

지속적인 싱글 골프게임 플레이를 위한 네트워크 시스템 설계

임승현[○], 박기현^{*}

^{○*}계명대학교 컴퓨터 공학과

e-mail: {burningwing, khp}@kmu.ac.kr[○]

Design of an Network System for the Continuous Single Golf Game Play

Seunghyeon Lim[○], KeeHyun Park^{*}

^{○*}Dept. of Computer Engineering, Keimyung university

● 요약 ●

본 논문에서는 지속적인 싱글 골프게임 플레이를 위한 네트워크 시스템 설계를 제안한다. 본 설계는 스크린 골프 시뮬레이터와 자료동기화 기술을 접합함으로써, 언제 어디서나 이전에 정지된 게임 플레이 환경을 재현하여 계속적인 플레이 할 수 있도록 하는데 그 목적이 있다. 플레이어가 스크린 골프장에서 게임 플레이하다가 게임이 완전히 끝나기 전에 정지하게 되면, 본 논문에서 제안한 네트워크 시스템이 해당 게임 환경 정보를 서버에 저장한다. 차후 플레이어가 게임 재개를 원할 경우, 종전에 정지한 게임 환경 정보를 그대로 재현할 수 있도록 다운로드하여 사용자에게 제공함으로써, 지속적인 게임 플레이를 가능하게 한다. 본 논문의 시스템은 PC콘솔 클라이언트, 프록시 서버, 사용자 데이터 서버로 구성된다. PC콘솔 클라이언트는 콘솔 기기에 탑재되며, 사용자에게 스크린 골프게임을 제공하고, 게임 환경 정보를 프록시 서버로 전송한다. 프록시 서버는 닫힌 네트워크에 존재하는 사용자 데이터 서버와 PC콘솔 클라이언트를 연결하며, 해당 세션을 관리한다. 사용자 데이터 서버는 실제 플레이어의 인증, 데이터 저장 및 불러오기와 같은 실질적인 서비스를 제공한다.

키워드: 지속적인 골프게임(Continuous Golf Game), 자료동기화(Data Synchronization), 게임 재개(Game Resumption)

I. 서론

현재 우리나라는 고도의 산업 발달과 네트워크의 발달을 통해, 국민들의 일상생활이 윤택하게 되었음을 부인할 수 없을 것이다. 하지만 고도의 산업 발달과 노동구조의 변화로 인해 국민들의 여가 시간과, 육체적 활동의 감소를 초래하였고, 이에 따라 여가 생활의 욕구가 높아지고 있다. 또한 웰빙 문화의 확산에 따라 간편하고 효과적인 스포츠가 각광 받고 있다.

스크린 골프는 골프 연습장을 3D데이터로 모델링하고 시스템이나 장치를 이용하여 플레이어가 실내에서 골프를 치면서 실제 골프장에서 골프를 즐기는 것과 같은 재미를 느낄 수 있는 골프 시뮬레이터로써, 현대 직장인, 주부, 학생과 같은 많은 연령, 직업군에서 선호하고 있는 가상현실 스포츠이다.[1]

본 논문에서는, 플레이어가 언제, 어느 스크린 골프장에서나, 플레이어가 정지된 게임의 재개를 희망할 경우, 네트워크를 통하여 저장된 게임 환경 정보를 다운로드해 줌으로써, 정지된 게임 환경과 동일한 환경을 구축할 수 있도록 하는 네트워크 시스템을 설계하였다. 플레이어가 스크린 골프장에서 게임 플레이중 게임이 완전히 끝나기 전에 정지하게 되면, 본 논문에서 제안한 네트워크 시

스템이 해당 게임 환경 정보를 서버에 저장한다. 차후 플레이어가 게임 재개를 원할 경우, 종전에 정지한 게임 환경을 그대로 재현할 수 있도록 다운로드하여 사용자에게 제공함으로써, 지속적인 게임 플레이를 가능하게 한다.

본 논문은 다음과 같이 구성된다. 2장에서는 관련 연구에 대해서 설명하며, 3장에서는 설계된 네트워크 시스템을 제안한다. 마지막으로, 결론에서는 실제 시스템의 구축 방안과 향후 과제에 대하여 논의한다.

