

MMORPG에서 VML를 이용한 동적 지역분할

이정진*, 원동기*, 김법균*, 두길수*, 안동언*, 정성중*

*전북대학교 컴퓨터공학과

e-mail : jeongjin2@chonbuk.ac.kr

Dynamic Local-Partition Using VML in MMORPG

Jeong-Jin Lee*, Dong-ki Won*, Beob-Kyun Kim*,
Gil-Soo Doo*, Dong-Un Ann*, Seung-Jong Chung*

*Dept of Computer Engineering, Chon-Buk University

요 약

오늘날의 MMORPG 환경에서 수만~수십만 명의 유저들이 동시 접속하여 게임을 즐기고 있다. 수많은 인원이 같은 맵 상에서 활동하기 때문에 최근의 게임시스템들은 맵의 영역에 따라 유저를 분할하여 처리하고 있다. 이를 위해 맵 분할 시스템에 대한 연구가 다양하게 이루어지고 있으나 기존 맵 분할 시스템은 유동적인 유저들에 대해 대처하기엔 문제점을 가지고 있다. 본 논문에서는 기존 맵 분할 방법들에 대한 구조적 단점을 보완하고, 부하에 능동적으로 대처할 수 있도록 VML를 이용한 동적 맵 분할 방법을 제안한다.

1. 서론

최근 컴퓨터 시스템과 인터넷이 발전하고 게임 콘텐츠의 개발됨에 따라 게임 산업이 크게 발전하고 있다. 특히 여러 사람이 함께 즐기는 온라인 게임은 네트워크의 발전과 함께 급속도로 확산 보급되고 있으며, 고부가가치를 창출하는 미래의 핵심 산업으로 각광을 받고 있다. 온라인 게임 중에서도 RPG장르의 게임들은 최대한 많은 사용자를 수용하고, 사용자들에게 양질의 서비스를 안정적으로 제공하는 서버의 역할이 가장 중요하다고 볼 수 있다. 이를 위해서는 다수의 사용자들이 하나의 서버에 집중되는 구조를 피하고, 과 부하된 서버의 부하를 분산시키는 방법이 필요하며 이에 대한 연구가 활발하게 진행되어지고 있다.

본 논문에서는 다수의 사용자가 접속 가능한 MMORPG (Massively Multi-player Online Role Playing Game) 환경 하에서 VML (Virtual Map Layer)을 이용한 동적인 부하 분산 방법을 제시하였다. VML을 도입함으로써 동적 지역분할이 가능해

지고 사용자들의 이동 패턴에 따라 맵 분할에 대한 융통성을 제공할 수 있다.

2장에서는 기존 게임 시스템의 구조와 최근 연구되는 MMORPG의 부하 분산 방법들에 대해 언급할 것이고, 3장은 VML에 대한 이론적인 면을 설명하려고 하고, 4장에서는 VML를 적용한 시스템의 구조에 대해 제시할 것이다.

2. 관련 연구

기존 RPG 시스템 구조는 가상공간의 운용 방식에 따라 크게 두 가지로 나뉜다. 중앙집중식구조와 지역분할방식구조이다.[4]

중앙 집중식 구조는 하나의 서버에서 완전히 독립된 영역을 처리하도록 하는 구조이다. 독립된 서버를 수십대 두어 운영하는 채널 구조로 운용할 수도 있으며, 설계와 개발이 용이하나 확장성에 한계가 있어 MMORPG 환경에서는 적합하지 않다.

