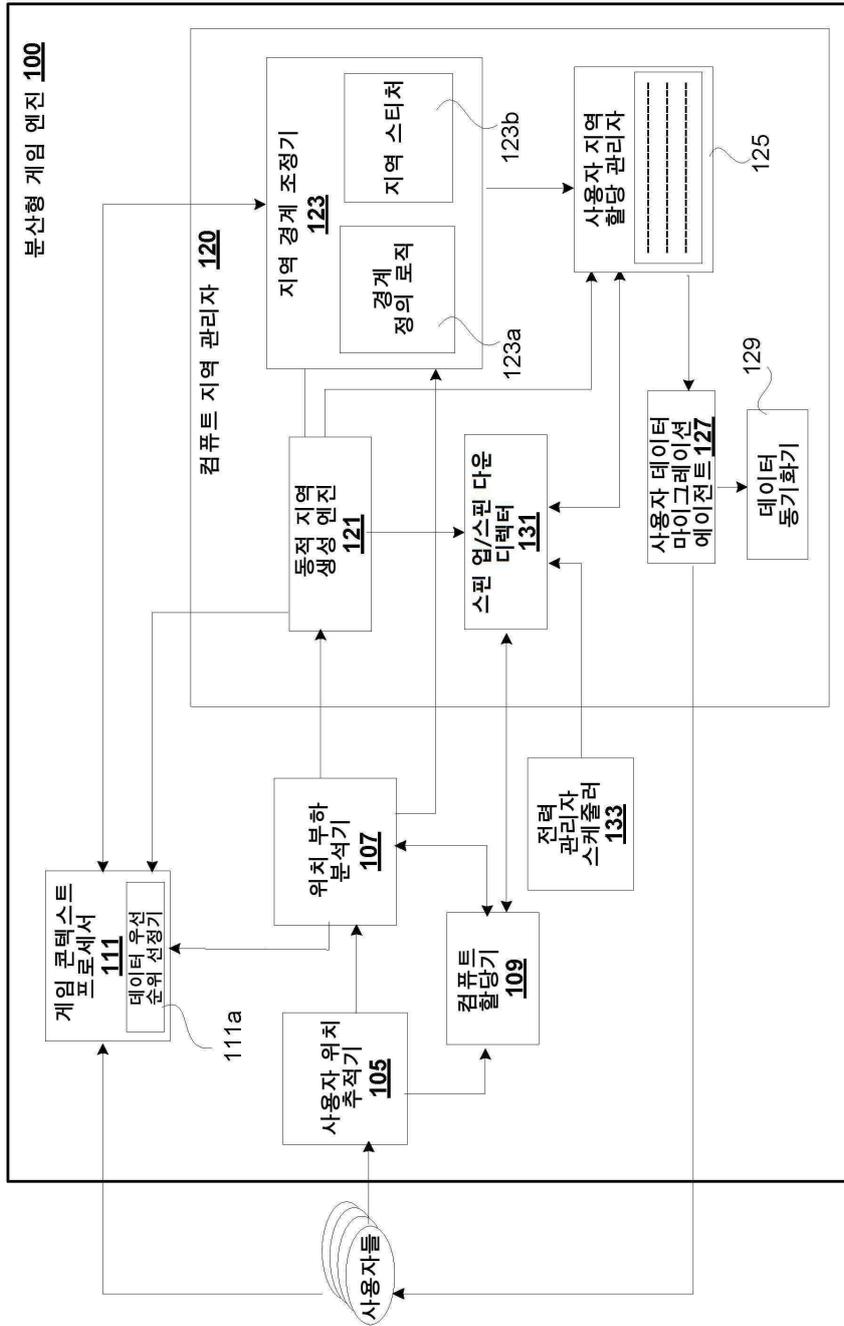
도면2



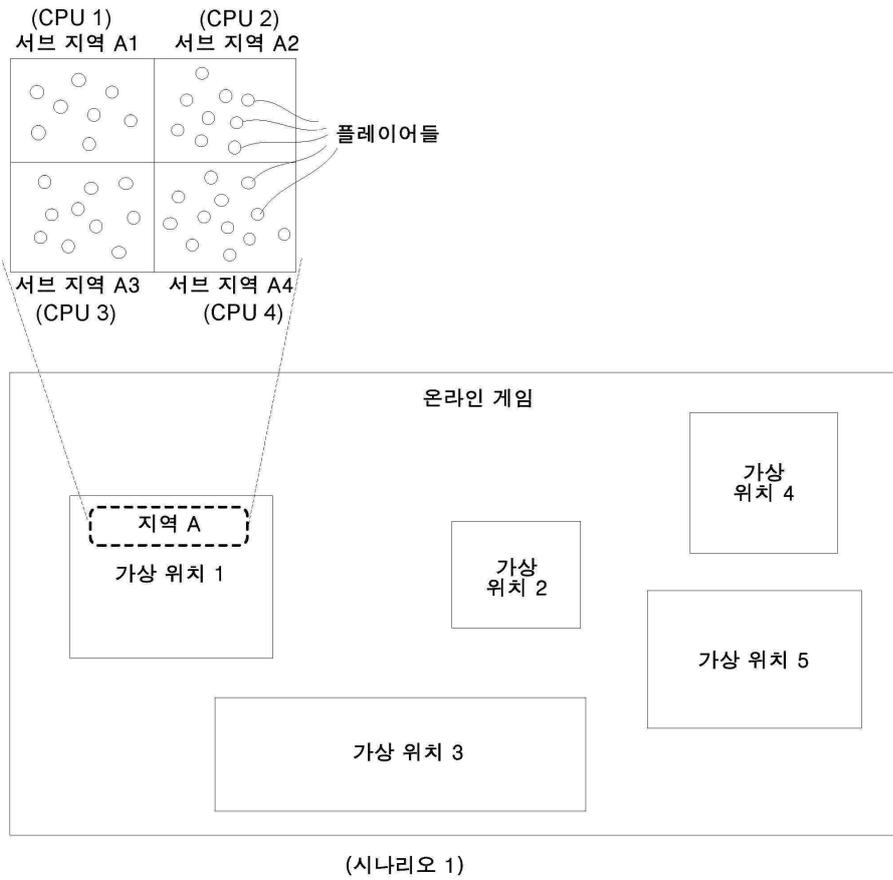
