

# 분산 게임서버 기술 동향

Technical Trend of Distributed Game Server

유비쿼터스 시대를 주도할  
디지털콘텐츠 기술 특집

임정열 (J.Y. Lim)	네트워크가상환경연구팀 기술원
박일규 (I.K. Park)	네트워크가상환경연구팀 연구원
정재용 (J.Y. Chung)	네트워크가상환경연구팀 연구원
심광현 (K.H. Shim)	네트워크가상환경연구팀 팀장

## 목 차

- .....
- I . 서론
  - II . 분산 게임서버 기술 분류
  - III . 분산 게임서버 기술 전망
  - IV . 결론

온라인 게임의 가장 큰 특성이자 매력은 사용자간 발생하는 상호작용이라고 할 수 있다. 사용자들은 게임상의 만남을 통해서 커뮤니티를 형성하고, 특정 목적을 위해서 협력하고 갈등한다. 이러한 특성으로 인해 온라인 게임의 성공여부가 동시에 게임을 즐기는 사용자 수로 평가되지만, 사용자 수의 증가와 사용자들간의 높은 상호작용성은 게임서버의 처리부담 급증을 가져 온다. 따라서, 다수의 사용자들에 대한 원활한 처리를 위해서 서버의 역할은 다수의 서버 시스템에 분산되어 처리되어야 하고, 이러한 기술을 분산 게임서버 기술이라고 한다. 본 고에서는 게임산업에서 사용되고 있는 분산 게임서버 기술을 분류하여 설명하고, 향후 기술 동향을 분석한다.

## I. 서론

온라인게임은 클라이언트(client)가 통신망을 통해서 서버에 접속, 다수의 사용자가 실시간으로 함께 진행되는 게임방식으로 정의할 수 있다. 일반적으로 클라이언트 사이의 연결만으로 진행되는 게임은 멀티플레이어 또는 네트워크 게임으로 분류할 수 있고, 여기에 서비스를 제공하는 게임서버와 이를 통한 사용자들의 만남이라는 요소가 추가되면 비로서 온라인게임으로 분류된다.

온라인게임에서 서버(server)가 필수 요소임은 확실하지만, 그 기능과 역할의 범위에 대해서는 뚜렷한 경계가 없다. 게임서버의 역할은 함께 게임을 즐길 사용자 그룹의 형성에서 시작해서, 사용자들이 게임을 즐길 환경의 제공이나 게임의 룰을 제공하는 기능을 수행할 수 있고, 더 나아가 내부적인 게임의 대부분을 처리하는 범위까지 확장할 수도 있다.

이러한 게임서버의 역할은 게임의 기획적인 특성에 따라 크게 달라진다. 소규모의 사용자가 한정된 장소에서 게임을 즐기는 방식이라면 게임서버는 이러한 유저들을 모아서 게임을 즐길 수 있도록 안내를 해주는 것으로 그 역할은 충분하다. 반면에, 대규모의 사용자가 거대한 세계에서 함께 탐험하는 게임 방식이라면 게임서버는 다수의 사용자가 서로를 인지하고 행동할 수 있도록 지원하고, 또한 탐험을 위한 장소와 이벤트도 제공할 수 있어야 한다.

이와 같이 게임서버는 상황에 따라 다른 역할을 담당하지만, 온라인게임 시장의 규모와 장르의 대중화 정도를 고려했을 때, 게임서버 1대가 서비스를 모두 전담하는 것은 거의 불가능하다. 따라서, 원활한 게임 서비스를 위하여 다수의 게임서버를 사용할 때, 전체 게임서버의 역할은 처리단계, 기능, 데이터 등의 기준에 나누어져 각 게임서버에 할당되어야 하고, 이렇게 역할이 분할되어 처리되는 방식을 분산 게임서버라고 한다.

분산 게임서버는 게임이 갖는 특성을 통해서 구성되는 것이 일반적이다. 제한된 소규모 사용자의 모임으로 진행되는 게임은 이러한 사용자의 모임을

처리단위로 분산 서버가 구성되고, 대규모 사용자 온라인게임(MMOG)에서는 채널(channel) 분리, 게임 월드(game world)의 분할 등을 통해 분산처리가 시도된다. 분산 게임서버는 그 동안 다양한 방면에서 시도되고 발전하였지만, 확장의 제약과 분산 게임서버간의 간섭에 따른 성능 감소 그리고 심리스(seamless) 월드 도입의 어려움과 같은 분명한 한계가 존재하기에 이후에도 지속적인 개선과 연구가 필요로 되고 있다.

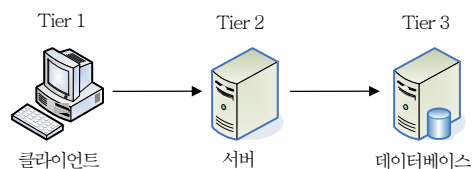
본 고에서는 게임산업에서 사용되고 있는 분산 게임서버 기술에 대해서 분류 및 설명하고, 이어서 분산 게임서버 기술의 추세와 전망을 제시하도록 하겠다.

