

실시간 게임정보 확인 서비스를 위한 SIP 기반의 게임 아키텍처의 설계

노강래, 김준일, 이종열
세종대학교 컴퓨터공학과

Design of Game Architecture for Report Service of the Real Time Game Information based on SIP

Noh Kang-rae, Kim jun-il, Lee Jong-youl
Sejong University

E-mail : krnoh@gce.sejong.ac.kr, junil@gce.sejong.ac.kr, leemaster@gce.sejong.ac.kr

요약

SIP는 새로운 서비스로의 확장에 유연하다는 장점을 가지고 있다. 본 논문에서는 SIP 프로토콜을 게임서버에 접목하여, 이를 통해 얻을 수 있는 부가서비스를 제안한다. 게임서버의 사용자 인증부분을 SIP Registrar 서버로 확장하여 해결하고, SIP Personal Mobility Service를 이용하여 게임에 접속되지 않은 사용자에게 게임 정보에 대한 피드백(feedback)이 올 수 있도록 설계하였다. 즉, 사용자가 이동 가능한 위치를 SIP Registrar 서버에 등록하면, 게임서버는 게임의 진행정보를 오프라인 상태의 사용자에게 전달한다. 이 부가 서비스는 단순히 게임엔진 뿐만 아니라 증권, 뉴스속보 서비스 등 실시간 정보를 요구하는 서비스에 접목하기에 충분한 장점을 가지고 있다. 본 논문에서는 SIP 프로토콜을 게임서버에 적용하기 위한 방안으로 인터넷상의 SIP 사용자에게 실시간 온라인 게임 정보 전송 서비스 아키텍처를 제안한다.

1. 서론

현대 경제에는 유형의 생산 산업보다도, 무형의 문화 미디어 산업이 더 많은 비중을 차지하고 있다. 영화, 만화, 게임 등이 이에 해당하며, 특히 첨단 기술을 사용한 미래의 게임 산업은 그 파급 효과와 경제적 가치가 매우 크다.

네트워크의 발달에 따라 게임산업의 가장 중요한 요소인 네트워크/온라인 게임 기술에 관한 연구가 현재 활발히 진행 중에 있다. 네트워크/온라인 게임서버의 모델로는 서버가 게임전체를 관장하는 중앙집중형 모델과, 각 클라이언트가 서버와 클라이언트의 역할을 모두 수행하는 peer-to-peer 모델, 그리고 각 서버별로 관리하는 지역을 나누는 방식인 지역분할 방식이 있다.

본 고에서는 SIP 프로토콜의 특징인 Personal

Mobility를 게임 시스템에 응용함으로써 SIP 프로토콜을 통해 게임의 세션을 열어주고, 연결된 UDP 채널을 통해 게임 데이터를 실시간으로 전송하는 방식의 게임 아키텍처를 제안하였다.[5,4]

본 논문의 구성은 다음과 같다. 2장의 관련연구에서 SIP 프로토콜과 게임서버기술에 대해 소개한다. 3장의 본론에서는 기존 온라인 게임 시나리오와 SIP를 적용한 게임 시나리오 및 이를 응용한 부가 서비스 아키텍처를 제안한다. 4장에서는 향후 과제 및 결론을 기술한다.

2. 관련연구

1. Session Initiation Protocol

SIP는 SMTP(Simple Mail Transfer Protocol), HTTP(HyperText Transfer Protocol)와 매우 유

사한 구조의 텍스트 기반의 응용계층 제어 프로토콜로서, 하나 이상의 참가자들이 함께 세션을 만들고, 수정하고 종료할 수 있게 한다. SIP를 이용한 서비스를 제공하기 위해서는 UAC(User Agent Client), UAS(User Agent Server), Redirect Server, Proxy Server, Registrar, Gateway 와 같은 SIP 구성요소가 필요하다.[1,2,3]

