

08

심화 클라우드 아키텍처

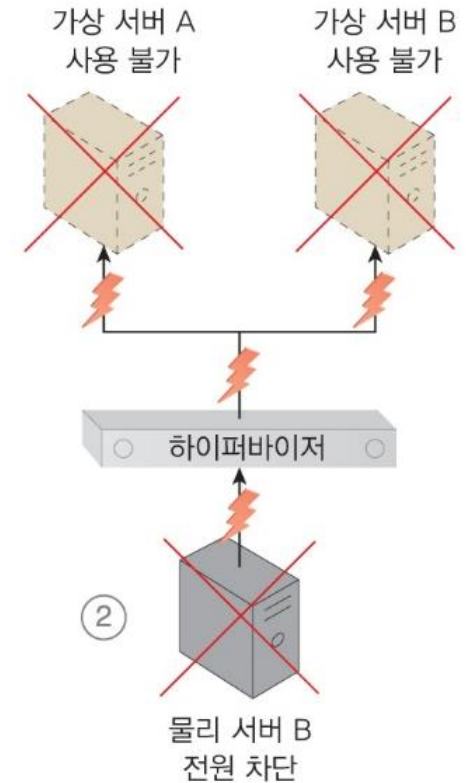
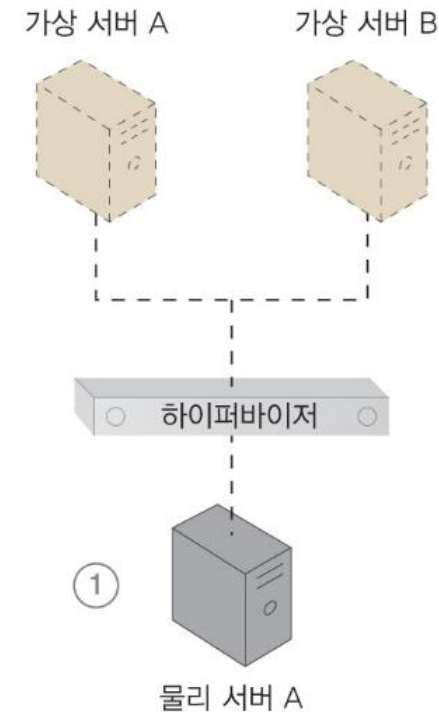
08 Deepening Cloud Architecture

심화 클라우드 아키텍처

- 하이퍼바이저 클러스터링 아키텍처
- 부하 분산 가상 서버 인스턴스 아키텍처
- 무중단 서비스 재배포 아키텍처
- 무정지 아키텍처
- 클라우드 밸런싱 아키텍처
- 자원 예약 아키텍처
- 동적 장애 감지 및 복구 아키텍처
- 베어 메탈 프로비저닝 아키텍처
- 신속한 프로비저닝 아키텍처
- 스토리지 워크로드 관리 아키텍처

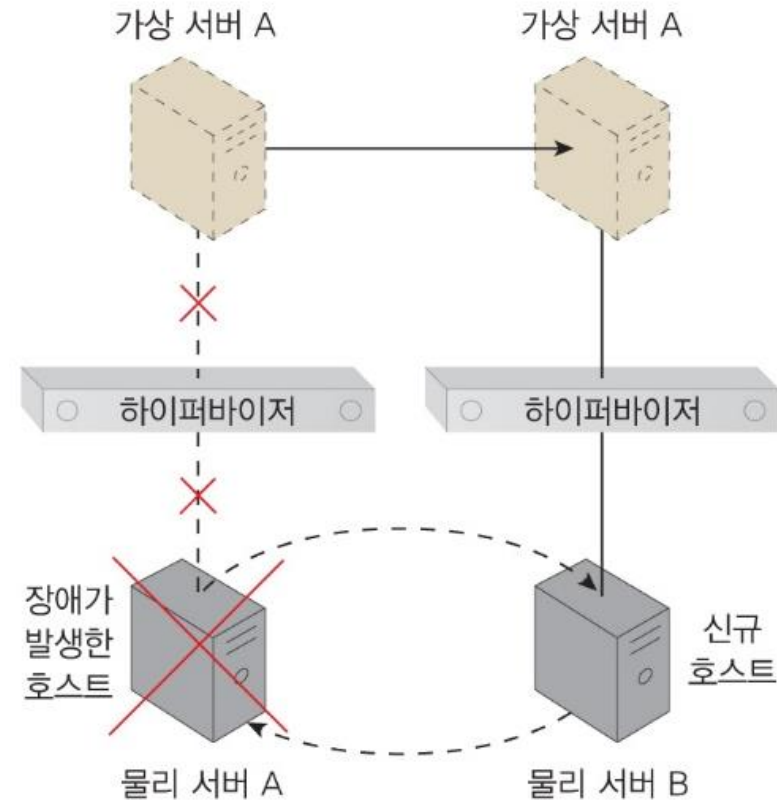
하이퍼바이저 클러스터링 아키텍처

- 하이퍼바이저는 다중 가상 서버를 생성 및 제공
- 하이퍼바이저에 영향을 미치는 장애 조건들은 하이퍼바이저 상의 가상 서버에도 파급효과를 미칠 수 있음



하이퍼바이저 클러스터링 아키텍처

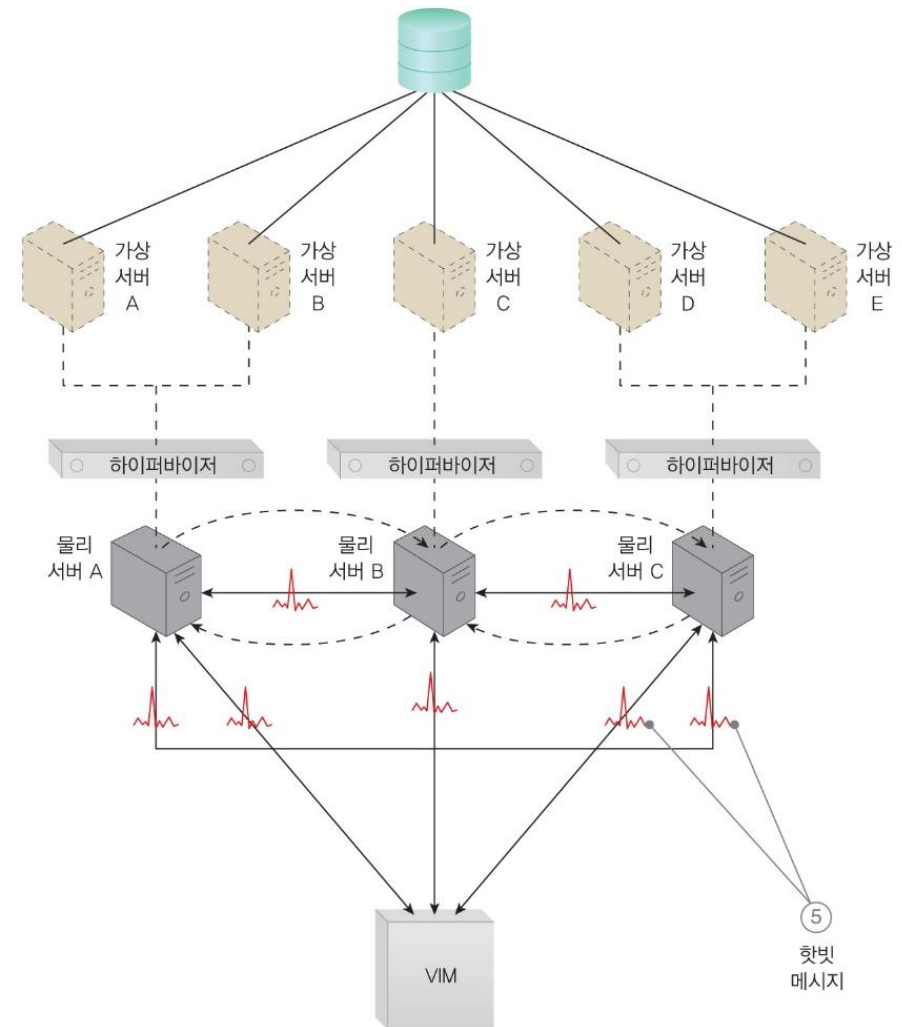
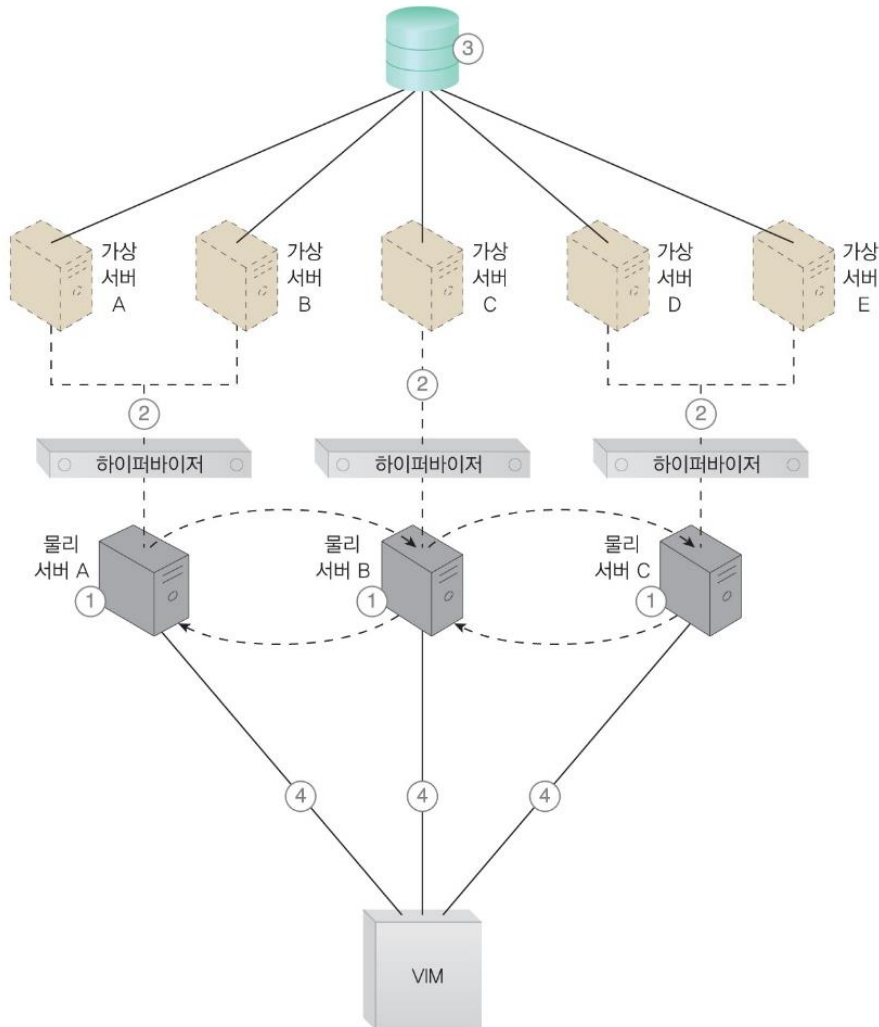
- 하이퍼바이저 클러스터링 아키텍처는 다중 물리 서버간 하이퍼바이저의 고가용성 클러스터를 형성
- 주어진 하이퍼바이저나 기존 물리 서버가 사용할 수 없게 되면, 제공되는 가상 서버는 런타임 동작을 유지하기 위해 이외의 물리 서버나 하이퍼바이저로 이관될 수 있음



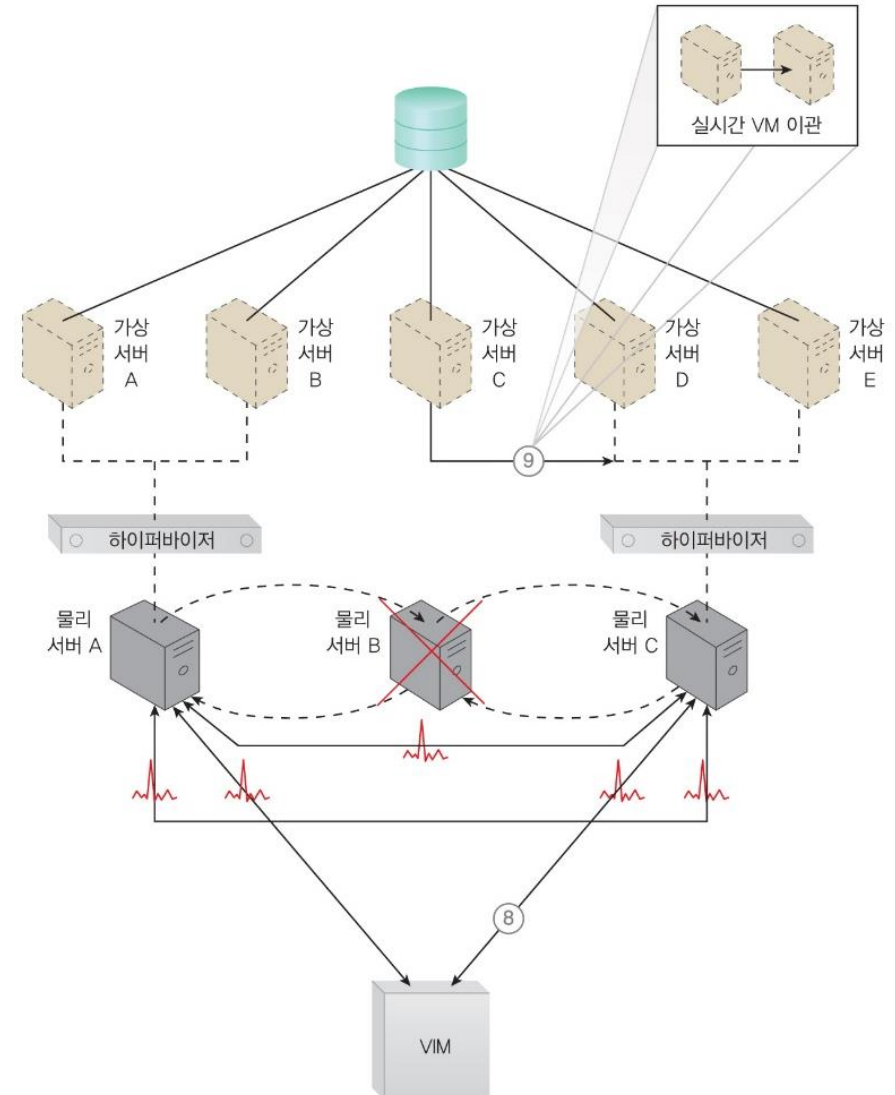
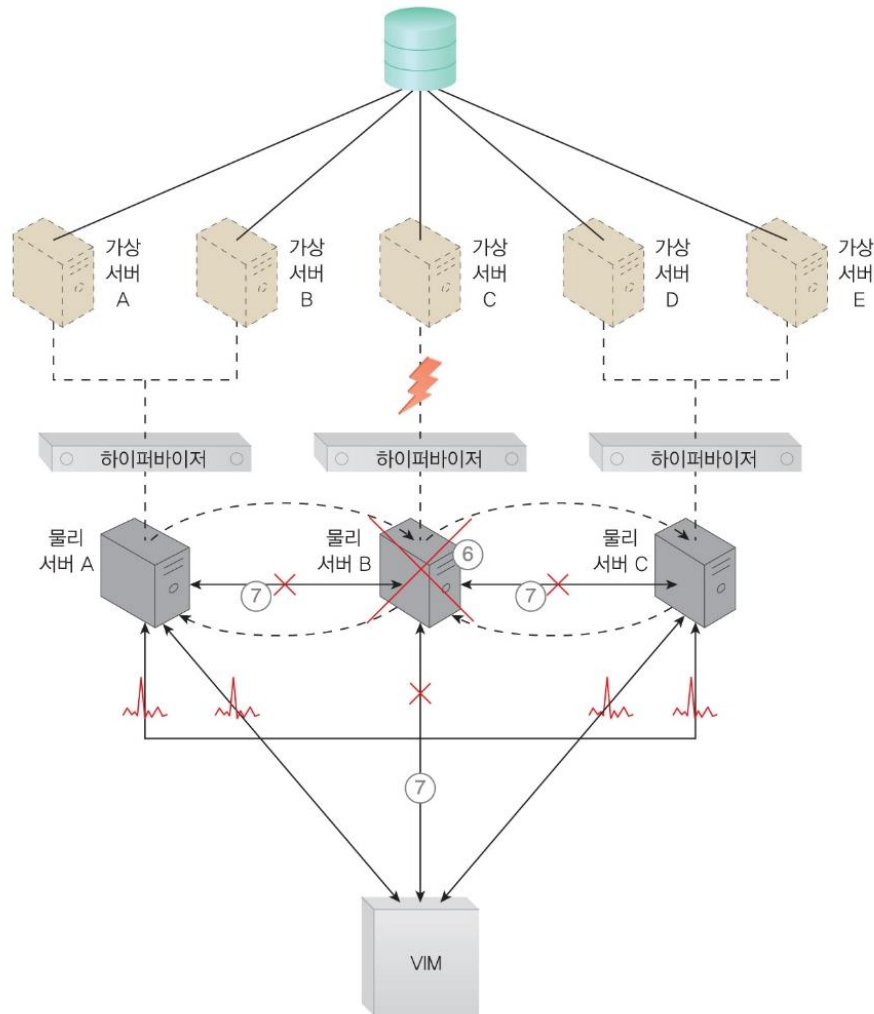
하이퍼바이저 클러스터링 아키텍처