도면3a



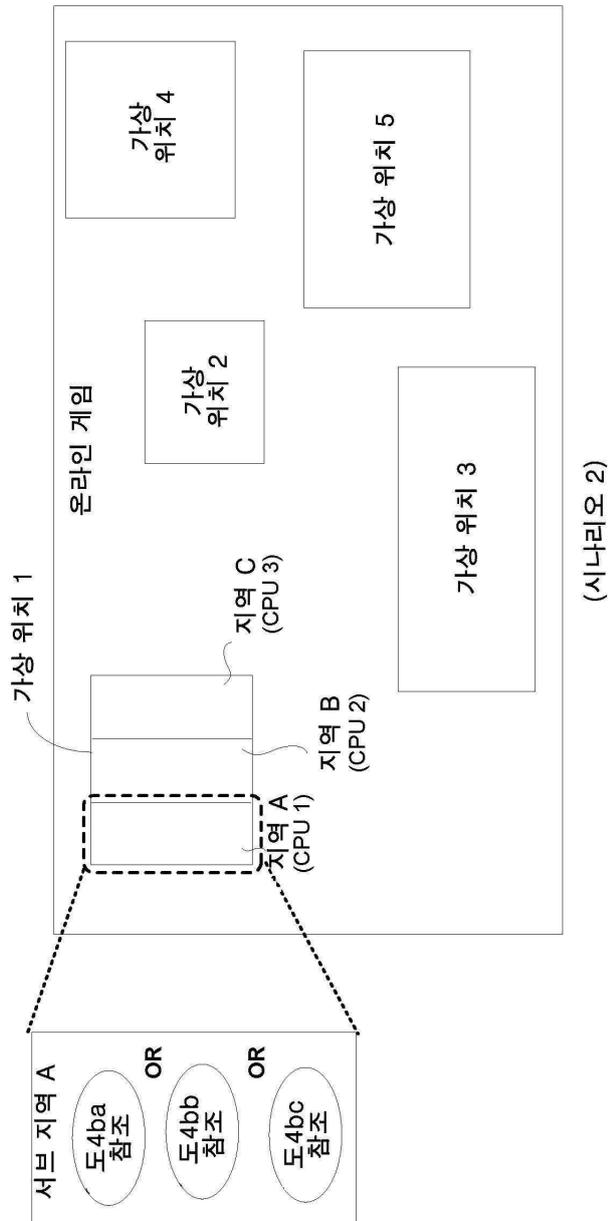
도면3b



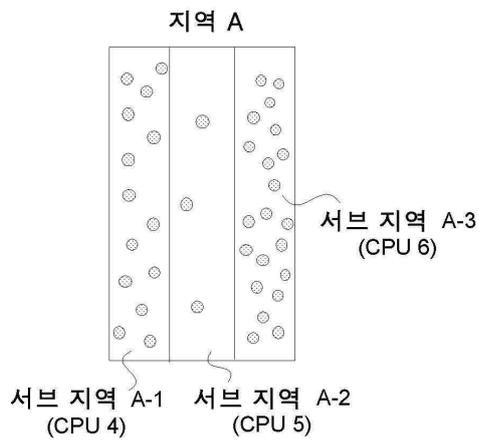