지역분할방식구조는 가상공간을 지역적으로 분할하고, 각 서버가 관리권을 갖는 방식이다. 가상공간을 기획 적으로 나누는 Zone 개념을 두어 엄격하게

공간을 분할하는 명시적 지역분할방식과 가상공간을 분할하여 관리 하기는 하지만, 관리영역을 기획 적으로 분할하지 않고, PC/NPC가 자유자제로 경계를 넘나들 수 있도록 하는 묵시적 지역분할방식, 마지막으로 서버의 부하에 따라 관리영역을 재조정하는 동적 지역분할방식으로 나뉜다.[4]

연구 사례로는 Butterfly.net에서 Globus Toolkit을 이용하여 Grid 기술을 도입한 Butterfly Grid에서 출발한 EMERGENT[1] 프로젝트가 있고, IBM Almden Research Lab에서는 FEM (Finite Element Modeling, 유한요소모델링)[2]을 기반으로 data의 연결 상관관계에 따라 분리하여 처리하는 방식을 도입한 Optimal Grid 프로젝트가 있다. 또한 지역 분할된 가상공간의 인접구역 연결에 대하여 Seamless Map[5] 연구가 진행되고 있다.

위의 연구 사례처럼 최근 MMORPG 시스템은 Seamless Map을 이용한 묵시적 지역분할방식을 적용하고 있지만, 유동적인 Client들에 대해 한 서버로의 집중현상에 대해서는 매우 정적인 구조를 가지고 있다. 따라서 본 논문에서는 그 해결방안으로 VML를 이용한 동적지역분할방식을 제안하려한다.

3. VML 이용한 동적 지역분할 방법

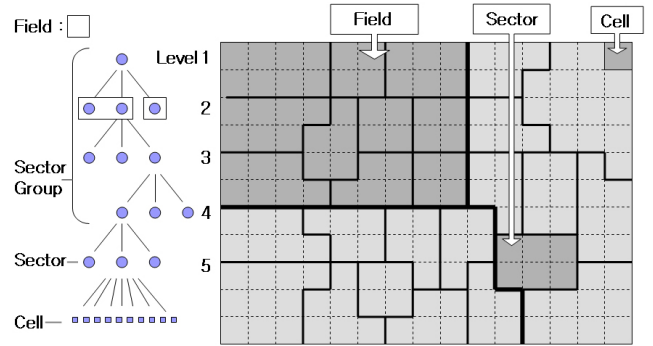
3.1 VML

Cell	지역 분할을 위해 구성되는 최소 단위의 영역
Sector	예상되는 부하량을 바탕으로 Cell들의 집합으로 구성되는 가상의 영역 Sector 단위로 Map의 분할 및 병합이 이루어짐
Sector Group	다수의 단계가 존재할 수 있음 Sector들로 이루어진 계층
Field	한 서버가 관리하는 영역 Field의 크기는 부하량에 따른 분할 및 병합으로 인해 유동적임

<표 1> VML의 용어들

본 논문에서는 기존의 Map 구조에 VML이란 가상의 Map Layer를 추가하고, 이 Layer에서 나눈 가상의 영역단위마다의 부하량을 기초로 지역분할 및 병합을 시도한다. 이 가상의 영역을 Sector라 부르며 이는 Optimal Grid의 OPC (Original Problem

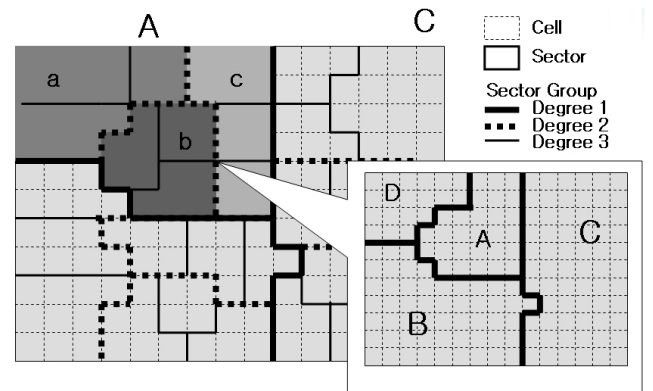
Cell)[2] 와 유사한 개념이다. 게임 설계 시부터 고려되어야 하며, Cell 이라는 더 작은 크기의 단위를 기초로 구성되고 그 모양 및 계층구조가 설계자 또는 관리자의 의도에 따라 융통성 있게 변할 수 있다는 점에서 차이가 있다.