- 하이퍼바이저 클러스터는 하이퍼바이저가 정상으로 운영되고 있음을 확인하기 위해 중앙 VIM에서 정기적으로 heartbeats 메시지를 보내 통제됨
- 응답받지 못한 heartbeats 메시지는 VIM에서 동적으로 가상 서버들을 신규 호스트로 이관하기 위해 실시간 VM 이관 프로그램을 초기화

하이퍼바이저 클러스터링 아키텍처



하이퍼바이저 클러스터링 아키텍처



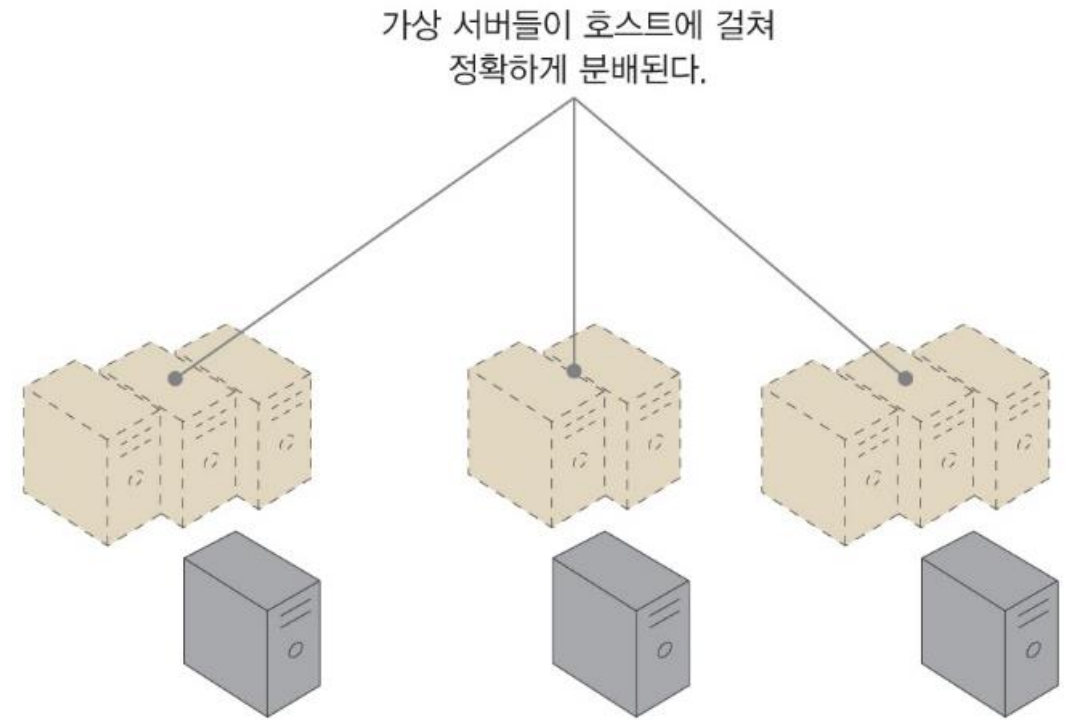
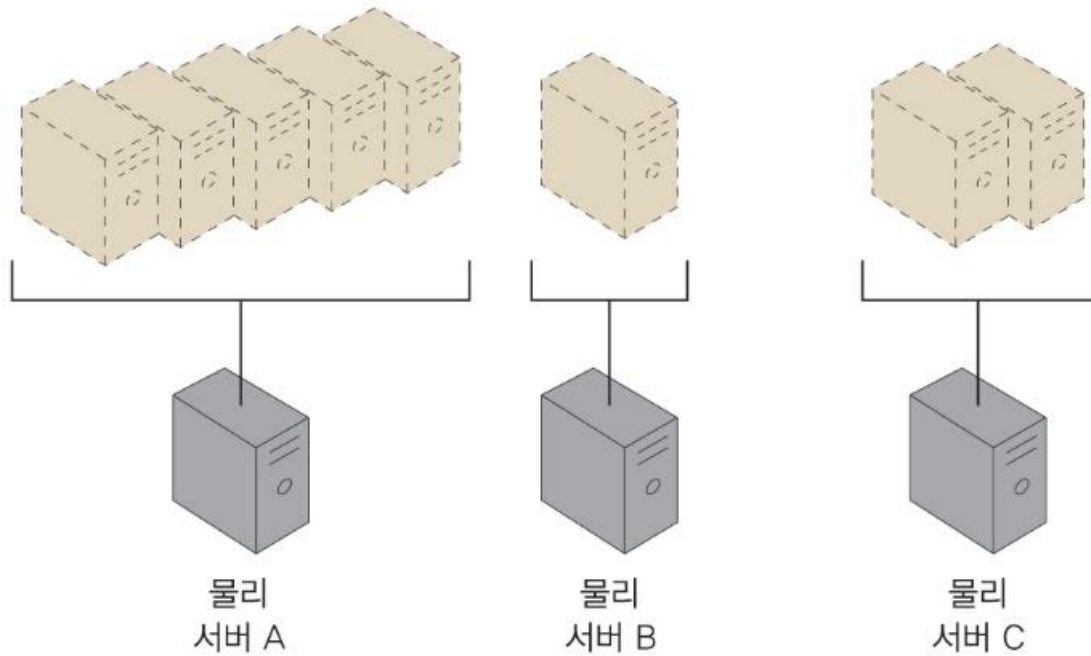
하이퍼바이저 클러스터링 아키텍처

- 클러스터 환경 내에서 보호되는 이러한 아키텍처 모델과 가상 서버들의 핵심은 하이퍼바이저와 자원 클러스터 메커니즘
- 논리 네트워크 경계 메커니즘에 의해 생성된 논리적 경계는 타 클라우드 소비자의 하이퍼바이저가 우연히 주어진 클러스터 내에 포함되지 않도록 보장
- 동일한 클러스터 내 하이퍼바이저들은 서로 그들의 상태와 가용성에 대한 정보를 주고 받는데, 가상 스위치의 생성이나 삭제와 같은 클러스터 내 발생한 변경 사항에 대한 갱신 정보는 VIM을 통해 모든 하이퍼바이저에 복제될 필요가 있음 (자원 복제 메커니즘)

부하 분산 가상 서버 인스턴스 아키텍처

- 운영 및 관리 작업이 분리된 물리 서버들간에 교차 서버 워크로드가 균등하게 분산되도록 유지하는 일은 까다로움
- 특정 물리 서버가 이웃한 물리 서버들보다 더 많은 가상 서버를 제공하거나, 더 많은 워크로드를 분배받는 결과가 쉽게 발생 가능
- 활용도가 적정 수준을 초과하거나 이에 미달하는 물리 서버들은 시간이 지남에 따라 엄청나게 많아질 수 있음
 - 초과 활용되는 서버들에게는 실제 성능상의 문제 발생
 - 활용도가 수준 이하인 서버들의 경우에는 잠재적인 처리 용량의 손실로 지속적인 낭비를 초래

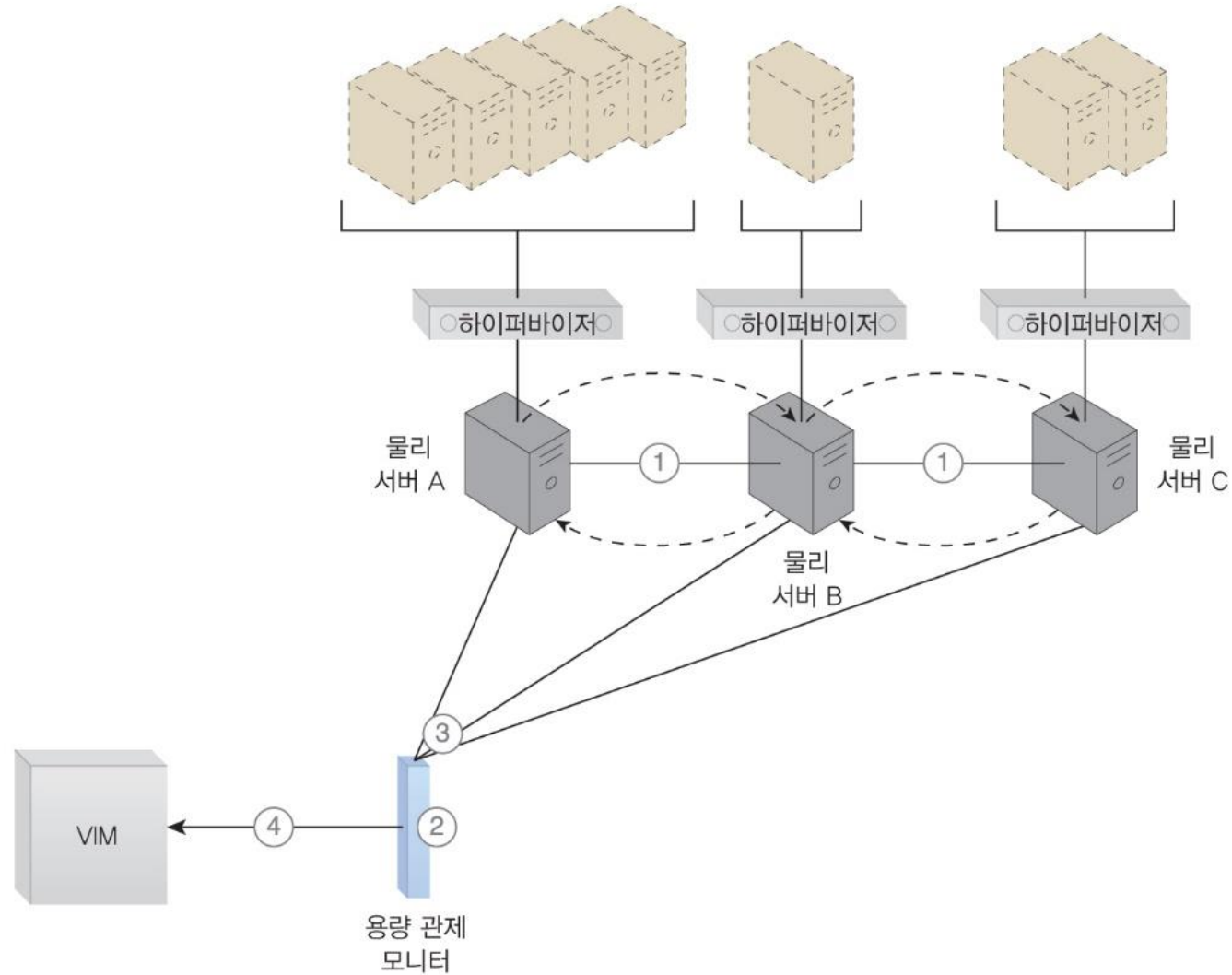
부하 분산 가상 서버 인스턴스 아키텍처



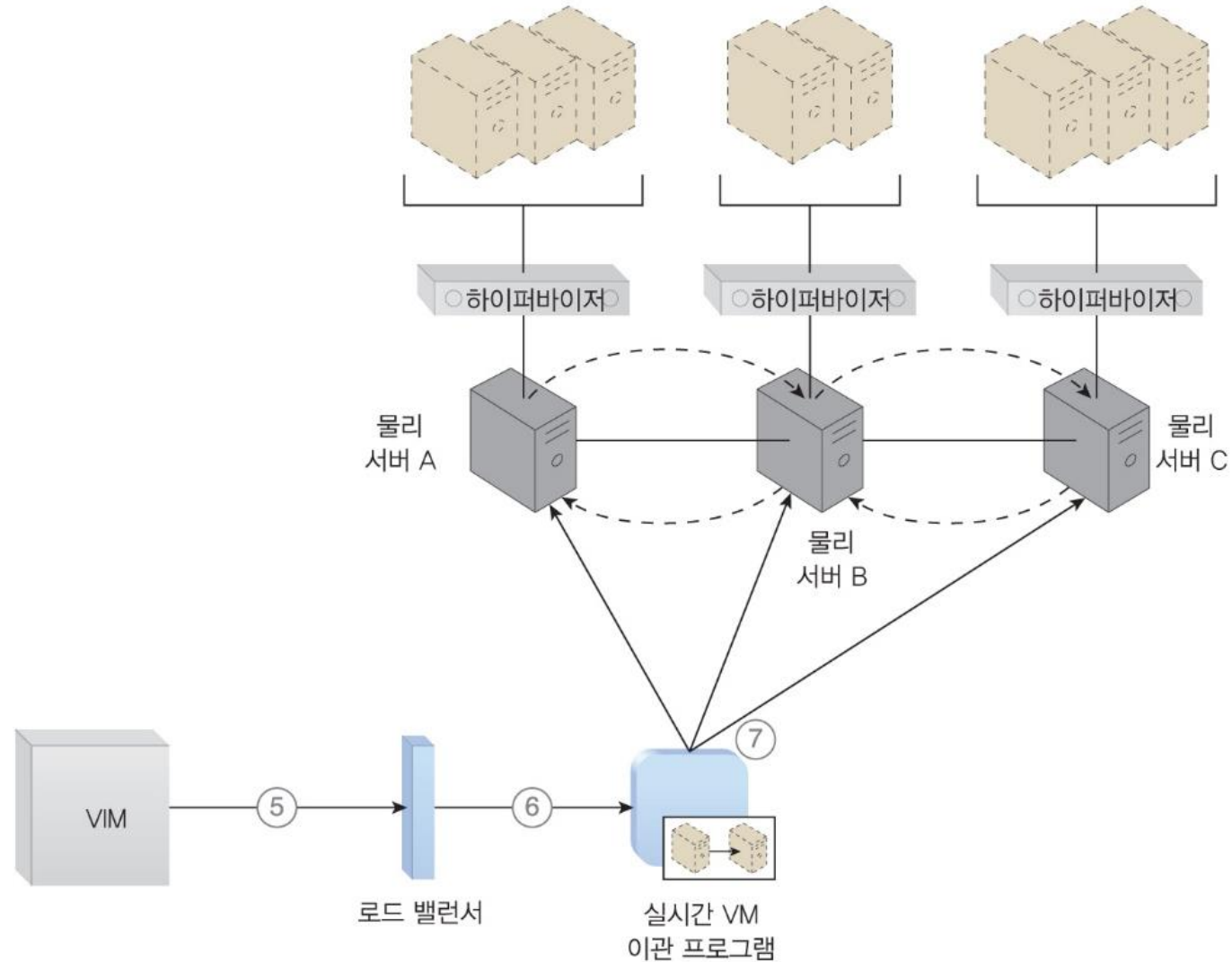
부하 분산 가상 서버 인스턴스 아키텍처

- 부하 분산 가상 서버 인스턴스 아키텍처는 가용 물리 서버 호스트 간 처리를 분산시키기 전에 동적으로 가상 서버 인스턴스와 관련 워크로드를 계산하는 용량 관제 시스템을 형성
- 용량 관제 시스템은 용량 관제 클라우드 사용 모니터, 실시간 VM 이관 프로그램, 용량 계획 도구로 구성
- 용량 관제 모니터는 물리 및 가상 서버 사용량을 추적
- 용량 계획 도구는 가상 서버 용량 요구사항에 대하여 동적으로 물리 서버 컴퓨팅 자원의 용량을 계산하여 변동 사항을 보고함
- 용량 계획 도구가 워크로드를 분배하기 위해 가상 서버를 또 다른 호스트로 이관하도록 결정하면 실시간 VM 이관 프로그램에 가상 서버를 이관하도록 신호를 보냄

