

클라우드 컴퓨팅 네트워크에서 Seamless 가상 환경 시스템 구축을 위한 마이그레이션 에이전트 (Migration Agent for Seamless Virtual Environment System in Cloud Computing Network)

원동현*, 안동언**

(Dong Hyun Won, Dong Un An)

요약

가상 환경 시스템의 대표적인 응용인 MMORPG에서는 아주 많은 사용자들이 더욱 현실적인 환경에서 게임하기를 원하고 있다. 하지만 많은 사용자에게 가상 환경을 서비스하면서 응답 지연이 없는 Seamless한 환경을 제공하는 것은 매우 어렵다. 하나의 서버가 아닌 여러 대의 서버로 실제 환경을 구성해야 하고 사용자가 특정 지역으로 이동할 때 실제 서버에서 데이터를 공유해야 하는 상황이 발생하기 때문이다. 이때 사용자는 서버 간 정보 동기화 과정에서 응답 지연을 경험하게 되고 이러한 상황은 가상 환경의 현실감을 떨어뜨린다. 이러한 문제를 해결하기 위해서는 서버 간 정보 동기화 과정에서 발생하는 응답 지연을 최소화해야 한다. 본 논문에서는 가상 환경에 대한 정보를 제공하는 필드 서버 간의 효율적인 정보 동기화 및 필드 서버와 PC(Player Character) 간의 응답 지연 최소화를 위한 마이그레이션 에이전트를 제안하고 클라우드 컴퓨팅 네트워크에서 구현하였다. 제안한 시스템은 PC의 수가 70,000 ~ 90,000인 구간에서 기존 시스템보다 필드 서버의 CPU 이용률이 6 ~ 13% 증가하였고 응답 시간은 5 ~ 10초 감소하였다.

■ **중심어** : 마이그레이션 에이전트 ; Seamless ; 클라우드 컴퓨팅 네트워크 ; PC(Player Character)

Abstract

In a MMORPG, a typical application of virtual environment systems, it is a common desire to play in a more realistic environment. However, it is very difficult to provide a latency-free virtual environment to a large user base, mainly due to the fact that the real environment must be configured on multiple servers rather than on single server and that data must be shared on the real server when users move from one region to another. Experiencing response delays continuously in the process of information synchronization between servers greatly deteriorates the degree of immersion. In order to solve this problem, it is necessary to minimize the response delay occurring in the information synchronization process between the servers. In this paper, we propose Migration Agent for efficient information synchronization between field servers providing information of virtual environment and minimizing response delay between Field Server and PC(Player Character) and implement it in cloud computing network. In the proposed system, CPU utilization of field server increased by 6 ~ 13%, and response time decreased by 5 ~ 10 seconds over the existing system in 70,000 ~ 90,000 PCs.

■ **keywords** : Migration Agent ; Seamless ; Cloud Computing Network ; PC(Player Character)

I. 서론

가상 환경은 개념적으로 가상 현실(Virtual Reality) [1]의 하위 개념이다. 1989년 미국의 Jason Lancer는 가상 현실이란 단어를 처음 사용하였다[2, 3].

가상 현실이 가장 활발하게 적용되는 분야는 게임 분야로 리니지[4], WOW(World of War craft)[5] 등과 같은

MMORPG(Massive Multiplayer Online Role Playing Game)[6, 7] 분야가 대표적이며 퀘이크[8]는 1인칭 슈팅 게임이지만 가상 환경 분야에서 높은 몰입도를 보이는 분야이다. 여러 분야의 가상 환경에서 현실 세계를 그대로 경험하고자 하는 사용자들의 요구를 충족시키기 위해 넓은 지형과 더 많은 PC(Player Character)를 지원하고 있다[9, 10, 11].