## II. 분산 게임서버 기술 분류

### 1. 기술 개요

온라인게임은 기본적으로 3계층 클라이언트-서버모델(3-tier client-server model)로 구성된다. 여기서 클라이언트는 사용자가 실제로 보고 조작하는 게임이고, 클라이언트는 서버에 접속하여 게임을 진행한다. 또한, 온라인게임은 데이터베이스를 통해서 게임에서 사용되는 데이터를 저장하고 사용한다. 이러한 3계층 구성은 데이터와 관련된 처리를 데이터베이스로 분리하는 역할을 하고, 클라이언트의 데이터베이스에 대한 직접적인 접근을 차단함으로써 게임 데이터가 서버에 의해 공정하게 처리되는 효과를 가져온다(그림 1) 참조).

이러한 3계층 구조에서 게임 사용자의 증가는 서버에 접속해서 게임을 진행하는 클라이언트의 증가



(그림 1) 3계층 클라이언트-서버 모델

〈표 1〉 서버 분산 기술의 분류

기능 분산처리	게임서버의 기본 구조와 특성 정의를 통한 서버 기능 분산
데이터 분산처리	확장성에 제약을 주는 주요 데이터에 대한 처리 병렬화
공간 분산처리	독립된 서버그룹 추가를 통한 즉각적인 처리능력 향상

를 가져오고, 효과적인 게임진행을 위해서는 서버의 역할을 다수의 서버시스템에 분산처리 되도록 조절 해주어야 한다. 다수의 서버 시스템을 사용하기 위한 게임서버 분산은 게임서버의 처리절차와 데이터 등 여러 면에서 이루어질 수 있지만, 제작할 게임 시스템의 특성을 고려하여 높은 확장성과 효율성을 유지하는 것이 중요하다. 게임이 목표로 하는 사용자 서버가 감당하지 못하거나, 서버 시스템의 확충이 만족할 만한 처리용량 향상을 가져오지 못하는 상황은 게임 분산서버의 구성에 문제가 있는 경우이다.

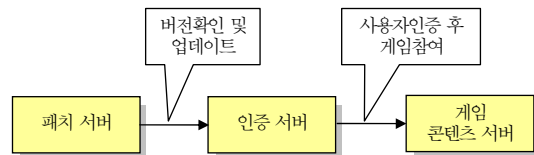
게임서버의 분산화 방향은 매우 다양하지만, 게임서버 제작에 널리 사용되는 방법은 <표 1>의 3가지로 분류할 수 있다. 이러한 서버 분산 기술은 분명 특성과 한계가 존재하기 때문에, 이러한 점에 대한 정확한 이해는 중요하고, 각 기술의 특성을 동시에 활용하여 게임서버 분산의 효과를 최대화하는 것이 바람직하다. 다음 절에서 각 서버 분산 기술에 관하여 고찰하자.

## 2. 기능 분산

게임서버를 다수의 시스템에 분산하여 처리하기로 결정했다면, 게임서버가 갖는 기본적인 구조와 게임의 특성에 맞는 세부 기능을 결정해야 한다. 이러한 정의는 게임서버가 이후 게임 콘텐츠 개발과 사용자 증가에 따라 유연성과 확장성을 동시에 지니도록 해야 하고, 이와 동시에 안정성 유지와 게임의 세부 기능에 대한 원활한 지원이 보장되도록 해야 한다. 이렇게 정의된 게임서버 구조에 따라 게임서버는 기본적인 분산처리를 하게 되고, 또한 정의된 구조를 기반으로 데이터 병렬화 기술을 도입함으로써 게임서버의 처리능력을 크게 향상시킬 수 있다.

게임서버의 구조에 따른 분산은 게임서버 기능 중 독립되어 처리 가능한 분산서버를 분리하고 정의하는 것이다. 이러한, 분산서버 분리는 일반적으로 분산서버가 담당할 기능에 따라 이루어진다고 나타낼 수 있지만, 분리방식을 세분하면 처리단계, 처리기능, 처리계층에 따른 분리로 나눌 수 있다. 처리단계, 처리기능, 처리계층에 따른 분산서버 분리는 다음과 같다.

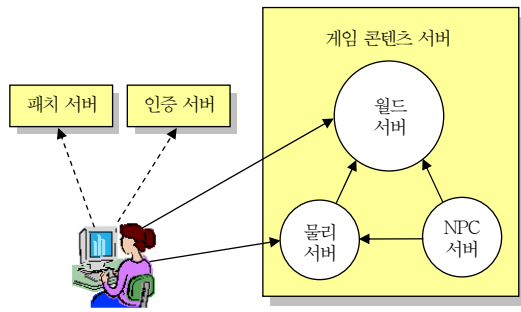
우선, 처리단계에 따른 분리는 클라이언트의 게임상태에 따른 분산서버의 기능을 분리하는 방식이다. 먼저 게임서버에 접속한 클라이언트는 버전 업데이트를 위해 게임패치(game patch)를 하고, 이어서 사용자 인증을 받게 된다. 또한, 게임에 따라 거대한 게임월드에 접속하기도 하거나 사용자들이 모여 있는 대기실에 접속하게 된다. 이러한, 클라이언트의 버전 확인, 사용자인증, 게임참여와 같은 사용자의 상태에 따라 게임서버의 기능을 분산하는 방식이 처리단계에 따른 분산서버 분리이다(그림 2) 참조.