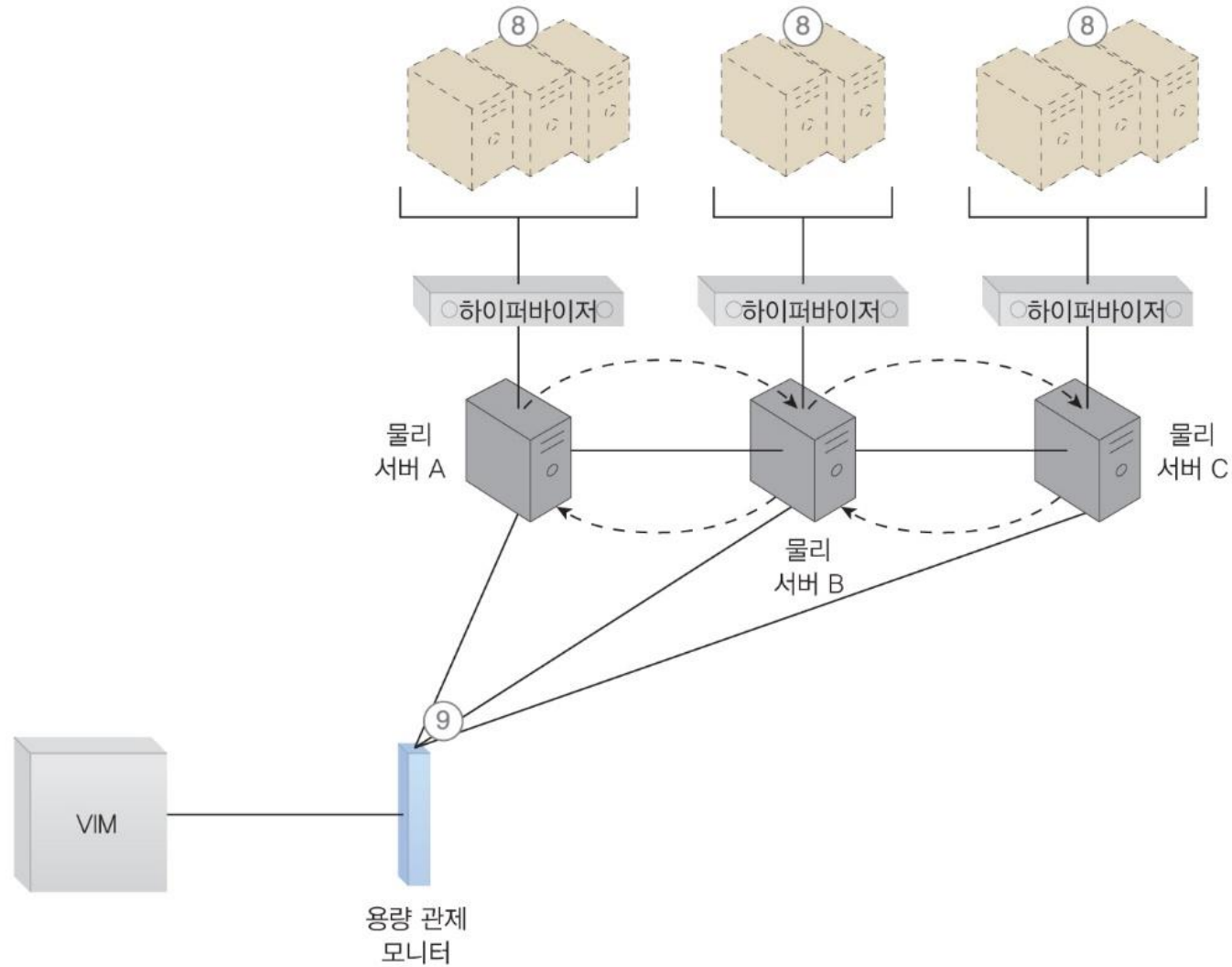
부하 분산 가상 서버 인스턴스 아키텍처



부하 분산 가상 서버 인스턴스 아키텍처



부하 분산 가상 서버 인스턴스 아키텍처



부하 분산 가상 서버 인스턴스 아키텍처

- 자동 확장 리스너는 부하 분산 프로세스를 초기화하고 하이퍼바이저를 통해 가상 서버들에게 오는 워크로드를 동적으로 감시하는데 사용
- 로드 밸런서 메커니즘은 하이퍼바이저 간 가상 서버들의 워크로드를 분배할 책임이 있음
- 논리 네트워크 경계는 재배치된 가상 서버의 목적지가 SLA와 개인 정보 보호 규제를 준수함을 보장
- 자원 복제는 가상 서버 인스턴스의 복제가 부하 분산 기능의 일부로서 필요하게 됨

무중단 서비스 재배치 아키텍처

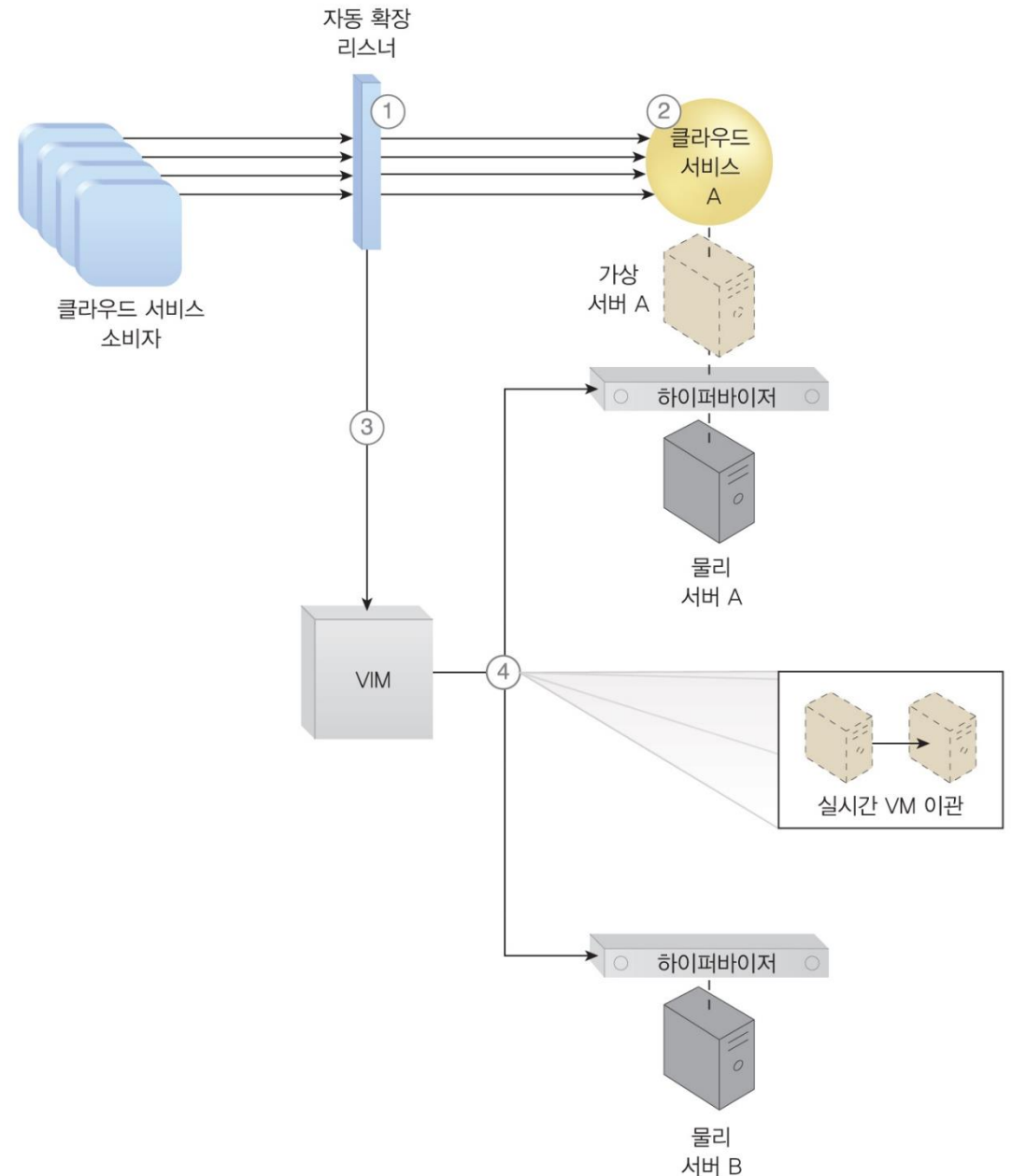
- 클라우드 서비스는 다양한 이유로 중단되는 경우가 발생
 - 런타임 사용 수요가 처리 용량을 초과하는 경우
 - 일시적인 다운 타임을 필수적으로 요구하는 유지 보수 갱신 작업
 - 신규 물리 서버 호스트로의 영구적인 이관
- 무중단 서비스 재배치 아키텍처는 정의된 이벤트 발생 시 자동으로 런타임에 클라우드 서비스 구현체를 복제 또는 이관하여 다운 타임을 방지하는 시스템을 확립
- 이중화된 구현체를 활용하여 클라우드 서비스를 확장하거나 축소하는 대신, 일시적으로 신규 호스트에 복제된 구현체를 추가해서 클라우드 서비스 활동이 런타임에 나머지 호스팅 환경으로 우회될 수 있음
- 유사하게 원본 구현체에서 계획된 다운 타임이 필요한 유지 보수 작업을 진행해야 할 경우 클라우드 서비스 소비자 요청을 일시적으로 복제된 구현체로 재전송할 수 있음

무중단 서비스 재배치 아키텍처

- 클라우드 서비스 구현체와 클라우드 서비스 활동의 재배치는 신규 물리 서버 호스트로 클라우드 서비스의 이관을 수용하기 위해 영구적으로 될 수 있음
- 아키텍처의 핵심적인 특징은 기존의 원본 클라우드 서비스 구현체가 비활성화되거나 제거되기 전에 신규 클라우드 서비스 구현체가 클라우드 서비스 소비자의 요청을 성공적으로 수용 및 대응할 수 있도록 보장
- 일반적인 접근법은 클라우드 서비스를 제공하고 있는 전체 가상 서버 인스턴스를 실시간 VM 이관 프로그램이 이관하도록 함
- 자동 확장 리스너와 로드 밸런서 메커니즘이 확장 및 워크로드 분배 요구사항에 대응하여 일시적으로 클라우드 서비스 소비자 요청의 재전송을 개시하는 데 사용될 수 있음

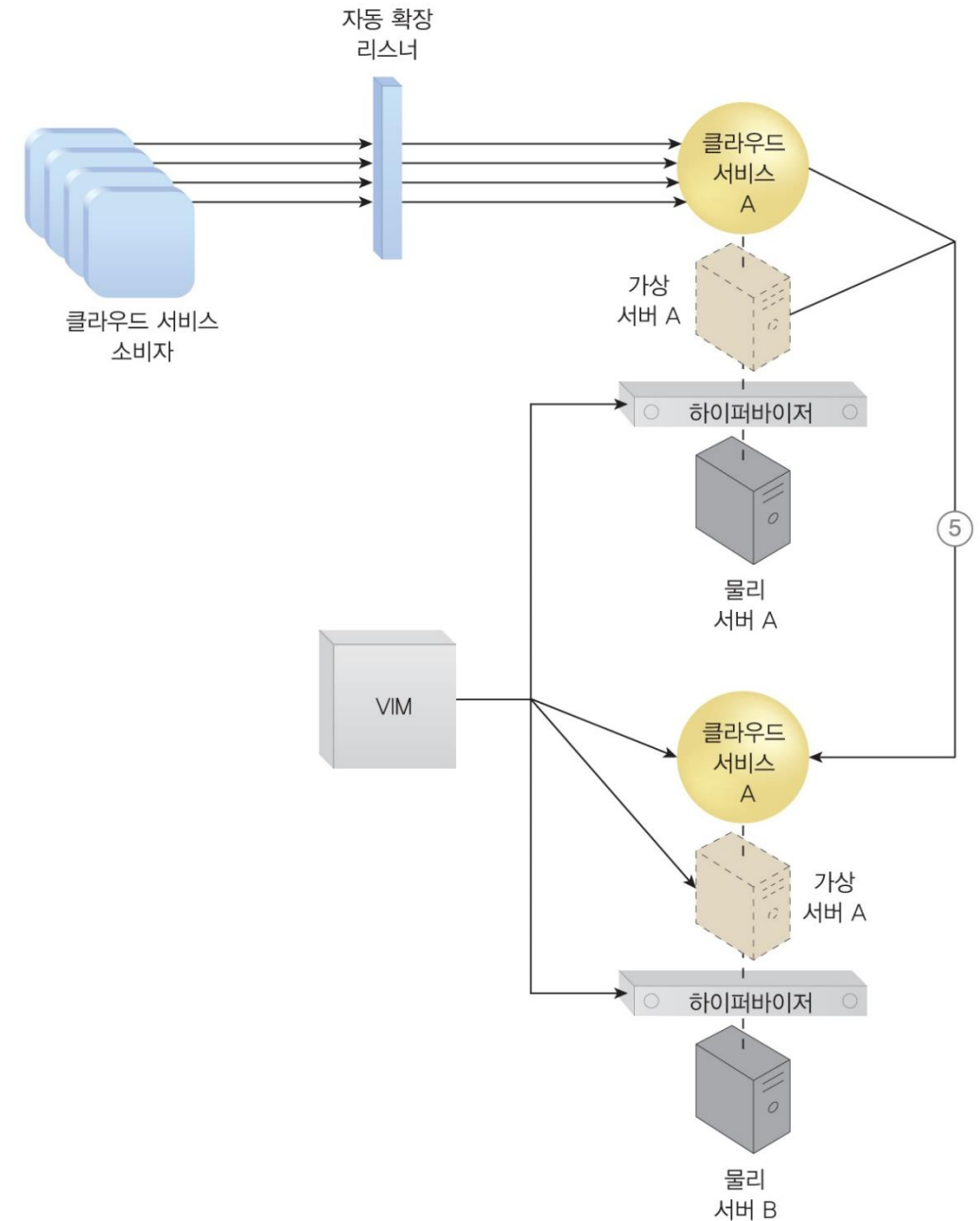
무중단 서비스 재배치 아키텍처

- ① 자동 확장 리스너가 특정 클라우드 서비스를 위해 워크로드 감시
- ② 워크로드가 증가함에 따라 클라우드 서비스의 선 정의된 임계치에 도달
- ③ 자동 확장 리스너가 재배치를 개시하기 위해 VIM에 신호를 보냄
- ④ VIM은 런타임 재배치를 수행하기 위해 기존 및 목적지 하이퍼바이저에 지시하는 데 실시간 VM 이관 프로그램 사용



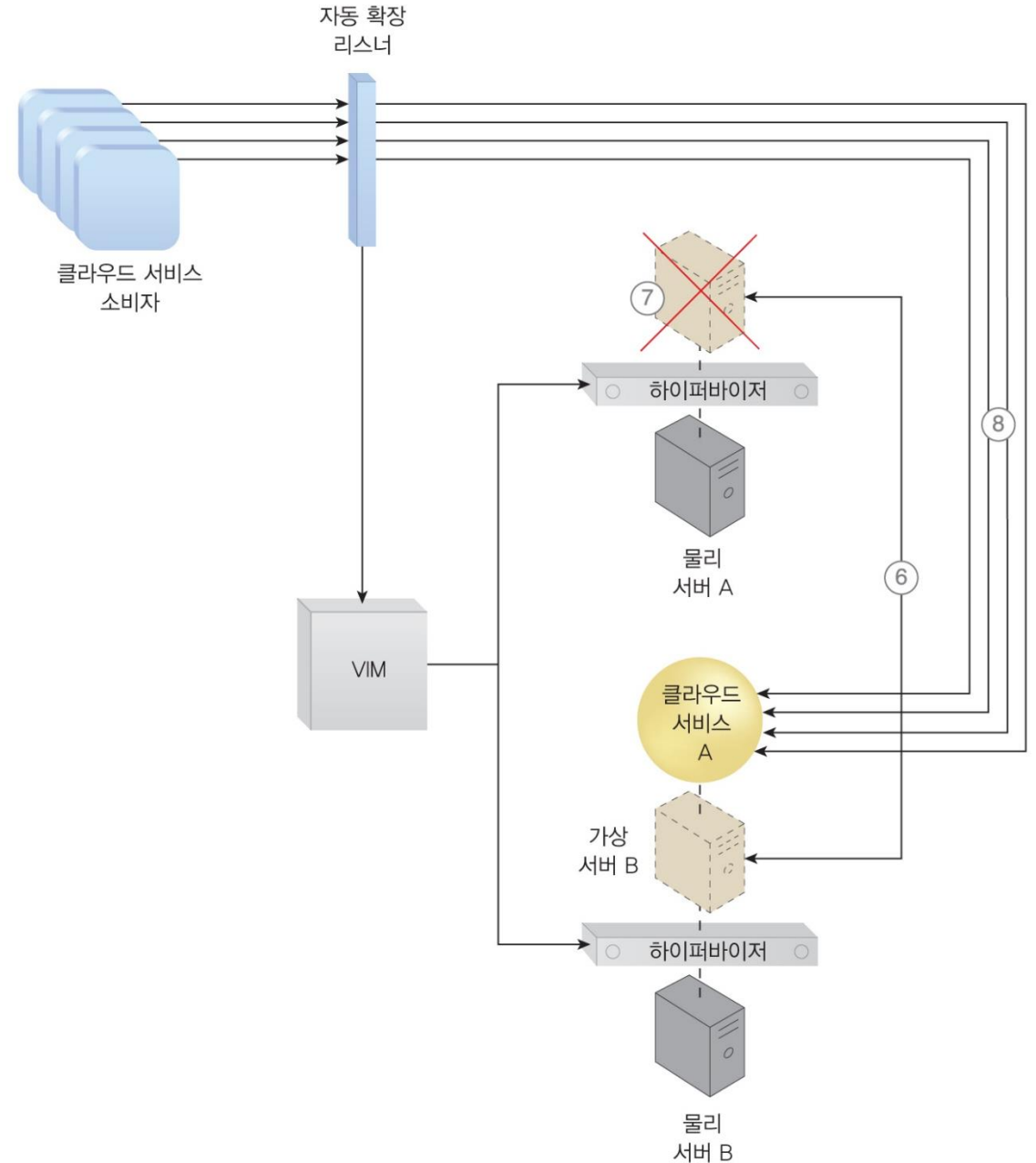
무중단 서비스 재배치 아키텍처

- ⑤ 가상 서버와 제공되는 클라우드 서비스의 복제본이 물리 서버 B 상의 목적지 하이퍼바이저를 통해 생성



무중단 서비스 재배치 아키텍처

- ⑥ 양쪽 가상 서버 인스턴스의 상태는 동기화
- ⑦ 첫 번째 가상 서버 인스턴스는 클라우드 서비스 소비자의 요청이 성공적으로 물리 서버 B에 상주하는 클라우드 서비스로 교체되어 처리됨을 확인 후 물리 서버 A에서 삭제
- ⑧ 클라우드 서비스 소비자의 요청은 이제 물리 서버 B 상의 클라우드 서비스로만 전달