이러한 가상 환경에서 PC는 서버들이 관리하는 지형 정보나

* 정회원, 전북대학교 IT정보공학과

** 정회원, 전북대학교 IT정보공학과

접수일자 : 2019년 02월 25일

게재확정일 : 2019년 09월 23일

수정일자 : 1차 2019년 06월 10일, 2차 2019년 08월 20일, 3차 2019년 09월 19일

교신저자 : 안동언, e-mail : duan@jbnu.ac.kr

PC의 위치 정보를 어떤 서버가 제공하는지 인식하지 않고 가상 환경을 하나의 게임 월드로 생각하고 게임을 할 수 있어야 한다. 하지만 이러한 과정 중에 Seamless[12] 환경을 담당하는 각각의 서버들은 PC들의 정보를 공유해야 한다. 또한, PC가 이웃한 다른 지역으로 이동하는 경우 각 지역을 관리하는 서버는 PC의 정보를 이전하는 복잡한 과정이 필요하다. 특히 서로 다른 서버 내에 존재하는 PC들 사이에서 이벤트가 발생하면 더욱 복잡한 데이터 공유 과정이 필요하다.

본 논문에서는 Seamless한 가상 환경을 제공하기 위한 마이그레이션 에이전트(Migration Agent)를 제안한다. 본 논문에서는 가상 현실 서비스를 위해서 Seamless 서버 구조를 클라우드(Cloud) 컴퓨팅 네트워크 기반에서 구축하였다. 구현된 가상 환경에서 서버들이 PC들의 정보를 공유할 때 발생할 수 있는 지연을 최소화할 수 있도록 서버 간 정보 동기화 과정 중 응답 시간을 단축하기 위한 캐시 서버 역할을 하는 에이전트 시스템을 제안한다. 본 논문에서는 이 서버를 마이그레이션 에이전트라 칭한다.

II. 관련 연구

1. Seamless 가상 환경 시스템

Seamless 분산 환경 시스템은 그림 1과 같이 크게 클라이언트, 서버로 이루어져 있으며 다음과 같은 역할을 한다[13, 14, 15].

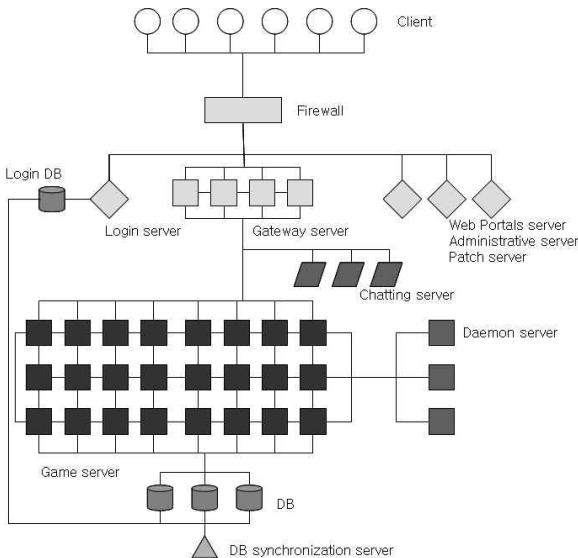


그림 1. Seamless 분산 게임 서버 구조

- 클라이언트: PC가 사용하는 프로그램이다. 필드 서버(Field Server)와 접속한 후 PC의 메시지를 필드 서버에 전달하고 이 메시지의 처리 결과를 받아 화면에 현재 게임 플레이 상

황을 표시한다. 클라이언트는 지속적으로 필드 서버와 통신을 한다.

- 서버: 클라이언트와 통신을 하는 서버군을 지칭한다.
- 로그인 서버: PC 접속에 대한 인증 및 계정 관리를 수행한다. PC가 최초로 게임 서버에 접속할 때 로그인 서버에 있는 데이터베이스에서 PC의 정보를 찾아서 요청한 계정의 유효성을 판단한다.
- 필드 서버: 게임 서버라고도 하며 가상 세계의 한 부분을 담당하면서 해당 영역에서 발생하는 PC의 행동을 처리하고 메시지를 전송하여 클라이언트와 통신을 수행한다. 필드 서버는 수많은 객체로 구성되며 각 객체의 상호 작용이 발생하는 곳이다. 게임의 객체들은 이 필드 서버 객체 안에서만 생성될 수 있으며, 필드 서버는 이러한 객체들의 규칙을 정의하고 관리하는 책임을 진다.

2. 클라우드 기반 가상 환경

서버 가상화는 크게 전가상화와 반가상화로 나뉜다[16].