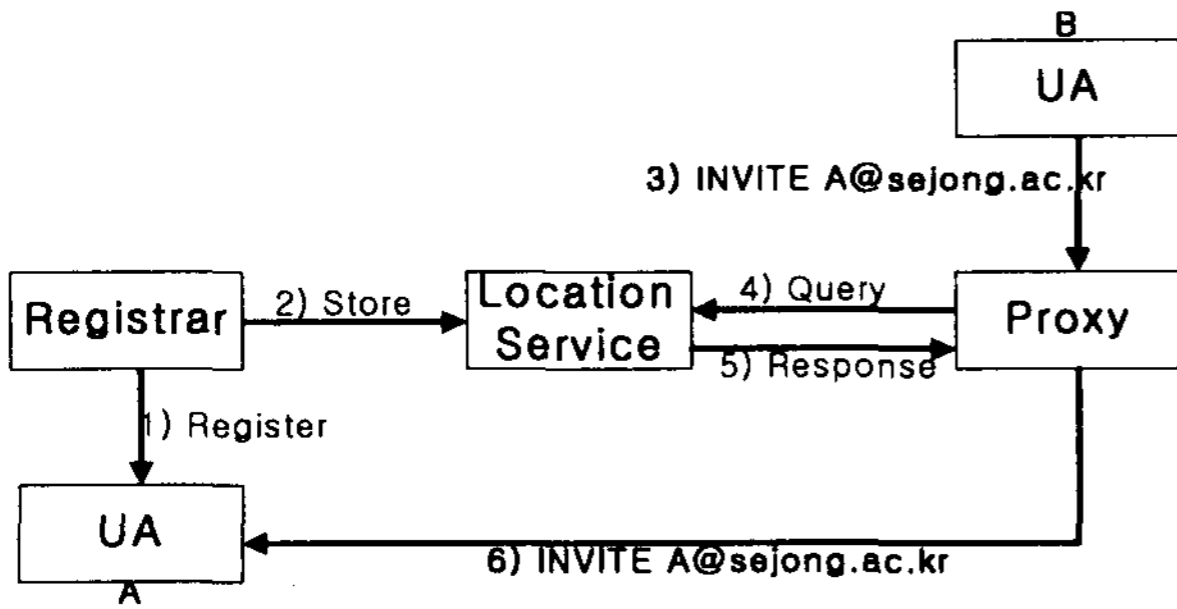


그림 1 Location Service

SIP 프로토콜의 Registrar Server는 이동성을 지원하기 위해 사용자의 위치를 등록하는 서버이다. [그림1]에서 사용자 A가 Registrar Server에 위치를 등록하게 되면, 사용자 B는 Location Service를 이용하여 사용자 A와 직접 연결될 수 있다.[1]

2. 네트워크 게임 기술

네트워크 게임은 크게 중앙 집중형 방식과 peer-to-peer 방식으로 나눌 수 있다. 본 절에서는 네트워크 게임의 방식에 대해서 알아보고, 온라인 게임 기술에는 어떤 방식이 있는지 알아본다.

2.1. 중앙 집중형 방식

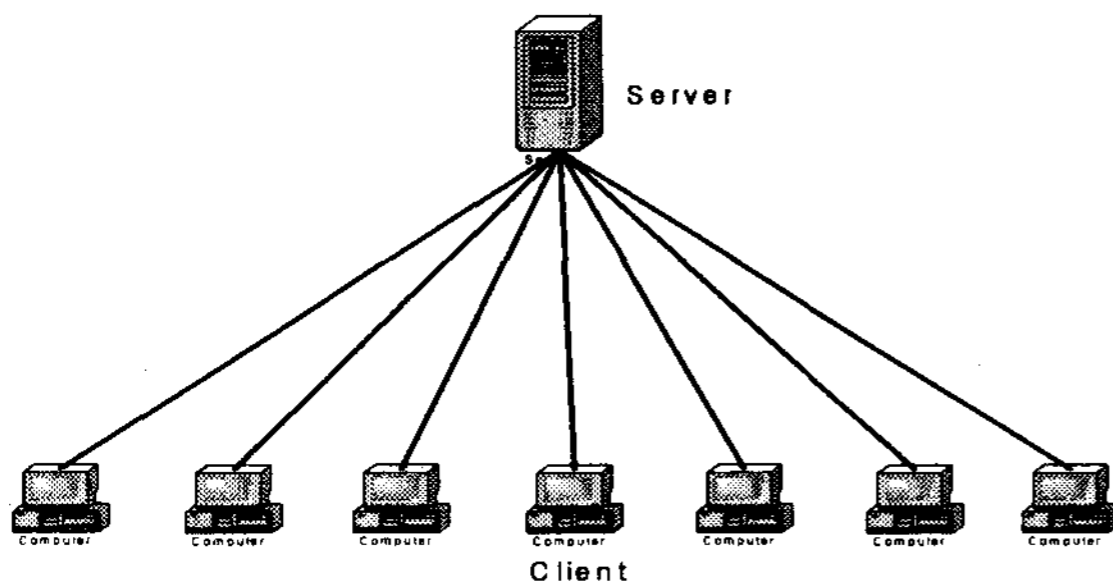


그림 2 Server-Client Model

중앙 집중형 방식은 [그림2]에서 보는 바와 같이 게임 서버가 모든 일을 처리하는 경우에 많이 사용되는 모델이다. 클라이언트는 게임 전체를 관장하는 것이 아니