II. 관련 연구

1. 스크린 골프

일반적으로 골프 시뮬레이터라고도 하며, 대한민국 업체들은 이미 세계시장을 석권하여 점유율 1위를 선정하고 있다. 스크린 골프는 실제 골프장에 방문하지 아니하고, 현장을 재구현 할 수 있는 콘솔기기, 사용자에게 체험감을 줄 수 있는 대형 스크린, 플레이어와 골프 클럽과 골프공의 움직임을 감지할 수 있는 센서로 이루어진 가상현실(Virtual reality) 기반의 운동의 일종이다.[2,3] 스크

린 골프 연습장은 날씨와 계절, 공간간의 영향을 받지 않고 골프 연습을 수행할 수 있는 시설을 의미하며, 현대의 직장인, 주부, 학생 등 남녀노소의 인기를 얻고 있는 스포츠의 한 종류로 자리매김 하고있다.[4]

2. OMA DS

OMA의 자료동기화(DS)는 모바일 환경에서 XML 형식의 메시지를 통하여 데이터 동기화를 수행하는 국제 표준안이다.[5-7] 본 논문에서 수행하고자 하는 실질적인 원리는 PC콘솔 클라이언트에서 수행되는 게임의 환경 데이터가 사용자 데이터 서버와의 동기화를 통해서, 서버는 항상 최신의 데이터를 유지하고, 이를 언제 어디서나 불러 올 수 있게 하는데 있으며, 이는 모바일 기기의 자료동기화와 큰 유사성을 지니고 있다. 또한 스크린 골프는 기존의 게임들과 달리 데이터의 전달이 빈번하지 않고 PC를 콘솔 기기로 사용하기 때문에, 모바일 기기와는 달리, 강력한 네트워크 성능과 컴퓨터 파워를 지니고 있으며, OMA DS와 같은 XML기반의 프로토콜을 사용하는 것이 차후, 프로토콜 확장과 버전 업데이트, 트래픽 관리에 유리하리라 판단된다.[8-10]

III. 네트워크 시스템 설계

본 논문에서 제안한 지속적인 싱글 골프 게임 플레이를 위한 네트워크 시스템은 <그림 1>과 같이 PC콘솔, 프록시 서버, 사용자 데이터 서버, XML 기반의 프로토콜로 구성된다.

PC콘솔은 사용자와 근접한 위치에서 플레이어의 게임 플레이 및 네트워크를 통한 데이터 통신의 클라이언트로써 동작한다. 프록시 서버는 사용자 데이터서버와 PC콘솔 클라이언트의 사이의 데이터 전달 및 세션의 생성 및 해제를 담당한다. 사용자 데이터 서버는 종단에서 플레이어의 현재 게임 정보를 저장하고, 게임 재개 시, 저장된 게임 환경 정보를 다운로드해주는 역할을 수행한다.

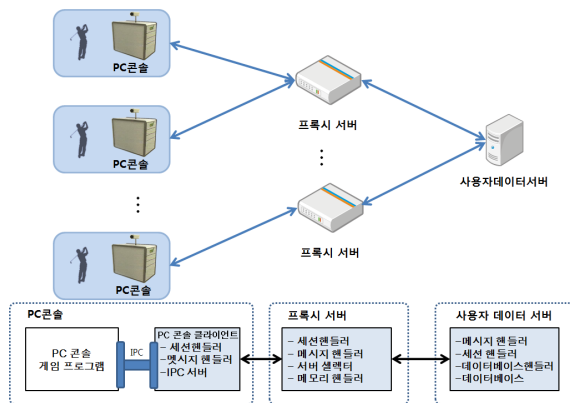


그림 1. 시스템 구조
Fig. 1. System Architecture

1. PC콘솔 클라이언트

<그림 2>는 PC콘솔 구조도를 나타낸다. PC콘솔 클라이언트는 PC콘솔 게임프로그램과 PC콘솔 클라이언트로 구성된다.

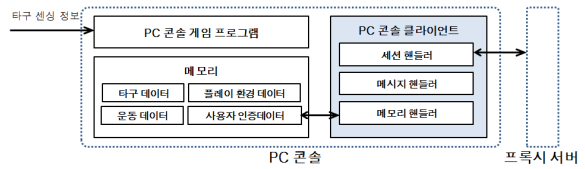


그림 2. PC 콘솔 클라이언트 구조
Fig. 2. PC Console Client Architecture

1.1 PC콘솔 게임 프로그램

PC콘솔 게임프로그램은 일반적인 골프 프로그램으로써, IPC통신을 통하여 PC콘솔에 탑재된 PC콘솔 클라이언트에게 현재 플레이어의 진행 정보를 전달하게 된다.

PC 콘솔 게임 프로그램은 프로그램의 흐름에 따라 <표 1>과 같은 5개 시점에서 데이터를 저장하고, 만약 종전에 저장된 (중지) 게임이 있다면, 로그인 직후에 이어서 게임을 재개할 것인지에 관해서 플레이어에게 입력받게 된다.

표 1. 게임 환경 정보 저장 시점

Table 1. Points to Save the Game Environment Information

저장 시점	저장 정보
새로운 게임 시작	필드, 홀, 플레이어정보
홀 시작	플레이어 순서, 홀 번호
타구 후	플레이어 순서, 공 위치
홀 종료	남은 홀수, 플레이어 정보
게임 종료	플레이어 정보

1.2 PC콘솔 클라이언트

PC콘솔 클라이언트는 PC콘솔 게임 프로그램에서 전달된 사용자 개인정보를 기반으로 공개된 프록시 서버와 세션을 설정하고, PC콘솔 게임 프로그램에서 전송된 플레이어의 진행 정보를 프록시 서버로 전달한다. PC콘솔 클라이언트는 IPC 핸들러, 세션 핸들러, 메시지 핸들러로 구성되며, 각 역할은 다음과 같다.