APP	APP	APP	APP	APP	APP	APP	APP
GuestOS		GuestOS		GuestOS		GuestOS	
H/W				H/W			
H/W							
<전가상화>							
APP	APP	APP	APP	APP	APP	APP	APP
HostOS		GuestOS		GuestOS		HostOS	
Hypervisor							
H/W							
<반가상화>							

그림 2. 서버 가상화 방식

그림 2는 서버 가상화 방식을 보여준다. 전가상화는 하드웨어를 완전히 가상화하는 방식으로 GuestOS의 수정 없이 여러 OS를 이용할 수 있다 하지만 CPU가 가상화 기술을 지원해야 하며 지원하지 않는 경우 성능 저하가 발생한다. VMware의 ESX, Microsoft Hyper-V 등이 이에 포함되며 Intel-VT 또는 AMD-V의 기능이 필수적으로 요구된다.

반가상화는 GuestOS의 커널이 수정되어 가상 하드웨어에 접근하거나 물리적인 하드웨어에 접근할 수 있도록 하는 방식으로 성능 향상이 쉽고 가상 하드웨어와 실제 하드웨어와의 대응이 단축되는 장점이 있으나 GuestOS의 커널을 수정해야 하는 단점이 있다. 하지만 일반적으로 사용하는 리눅스 및 윈도우즈는 반가상화를 지원하는 버전이나 드라이버를 제공한다. 대표적으로 Xen과 KVM이 있다[17].

그림 3은 일반적인 클라우드 기반의 가상 환경 구축을 보여준다. 클라우드 가상 환경에서는 서비스에 필요한 만큼의 서버를 이용할 수 있다[18]. 클라우드 기반에서는 PC가 늘어날 때마다 필요한 만큼의 서버를 사용하는 것이 가능하여 비용 절감 및 서비스 확장의 장점이 있다.

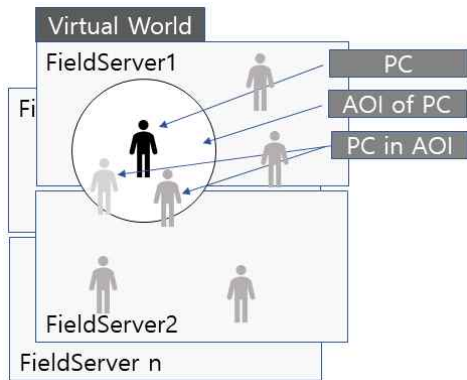


그림 3. 클라우드 기반 가상 환경

3. 마이그레이션 서버(Migration Server)

Seamless 가상 환경에서 필드 서버 간 PC의 이동을 전담하여 관리하는 서버이다. 필드 서버들은 지역 정보 및 PC의 이벤트 정보들을 처리하고 이웃하고 있는 필드 서버에 전달하는 임무를 수행하게 된다. 이때 PC의 필드 서버 간 이동 시 발생하는 응답 지연으로 인해 별도의 PC의 필드 서버 간 이전 작업을 위한 서버를 두는데 이러한 서버를 마이그레이션 서버라 한다 [19].

이 서버는 인접 영역에 있는 PC들의 메시지를 처리하고 PC들이 이웃하는 필드 서버로 Seamless하게 이동할 수 있도록 필드 서버와 PC 간 데이터 전송 및 이동을 처리하고 데이터베이스를 동기화한다. 마이그레이션 서버는 네트워크 지연 현상을 줄이기 위해 필드 서버와 이웃하는 필드 서버 내 PC 정보를 관리하며 각 필드 서버의 데이터베이스에 있는 정보를 업데이트한다.

III. 클라우드 컴퓨팅 네트워크에서의 마이그레이션 에이전트 제안

1. 기존 시스템

마이그레이션 서버는 플레이어 PC의 AOI(Area of Interest)를 공유하는 PC들의 정보 동기화를 위한 정보를 관리한다. PC의 정보가 필드 서버에서 필요하다면 마이그레이션 서버는 PC의 정보를 제공한다. 그림 4는 마이그레이션 서버가 필드 서버 사이의 정보를 동기화하는 과정을 보여준다. 클라우드 컴퓨팅

네트워크 환경에서는 시스템의 과부하가 발생하면 추가로 필드 서버를 추가한다. 이때 새로 추가된 필드 서버와 기존 필드 서버 간에 정보가 동기화되지 않으면, 새로 추가된 필드 서버에 있는 PC들은 기존 필드 서버 내 PC와 함께 플레이할 수 없게 된다.