라 서버로부터 현재의 상황을 받아 출력하는 역할, 또는 클라이언트가 하려는 일을 서버에 전달해 주는 역할을 하게 된다. 그러므로 [그림2]와 같은 방식의 클라이언트는 서버에서 받은 정보를 눈에 보이는 형태로 바꿔 사용자에게 보여 주는 터미널 역할을 하게 된다.[5]

2.2. peer-to-peer 방식

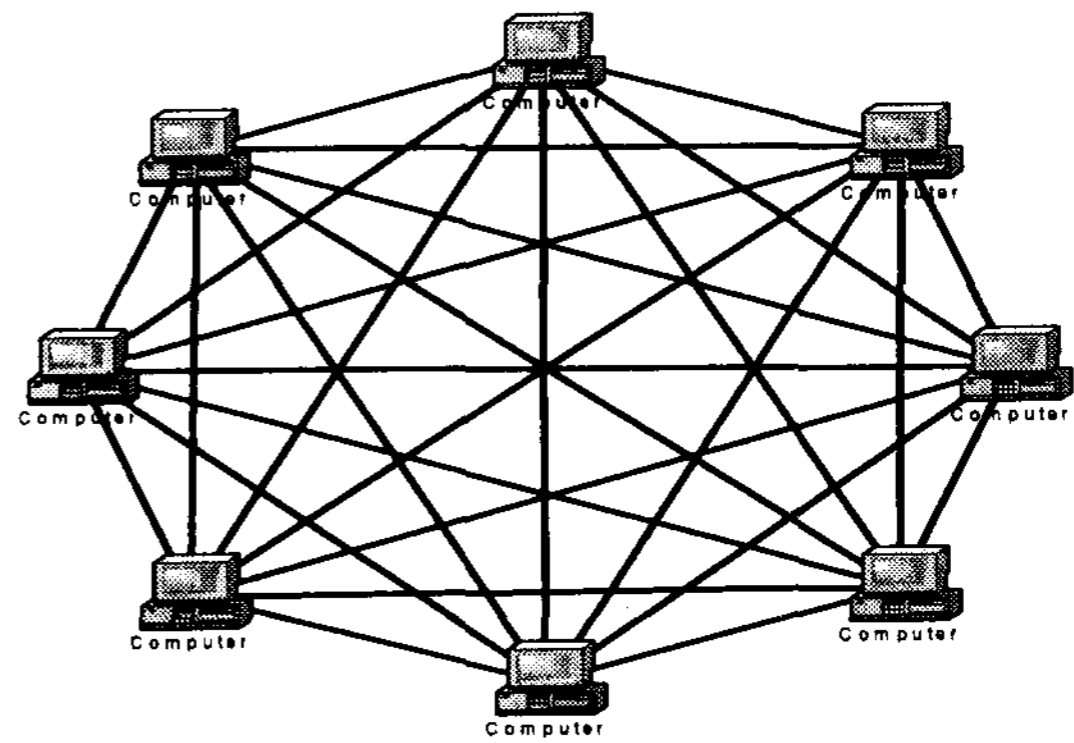


그림 3 peer-to-peer Model

[그림3]의 peer-to-peer 방식은 서버의 역할을 줄여 서버의 부담을 줄이는 방식이다. peer-to-peer 방식에서의 서버는 일종의 중계자 역할을 하게 된다. 게임을 원하는 클라이언트들은 우선은 게임서버에 접속하고, 게임이 시작되게 되면 게임을 시작한 클라이언트들은 서버와 독립하여 서로 연결을 하게된다. 서버를 배제한 채 동일한 레벨에서 서로 연결되어 있는 상태가 되어 패킷을 주고받으며 게임이 실행된다. 게임이 끝난 후에는 다시 서버로 연결되어 승패를 기록하고 다른 게임을 시작할 준비를 하게된다.[4,5]