(그림 1) VML의 구조

3.2 VML 설계

Sector는 예상되는 부하량을 기초로 Cell 단위로 구성되며 이를 바탕으로 Field가 구성되어 물리적으로 독립된 Field Server (Game Server, Processing Server라 부르기도 함)가 관리한다. Field는 Field Server의 분할 여부에 따라 작아지기도 하고, 다시 처음 상태로 돌아오기도 한다.



(그림 2) VML 분할

3.3 분할 및 병합 알고리즘

특정 Field에 임계치 이상의 부하가 발생하면 Field를 구성하는 Sector들의 부하량을 고려한 적절한 조합을 선택한 후 분할한다<표2>. 병합알고리즘도 이와 유사하며 맵 자체의 특성에 따라 Sector 단위의 가중치를 부여하여 적용할 수도 있다. 분할 및

병합 시 효율성 및 공간의 안정성을 위해 3개의 Sector Group으로 상위 Level의 Sector 또는 Field를 구성한다.

<표 2> FieldPartition Algorithms

```

algorithm val FieldPartition (field A)

s1, s2, s3 = get sub-sectors
ls1, ls2, ls3 = get load of s1, s2, s3
(sc.a, sc.b) = get best combination of s1, s2, s3
              based on ls1, ls2, ls3

allocate new field (sc.a)
reduce the existing field (sc.b)
synchronize field servers (sc.a, sc.b)
    
```

조에 VML Management Server가 추가하였고, VML을 위한 정보를 Map DB에 새로 추가하였으며 일정량의 Waiting Field Server(대기필드서버)를 두어 동적 지역분할을 준비토록 하였다.

4.2 시스템의 기능

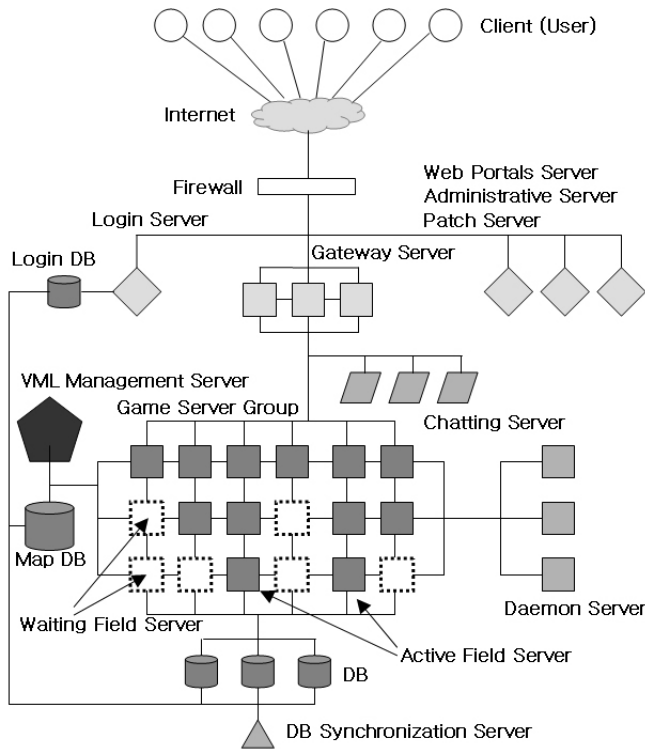
Map DB에서는 Cell, SectorGroup, Sector등의 정적인 Map Layer 정보들과 Field, FieldServer등의 유동적인 Map Layer 정보들을 가지고 있다.

Waiting Field Server는 Active Field Server(활성 필드서버)가 지역분할을 하고자할 때 일정량의 가상 공간을 할당받아 관리 권한을 이전받은 후 게임연산을 처리한다.