그러므로 필드 서버들은 PC의 정보에 따라 동기화가 필요한 필드 서버들을 찾아 동기화할 필요가 있다. 이 과정은 일반적으로 데이터베이스 접속 등의 추가적인 작업이 필요하므로 PC에 대한 응답 지연이 발생한다. 마이그레이션 서버는 필드 서버들의 동기화 과정을 필드 서버 대신 수행하며 응답 지연 시간을 감소시킨다. 마이그레이션 서버는 AOI가 중복되는 모든 PC를 관리한다. 정보 공유가 필요한 해당 필드 서버 간 PC의 정보를 마이그레이션 서버가 필드 서버와 함께 관리하여 정보 동기화 과정에서 발생하는 응답 지연을 줄이게 된다.

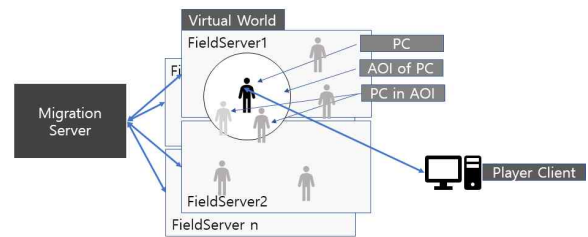


그림 4. 마이그레이션 서버와 필드 서버 동기화

2. 제안하는 시스템

본 논문에서는 기존 마이그레이션 서버의 기능을 축소 한 마이그레이션 에이전트를 제안한다. 마이그레이션 에이전트는 AOI를 공유하는 다른 필드 서버에 있는 PC 간의 위치 정보만을 동기화한다.

기존 시스템은 PC 간 정보 교환이 필요한 이벤트가 발생하지 않아도 PC의 전체 정보를 중복으로 관리해야 하는 부담이 있다. 마이그레이션 에이전트는 PC의 위치 정보만을 동기화하고 있다가 실제 PC 간 이벤트가 발생하면 필드 서버 간 동기화를 진행한다. 가상 환경 시스템 접속자는 자신이 어떤 필드 서버에 있는지 상관없이 마이그레이션 에이전트를 통해서 PC의 정보를 전달받는다.

그림 5는 마이그레이션 에이전트와 필드 서버, 플레이어 클라이언트(Player Client)의 관계를 보여준다. 플레이어 클라이언트는 마이그레이션 에이전트를 통해 현재 플레이하는 PC와 인접한 PC들의 정보를 필드 서버 간의 정보 동기화 과정 없이 전달받을 수 있다. 마이그레이션 에이전트는 PC의 위치 정보만 관리하여 가상 환경 시스템 접속자(Player Client)에 정보를 전송한다. 이웃하는 필드 서버의 PC 간 상호 이벤트가 발생하면 필드 서버는 PC 정보를 동기화하고, 마이그레이션 에이전트는 가상 환경 시스템 접속자에게 결과를 전송한다.

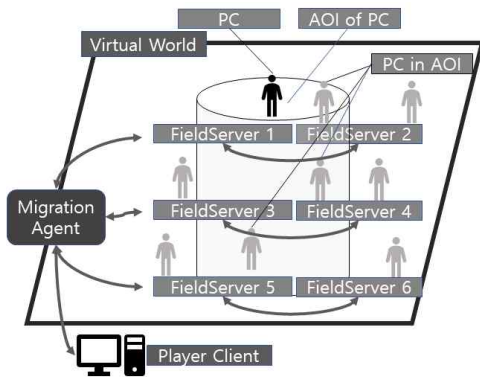


그림 5. 클라우드 컴퓨팅 네트워크에서 Seamless 필드 서버 및 마이그레이션 에이전트 구성

3. 시스템의 비교

표 1은 마이그레이션 서버와 마이그레이션 에이전트에서의 마이그레이션 방식을 비교한 것이다. 동기화 대상은 두 방식 모두 AOI 내 PC로 같다. 하지만 다음과 같은 차이점이 있다.