2.3. 온라인 게임기술

온라인 게임의 서버는 사용자의 과도한 접속으로 인한 장애를 막기 위해 지역 분할 방식의 서버기술을 사용하고 있다. 지역 분할 방식에는 명시적(Explicit Region Partition)방식, 묵시적(Implicit Region Partition)방식과 지역 분할 동기화 방식(Region Partition Synchronization)이 있다.[4]

■ 명시적 지역 분할 방식: 가상공간을 지역적으로 분

할하여, 각 서버가 관리권을 맡는 방식이다. 기획 관점에서 가상 공간을 분할하는 경우이며, 때문에 서버간 동기화가 거의 없어져 개발이나 운영에 장점이 있다. 하지만 기획에 의존적이므로 확장성의 한계가 있다. 일반적으로 존(Zone)개념을 두어 엄격하게 공간을 분할하거나, 울티마 온라인(Ultima Onlin)처럼 비교적 덜 엄격하게 공간을 분할하는 방법이 있다.[5,6]

■ 묵시적 지역 분할 방식: 은 서버가 가상 공간을 분할하여 관리하기는 하지만, 관리영역에 따라 가상 공간을 정적으로 분할하지는 않은 경우이다. 관리영역의 경계를 넘나들거나, 경계 부근에서 머무르는 경우, 지역에 대한 관리권 이전 문제가 서버간에 빈번하게 발생 될 수 있다.[4,5,6]

■ 지역 분할 동기화 방식: 가상 공간을 지역적으로 분할하여, 각 서버가 관리권을 맡는 방식이다. 기존 지역 분할 방식과는 달리, 지역의 분할이 명시적으로 나누어지지 않는 경우이다. 때문에 서버간 동기화가 많이 요구된다. 서버가 지역적으로 분산되어 있는 네트워크 분산 환경들에서 볼 수 있다.[4,5,6]

3. SIP 기반의 게임서버의 구성

1. 개요

네트워크/온라인 게임서버 기술은 게임과 관련된 모든 정보를 가지고 있는 게임서버를 어떻게 구성하느냐에 따라 네트워크 퍼포먼스가 달라진다. 그러므로, 게임서버는 서버의 사용목적과 방식에 따라 상이하게 구성하게 된다.[5]

게임서버를 기능적인 측면에서 분석해보면 게임의 세션을 만들어주는 부분과 게임의 정보를 전송해 주는 부분으로 나눌 수 있다. 게임 자체의 정보를 교환하는 방식은 각 게임마다 상이하지만, 게임의 세션을 만드는 방식은 어떤 네트워크/온라인게임서버에도 모두 적용할 수 있다. 기존 게임서버는 세션을 관리하는 서버, 사용자의 정보를 관리하는 서버, 클라이언트의 버전을 체크하는 서버, 그리고 게임의 정보를 담고 있는 서버로 구

성된다. 하지만, SIP 인프라가 구성되어있고, 사용자가 SIP 클라이언트를 사용하고 있다면, SIP의 Registrar 서버를 이용하여 사용자 데이터베이스를 관리할 수 있다. 기존 Registrar 서버는 사용자의 위치정보만을 가지고 있지만, Registrar 서버를 확장하면 등록된 사용자의 개인정보를 저장할 수 있다. 확장된 Registrar 서버를 이용하여 사용자의 로그인 정보를 가져온다. 게임 세션서버는 게임세션에 참여하려는 사용자가 Registrar 서버에 등록된 사용자와 일치하는지 만 확인하면 된다. 또한, 게임 서버에 가상의 SIP UA를 적용하여 현재 게임 사용자의 위치를 추적할 수 있다. 게임서버의 UA는 게임의 진행정보를 가지고 있고, 각 사용자 아이템이 네트워크 게임 내부에서 일어나는 일에 대해 반응할 수 있다. 가령, 사용자는 온라인 게임의 모든 정보를 알기 위해서는 24시간 게임서버에 접속해 있어야 한다.[7] 또한 사용자가 게임에 접속하지 않은 상황이라면 다른 사용자에게 의해서 자신의 아이템이나 목숨을 빼앗길 수도 있다. 하지만, 게임 서버의 가상의 UA와 SIP 클라이언트 사이에 SIP 세션이 성립되어 있다면, 게임서버는 게임 서버의 가상의 UA를 통해 현재 게임 세션이 오프라인 된 SIP 사용자에게 침입 상황을 알려 줄 수 있다. 이 후, 사용자는 게임서버에 접속하여 자신을 방어할 수 있는 기회가 주어진다.