도면4a



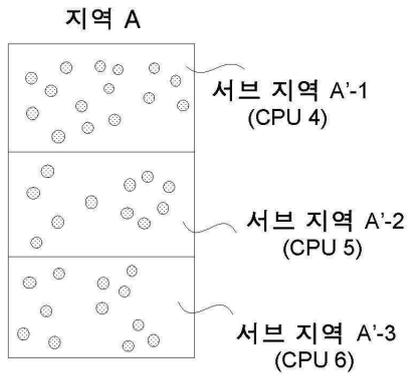
도면4b



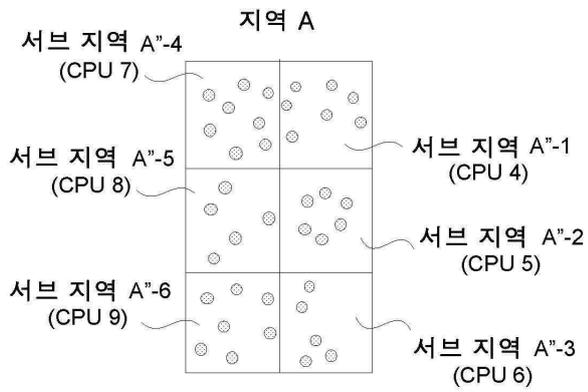