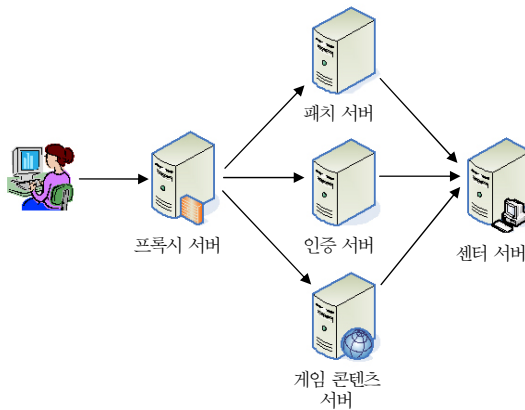
(그림 2) 처리단계에 의한 서버분산

이어서, 처리기능에 따른 분리는 처리단계에 따라 분산 정의된 게임서버가 클라이언트 요청에 대한 처리의 효율성과 확장성을 높이기 위해 기능적으로 분산처리 하는 방식이다. 만약, (그림 2)의 게임 콘텐츠 서버가 다수의 사용자가 동일한 공간에서 만나는 대규모 사용자 온라인게임일 경우, 처리기능에 따른 서버기능의 분리는 NPC 서버, 물리 서버와 같은 방식으로 될 수 있다. 하지만, 이러한 방식으로 분리된 서버들은 분산처리의 이점보다 높은 커뮤니케이션 비용을 발생시키는 상황이나 이후 적용될 데이터 분산과 상충되는 문제를 발생시킬 가능성이 높기 때문에 주의해서 적용해야 한다(그림 3) 참조.

마지막으로, 처리계층에 따른 분리는 처리단계와



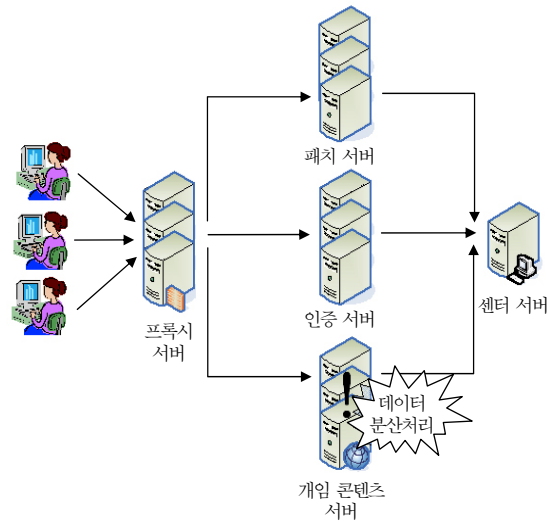
(그림 3) 처리기능에 의한 서버분산



(그림 4) 처리계층에 의한 서버분산

처리기능에 의해서 분산된 게임서버의 연결 또는 관리를 위한 계층을 구성하거나 기능 전담을 위한 계층을 추가하여 처리계층에 대해서 정의하는 방식이다. (그림 4)는 (그림 2)에서 분산서버를 관리하고 연결하는 센터서버(center server)와 클라이언트와 분산서버를 중계함으로써 내/외부망을 분리하고 초기접속을 계속적으로 활용할 수 있도록 하는 프록시 서버(proxy server)를 추가하였다. 이러한, 처리계층에 따른 분리는 분산처리의 연결성과 통합성을 보장하고, 추가적인 기능적 분산효과를 가져올 수 있는 만큼 분산처리 서버 구성요소의 변경 시에 처리계층의 적절성을 다시 고려해야 한다.

이러한 기능에 의한 분산은 병렬처리를 구성하는 방법으로 분산처리의 처리능력을 더욱 높일 수 있다. 특히, 사용자 수에 따라 지속적으로 부하가 상승하는 경우는 병렬처리를 통해서 수용성을 계속적으



(그림 5) 분산서버의 병렬처리 구성

로 향상시킬 수 있기 때문에, 더욱 효과가 크다고 말할 수 있다. 하지만, 기능에 의해서 분리된 모든 분산서버에 바로 병렬처리를 적용시킬 수 있는 것은 아니다. 패치서버와 인증서버처럼 클라이언트가 게임에 참여하기 위해서 임시적으로 거쳐가는 서버의 경우는 병렬처리를 어렵지 않게 적용할 수 있지만, 사용자들이 실제로 게임을 진행하는 게임 콘텐츠 서버에서는 사용자들이 동일한 공간에 존재하는 데 따르는 어려움과 온라인게임의 가장 큰 특징인 상호작용성이 극명하게 나타나기 때문에, 병렬처리로의 전환이 어려운 경우가 흔히 존재한다. 이렇게 병렬처리 구성에 어려움이 있을 경우에는 게임 데이터를 나누어 처리하는 방식으로 게임서버의 확장성을 향상시킨다. 다음 절은 이러한 데이터 분산을 통한 분산 서버 구성에 대해서 설명하도록 하겠다(그림 5 참조).

### 3. 데이터 분산

온라인 게임서버가 높은 확장성을 갖기 위해서는 사용자의 증가에 따라 로드(load)가 높아지는 서버에 대한 분산 처리가 뒷받침되어야 한다. 하지만, 분산 처리되어야 할 서버가 온라인 게임이 갖는 특정

한 제약에 의해서 분산처리 될 독립적인 요소를 찾기 어려울 경우, 사용자 수에 영향을 받는 데이터에 대한 분산을 시도하게 된다.