2. 기존 온라인 게임서버 로그인 시나리오

현재 네트워크/온라인 게임서버에 접속하기 위해서는 아래의 과정을 거친다.[7]

- ①클라이언트가 실행되어 login을 하려면 제일 먼저 세션 서버에 현재 클라이언트의 게임버전을 보낸다.
- ②세션서버는 버전을 체크하고 버전이 동일하면 login의 다음 과정으로 진행하고, 틀릴 경우에는 업그레이드 서버와 통신하여 새로운 버전의 클라이언트를 클라이언트로 전송한다.
- ③세션서버에서 버전 확인 단계를 통과하면 사용자 데이터 베이스는 클라이언트에서 받은 사용자 계정 정보를 분석하여 인증과정을 진행한다. 유료 게임일 경우 접속을 시도하는 사용자의 과금 여부를 확인 하여, 접속자의 접속 여부를 결정한다.

④사용자인증이 완료된 후 게임서버가 여러 개일 경우 사용자는 접속할 서버를 결정하고, 그렇지 않을 경우 다음 단계로 넘어간다.

⑤해당 서버에서 클라이언트가 접속을 허용하면, 곧 게임이 진행된다.

3. SIP를 이용한 온라인 게임서버 아키텍처

기존 SIP망 위에 있는 사용자는 사용자의 UA가 구동됨과 동시에 Registrar서버에 자신의 위치를 등록한다. 등록된 사용자의 정보는 게임 세션을 관리하는 게임 세션 서버와 통신을 통하여 접속하려는 사용자가 어떤 사용자인지에 관한 정보를 교환하고, 게임 세션서버는 교환된 정보를 바탕으로 유·무료 서비스와 성인제한 등 제반사항을 결정한다. [그림4]는 SIP를 이용한 온라인 게임 서버 시스템의 구성을 도식화하였다.