표 1. 마이그레이션 방식 비교

	기존 시스템	제안 시스템
동기화 대상	AOI 내 모든 PC	AOI 내 모든 PC
동기화 정보	PC의 전체 정보	PC의 위치 정보
동기화 수행	마이그레이션 서버	필드 서버
시스템 접속자의 가상 환경 접근	필드 서버	마이그레이션 에이전트

- 동기화 정보: 기존 시스템에서는 필드 서버와 동일한 정보를 관리하여 필드 서버의 동기화 과정에서 중복된 정보가 발생한다[20]. 제안하는 시스템에서는 PC의 위치 정보만을 관리하고 필요할 때 PC 정보를 동기화한다.
- 동기화 수행: 기존 시스템에서는 PC가 인접한 필드 서버의 정보가 필요 없을 때까지 마이그레이션 서버가 필드 서버의 정보를 동기화한다. 제안하는 시스템에서는 동기화가 필요한 PC들의 정보를 마이그레이션 에이전트로부터 전달받아 필드 서버가 동기화함으로써 중복 정보를 최소화한다.
- 시스템 접속자의 가상 환경 접근: 기존 시스템에서 가상 환경 시스템 접속자는 로그인 서버를 통해 접속한 뒤, 자신이

접속한 가상 환경을 담당하는 필드 서버로부터 정보를 전달 받는다. 제안하는 시스템에서는 시스템 접속자 PC의 AOI를 공유하는, 주변 PC들을 관리하는 필드 서버가 수시로 변할 수 있으므로, 시스템 접속자는 마이그레이션 에이전트를 통해 가상 환경에 접근하도록 하였다.

IV. 실험 및 평가

1. 실험 환경

실험을 위한 PC들은 맵 전체에 고르게 분포되어 있다고 가정하고 초기화를 위해 다음과 같은 변수를 적용하였다.

- 접속 시간 : PC가 접속 후 2시간 후에 플레이를 종료하는 것으로 적용하였다. MMORPG의 이용분석에 근거하면 일반적으로 평균 2시간의 접속 시간을 갖는다[21].
- 맵 크기: 한 셀(Cell) 당 50cm로 가로 10Km, 세로 10Km의 가상 맵을 구성하였다.
- AOI: 상하좌우 각 70 셀의 크기를 적용하였다. 실제 음성이 0.1초 내 전달 가능한 거리인 34.4m를 기준으로 하였다.
- PC: 10,000, 30,000, 50,000, 70,000, 90,000
- 서버 환경: 실험을 위해 클라우드 환경을 구축하고 총 4대의 가상 서버를 생성하였다. 2대의 Intel Xeon E3-1280V 32GB RAM으로 구성된 시스템에 Ubuntu 16.04 LTS를 설치하고 클라우드 환경을 위한 가상 서버로 KVM을 적용하였다. 서버별로 3대의 필드 서버를 구축하고 KVM을 통해 CPU core 2개와 RAM 4GB를 할당했다. 전체 구성된 6대의 가상 서버 중 필드 서버는 4대를 설정했으며 마이그레이션 에이전트 및 마이그레이션 서버 1대, 그리고 PC 접속을 위한 트래픽 발생 서버 1대로 구축하였다.

제안한 마이그레이션 에이전트의 성능을 실험하기 위해 마이그레이션 서버와 마이그레이션 에이전트가 주고받는 메시지는 표 2와 3과 같이 동일하게 구성하였으며 각 이벤트는 무작위로 발생시켰다.

표 2. 필드 서버와 PC 사이의 메시지 (Server->Client)

메시지 (Server-> Client)	내용
C_NAME	접속한 클라이언트 계정 정보
C_MOVE	필드 서버내 PC 이동 정보
C_MESSAGE	서버와 클라이언트간 메시지

표 3. 필드 서버와 PC 사이의 메시지 (Client->Server)

메시지(Client->Server)	내용
S_MOVE	PC의 이동 요청 메시지
S_VELOCITY_CHANGE	PC 이동 속도 처리 메시지
S_STOP	PC의 정지 요청 메시지
S_MESSAGE	메시지 요청
S_REQUEST_MIGRATION	필드 서버 간 PC 이동에 대한 요청
S_RESPONSE_MIGRATION	필드 서버 간 PC 이동에 대한 요청 응답
S_REQUEST_FIELD	인접한 필드 서버 요청
S_RESPONSE_FIELD	인접한 필드 서버의 응답
S_REGIST_MIGRATION	마이그레이션 서버에 등록 요청
S_REPLY_REGIST_MIGRATION	마이그레이션 서버 등록 요청 응답
S_UPDATE_STATUS	PC 변경 정보의 필드 서버 정보 변경