일반적으로 온라인 게임서버의 로드는 사용자 수에 따라 급증하기 때문에 데이터 분산은 실질적으로 게임 사용자를 효과적으로 분산하는 기술이라고 할 수 있다. 사용자를 분산할 데이터의 선택과 분할은 게임 특성에 따라 다르지만, 사용자간 상호작용이 가장 빈번하게 발생하는 요소를 모으고, 낮게 발생하는 요소를 기준으로 분할하는 것이 효과적이다.

이러한 기준에 따라 온라인 게임은 인접한 사용자간의 상호작용이 빈번한 특성으로 인해 게임의 공간을 분산될 데이터로 빈번하게 사용한다. 즉, 소규모 사용자가 독립된 공간에서 즐기는 게임이라면 각 공간을 기준으로 데이터가 분산되고, 대규모 사용자가 단일 공간에 참여하는 게임이라면 공간의 분할을 통해서 데이터가 분산된다. 특히, 단일 공간에서 진행되는 게임의 공간 분할 처리는 게임 기획과 구현 기술적인 요소를 필요로 하기 때문에 상대적으로 어려운 분산 기법으로 분류되고, 일반적으로 말하는 분산서버가 이러한 처리 기법을 나타내고 있다. 이에 따라 이번 절은 단일 공간을 기준으로 데이터 분산 기법에 대해서 설명하도록 하겠다.

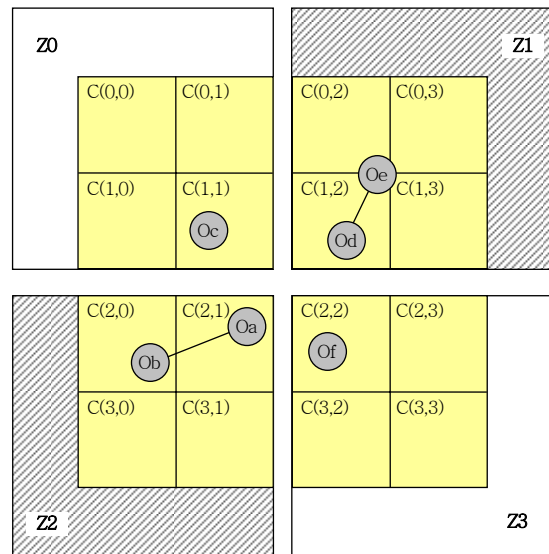
단일 공간의 분할을 통한 사용자 분산은 존 분할 서버 모델(zone-based server model)과 심리스 서버 모델(seamless server model)의 두 가지로 분류[1]된다. 우선 존 분할 서버 모델은 게임 공간을 미리 분할하여, 분할 영역별로 서버에 할당하는 방식을 사용한다. 여기서 사용자는 하나의 존에만 속할 수 있고, 존 간의 상호작용은 없거나 거의 없는 특성을 가지고 있다. 심리스 서버 모델은 마찬가지로 전체 월드를 분할하여 서버에 할당하여 처리하지만, 분할된 경계가 명시적으로 사용자에게 드러나지 않고, 인접한 분할 영역간의 모든 상호작용을 지원한다. 이러한 두 가지 데이터 분산 모델에 대한 선택은 게임의 제작 및 구현방식에 크게 영향을 줄 수 있는 만큼 게임 기획 초기에 이루어져야 하며, 그 장단점에 대한 고민이 선행되어야 한다. 존 분할 서버 모

델과 심리스 서버 모델에 대해서 조금 더 자세히 알아보자.

존 분할 서버 모델은 월드를 각각의 존으로 분할하여 제작하고, 각 존에 대한 처리를 서버에 분산하는 방식이다. 이러한 방식은 존 간의 상호작용이 없고, 사용자들은 게임의 이벤트를 통해서 존의 경계를 이동한다. 이러한 방식은 데이터 분산의 기술적인 어려움을 낮춰서 게임의 제작 시간을 단축시키는 장점을 가지고 있지만, 존의 영역 이동에 따른 게임의 몰입감 감소와 존 간의 영역을 재설정의 어려움에 따른 로드불균형 그리고, 서버 경계지점에서의 데이터처리 약용 가능성 등을 가지고 있다. 하지만, 로컬처리에 필요한 대부분의 오브젝트에 대한 접근이 보장되기 때문에 개발이 쉬워지는 장점은 여전히 매력적이어서, 현재까지의 대부분의 대규모 사용자 온라인 게임에서는 이러한 방식을 사용하고 있다.

(그림 6)은 존 분할 서버 모델을 보여주고 있다. 16개의 셀로 분할된 월드를 Z0~Z3의 4개의 존에 4개씩 할당해서 분산처리 되도록 하였다. 월드에는 Oa~Of의 6개의 객체(object)가 존재하고, 각 객체는 동일한 존의 객체와만 상호작용 할 수 있다.