무중단 서비스 재배치 아키텍처

- 가상 서버 이관은 가상 서버 디스크의 위치와 구성에 따라 두 가지 방식으로 진행
- 가상 서버 디스크가 로컬 스토리지 장치에 저장되거나, 비공유 원격 스토리지 장치가 소스 호스트에 장착되어 있을 경우
 - 가상 서버 디스크의 복제본이 목적지 호스트상에 생성
 - 복제본이 생성되고 나면, 양쪽 가상 서버 인스턴스는 동기화되고 가상 서버 파일이 기존 호스트에서 제거
- 가상 서버의 파일들이 기존 및 목적지 호스트 간 공유된 원격 스토리지 장치에 저장되는 경우
 - 가상 서버 디스크를 복제할 필요는 없음
 - 단순히 가상 서버의 소유권이 기존에서 목적지 물리 서버 호스트로 이전되고, 가상 서버의 상태가 자동으로 동기화

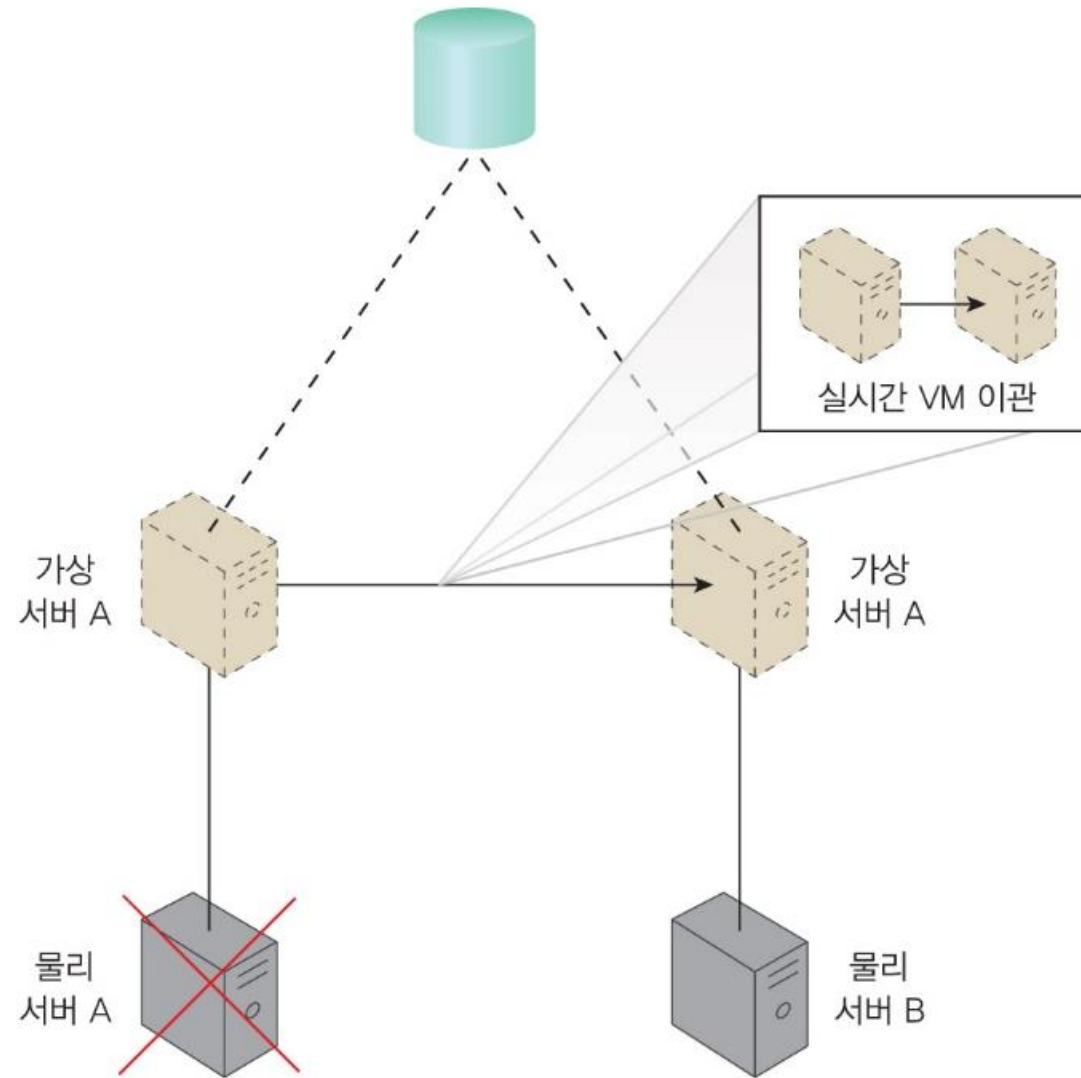
무중단 서비스 재배치 아키텍처

- 여러 종류의 클라우드 사용 모니터가 지속적으로 IT 자원 사용량과 시스템 활동을 추적하는 데 사용될 수 있음
- 사용량당 과금 모니터는 기존 및 목적지의 IT 자원에 대한 서비스 사용 비용을 계산하기 위해 데이터를 수집하는 데 사용
- 자원 복제 메커니즘은 목적지의 클라우드 서비스 복제본을 인스턴스화 하는데 사용
- SLA 관리 시스템은 클라우드 서비스 복제나 재배치 도중 및 이후에 클라우드 서비스의 가용성을 보장하기 위해 SLA 모니터에 의해 제공되는 SLA 데이터를 처리할 책임이 있음
- SLA 모니터링 메커니즘은 가용성 보장을 위해 연관성이 있을 수 있는 SLA 관리 시스템이 필요로 하는 SLA 정보를 수집

무정지 아키텍처

- 무정지 아키텍처는 기존 물리 서버 호스트가 중단되는 이벤트가 발생했을 때, 다른 물리 서버 호스트로 가상 서버들을 동적으로 이관할 수 있도록 하는 정교한 대체 작동 시스템을 확립
- 여러 물리 서버들이 하나의 그룹을 이루어 중단 없이 한 물리 서버에서 다른 물리 서버로 전환하는 활동을 수행할 수 있는 장애 방지 시스템을 통해 통제
- 실시간 VM 이관 컴포넌트는 일반적으로 이런 형태의 고가용성 클라우드 아키텍처의 핵심적인 부분이 됨
- 장애 방지 시스템은 물리 서버가 중단되는 경우 해당 서버에서 제공되던 가상 서버들이 보조 물리 서버로 이관될 것을 보장
- 모든 가상 서버들은 공유된 스토리지 볼륨에 저장되어 같은 그룹 내에 속한 다른 물리 서버들이 관련 파일에 접근할 수 있음

무정지 아키텍처



무정지 아키텍처

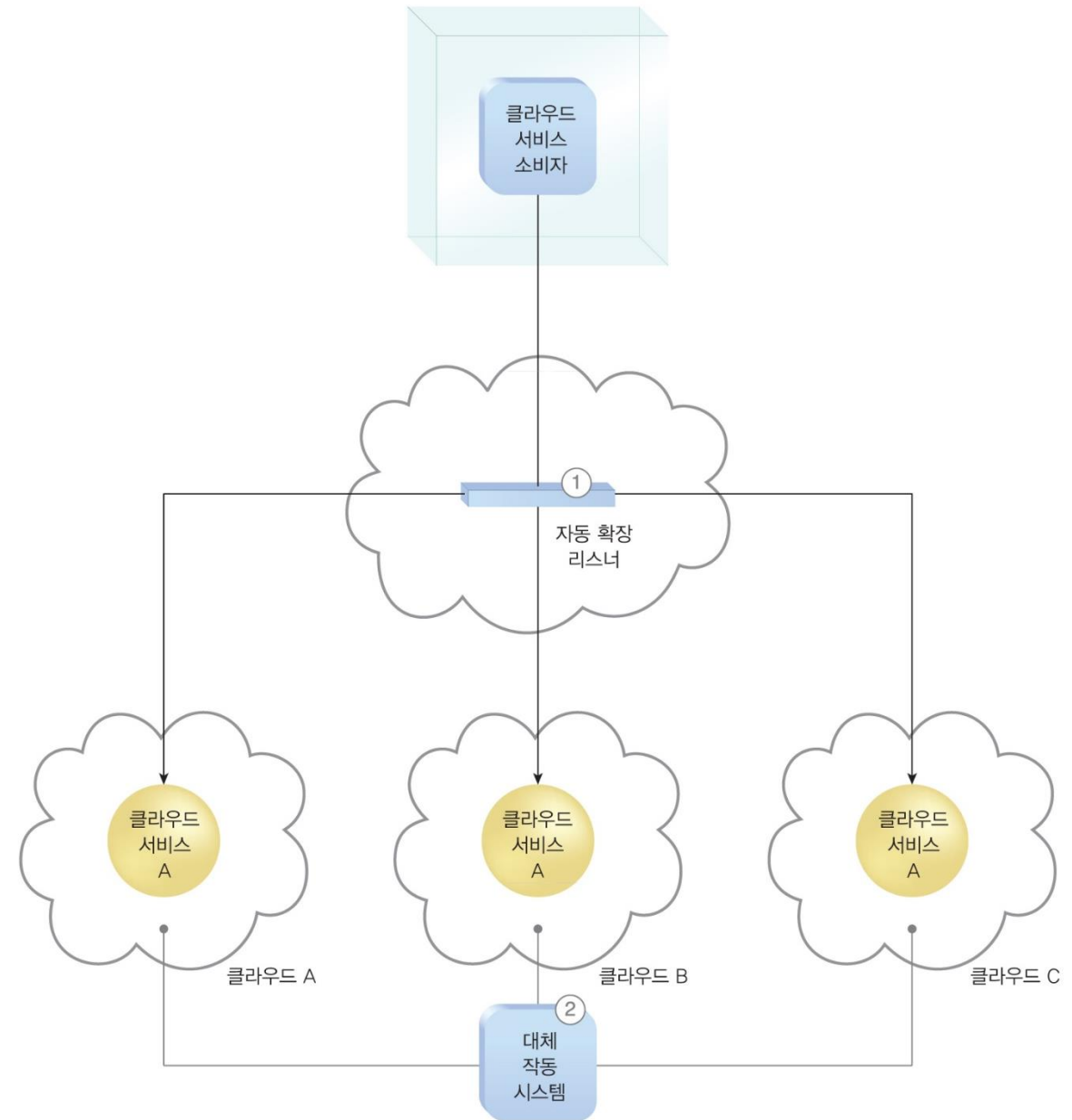
- 감사 모니터 아키텍처는 가상 서버의 재배포로 인해 제공되던 데이터 역시 금지된 영역에 재배포되는지의 여부를 확인할 필요가 있음
- 클라우드 감사 모니터 메커니즘의 다양한 변형은 가상 서버 용량이 초과되지 않도록 보장하기 위해 클라우드 소비자의 실제 IT 자원 사용량을 감시하는데 사용
- 장애의 영향을 받는 물리 서버의 하이퍼바이저가 물리 서버의 장애에 영향을 받는 가상 서버들을 제공
- 논리 네트워크 경계가 가상 서버의 재배포 이후 각 클라우드 소비자가 고유의 논리적 경계 내에 머물러 있도록 보장하는 데 필요한 분리 메커니즘을 제공 및 유지
- 자원 클러스터 메커니즘은 가상 서버가 제공하는 IT 자원의 가용성을 협력 하에 개선하는 다양한 종류의 능동-능동 클러스터 그룹을 생성하는데 적용
- 자원 복제 메커니즘은 가상 서버가 중단될 때 신규 가상 서버와 클라우드 서비스 인스턴스를 생성할 수 있음

클라우드 밸런싱 아키텍처

- IT 자원은 다중 클라우드간에 걸쳐서도 부하가 분산될 수 있음
- 클라우드 서비스 소비자의 요청을 다중 클라우드 간 교차 분산시키는 것의 장점
 - IT 자원의 성능과 확장성을 개선
 - IT 자원의 가용성과 신뢰성을 향상
 - 부하 분산과 IT 자원 최적화를 개선
- 클라우드 밸런싱 기능은 주로 자동 확장 리스너와 대체 작동 시스템 메커니즘의 조합에 기반하고, 더 많은 컴포넌트들이 전체적인 클라우드 밸런싱 아키텍처의 일부가 될 수 있음

클라우드 밸런싱 아키텍처

- ① 자동 확장 리스너가 다중 클라우드에 걸쳐 분산된 클라우드 서비스 A의 중복된 이중화 구현체에 클라우드 서비스 소비자 요청을 라우팅함으로써 인해 클라우드 밸런싱 프로세스를 통제
- ② 대체 작동 시스템은 클라우드 간 교차 대체 작동(cross-cloud failover)을 통해 자동 복구 능력을 제공



클라우드 밸런싱 아키텍처

- 자동 확장 리스너는 현재의 확장 및 성능 요구사항을 토대로 여러 이중화 IT 자원 구현체 중 하나에 클라우드 서비스 소비자 요청을 전달
- 대체 작동 시스템은 이중화 IT 자원들이 특정 IT 자원이나 호스팅 환경 내의 장애 이벤트 발생 시 클라우드 간 교차 대체 작동이 가능함을 보장
- IT 자원의 장애가 보고되면 자동 확장 리스너가 우연히 클라우드 서비스 소비자의 요청을 사용할 수 없거나 불안정한 IT 자원으로 라우팅하는 것을 방지
- 클라우드 밸런싱 아키텍처가 효율적으로 작동하려면, 자동 확장 리스너는 클라우드 밸런싱을 이루는 아키텍처 범위 내의 모든 이중화 IT 자원 구현체를 인지해야함
- 다중 클라우드에 걸친 IT 자원 구현체들 간 수동 동기화가 불가능할 경우, 자원 복제 메커니즘이 동기화를 자동화하기 위해 포함될 필요가 있음