2. 평가

표 4는 기존 시스템과 마이그레이션 에이전트의 응답 시간 차이를 보여준다. PC의 수가 50,000이 될 때까지는 기존 시스템과 제안 시스템의 응답 시간이 모두 약 0.3초 이내로 성능에 미치는 영향은 거의 없었으나 PC의 수가 70,000이 되었을 때부터 제안 시스템의 응답 시간이 개선되는 것을 확인할 수 있었다. 특히, PC의 수가 90,000일 때 제안 시스템과 기존 시스템의 차이가 매우 증가함을 볼 수 있었다. PC의 수가 증가한다고 해도 실제 동기화가 필요한 PC의 수가 많지 않았기 때문으로 보인다.

표 4. 마이그레이션 방식에 따른 응답 시간(초)

PC	마이그레이션 서버	마이그레이션 에이전트
10,000	0.281	0.252
30,000	0.287	0.298
50,000	0.343	0.353
70,000	5.55	0.439
90,000	14.5	2.816

표 5. 마이그레이션 방식에 따른 필드 서버 CPU 이용률

PC	기존 방식	제안 방식
10,000	56%	52%
30,000	66%	64%
50,000	71%	66%
70,000	53%	66%
90,000	56%	62%

표 5는 마이그레이션 방식에 따른 필드 서버의 CPU 평균 이용률을 비교한 것이다. 기존 시스템에서는 PC의 수가 10,000에서 50,000이 될 때까지 증가할수록 평균 CPU 이용률이 증가하다가 그 이후부터는 다시 감소하였다. 필드 서버의 프로세스 사

용 정보를 분석해보면 데이터베이스의 응답 대기 시간이 증가하고 네트워크 연결이 지연되어 CPU 이용률이 낮아짐을 확인할 수 있었다.

제안한 시스템에서는 PC의 수가 증가할수록 뚜렷하게 응답 시간이 감소하였다. 특히 PC의 수가 70,000 ~ 90,000인 구간에서 기존 시스템보다 필드 서버의 CPU 이용률이 6 ~ 13% 증가하였고 응답 시간은 5 ~ 10초 감소하였다. 제안 시스템은 많은 사용자가 함께 플레이하지만, PC 동기화가 많이 필요하지 않은 가상 환경에서 더욱 좋은 성능을 보일 것임을 확인할 수 있었다.

V. 결론

본 논문에서는 클라우드 컴퓨팅 네트워크의 Seamless 가상 환경에서 PC에 대한 응답 시간을 단축하고 필드 서버 간의 효율적인 운용을 위해 마이그레이션 에이전트를 제안하고 테스트 하였다. 마이그레이션 에이전트를 사용함으로써 CPU 이용률을 높일 수 있었고 그 결과 응답 지연 시간을 줄일 수 있었다.

앞으로는 필드 서버 간의 동기화에 관한 연구와 필드 서버 간의 실질적인 데이터양을 줄임으로써 마이그레이션 에이전트의 효율을 높이기 위한 연구와 함께, 필드 서버 부하 정도에 따른 필드 서버 증가 또는 감소가 가상 세계에 어떤 영향을 주는지를 연구하고자 한다.

REFERENCES

- [1] Virtual reality(2019). https://en.wikipedia.org/wiki/Virtual_reality (accessed Jan., 3, 2019).
- [2] R.A. Earnshaw, *Virtual Reality Systems, Academic Press*, pp. 3-10, 2014.
- [3] K.B. Park and J.Y. Lee, "Comparative Study on the Interface and Interaction for Manipulating 3D Virtual Objects in a Virtual Reality Environment," *Transactions of the Society of CAD/CAM Engineers*, vol. 21, no. 1, pp. 20-30, 2016.
- [4] Lineage(2019). <https://lineage.plaync.com> (accessed Jan., 5, 2019).
- [5] WOW(2019). <https://worldofwarcraft.com> (accessed Jan., 3, 2019).
- [6] H. Cole and M.D. Griffiths, "Social Interactions in Massively Multiplayer Online Role-Playing Gamers," *CyberPsychology & Behavior*, pp. 575-583, 2007.
- [7] ZhangHui, "Head-mounted display-based intuitive virtual reality training system for the mining industry," *International Journal of Mining*