도면4ba



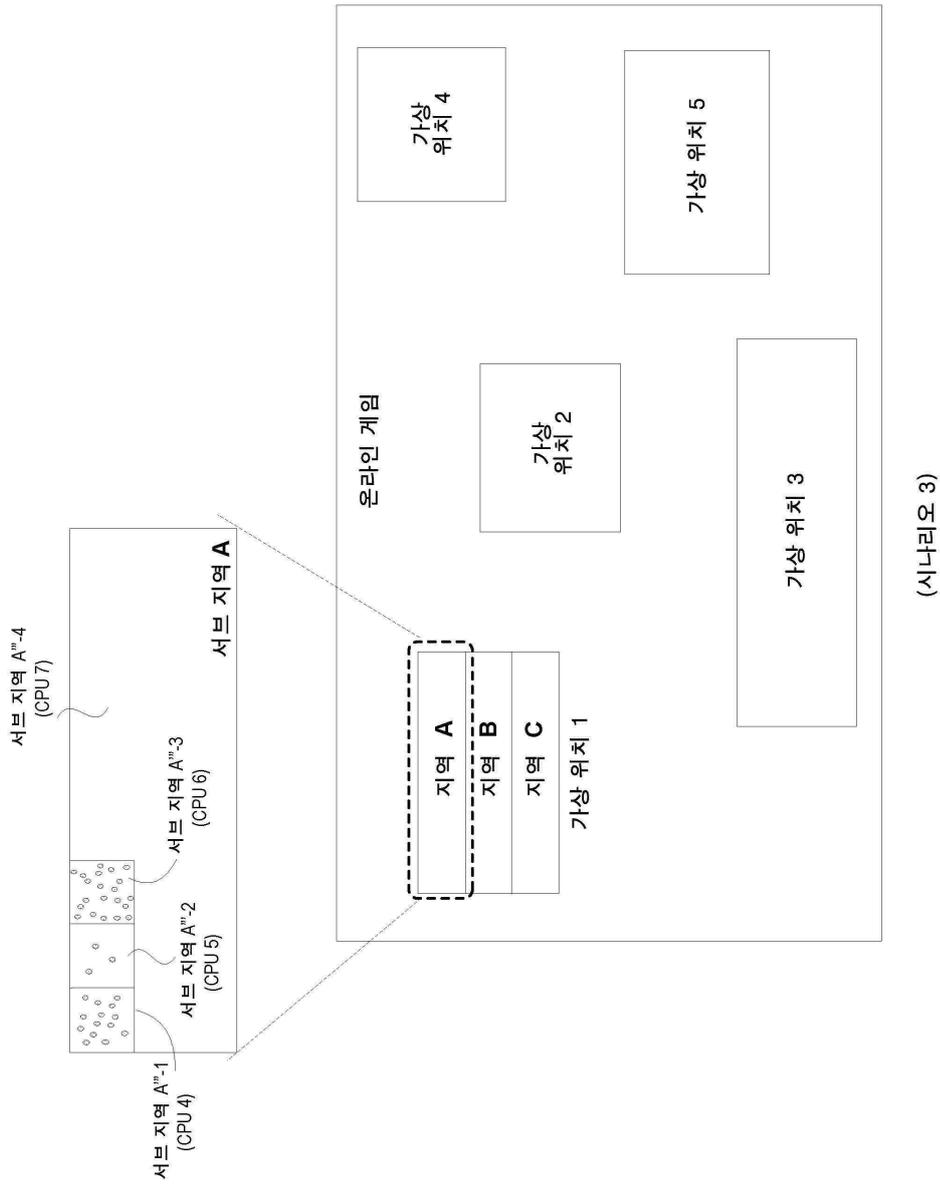
도면4bb



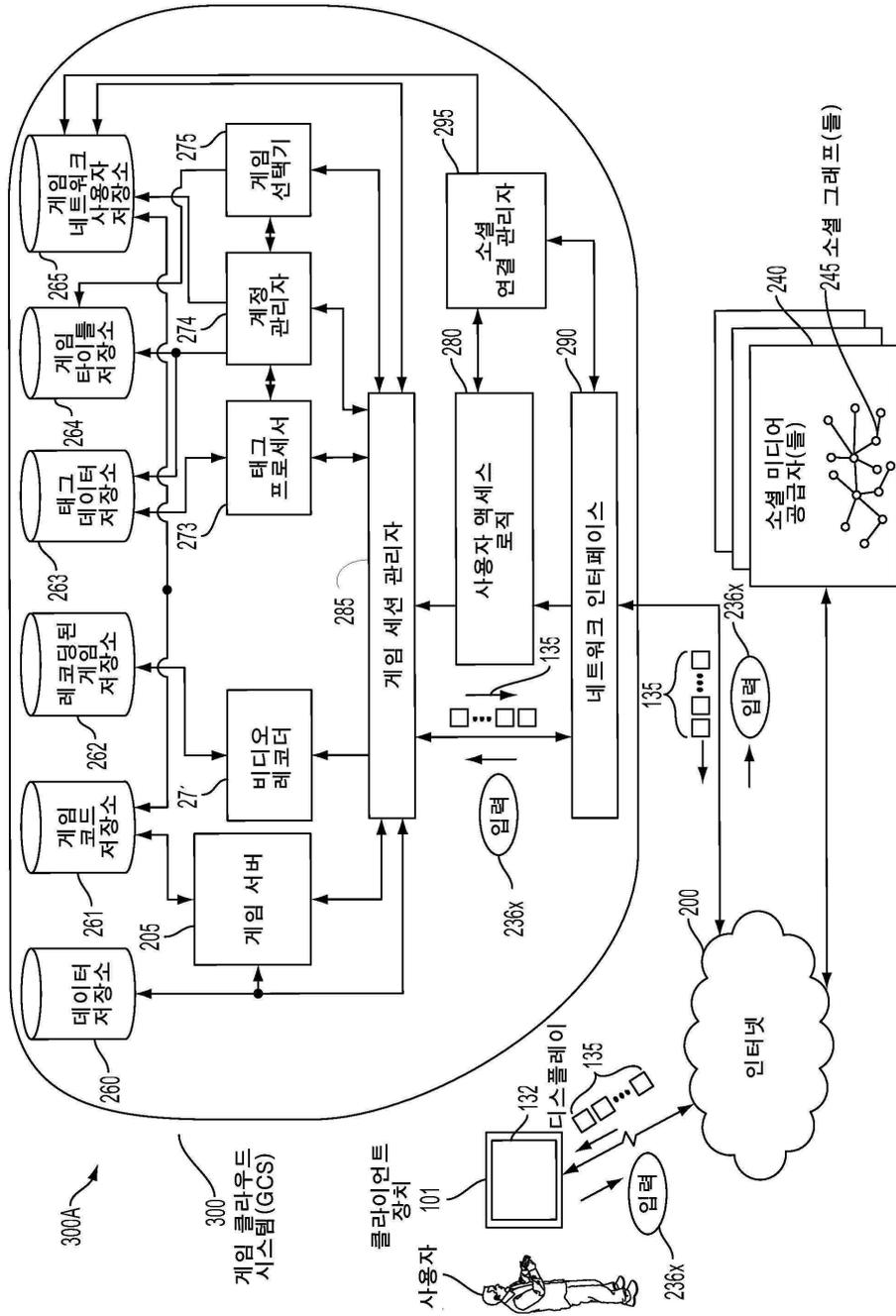