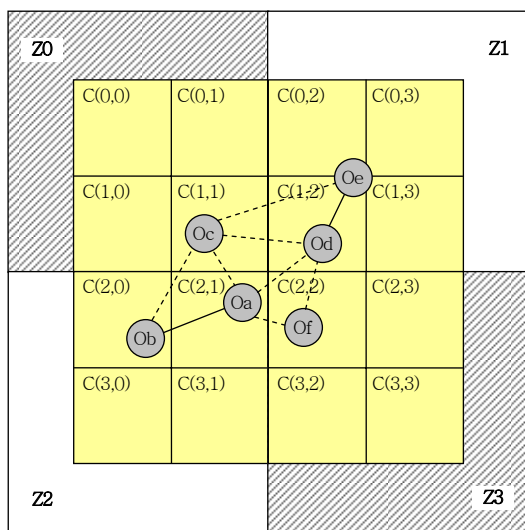
심리스 서버 모델은 월드가 분산되어 처리되지만



(그림 6) 존 분할 서버 모델

게임상에서는 한 곳에서 처리되는 것처럼 보이게 한다. 사용자들은 경계지점에 대해서 명확히 인지하지 못하고, 서로 다른 서버에서 처리되는 사용자들도 인접한 지역에서의 상호작용은 모두 보장하도록 한다. 이러한 요소들은 사용자들에게 더욱 몰입감 있는 게임환경을 제공할 수 있다는 장점을 제공하고, 대규모 사용자가 한 장소에 모일 수 있는 점은 게임 기획에 적극 활용될 수 있는 요소임이 분명하다. 하지만, 심리스 서버 모델은 게임 개발과 유지에 있어서 모든 점이 복잡[1]해지는 큰 단점을 지니고 있다. 특히 경계지역에서의 객체간 상호작용의 지원을 위해서 오브젝트들에 대한 구현이 여러 서버나 복잡한 전달 계층을 통해서 이루어질 가능성[2]이 높고, 이러한 요소의 많은 부분은 비동기적으로 구현되어야 하기 때문에 수 많은 처리 실패 가능성과 이에 따른 버그들이 발생할 가능성이 높다. 이러한 어려움에도 불구하고 심리스 서버 모델은 실제 세계와 유사한 게임 환경 제공이나 동적인 경계할당과 재조정을 통한 서버활용 최적화와 같은 강점을 제공하기 때문에 지속적인 개발과 연구가 이루어지고 있다.

(그림 7)은 심리스 서버 모델을 보여주고 있다. (그림 6)에서와 마찬가지로 월드가 분산되어 처리되고 있지만, 월드에 존재하는  $O_a \sim O_f$ 의 6개 객체는



(그림 7) 심리스 서버 모델

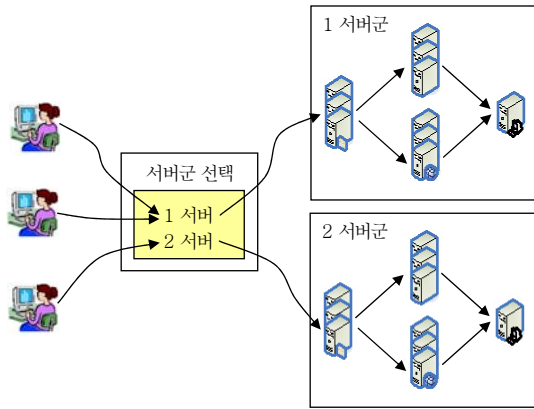
서로 다른 존에 존재하더라도 상호작용을 할 수 있다. 하지만, 점선으로 표현된 경계지역에서의 상호작용은 게임 개발과 유지의 어려움을 가중시킨다.

존 분할 서버 모델과 심리스 서버 모델에 대한 선택은 게임개발 규모와 게임 기획의 특성, 그리고 구현 가능성 등에 대한 전체적인 고려가 선행되어야 한다. 이러한 공간 분산이 갖는 장단점에 대해서 정확히 인식하고 사용해야 하고, 이러한 분산모델이 전부는 아니기 때문에 데이터 분산에 대한 계속적인 논의와 연구가 필요하다. 이어서, 다음 절은 비교적 손쉽게 적용할 수 있는 공간 분산에 대해서 설명하도록 하겠다.

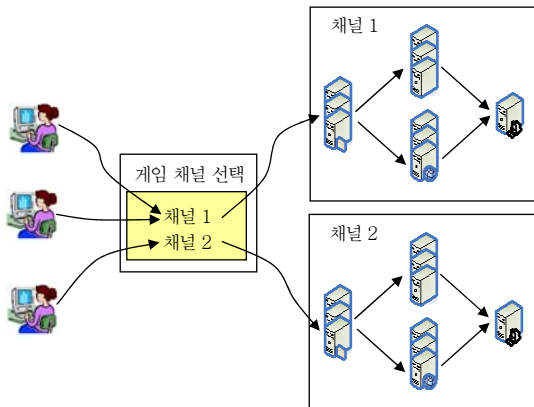
#### 4. 공간 분산

앞 절에서 기능 분산과 데이터 분산을 통한 분산 게임서버에 대해서 설명을 했지만, 가장 손쉬운 분산서버 구성은 게임의 공간을 병렬적으로 설치하는 공간 분산기술이다. 공간 분산은 서비스가 가능한 독립된 게임서버 그룹을 다수 설치함으로써 게임서버의 수용능력을 손쉽게 높일 수 있고, 특정 공간에 사용자가 밀집할 가능성을 낮출 수 있기 때문에 서버 부하를 전반적으로 낮출 수 있다. 이러한 방법은 온라인게임에서 다수의 서버군(server group) 제공 또는 다수의 게임 채널(game channel) 제공과 같은 방법으로 활용되고 있다.