자원 예약 아키텍처

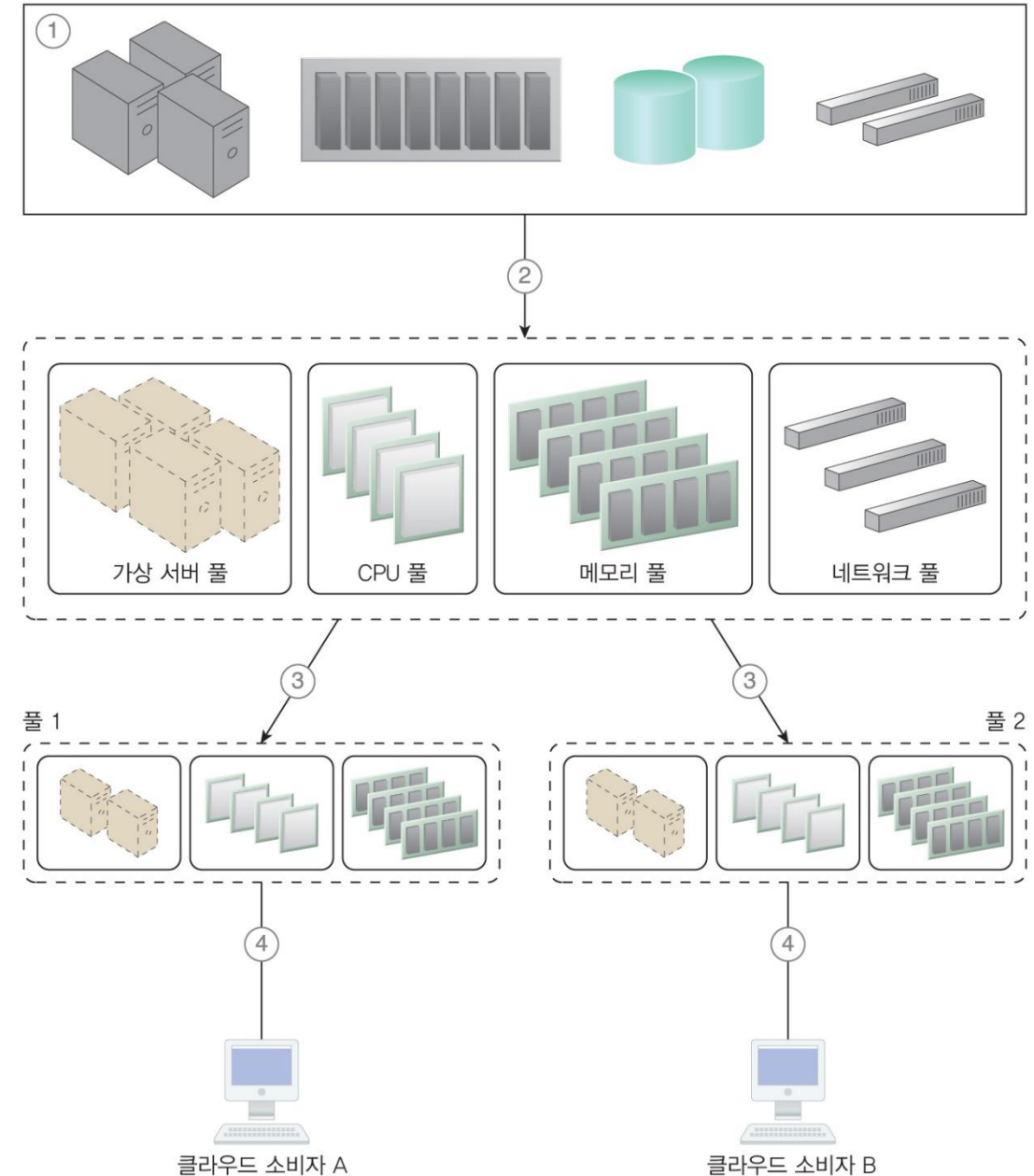
- IT 자원이 공유된 활용을 위해 설계된 방식과 용량의 가용 수준에 따라 동시 사용자 접근이 자원 제약이라 불리는 런타임 예외 상황에 도달할 수 있음
- 자원 제약이란 둘 이상의 클라우드 소비자들이 그들의 총 처리 요구사항을 수용할 수 없는 용량을 가진 IT 자원을 공유하도록 할당된 경우 발생하는 상태
- 그러므로 하나 이상의 클라우드 소비자들이 성능 저하를 경험하거나 모두 처리 거부 상태가 될 수 있음
- 또한 클라우드 서비스 자체가 중단되어 모든 클라우드 소비자들의 요청이 처리 거부 되는 결과에 이를 수도 있음
- 특정 IT 자원이 다른 클라우드 서비스 소비자에 의해 동시에 접근될 경우에 기타 다른 종류의 런타임 충돌이 일어날 수 있음

자원 예약 아키텍처

- 자원 예약 아키텍처는 주어진 클라우드 소비자를 위해 다음 항목들 중 하나를 전용으로 예비해두는 시스템을 형성
 - 단일 IT 자원
 - IT 자원의 일부
 - 다중 IT 자원들
- 클라우드 소비자들이 서로 앞서 언급한 자원 제약 및 자원 임대 상황에 빠지는 것을 방지하도록 보호

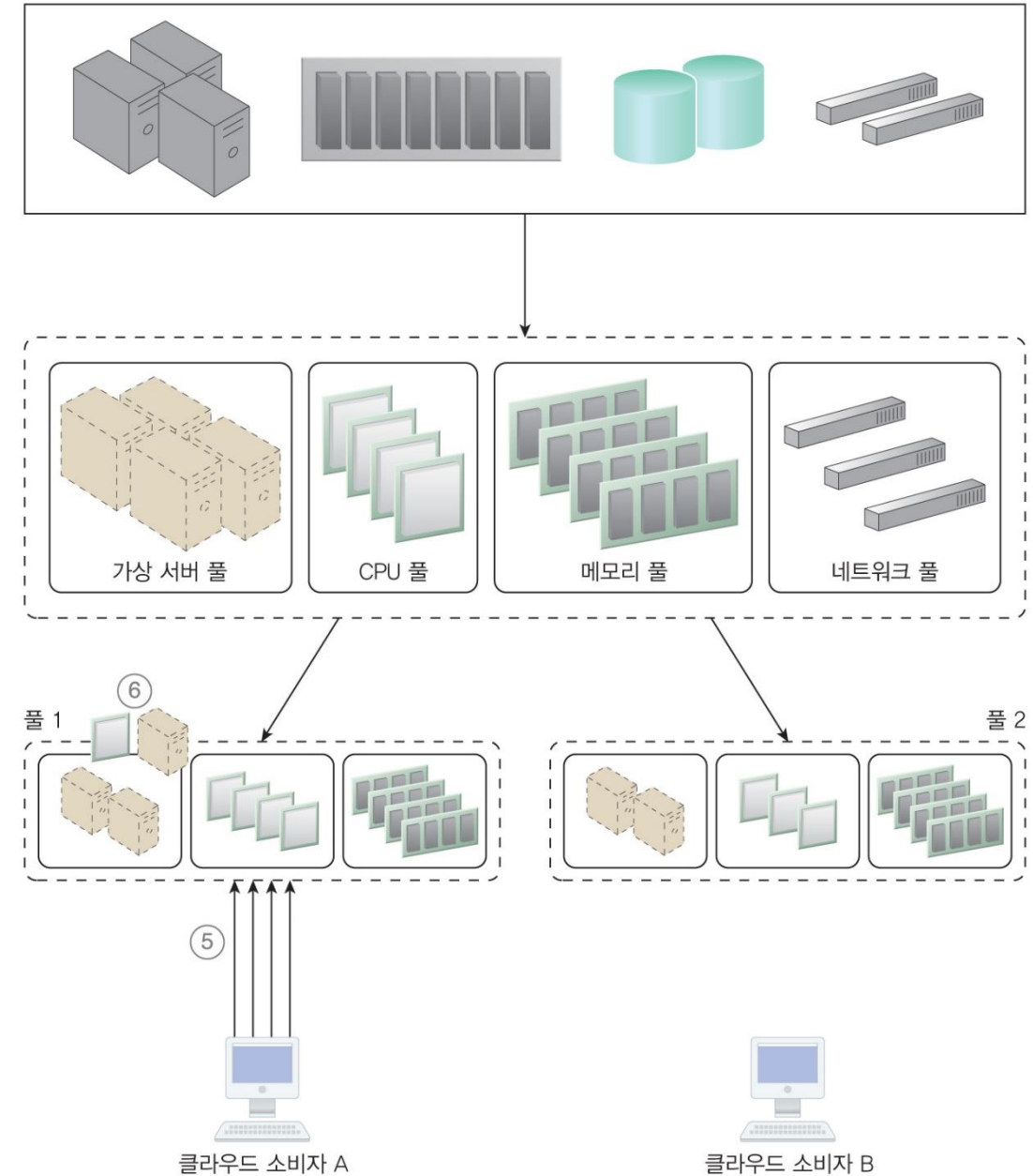
자원 예약 아키텍처

- ① 물리 자원 그룹이 생성
- ② 자원 풀링 아키텍처에 따라 만들어진 부모 자원 풀로부터 생성
- ③ 부모 자원 풀로부터 두 개의 작은 자식 풀이 생성되고, 자원 관리 시스템을 활용하여 자원 한계점이 정의
- ④ 클라우드 소비자에게는 그들의 고유한 전용 자원 풀에 대한 접근이 제공



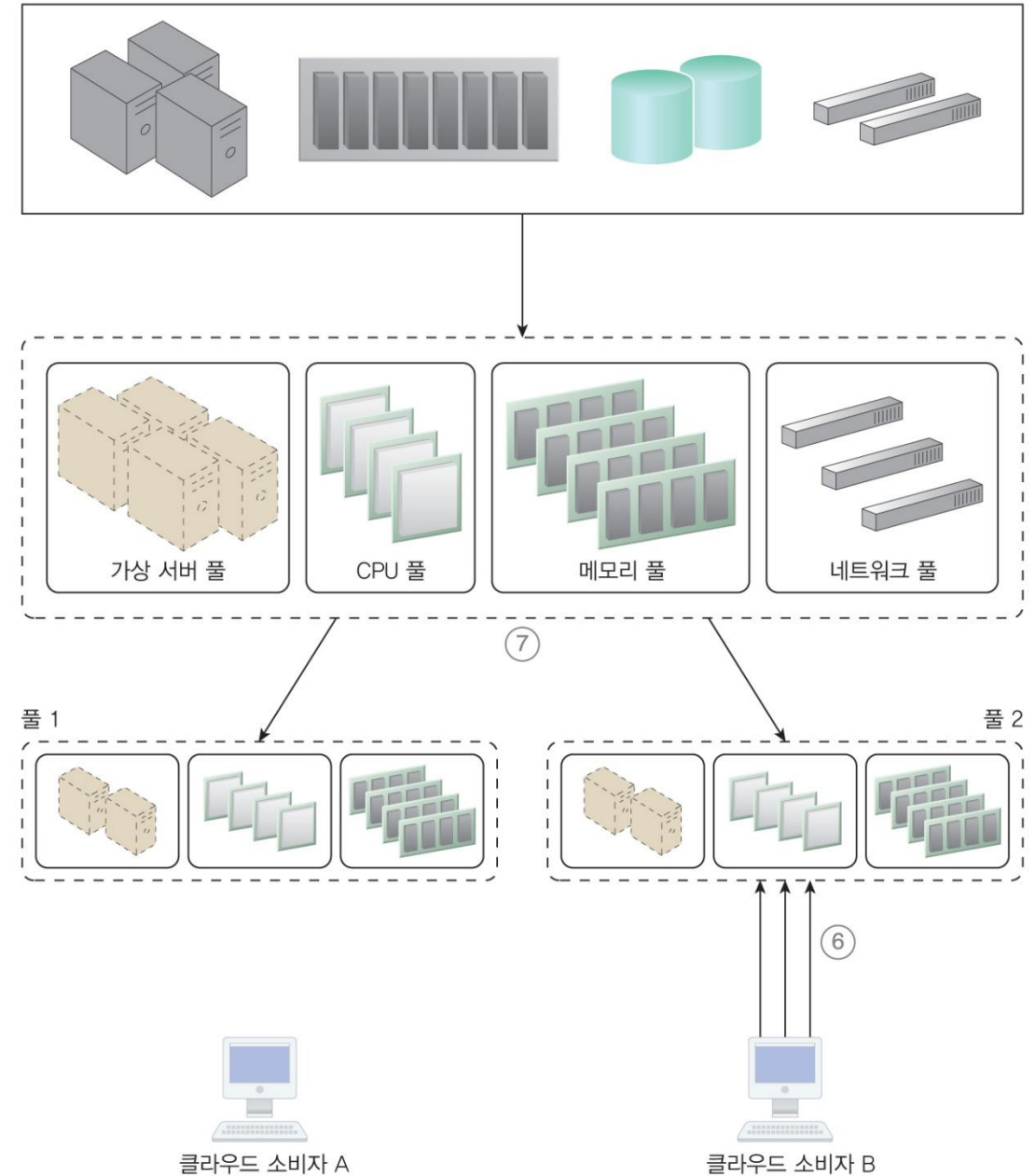
자원 예약 아키텍처

- ⑤ 클라우드 소비자 A로부터 오는 요청이 증가하면 해당 클라우드 소비자에게 더 많은 IT 자원이 할당되는 결과를 초래
- ⑥ 풀 2로부터 일부 IT 자원을 임대할 필요가 있다는 것을 의미하고, 빌리는 IT 자원의 양은 클라우드 소비자 B가 자원 예약 상황에 도달하지 않도록 보장하기 위해 정의된 자원 한계점 내에 있어야 함



자원 예약 아키텍처

- ⑥ 클라우드 소비자 B가 더 많은 요청과 사용 수요를 발생시키면서 곧 풀 내의 모든 가용 IT 자원을 활용해야 할 수 있음
- ⑦ 자원 관리 시스템은 클라우드 소비자 B가 사용할 수 있도록 풀 1로 하여금 점유한 IT 자원을 풀 2로 반납하도록 강요



자원 예약 아키텍처

- IT 자원 예약 시스템의 생성은 각 독립된 IT 자원과 자원 풀에 대한 사용 임계치를 정의하는데 사용되는 자원 관리 시스템 메커니즘이 포함하도록 요구할 수 있음
- 자원 예약은 해당 풀의 IT 자원을 여전히 공유 및 임대용으로 사용할 수 있도록 균형을 이루면서도 각 풀이 유지해야 할 필요가 있는 IT 자원의 양을 고정하여 잠금 장치를 해둠
- 원격 관리 시스템 메커니즘 또한 클라우드 소비자가 예약한 IT 자원의 할당을 관리하기 위해 관리 통제권을 가질 수 있도록 활용
- 감사 모니터는 자원 예약 시스템이 클라우드 소비자 감사, 개인 정보 보호, 기타 규제 요구사항을 준수하는지 여부를 확인하는데 사용

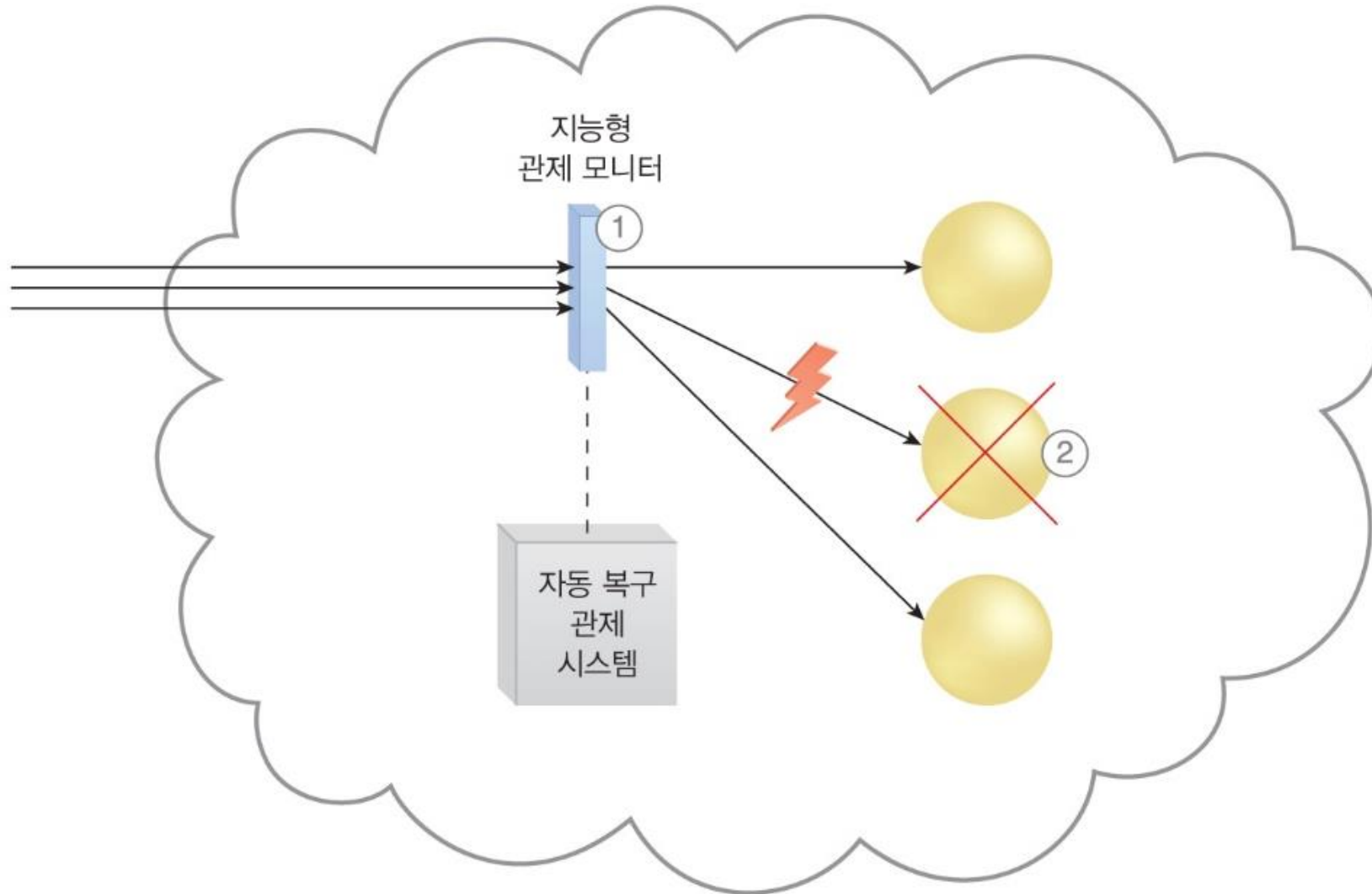
자원 예약 아키텍처

- 클라우드 사용 모니터는 예약된 IT 자원의 할당을 유발하는 임계치를 관제할 수 있음
- 하이퍼바이저 메커니즘은 여러 클라우드 소비자들이 예약한 IT 자원을 그대로 정확히 할당받을 수 있도록 보장하기 위해 예약 설정을 적용할 수 있음
- 논리 네트워크 경계 메커니즘은 예약된 IT 자원이 각 클라우드 소비자에게 전용으로 쓰일 수 있도록 보장하는 데 필요한 경계를 정립
- 자원 복제 컴포넌트는 편의상 신규 IT 자원 인스턴스를 복제 및 프로비저닝하기 위해 IT 자원 소비에 대한 각 클라우드 소비자의 한계점에 대한 정보를 지속적으로 제공받을 필요가 있음