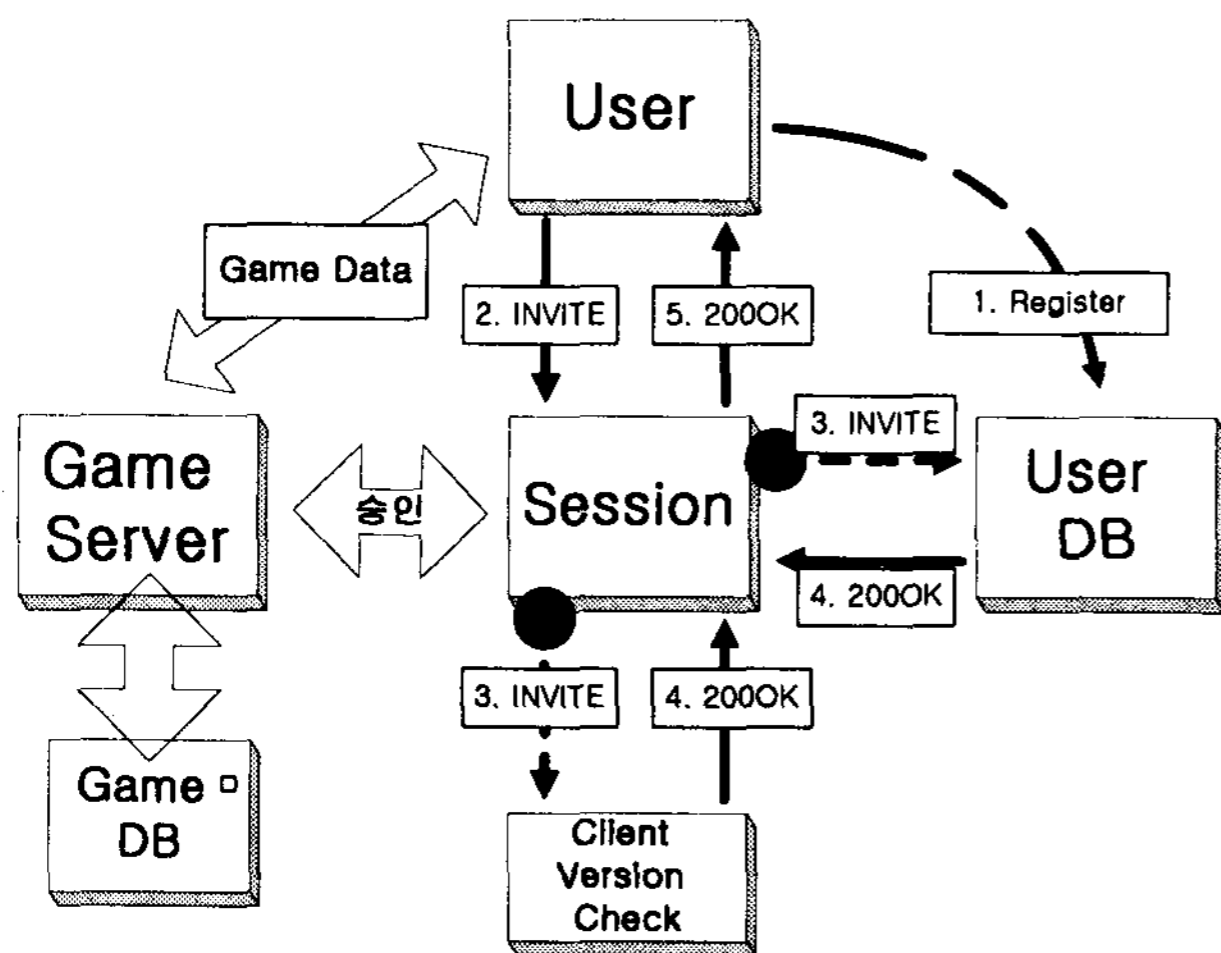


그림 4 SIP를 이용한 온라인 게임 서버

3.1 SIP를 적용한 게임 세션으로의 로그인

SIP를 게임서버에 응용하려면, 사용자와 게임서버는 SIP 인프라에 존재하여야 한다. SIP 프로토콜은 게임 세션 성립을 위한 부분만을 책임지고, 게임진행과 관련된 부분은 신경 쓰지 않는다. 또한, 확장된 SIP Registrar 서버는 사용자의 인증을 위해서 사용자의 위치정보 이외에 신상정보를 담고 있다. 게임자체의 부하를 줄이는 방법은 게임 서버 영자가 결정한다. 로그인 시나리오는 다음과 같다.

①사용자가 UA를 구동시킨다.

②UA가 구동되는 순간 사용자의 위치정보와 사용자 개인정보가 Registrar(User DB)서버에 등록된다.

③사용자가 게임세션에 참여하기 위해 Session 서버에 INVITE 메시지를 보낸다.

④세션서버는 사용자로부터 받은 INVITE 메시지를 Registrar 서버와 Client Version Check 서버에 보낸다.

⑤ Registrar 서버는 Register된 사용자와 게임세션에 참가하려는 사용자가 같은 사용자인지를 확인한다.

⑥Client Version Check 서버는 접속하려는 사용자의 클라이언트의 버전을 확인하여, 게임 버전과 일치하는지 확인한다. 일치하지 않으면, 세션서버에 버전이 일치하지 않는다는 메시지를 보낸 후 사용자를 새로운 버전으로 업그레이드 해준다.

⑦사용자의 모든 인증과정이 끝난 후, 게임 서버에 승인을 해줌으로서 사용자와 게임서버간의 UDP채널이 열리게 된다.

⑧UDP 채널을 통해 게임데이터를 전송하고 게임을 시작한다.

3.2 실시간 게임 정보 확인 서비스

네트워크/온라인 게임서버가 가상의 UA를 적용하게 된다면 SIP프로토콜의 Personal Mobility Service를 이용할 수 있다. 네트워크/온라인 머그게임인 리니지를 예로 Personal Mobility Service의 아키텍처를 설명한다.[7] 온라인 머그게임인 리니지는 전통적인 그래픽 머그 게임으로, 여러 명의 사용자가 수시로 접속하여 가상의 세계에서 탐험 및 전쟁을 진행하는 게임이다. 게임은 서로의 영역 및 아이템을 전쟁을 통해 쟁취하는 방식으로 진행된다. 게임의 특성상 게임서버에 접속해 있지 않은 사용자는 다른 사용자의 공격이나 침입 정보를 알 수가 없다. 하지만, 게임서버에 SIP UA가 탑재되

어 있다면 게임서버가 전송해주는 게임정보를 실시간으로 전송 받아 침입자에 대한 대처를 할 수 있다. 서비스 시나리오는 다음과 같다.