(그림 8)은 다수의 서버군을 설치하여 사용자에게 제공하는 방법의 예시이다. 사용자가 게임에 접속할 때 참여할 서버군을 선택하면, 선택한 서버군에 접속하여 게임을 즐기는 방식이고, 다른 서버군에 참여중인 사용자는 완전히 별개의 공간에서 게임을 즐기는 것과 같기 때문에 함께 게임을 즐길 수 없다. 이러한 방식은 완전히 독립된 다수의 공간을 사용자에게 제공함으로써 서비스 질을 높이고, 사용자에게 더 많은 기회를 제공하는 추가적인 효과를 제공하지만, 서버군 사이의 사용자 수 불균형과 사용자간의 커뮤니티 활성화 실패와 같은 문제점을 발생시킬 수 있다.



(그림 8) 다수의 서버군 제공



(그림 9) 다수의 게임채널 제공

(그림 9)는 다수의 서버군 설치를 통한 분산과 유사한 방식으로 구성된 다수의 게임 채널 분리에 대한 예시이다. 이러한 방식은 다수의 서버군 설치와 유사한 특성을 갖지만, 채널은 일시적으로 제공되는 게임 공간이므로 서로 다른 채널을 사용중인 사용자가 같은 채널에 다시 참여를 할 경우는 함께 게임을 즐길 수 있다. 대체로 게임 채널의 활용은 소규모 사용자가 모여서 즐기는 방식의 게임에서 자주 활용된다.

지금까지 분산서버 구성 기술을 분류하고 그 각각에 대해서 알아보았다. 다음 절에서는 이러한 분산 서버기술의 활용 추세와 그 발전 방향에 대해서 전망하겠다.

### Ⅲ. 분산 게임서버 기술 전망

지금까지 설명한 분산 서버 기술은 현재 개발되는 온라인 게임에도 계속적으로 활용되고 있다. 여전히, 존 분할 서버 모델은 많은 온라인 게임에 활용되고 있고, 심리스 서버 모델의 기술 활용과 동시에 기술이 개발되고 있다. 또한, 서버군 또는 채널 분산은 기획에서 적극적으로 활용되는 추세이고, 게임서버의 전체적인 구조는 기본적인 형태에서 각 온라인 게임에 적합한 모습으로 보완되어 활용되고 있다.

특히, 존 분할 서버 모델과 심리스 서버 모델은 서로 상충되는 특성을 지니고 있기 때문에 개발되는 온라인게임에서 장점을 살리고, 단점을 보완하는 노력이 계속되고 있다. 이에 따른 각 분산 서버 모델의 활용 추세 및 전망을 게임 기획, 서버 시스템, 분산 서버 기술로 나누어 제시하겠다.

#### 1. 게임 기획

분산 게임서버 구조는 게임 기획에 따라 크게 달라진다. 하지만, 주된 게임의 장르와 기획에 따라 서버의 구조가 일정수준 정해진 경우라도 기획에 분산 처리 가능한 요소를 추가할 경우 게임서버의 분산에 활용될 수 있다.

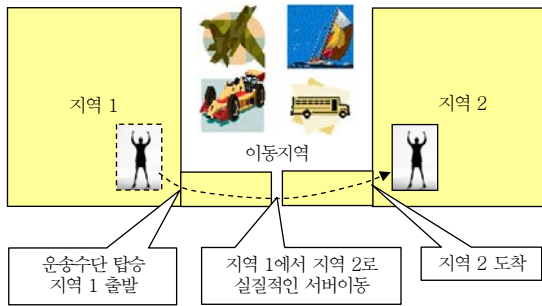
예를 들어, 사용자가 자신만의 공간을 원하는 점을 활용한 집 짓기 시스템의 도입이나, 일정 수의 사용자만을 위한 독립된 탐험 공간의 제공 기획의 추가는 게임서버의 분산에 적극 활용될 수 있다. 또한, 서로 다른 서버군에 존재하는 사용자들이 한 장소에서 만나서 게임을 즐기길 원하는 점을 활용해서 일정 수의 사용자가 서버군에 상관없이 참여할 수 있는 미니게임을 추가하는 것도 예가 될 수 있다.

분산처리가 가능한 기획요소 추가와는 별도로 분산 게임서버의 구조에 따른 어려움을 게임 기획을 통해서 보완하는 것도 가능하다.

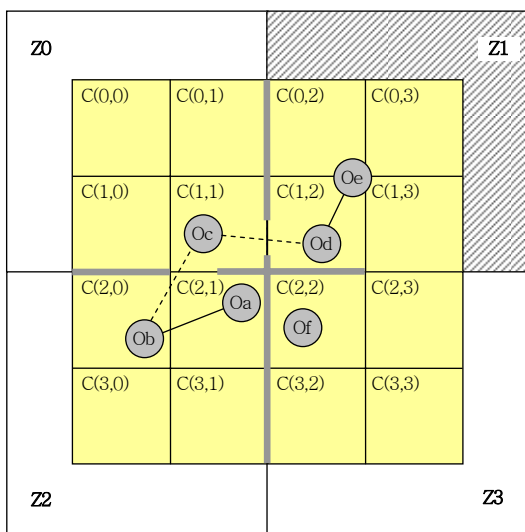
가령, 존 분할 서버 모델의 경우는 월드의 인위적인 분할로 인해 게임의 사실성이 감소하는 단점이 있다. 하지만, 존 이동이 포털을 통해서 이루어지는

것이 아니라 배와 비행기 같은 합리적인 수단을 통해서 이루어지도록 한다면 게임의 사실성을 향상시킬 수 있다. 게다가, 이러한 부분의 그래픽 효과를 높이거나 간단한 게임을 추가해서 사용자를 즐겁게 한다면 그 효과는 더욱 커질 수 있다(그림 10) 참조.