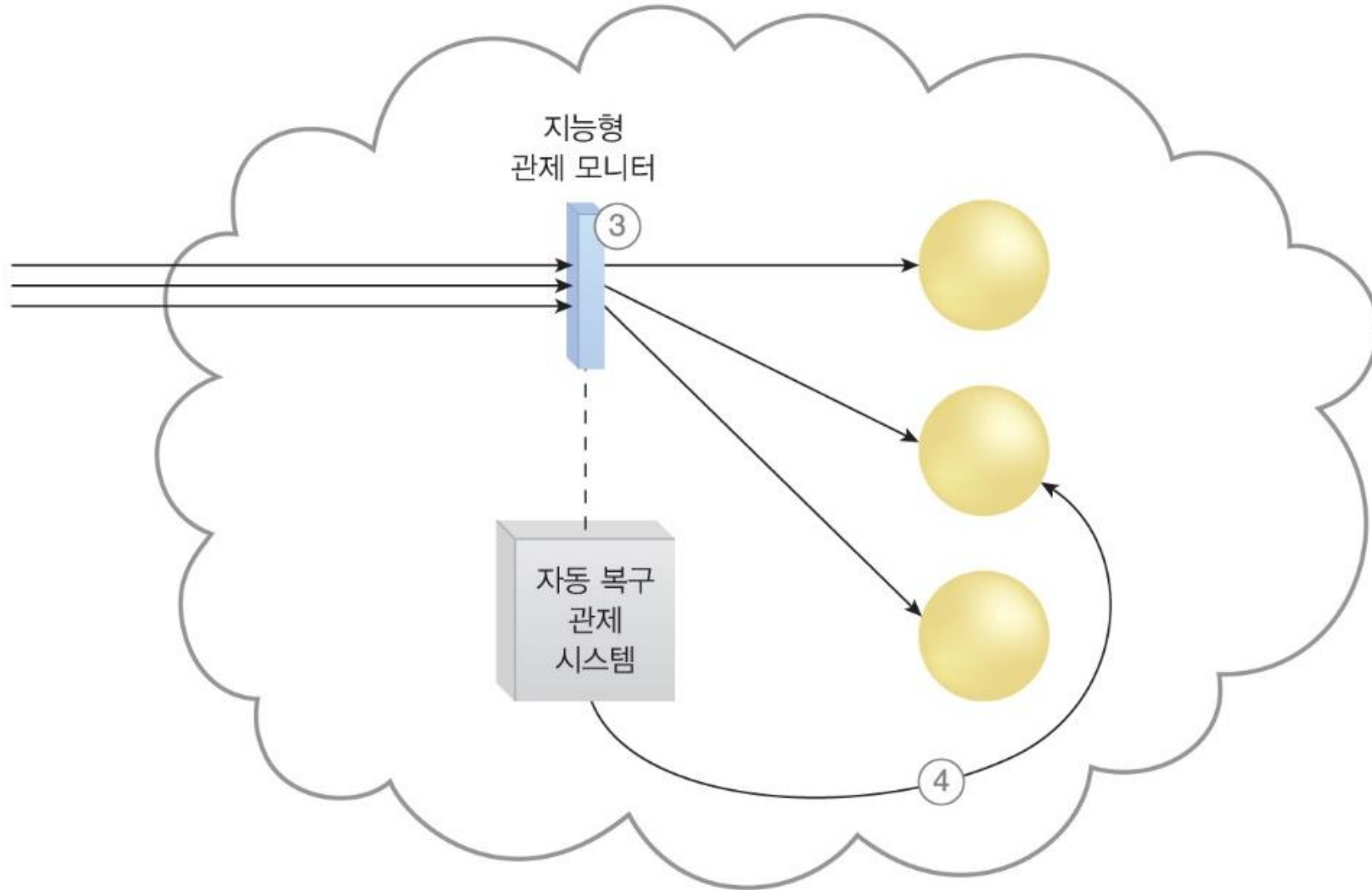
동적 장애 감지 및 복구 아키텍처

- 클라우드 기반 환경은 수 많은 클라우드 소비자가 동시 다발적으로 접근하는 방대한 양의 IT 자원들로 구성될 수 있음
- IT 자원들 중 어떤 것이라도 장애 상황에 처하여 인간의 수동적 개입으로 해결해야할 수 있음
- IT 자원의 장애를 수동으로 관리 및 해결하는 것은 일반적으로 비효율적이며 비실용적인 일
- 동적 장애 감지 및 복구 시스템은 광범위하게 정의된 장애 시나리오를 감시 및 대응하는 자동 복구 능력이 있는 관제 시스템을 확립하고, 만약 직접 자동으로 해결할 수 없는 상황이 발생하면 통지함
- 활발하게 IT 자원을 추적하고 정의된 이벤트에 대응하여 정의된 활동을 실행하는 지능형 관제 모니터라 불리는 특화된 클라우드 사용 모니터에 의존

동적 장애 감지 및 복구 아키텍처



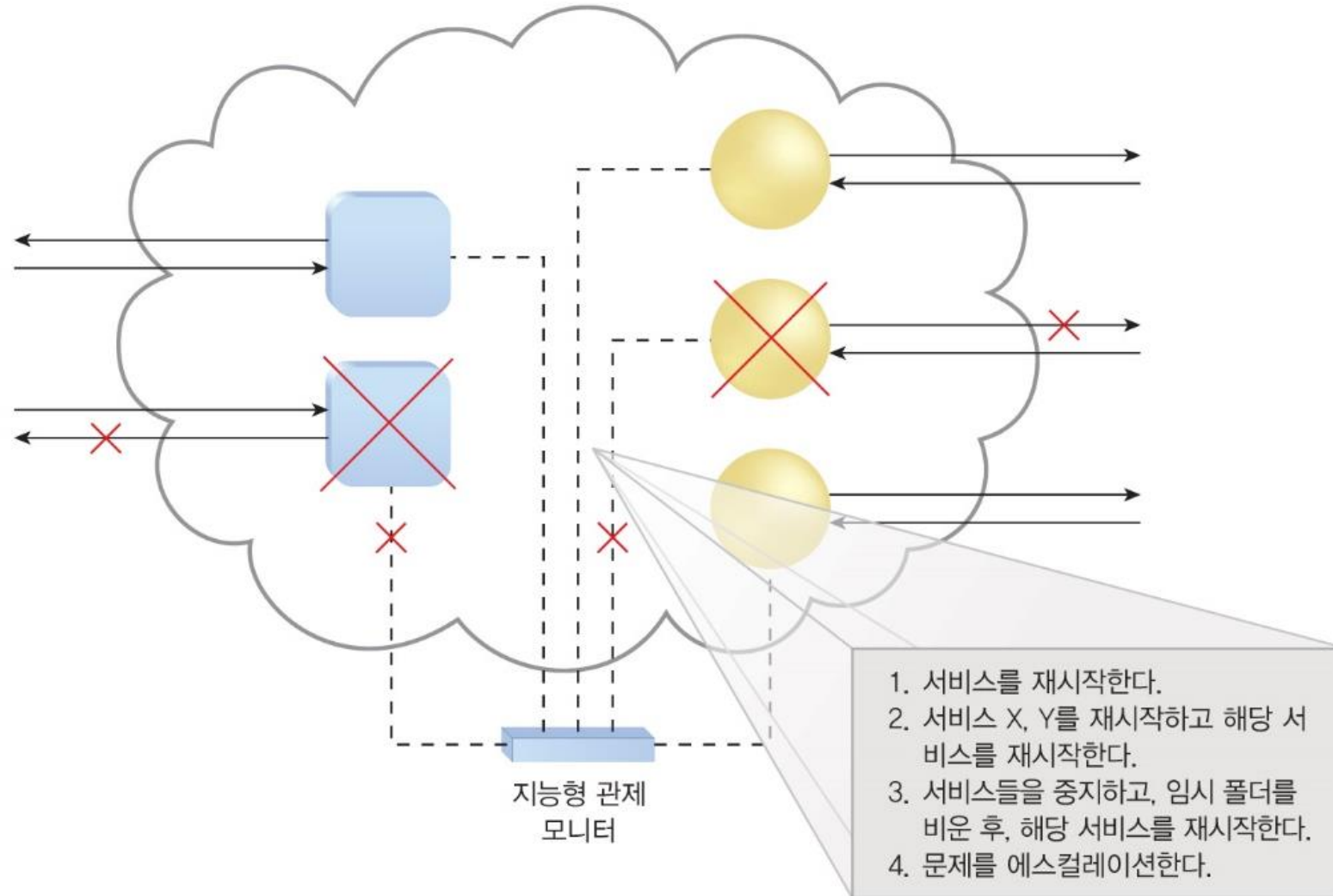
동적 장애 감지 및 복구 아키텍처



동적 장애 감지 및 복구 아키텍처

- 자동 복구 관제 시스템의 핵심 기능
 - 관찰 및 감시
 - 이벤트 발생 시 대응 행위를 결정
 - 이벤트 발생 시 대응 행위를 수행
 - 보고
 - 에스컬레이션^{escalation}
- 장애 상황이 발생한 경우 지능형 관제 모니터가 취할 단계적 행동을 결정하기 위해 순차적인 복구 정책이 각 IT 자원용으로 정의될 수 있음
- 예를 들어, 복구 정책은 장애 통지를 발행하기 이전에 한 가지 복구 시도가 자동으로 수행될 필요가 있음을 명시

동적 장애 감지 및 복구 아키텍처



동적 장애 감지 및 복구 아키텍처

- 지능형 관제 모니터가 특정 이슈를 에스컬레이션 할 때 수행되는 활동
 - 배치 파일 실행
 - 콘솔 메시지 전송
 - 텍스트 메시지 전송
 - 이메일 메시지 전송
 - SNMP 트랩 전송
 - 티켓 로깅

동적 장애 감지 및 복구 아키텍처

- 감사 모니터 메커니즘은 데이터 복구가 법적, 정책적 요구사항을 준수하여 수행되는지의 여부를 추적하는 데 사용
- 대체 작동 시스템 메커니즘은 보통 장애가 발생한 IT 자원을 복구하기 위한 초기 시도를 실행하는 도중에 사용
- SLA 관리 시스템 및 SLA 모니터 메커니즘을 통해 SLA 보장 요건을 잘 지켜지는데 관리 및 처리

베어 메탈 프로비저닝 아키텍처

- 베어 메탈 서버, 즉 미리 운영체제나 기타 소프트웨어가 설치되지 않은 물리 서버에는 원격 관리 프로그램에 접근하는 것 자체가 불가능
- 대부분의 최신 물리 서버들은 원격 관리 지원을 해당 서버의 ROM에 설치하는 옵션을 제공
- 어떤 업체들은 확장 카드를 통해서 제공하고, 기타 업체들은 미리 칩셋에 통합
- 베어 메탈 프로비저닝 아키텍처는 효과적으로 전체 운영체제를 원격으로 식별 및 프로비저닝 하는데 사용되는 특화된 서비스 에이전트를 통해 시스템 형성

베어 메탈 프로비저닝 아키텍처

- 서버의 ROM에 통합된 원격 관리 소프트웨어는 서버가 시작할 때 사용 가능한 상태가 됨
- 원격 관리 시스템을 통해 제공되는 포털과 같은 웹 기반, 혹은 사설 사용자 인터페이스가 일반적으로 물리 서버의 원격 관리 인터페이스에 연결하는 데 사용
- 원격 관리 인터페이스의 IP 주소는 기본 IP를 통해 수동으로 구성되거나 DHCP 서비스의 구성을 통해 대체 설정 가능
- IaaS 플랫폼 내 IP 주소는 클라우드 소비자들에게 직접적으로 전달되어 클라우드 소비자들이 독립적으로 베어 메탈 운영체제 설치를 수행할 수 있도록 함
- 원격 관리 소프트웨어가 물리 서버 콘솔에 연결되거나 운영체제 배치를 가능하게 하지만, 여러 서버에 수동으로 배치하는 것은 우연한 실수 및 구성 오류에 취약하고, 원격 관리 소프트웨어는 시간 소모적이고, 상당한 런타임 IT 자원 처리를 요구

베어 메탈 프로비저닝 아키텍처

- 식별 에이전트: 클라우드 소비자에게 할당될 가용한 물리 서버를 탐색하고 발견하는 모니터링 에이전트의 한 종류
- 배치 에이전트: 베어 메탈 프로비저닝 배치 시스템의 클라이언트로서 물리 서버와 메모리에 설치되는 관리 에이전트
- 식별 영역: 네트워크를 정밀 검사하고 연결할 가용한 물리 서버의 위치를 확인하는 소프트웨어 컴포넌트
- 관리 로더: 물리 서버에 연결하고 클라우드 소비자용 관리 옵션을 로드하는 컴포넌트
- 배치 컴포넌트: 선택된 물리 서버 위에 운영체제를 설치할 책임이 있는 컴포넌트