VML Management Server는 Field Server들의 Sector당 부하량을 측정하고<표3>, 동적지역분할을 할 것인지 결정한 후, 분할하게 된다면 Waiting Field Server를 가동시켜 가상공간을 분할하도록 하고, Data를 동기화시켜주는 역할을 한다.

<표 3> 부하량 계산에 필요한 요소들

SectorID	: 섹터당 부하량 산출
NumOfUser	: 유저의 수
Activity	: 유저의 활동성
Performance	: 서버의 성능
MapWeight	: 맵 영역별 가중치



(그림 3) 제안한 MMORPG 시스템의 구조

4. 시스템의 구조

4.1 시스템의 구조

(그림3)은 본 논문에서 제안하고자 하는 시스템의 전체적인 구조이다. 기존 MMORPG 시스템의 구

5. 결론

MMORPG 환경 하에서 동적 지역분할 방식을 운용함으로써 기존의 정적 지역분할 방식과는 달리 Map의 설계 시부터 동적인 지역분할을 염두 해두고 설계해야 한다는 단점이 있지만, 게임 중 유동적인 사용자에게 의한 몰림 현상에 대해 대응할 수 있고, 맵 영역과 유저 수를 크게 늘릴 수 있다는 장점이 있다.

또한 제안된 시스템은 시스템 분석 자료, 맵의 추가/변경, 사용자의 이용 패턴 분석 등의 자료에 의해 분할방법이 변경될 수 있으며 이는 VML의 수정을 통해 가능하다. 즉, 기존의 안정화된 시스템 구조를 바꾸지 않고, 몇몇 모듈의 추가/변경 등으로 가능하다는 장점이 있다.

참고문헌

- [1] IBM Grid Research Lab "Butterfly.net: Powering Next-Generation Gaming with Computing On-Demand" 『Butterfly.net』
http://www.butterfly.net
- [2] Tobin J, Lehman, James H. Kaufman. "OptimalGrid: Middleware for Automatic Deployment of Distributed FEM Problems on an Internet-Based Computing Grid", 『IEEE International Conference on Cluster Computing(CLUSTER'03)』
- [3] 이철민, 박홍성. "다중 사용자 게임 성능 향상을 위한 데이터 가상 그룹핑 방법", 『정보과학회논문지』 소프트웨어 및 응용 제 30권 제3호
- [4] 신동원 "A Study on Online Game Server Architecture" Synczone.Net
- [5] 문성원 "분산 seamless 게임 서버에서의 효율적인 게임 공간 관리 기술" 서강대 정보통신대학원 학위 논문
- [6] 이남재 곽훈성 "진화하는 온라인 롤플레이팅 게임을 위한 분산형 게임 서버 모델" 한국게임학회 논문집
- [7] 이정진 외 3명 "MMORPG 부하 분산을 위한 동적 맵 분할 시스템 설계" 정보과학회 2005년 춘계 학술대회 논문집
- [8] 김훈 "온라인 게임서버를 위한 미들웨어의 분산 처리 기법 설계" 숭실대 정보과학대학원 학위 논문
- [9] P.Morillo, J.M. Orduna, M.Fernandez "A Comparison Study of Evolutive Algorithms for Solving the Partitioning Problem in Distributed Virtual Environment Systems" Parallel Computing Vol.30
- [10] C.M.Greenhalgh, and S.D.Benford, MASSIVE "A Distributed Virtual Reality System Incorporation Spatial Trading, Proceedings of 15th International Conference on Distributed Computing Systems" Los Alamitos CA, ACM Press
- [11] P.T Tam, Communication Cost Optimization and Analysis in Distributed Virtual Environment", Technical Report Rm1026-TR98-0412, Department of Computer Science & Engineering, The Chinese University of Hong Kong
- [12] 신동훈 "온라인 게임 서버 프로그래밍" 대림
- [13] 남재욱 "온라인 게임 서버 프로그래밍" 한빛미디어