도면4bc



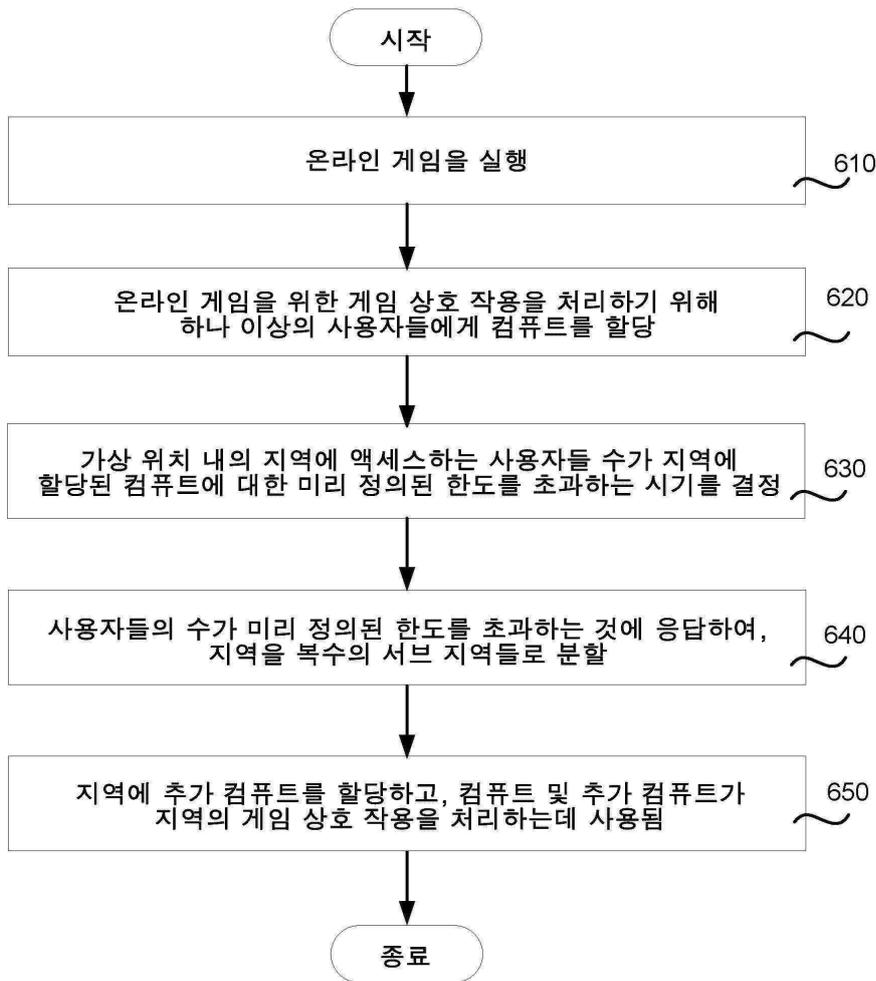
도면4c



도면5



도면6



도면7