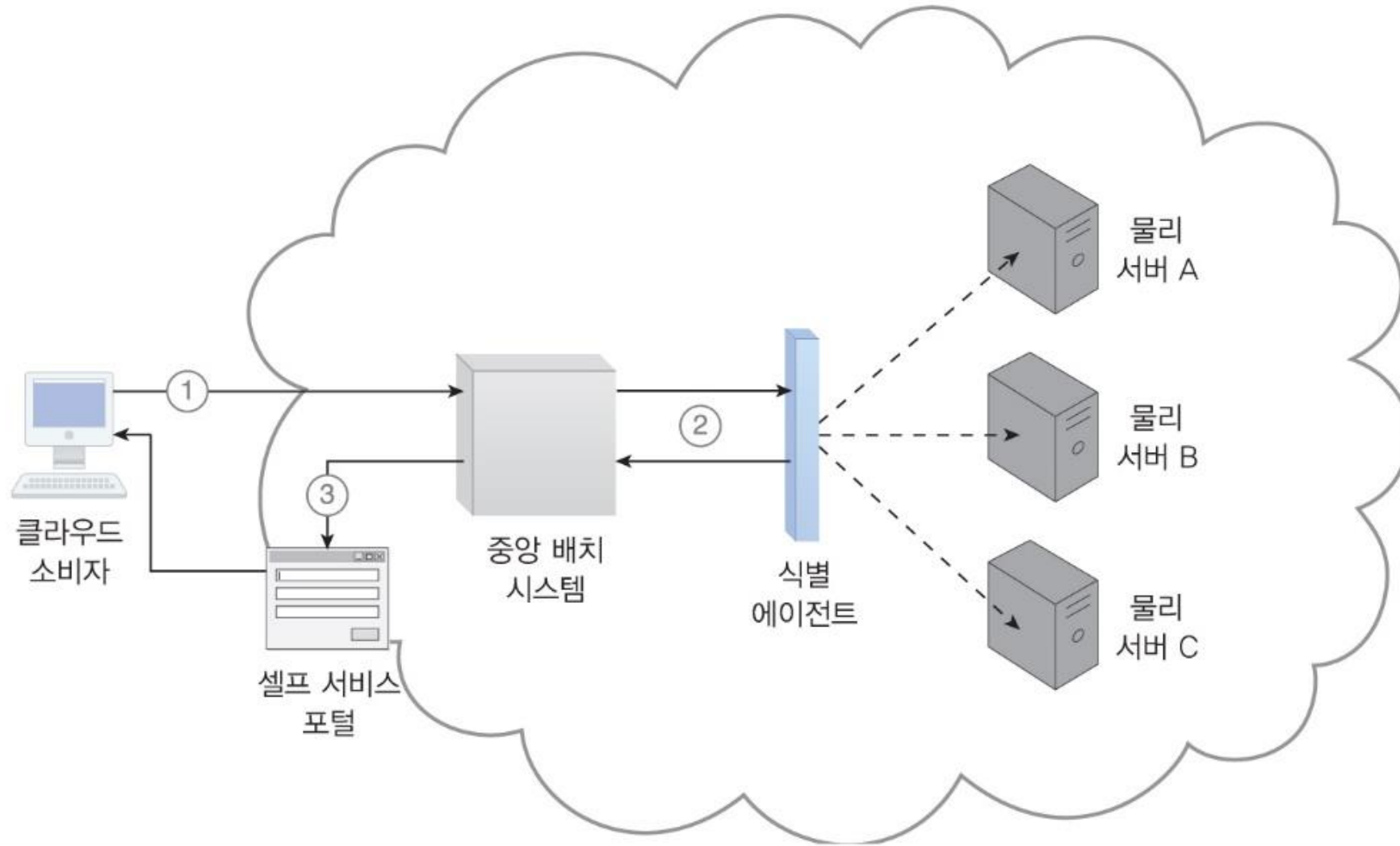
베어 메탈 프로비저닝 아키텍처

- 베어 메탈 프로비저닝 시스템은 클라우드 소비자가 배치 소프트웨어에 연결하여 동시에 하나 이상의 서버나 운영체제를 프로비저닝할 수 있도록 하는 자동 배치 기능을 제공
- 중앙 배치 시스템은 관리 인터페이스를 통해 서버로 연결하며, 물리 서버의 RAM 안에 업로드하고 에이전트로서 동작하는 데 동일한 프로토콜을 사용
- 베어 메탈 서버는 이제 관리 에이전트가 설치된 저 수준 클라이언트가 되고, 배치 소프트웨어는 운영체제를 배치하기 위해 필요한 설정 파일들을 업로드

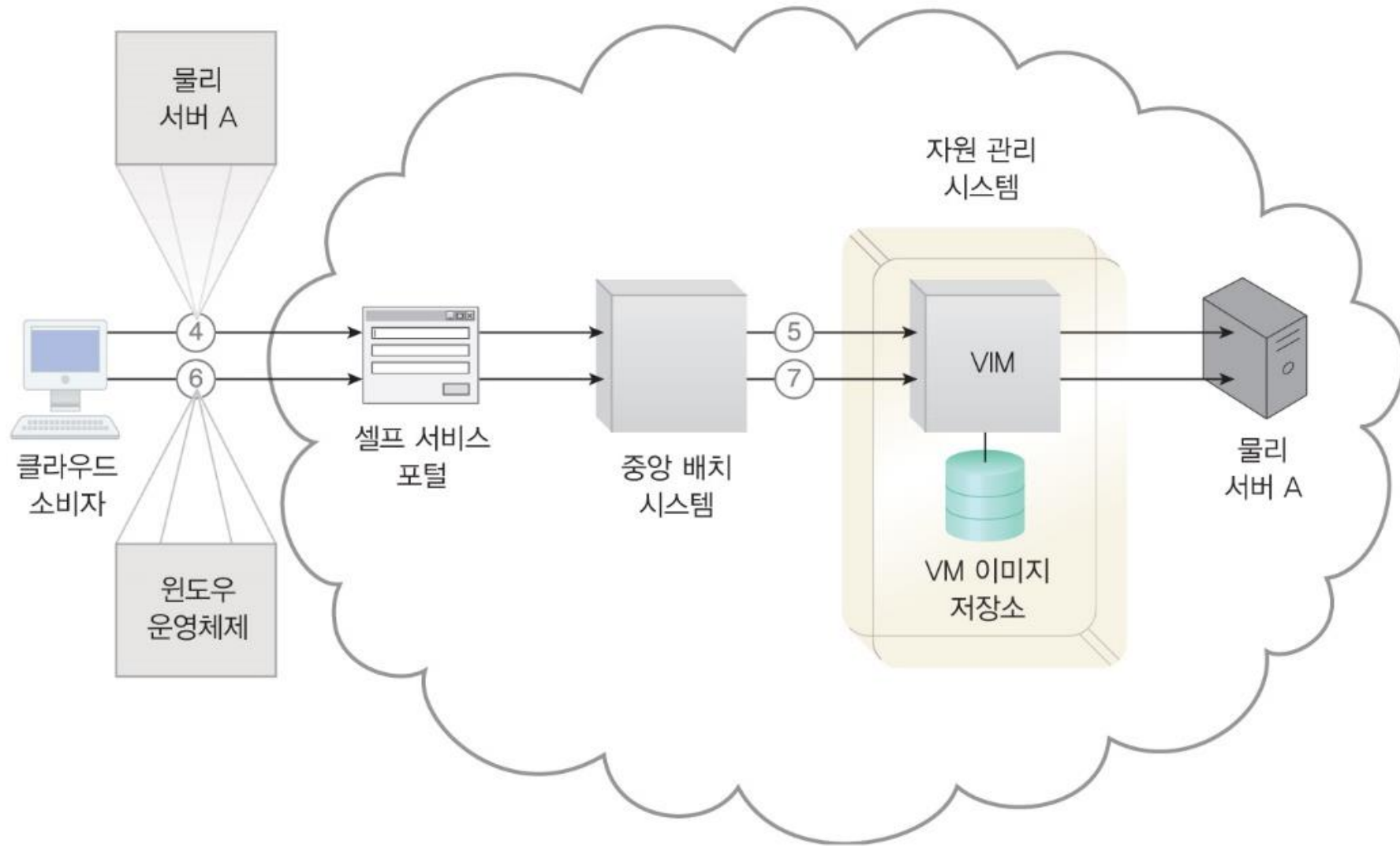
베어 메탈 프로비저닝 아키텍처

- 클라우드 스토리지 장치 메커니즘은 운영체제 템플릿과 설치 파일을 저장할 뿐 아니라, 프로비저닝 시스템을 위해 배치 에이전트와 배치 패키지를 저장
- 하이퍼바이저는 운영체제 프로비저닝의 일부로서 물리 서버 상에 하이퍼바이저를 배치하는 것이 필요할 수 있음
- 논리 네트워크 경계가 저 수준의 물리 서버들에 인가받은 클라우드 소비자만 접근할 수 있도록 보장
- 자원 복제 메커니즘은 프로비저닝 도중 혹은 이후에 하이퍼바이저 워크로드를 분배하기 위해 물리 서버상에 신규 하이퍼바이저를 배치해서 IT 자원 복제용으로 구현
- SLA 관리 시스템은 물리 베어 메탈 서버의 가용성이 정의된 SLA 조항을 따른다는 것을 보장

베어 메탈 프로비저닝 아키텍처



베어 메탈 프로비저닝 아키텍처



신속한 프로비저닝 아키텍처

- 전형적인 프로비저닝 프로세스는 기존에 관습적으로 관리자와 기술 전문가들이 수동으로 완료하던 다수의 작업들, 즉 미리 패키징된 명세나 고객 맞춤 요구사항에 따라 필요한 IT 자원을 준비하는 작업을 포함
- 더 많은 수의 고객을 대상으로 서비스하고, 평균적으로 고객들이 더 많은 양의 IT 자원을 요청하는 클라우드 환경에서는 수동 프로비저닝 프로세스는 부적절함, 나아가 인적 오류 및 비효율적인 대응 시간으로 인해 비합리적인 위험을 감당하게 됨
- 신속한 프로비저닝 아키텍처는 독자적으로, 혹은 협력 하에 광범위한 IT 자원의 프로비저닝을 자동화하는 시스템을 형성
- 신속한 자원 프로비저닝의 기술적 아키텍처는 정교하고 복잡할 수 있고, 자동화 프로비저닝 프로그램, 신속한 프로비저닝 엔진, 온디맨드 프로비저닝용 스크립트와 템플릿으로 구성된 시스템에 의존

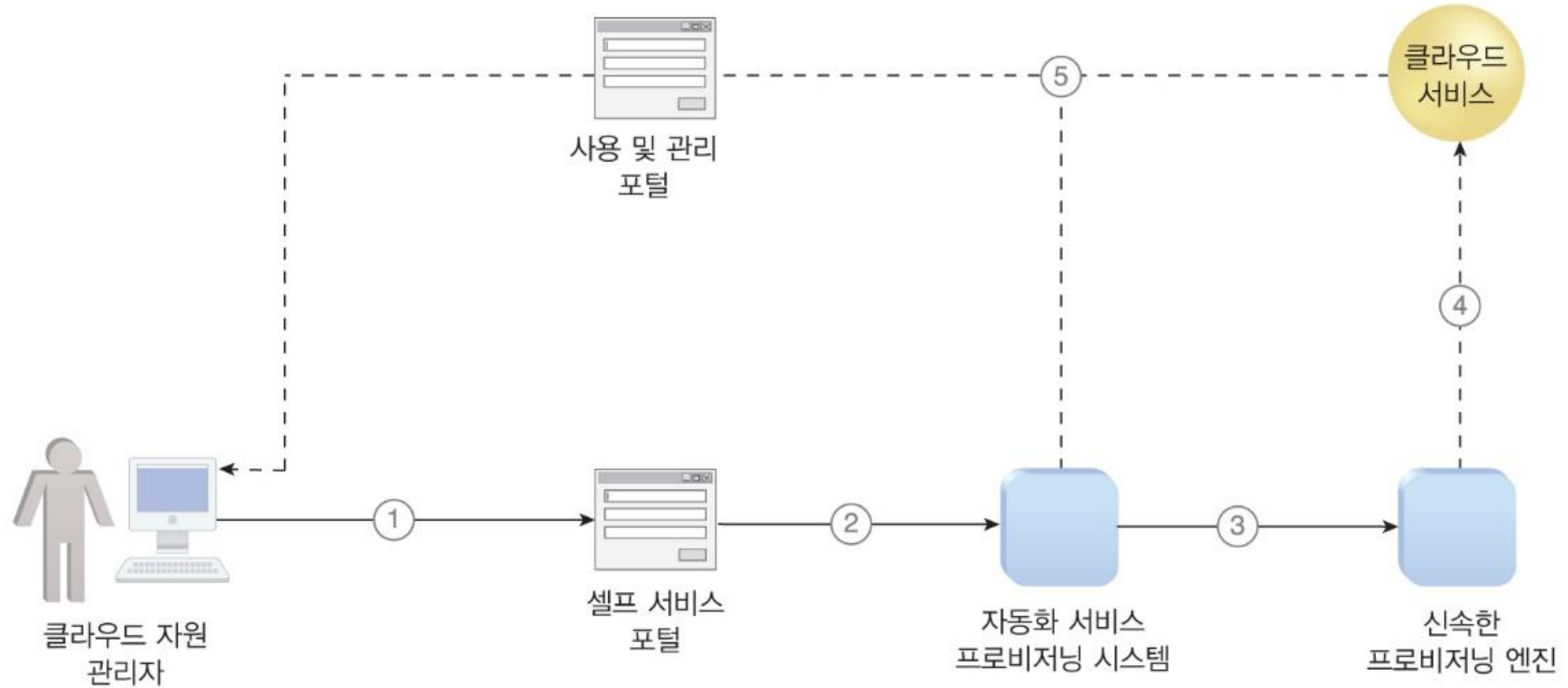
신속한 프로비저닝 아키텍처

- 서버 템플릿: 신규 가상 서버의 인스턴스화를 자동화하는 데 사용되는 가상 이미지 파일의 템플릿
- 서버 이미지: 이 이미지들은 가상 서버 템플릿과 유사하지만 물리 서버를 프로비저닝하는데 사용
- 애플리케이션 패키지: 자동 배치용으로 패키징된 애플리케이션 및 기타 소프트웨어의 모음
- 애플리케이션 패키지: 애플리케이션 패키지를 생성하는 데 사용되는 소프트웨어
- 맞춤 스크립트: 지능형 자동화 엔진의 일부로 관리 작업을 자동화하는 스크립트

신속한 프로비저닝 아키텍처

- 순서 배열 관리자: 자동화 프로비저닝 작업의 순서를 조직화하는 프로그램
- 순서 이력 기록 장치: 자동화 프로비저닝 작업 순서의 실행 이력을 기록하는 컴포넌트
- 운영체제 베이스라인: 운영체제가 설치된 후에 이를 신속하게 활용 가능하도록 준비하기 위해 적용되는 구성 템플릿
- 애플리케이션 구성 베이스라인: 신규 애플리케이션 사용 준비에 필요한 설정 및 호나경 변수가 포함된 구성 템플릿
- 배치 데이터 스토어: 가상 이미지, 템플릿, 스크립트, 베이스라인 구성 및 기타 관련 데이터를 저장하는 저장소

신속한 프로비저닝 아키텍처



신속한 프로비저닝 아키텍처

■ 신속한 프로비저닝 엔진의 내부 원리

1. 클라우드 소비자가 셀프 서비스 포털을 통해 신규 서버를 요청
2. 순서 배열 관리자가 운영체제의 준비를 위해 해당 요청을 배치 엔진으로 전달
3. 배치 엔진은 해당 요청이 가상 서버용인 경우 프로비저닝을 위해 가상 서버 템플릿을 사용, 가상 서버용이 아닌 경우, 배치 엔진은 물리 서버를 프로비저닝하도록 요청
4. 사용할 수 있는 경우, 요청된 종류의 운영체제용으로 제작된 이미지가 운영체제의 프로비저닝에 사용, 이미지를 사용할 수 없는 경우에는 운영체제를 설치하기 위해 정규 배치 프로세스가 수행
5. Heartbeat은 하이퍼바이저 간, 하이퍼바이저와 가상 서버 간, 하이퍼바이저와 VIM 간 교환되는 시스템 수준의 메시지. 배치 엔진은 순서 배열 관리자에게 운영체제의 사용 준비가 완료되면 이를 통지
6. 순서 배열 관리자는 저장을 위해 순서 이력 기록 장치에 로그를 갱신 및 전송
7. 순서 배열 관리자는 배치 엔진이 프로비저닝된 운영체제에 운영체제 베이스라인을 적용하도록 요청