또한, 심리스 서버 모델은 경계지역에서 상호작용이 높은 개발 부담이 발생하므로 기획적으로 경계지역 관련된 부담을 최소화하는 방법이 사용될 수 있다. 경계지역에 산과 강과 같은 요소를 배치해서 경계 영역을 최소화하거나 기획적으로 경계영역에서 상호작용 발생빈도를 낮춘다면, 처리 성능이 높지 않더라도 빠르게 구현될 수 있는 경계지역 처리 기술을 사용하는 것이 가능하다. 이러한 기획적인



(그림 10) 이동수단을 이용한 존 이동



(그림 11) 심리스 서버 모델의 경계지역 축소

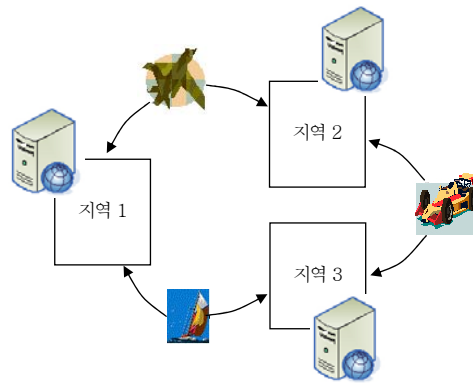
고려는 심리스 서버 모델 적용에 따른 부담을 줄여 줄 수 있다(그림 11) 참조.

## 2. 서버 시스템

분산서버의 목적이 단일 서버 시스템이 갖는 수용성의 한계를 극복하기 위해 다수의 서버 시스템을 사용하는 것이라고 했을 때, 서버 시스템의 성능향상은 분산 게임서버의 수용성 향상을 가져온다. 이러한 상황은 다르게 생각하면 게임이 목표로 하는 일정 수준의 수용성을 갖추는 데 필요한 서버 시스템 수가 적어지는 효과를 가져온다.

이러한 서버 시스템 처리능력 증가에 따라 일정 수준의 사용자 수용을 목표로 한정된 개수의 서버 시스템을 이용하여 분산서버를 구성하는 게임이 많아졌고, 고성능의 서버 시스템을 활용해서 서버 시스템의 수를 최소화하려고 노력하는 게임도 있다.

(그림 12)는 3개의 지역으로 구성된 온라인 게임을 나타내고 있다. 각 지역은 1개의 서버시스템에 의해서 처리되고, 지역간 이동은 비행기, 자동차, 배와 같은 이동수단에 의해서 자연스럽게 이루어진다. 각 지역을 처리하는 일반적인 서버시스템의 처리능력이 사용자 500명이라면, 위의 게임은 최소 500명에서 최대 1500명의 수용능력을 가지고 있다. 하지만, 처리능력이 사용자 1500명인 고성능의 서버시스템을 도입하게 되면, 최소 1500명에서 최대 4500명의 수용능력을 확보할 수 있다. 또한, 기획요소에 의



(그림 12) 고성능 서버를 이용한 분산처리



한 고른 사용자 분산은 최대에 가까운 수용능력 발휘로 이러한 방식의 서버구성의 효율성을 높여줄 수 있다.

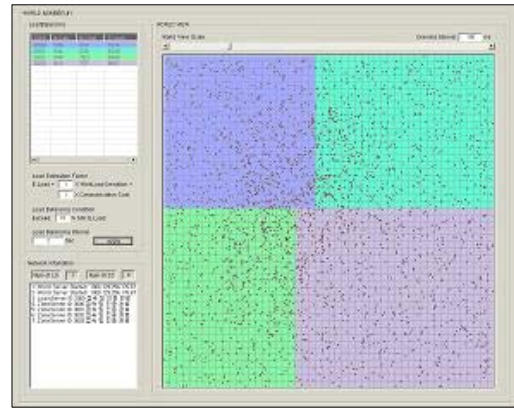
고성능 서버 시스템의 적극적인 활용은 게임의 수용능력을 손쉽게 상승시키고, 분산 구성에 따른 노력을 경감시켜서 게임 제작시간을 단축시켜 주지만, 고성능 서버 시스템 도입에 따라 높은 서비스 비용이 발생하고, 각 서버가 많은 부분을 담당하는 만큼 높은 서버 안정성이 요구된다. 따라서, 서비스 비용보다는 빠른 게임 제작과 출시가 요구된다면 이러한 시스템을 도입하는 것이 좋은 전략이 될 수 있지만, 합리적인 서비스 비용 역시 요구된다. 이러한 고성능 서버 시스템의 도입은 게임의 기획과 규모, 출시시기 등을 종합적으로 고려해서 그 도입 수준이 결정되어야 한다.

서버 시스템의 성능은 계속적으로 발전하고, 이에 따라 게임서버의 수용력은 자연스럽게 상승한다. 하지만, 기술 발전은 새로운 요구와 서비스에 대한 요구로 이어지기 때문에, 계속적인 서버 시스템의 성능 개선에 대한 의존보다는 보편적으로 적용할 수 있는 분산서버 기술에 대한 연구가 필요하다.