②사용자는 리니지 게임에서, 성주라는 계급을 가지고 있다.

③사용자는 Registrar Server에 자신의 위치를 등록한다.

④사용자가 온라인 되어 있지 않은 상황에서 적군이 공격해 들어오고 있다고 가정,

⑤게임엔진은 SIP 프로토콜을 이용하여, 사용자의 위치를 검색한다.

⑥검색된 사용자의 위치에 적군의 공격상황을 실시간으로 보내준다.

⑦사용자는 SIP 클라이언트를 통해 공격정보를 받은 후, 대처 방안을 세운다.

[그림5]에서는 사용자가 게임에 로그인 되어있지 않을 때, SIP 프로토콜을 통해서 게임의 상황을 전송하는 메커니즘을 나타낸다. [그림5]는 게임서버가 게임상황의 변화를 감지하여 게임서버에 있는 UA에게 정보를 보내면, 게임상황 정보를 받은 UA는 SIP 아키텍처를 통해 사용자가 미리 저장해놓은 위치에 정보를 보낸다.

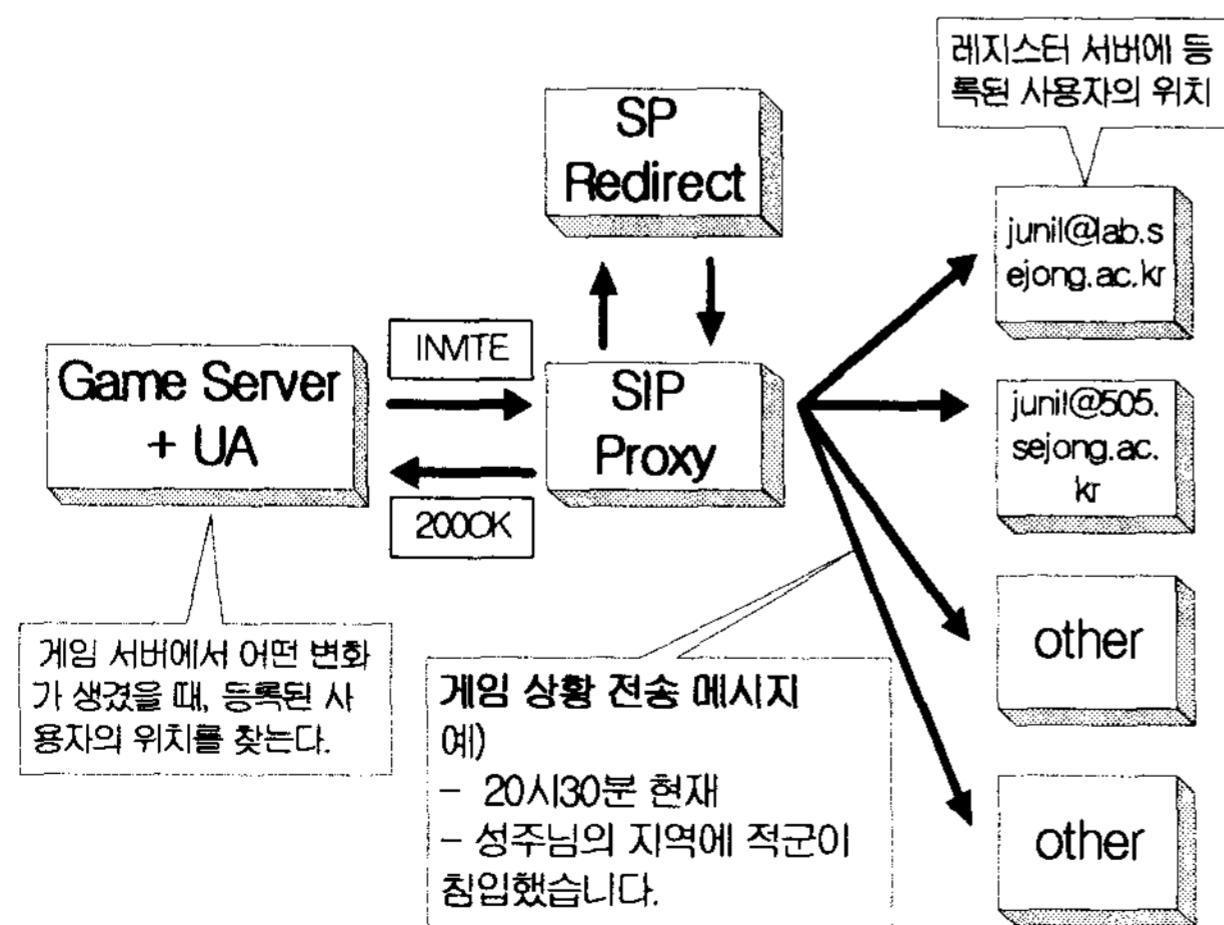


그림 5 Game Server+UA를 이용한 부가가치서비스

3.2 서비스의 장점

기존 리니지 게임과 같은 머그 게임에서는 사용자의 개인 사정으로 인한 무관심을 허용하지 않는다. 사용자가 온라인 되어있지 않다면, 그 사용자는 다른 팀의 공격을 막아낼 방법이 없다. 하루 24시간을 PC방에 앉아서 자신의 성을 지키려는 성주와 그 성주를 보호하기 위해 밤을 새우는 중독자들은 자신의 생활을 할 여유가 없다.[7] 하지만, SIP의 Personal Mobility Service를 이용한다면 게임을 위해 하루 24시간을 소비하지 않아도 될 것이다. 또한 개인적인 사정으로 자리를 비웠다 하더라도, 미리 등록해놓은 사용자의 위치를 찾아 게임상황을 전송해 줌으로 준비되지 않은 상황에 대처할 수 있는 기회를 제공한다. 또한 게임을 못하게 되는 시간동안 공급할 수 있는 게임 진행정보를 SIP 프로토콜의 Personal Mobility Service를 통해 전달받을 수 있다. 사용자는 이 서비스를 통해 게임에 접속하지 않은 상황에도 게임에 참가할 수 있다는 장점을 갖는다.

4. 결론