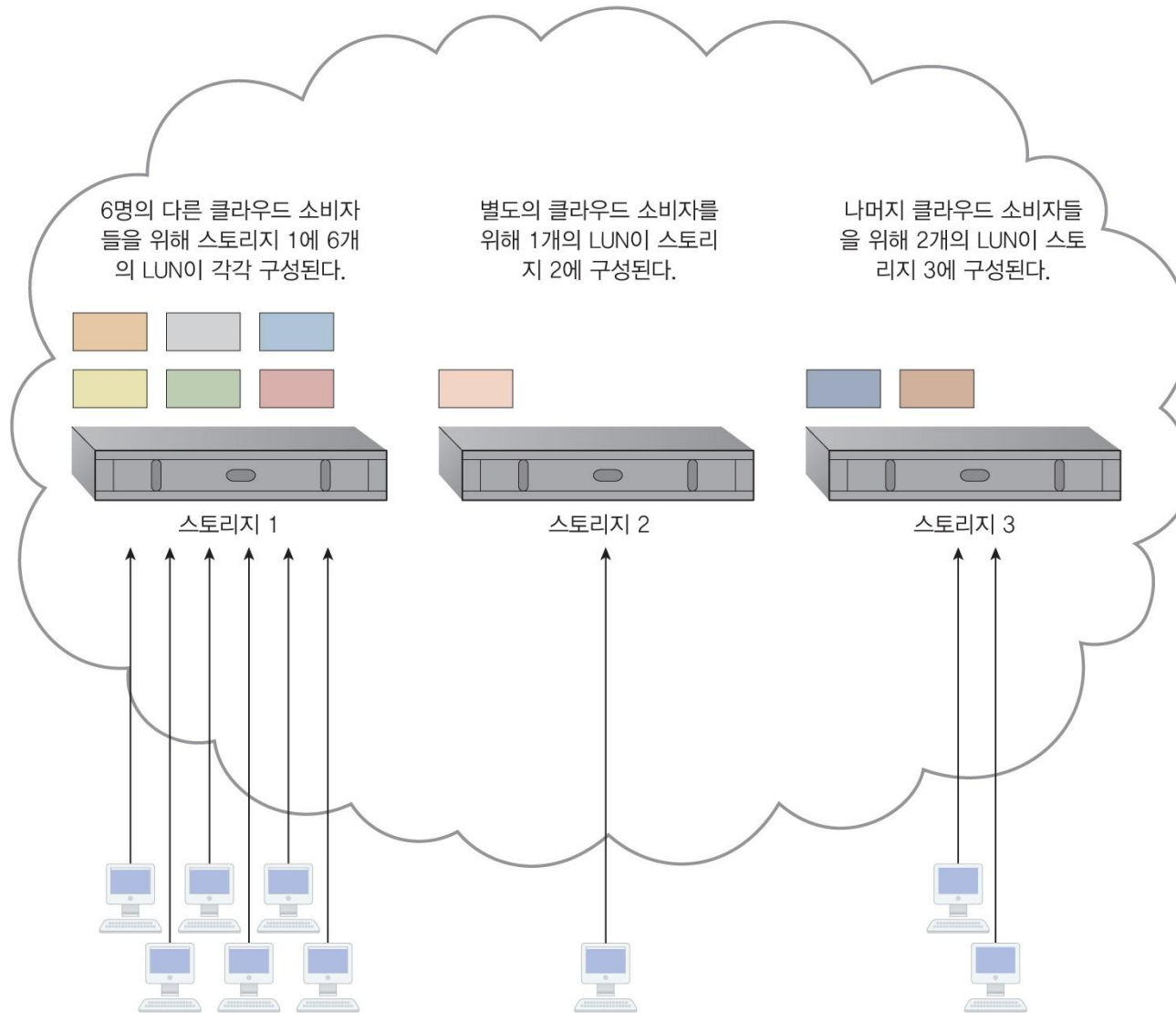
신속한 프로비저닝 아키텍처

8. 배치 엔진은 요청된 운영체제 베이스라인을 적용
9. 배치 엔진은 순서 배열 관리자에게 운영체제 베이스라인이 적용되었음을 알림
10. 순서 배열 관리자는 저장을 위해 순서 이력 기록 장치에 완료된 단계의 로그를 갱신 및 전송
11. 순서 배열 관리자는 배치 엔진이 애플리케이션을 설치하도록 요청
12. 배치 엔진은 해당 애플리케이션을 프로비저닝된 서버에 배치
13. 배치 엔진은 순서 배열 관리자에게 애플리케이션이 설치되었음을 알림
14. 순서 배열 관리자는 저장을 위해 순서 이력 기록 장치에 완료된 단계의 로그를 갱신 및 전송
15. 순서 배열 관리자는 배치 엔진이 애플리케이션 구성 베이스라인을 적용하도록 요청
16. 배치 엔진은 해당 구성 베이스라인을 적용
17. 배치 엔진은 구성 베이스라인이 적용되었음을 순서 배열 관리자에게 알림
18. 순서 배열 관리자는 저장을 위해 순서 이력 기록 장치에 완료된 단계의 로그를 갱신 및 전송

스토리지 워크로드 관리 아키텍처

- 사용 수준이 적정 수준을 초과한 클라우드 스토리지 장치는 스토리지 컨트롤러상의 워크로드를 증가시키고, 다양한 성능 이슈를 야기할 수 있음
- 역으로 사용 수준이 미달된 클라우드 스토리지 장치는 잠재적인 처리 및 저장 용량의 손실로 인해 낭비가 심함
- 스토리지 워크로드 관리 아키텍처는 쓸 수 있는 클라우드 스토리지 장치에 걸쳐 LUN이 균등하게 분배되도록 하며, 동시에 스토리지 용량 시스템이 확립되어 런타임 워크로드가 여러 LUN에 걸쳐 균등하게 분배되도록 보장
- 클라우드 스토리지 장치를 하나의 그룹으로 묶으면 LUN 데이터가 사용할 수 있는 스토리지 호스트에 균등하게 분배될 수 있게 됨
- 스토리지 용량 시스템은 LUN의 접근 빈도가 낮거나 유휴 상태인 기간에 운영중인 스토리지 장치를 전원 절약 모드로 유지

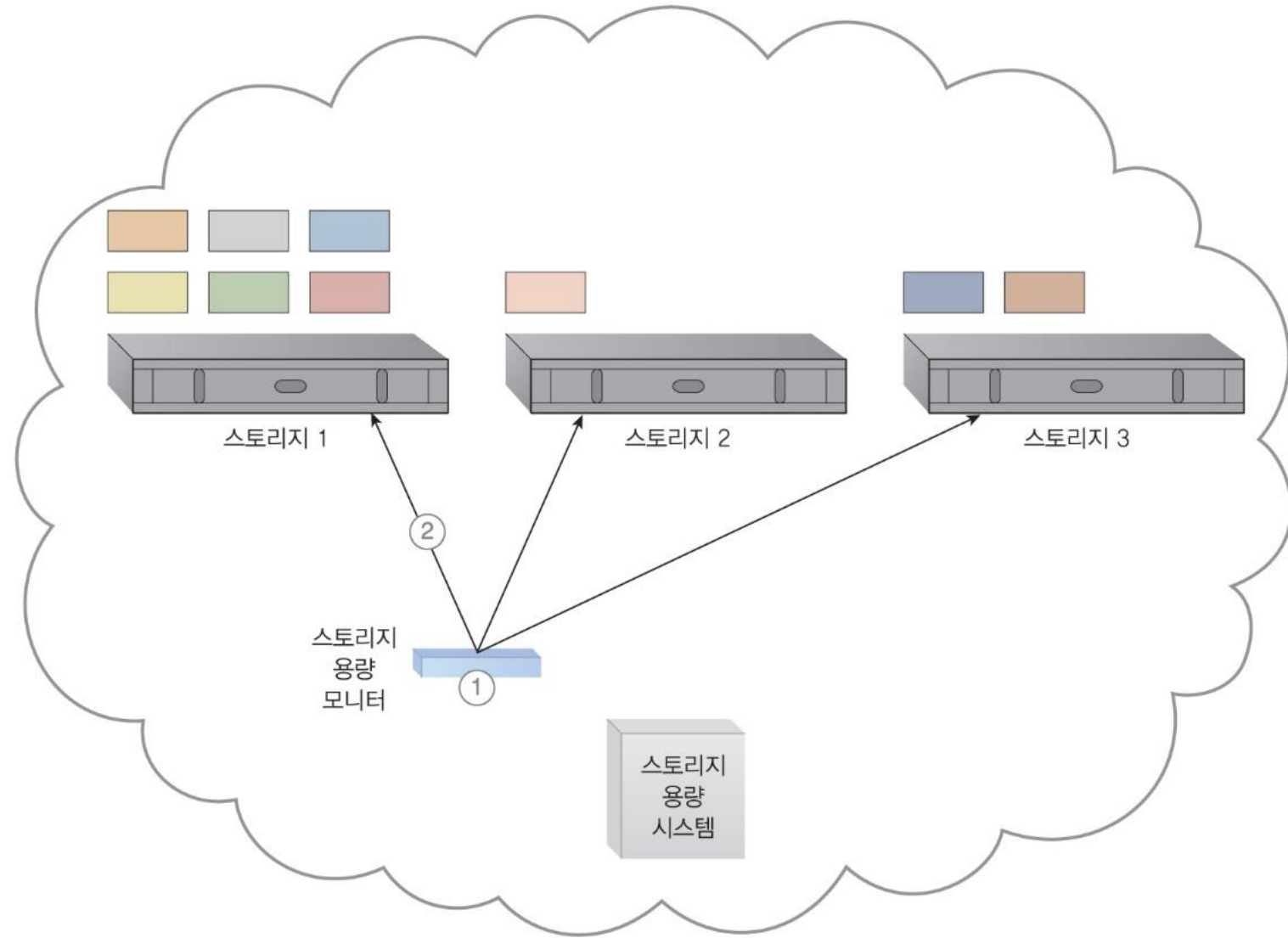
스토리지 워크로드 관리 아키텍처



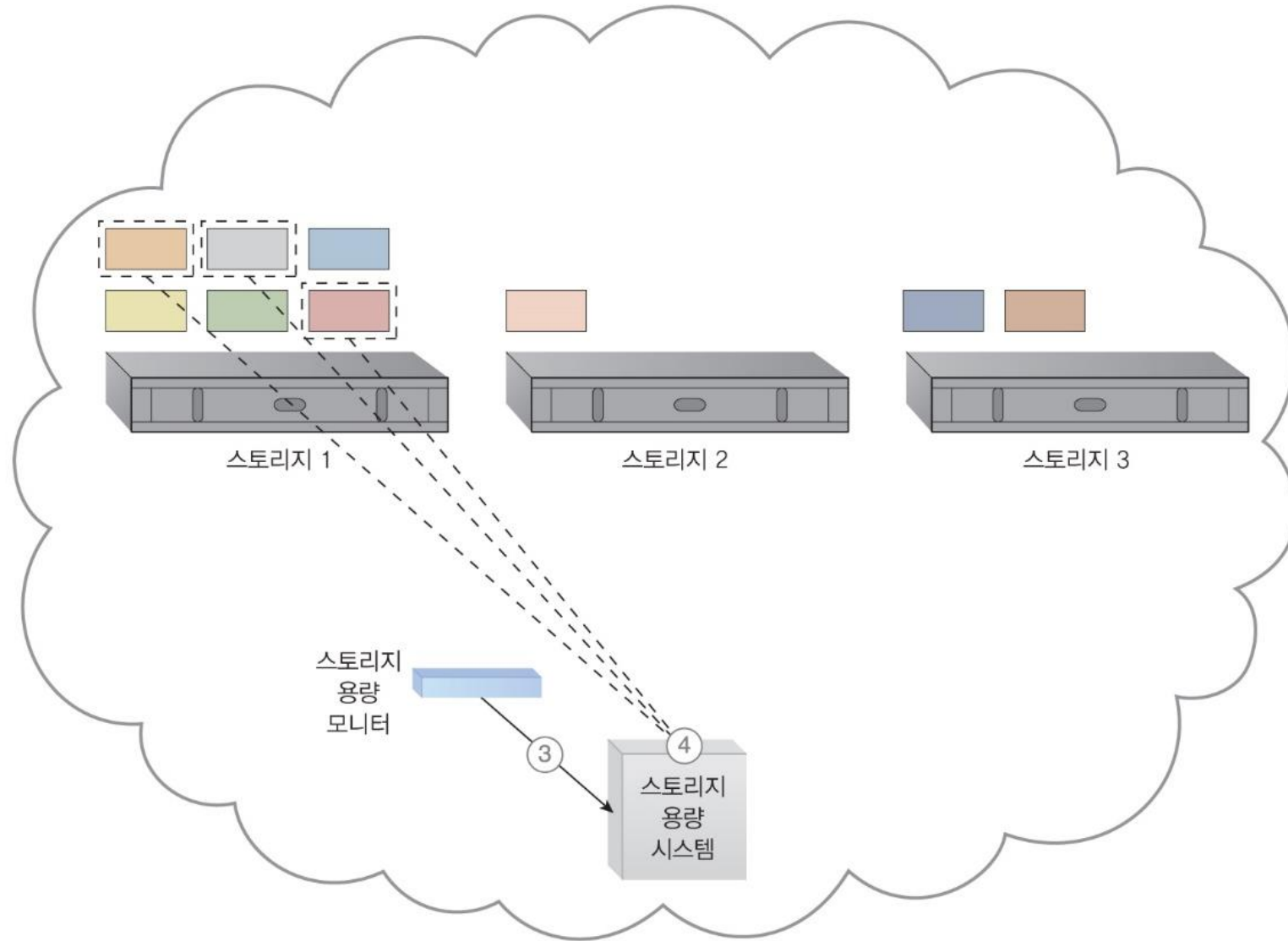
스토리지 워크로드 관리 아키텍처



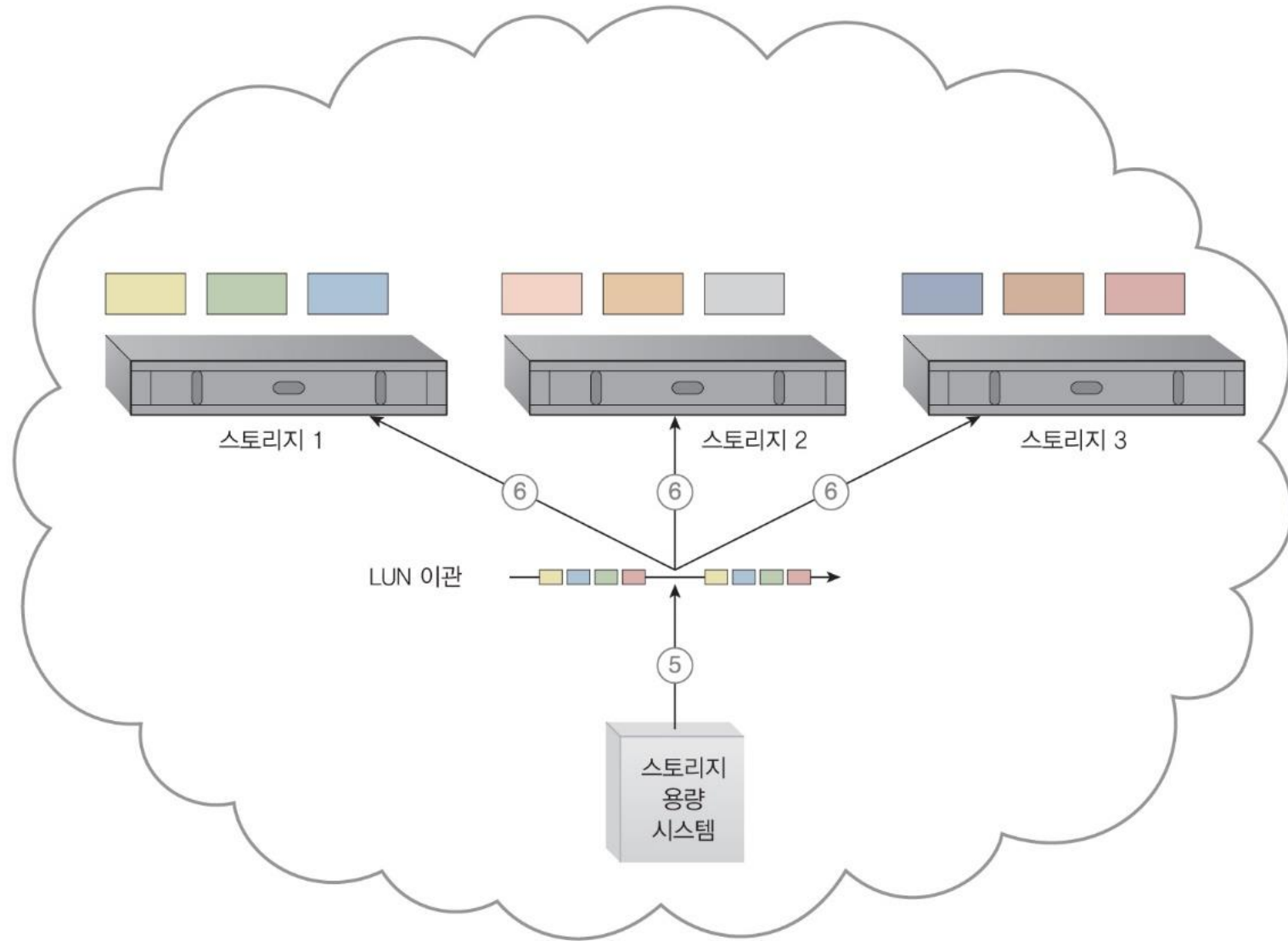
스토리지 워크로드 관리 아키텍처



스토리지 워크로드 관리 아키텍처



스토리지 워크로드 관리 아키텍처



스토리지 워크로드 관리 아키텍처

- 감사 모니터는 시스템이 물리적으로 데이터를 재배치하는지와 규제, 개인 정보 보호 및 보안 요구사항을 준수하는지 확인
- 자동 확장 리스너가 워크로드의 변동을 관제 및 대응하는 데 사용
- 클라우드 사용 모니터가 LUN의 이동을 추적하고 워크로드 분배 통계치를 수집하는 데 사용
- 로드 밸런서 메커니즘이 사용할 수 있는 클라우드 스토리지 장치들 간 워크로드를 수평적으로 분배하는 데 추가될 수 있음
- 논리 네트워크 경계는 다양한 수준의 분리 기능을 제공하여 재배치의 영향을 받는 클라우드 소비자의 데이터에 허가되지 않은 사용자가 접근할 수 없도록 보호