- Science and Technology*, vol. 27, pp. 717-722, 2017.
- [8] Quake(2019). <https://quake.bethesda.net> (accessed Jan., 5, 2019).
- [9] T. Ni, H. Zhang, C. Yu, D. Zhao, and S. Liu, "An International Journal for All Aspects of Design," *Computers & Electrical Engineering*, vol. 39, no. 7, pp. 2112 - 2123, 2013.
- [10] G.H. Choi and M.J. Kim, "Analysis of Players' Eye-Movement Patterns by Playing Experience in FPS Game," *Smart Media Journal*, vol. 5, no. 2, pp. 33-41, 2016.
- [11] H.L. Kim and H.I. Kim, "An Analysis of Game Scenario Variations in Literature Using Actant Model - Focused on "Hamlet" and "Mabinogi"," *Smart Media Journal*, vol. 6, no. 3, pp. 64-67, 2017.
- [12] Seamless(2019). <http://research.microsoft.com/en-us/um/people/sdrucker/papers/intermedial.pdf> (accessed Jan., 5, 2019).
- [13] H. Cole and M.D. Griffiths, "Social Interactions in Massively Multiplayer Online Role-Playing Gamers," *CyberPsychology & Behavior*, pp. 575-583, 2007.
- [14] 이정진 외 3명 "MMORPG 부하 분산을 위한 동적 맵 분할 시스템 설계", *정보과학회 2005년 춘계 학술대회 논문집*, vol. 32, pp. 802-804, 제주도, 대한민국, 2005년 2월
- [15] 이남재, 곽훈성 "진화하는 온라인 롤플레이팅 게임을 위한 분산형 게임 서버 모델," *한국게임학회 논문집*, 제2권, 제1호, 36-41쪽, 2002년
- [16] K.T. Seo, H.S. Hwang, I.Y. Moon, O.Y. Kwon, and B.J. Kim, "Performance Comparison Analysis of Linux Container and Virtual Machine for Building Cloud," *Advanced Science and Technology Letters*, vol. 66, pp. 105-111, 2014.
- [17] E. Dhib, N. Zangar, N. Tabbane, and K. Boussetta, "Resources allocation trade-off between cost and delay over a distributed Cloud infrastructure," *Sciences of Electronics Technologies of Information and Telecommunications (SETIT) 2016 7th International Conference on*, pp. 486-490, Hammamet, Tunisia, 2016. 6.
- [18] K.S. Kim, J.H. Jang, and J.M. Hong, "Loan/Redemption Scheme for I/O performance improvement of Virtual Machine Scheduler," *Smart Media Journal*, vol. 5, no. 4, pp. 18-25, 2016
- [19] 문성원, "분산 seamless 게임 서버에서의 효율적인 게임 월드 관리 기술", *서강대 정보통신 대학원 석사학위 논문*, 2004. 2
- [20] H. Cole and M.D. Griffiths, "Social Interactions in Massively Multiplayer Online Role-Playing Gamers," *CyberPsychology & Behavior*, vol. 10, no. 4, pp. 575-583, Aug. 2007.
- [21] W.C. Kuo, S.T. Wang, and J.C. Yang, "An Empirical Analysis of the Playing Time by Different Genders and Ages in an MMORPG," *Digital Game and Intelligent Toy Enhanced Learning (DIGITEL)*, pp. 27-30, Takamatsu, Japan, 2012. 3.

 저자 소개



원동현(정회원)

2003년 전북대학교 컴퓨터학과 (공학사)

2006년 전북대학교 컴퓨터공학과 (공학석사)

2017년 전북대학교 컴퓨터공학과 (공학박사)

<주관심분야 : 분산시스템, 빅데이터, 블록체인>



안동언(정회원)

1981년 한양대학교 전자공학과(공학사)

1987년 KAIST 전산학과(공학석사)

1995년 KAIST 전산학과(공학박사)

1995년~현재 전북대학교 IT정보 공학과 교수

<주관심분야 : 정보검색, 한국어정보처리, 분산시스템>