본 논문에서는 SIP 프로토콜을 이용한 부가적인 게임 서비스를 제안하였다. 그 동안의 네트워크/온라인 게임은 TCP나 UDP 또는 다른 운영체제에서 기본으로 제공하는 프로토콜 위에 직접 필요한 프로토콜을 구축해야 했다. 그러나, SIP망이 구축된 네트워크에 UA를 탑재한 게임서버와 게임 클라이언트를 가지고 있다면, 사용자의 인증부분을 해결할 수 있고, 사용자에게 게임정보를 실시간으로 알려주는 서비스와 같은 부가서비스로의 확장이 가능하다. 그 예로, SIP 프로토콜을 접목한 게임에서 Personal Mobility Service를 아이템 형태, 또는 유료서비스 형태로 제공하게 된다면 새로운 부가가치를 창출할 수 있는 방안이 될 것이다. 또한, 본 서비스를 증권정보, 뉴스속보와 같이 실시간 정보 이동이 필요한 분야에 접목한다면 새로운 부가가치를 창출할 수 있다. 네트워크/온라인 게임서버에 있어서 게임세션을 이루는 부분은 극히 일부분에 불과하다. 세션을 이룬 후에도 게임서버를 어떻게 구성하느냐에 따라 서버의 성능의 차이가 크게 틀

리게 된다. 본 고에서는 게임의 성격에 따라 서버 구성이 상이하게 설계되기 때문에 서버의 구성에 관한 부분은 언급하지 않았다. 또한, 하드웨어적 병목현상이나 회선이상과 같은 서버기술 이외의 문제점에 관한 언급은 하지 않았다. 향후, 서버구성간의 퍼포먼스나 하드웨어적 퍼포먼스 측정이 이루어져야 한다.

[참고문헌]

- [1] M.Handley, H. Schulzrinne, "SIP: session initiation protocol", Mar,1999.
- [2] Alan B.Johnston "SIP: Understanding the Session Initiation Protocol"
- [3] M.Handley, V.Jacobson "SDP: Session Description Protocol" April,1998
- [4] 장세윤, "네트워크게임의 현장", 프로그램세계 특집기사 2000.12
- [5] 배현직, "스타트라인에 서려면 이런 준비부터", 프로그램세계 특집기사 2002.03
- [6] 신동원 "미래 첨단 게임의 흐름 - Network Game"
- [7] <http://www.lineage.co.kr/> - 리니지 공식 홈페이지