### 3. 분산서버 기술

분산서버 기술은 서버 구조, 네트워크 프로토콜, 게임 데이터의 분산 등 여러 분야에 걸쳐서 활발하게 연구되고 있다. 이러한, 분산 기술의 연구는 수많은 사용자들에게 다양한 서비스를 원활하게 제공하는 것이 주된 목적이라고 할 수 있다.

최근에는 온라인 게임 엔진 개발과 판매를 위한 연구도 활발하게 이루어지고 있다. 이러한 게임 엔진에서 개발되는 분산서버 기술은 대체로 심리스 서버 모델과 관련이 있다. 앞장에서 심리스 서버 모델은 다수의 사용자가 단일 월드에서 함께 게임을 즐기는 장점을 제공하지만, 개발과 유지의 어려움을 유발하기 때문에 쉽게 사용되지 못하고 있다고 설명했었다. 이에 따라, 심리스 서버 모델의 구현과 관련된 별도의 라이브러리(library)를 제작하거나[3] 스



(그림 13) 심리스 서버와 동적 로드밸런싱

크립트를 이용한 처리[2]로 분리하는 방법을 통해서 그 적용 난이도를 낮추는 개발이 되고 있다. 또한, 심리스 서버 모델은 동적인 관리영역 조절로 분산서버의 처리부하를 조절하는 동적 로드밸런싱(DLB)과 일부 서버의 장애 발생 시에도 서비스를 계속적으로 할 수 있도록 하는 시스템 무결성 적용이 용이한 시스템으로 평가되고 있고, 실제로 이러한 기능들이 함께 연구되고 있다(그림 13) 참조).

위와 같은 심리스 서버 모델에 대한 연구뿐만 아니라, 다양한 서버 구조에 대한 연구 또한 진행되고 있다. 이러한 연구로는 기존의 지역기반 분산을 사용자 기반 분산으로 재구성하거나 대규모 사용자 온라인에서 P2P를 적극적으로 활용하기 위한 구조 개선 등이 있다. 이러한 방식의 서버 구조에 대한 변화는 온라인게임에 일반적으로 적용하기에는 무리가 있을 수 있지만, 다양한 서버 구조 제시 이후의 새로운 게임에 주도적으로 사용될 수 있는 측면에서 이러한 시도들의 높은 가치를 찾아볼 수 있다.

## IV. 결론

다수의 사용자를 대상으로 하는 온라인 게임은 분산처리가 필수적이다. 하지만, 제작되는 게임에 따라 분산처리의 적용방식에는 많은 차이가 있기 때문에 분산처리의 과정을 이해하고, 분산처리와 관련

된 사례를 다수 접하는 것이 효과적인 분산서버 구성에 큰 도움이 될 것이다.

온라인 게임산업은 지속적인 발전과 시장확대로 그 어느 때보다 경쟁이 치열하다. 이러한 시장상황에서 온라인 게임은 살아남기 위해 신선한 기획을 바탕으로 한 빠른 게임제작과 대규모의 인력과 기술을 투입하여 제작하는 게임으로 양분되고 있다.

하지만, 어떤 방식이더라도 온라인 게임 기획이 현실화되기 위해서는 게임 기술을 구축하여야 할 필요성이 생긴다. 이러한 기술은 외부에서 엔진 또는 라이브러리 형태로 도입되거나 자체적으로 개발될 수 있다. 하지만, 사용되는 제작기술은 게임의 제작 기간과 품질에 영향을 줘서 결과적으로 게임의 성공 여부를 결정하기 때문에 어떠한 기술을 적용할지, 어떠한 방식으로 기술을 확보할 것인지 충분히 고민하여 결정해야 한다.

본 고에서는 온라인게임에서 사용중인 분산서버 기술에 대해서 분류하고 그 동향에 대해서 기술하였다. 다른 온라인게임 기술과 마찬가지로 분산서버 기술은 게임의 성공 여부에 영향을 주겠지만, 게임이 제작된 후 서비스 비용 수준과 사용자 증가에 따른 서비스 품질을 결정하는 점에서 중요한 부분이라고 할 수 있다.

분산서버는 본 고에서 설명한 처리 기능, 단계,

계층과 데이터 분산, 공간 분산을 통해서 구축되고, 적용 방식의 체계화와 새로운 적용 기술 개발로 그 효율성을 증가시킬 수 있다. 하지만, 이러한 방식 이외에도 분산서버는 네트워크 프로토콜의 개선, 멀티캐스트 관련 연구 등을 통해서 그 성능이 개선될 수 있으며, 이러한 기술연구와 함께 그 적용방식이 크게 달라질 수 있는 만큼 다양한 기술 변화를 주시해야 하고, 개발과 연구를 통한 기술 선도의 노력이 필요한 시점이다.

## 약어 정리

DLB	Dynamic Load Balancing
MMOG	Massively Multiplayer Online Game
NPC	Non Playable Character
P2P	Peer to Peer

## 참 고 문 헌

- [1] Massively Multiplayer Game Development, Charles River Media, 2003, pp.213-227.
- [2] [http://www.gdconf.com/archives/2004/johnson\\_jeff.doc](http://www.gdconf.com/archives/2004/johnson_jeff.doc)
- [3] Zona, inc, <http://www.zona.net>