- 세션 핸들러 : 프록시 서버와의 네트워크 세션 생성 및 해제
- 메시지 핸들러 : XML기반의 저장 및 불러오기 메시지 생성 및 해석
- IPC 핸들러 : PC 콘솔 게임 클라이언트와의 데이터 통신

2. 프록시 서버

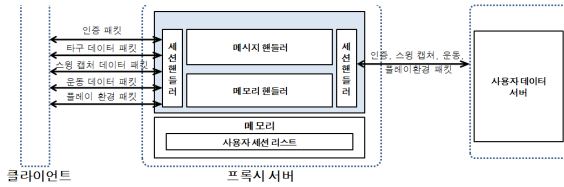


그림 3. 프록시 서버 구조
Fig. 3. Proxy Server Architecture

<그림 3>은 프록시 서버의 구성을 나타낸다. 프록시 서버는 PC 콘솔 클라이언트와 사용자 데이터 서버의 중간에 위치한다. PC 콘솔 클라이언트로부터 수신된 인증 패킷, 타구 데이터 패킷, 플레이 환경 정보 패킷 등을 수신하고, 이를 단힌 네트워크에 존재하는 사용자 데이터 서버로 전송하는 역할을 수행한다.

사용자 인증 패킷이 사용자 데이터 서버 서버로부터 적합한 사용자 인증될 경우, 프록시 서버는 새로운 세션을 생성하게 되며, 이는 게임 플레이가 종료 될 때까지 유지하게 된다.

프록시 서버는 메시지 핸들러, 메모리 핸들러, 세션 핸들러로 구성된다.

- 메시지 핸들러 : XML기반의 패킷의 암호화 및 복호화를 담당한다.
- 메모리 핸들러 : 사용자 데이터 서버로부터 인증된 사용자 및 PC 콘솔 클라이언트의 세션 리스트를 저장한다.
- 세션 핸들러 : 세션 핸들러는 프록시 서버 양 종단에 위치하며, 일반적인 네트워크상의 PC콘솔 클라이언트와 단힌 환경의 사용자 데이터 서버의 세션을 관리한다.

3. 사용자 데이터 서버

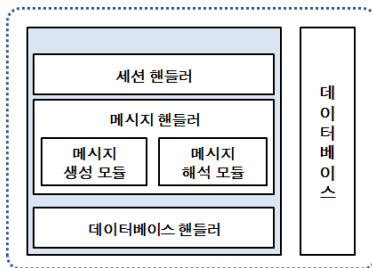


그림 4. 사용자 데이터 서버 구조
Fig. 4. User Data Server Architecture

<그림 4>는 사용자 데이터 서버의 구성을 나타낸다.

사용자 데이터 서버는 PC콘솔 클라이언트의 인증, 데이터 저장, 데이터 불러오기 기능의 기능을 수행하는 프로그램과 단힌 네트워크 환경의 서버 컴퓨터로 구성된다. 모든 PC콘솔 클라이언트 및 서비스를 요청하는 주체는 예외 없이 프록시 서버를 거쳐, 단힌 네트워크상에 존재하는 사용자 데이터 서버로 데이터를 보내게 된다, 이는 사용자 데이터 서버의 독립성을 보장하여 외부로부터의 공격을 방어하는 수단으로 이용된다.

사용자 데이터 서버는 세션 핸들러, 메시지 핸들러, 데이터베이스 핸들러, 데이터베이스로 구성된다.

- 세션 핸들러 : 단힌 네트워크 상에 존재하는 다수의 프록시 서버와의 세션을 관리 한다.
- 메시지 핸들러 : 프록시 서버를 통하여 전달된 PC콘솔 클라이언트의 메시지를 해석하고, 그에 대한 결과 메시지를 생성한다.
- 데이터베이스 핸들러 : PC콘솔 클라이언트로부터 수신한 게임 환경 정보 등을 데이터베이스에 저장하거나, 게임 환경 정보에 대한 불러오기 요청이 있을시, 종전의 게임 환경 정보를 인출한다.
- 데이터베이스 : 사용자 인증 정보, 게임 환경 정보 등을 저장 한다.

4. XML 프로토콜

```
<ELEMENT OneClickGolf (Auth,Body)>
<!--인증정보, 멀티플레이를 위한 Prototype-->
<ELEMENT Auth (ClientVer,SessionID,AuthHash,ProType,InstansKey?)>
<!--바디는 실제 클라이언트 또는 암호화된 문자열로 구성-->
<ELEMENT Body (PCDATA|PlayerInfo|ShotInfo|GameContinueInfo)>
<!--세션 정보-->
<ELEMENT SessionID (PCDATA)>
<!--암호화된 인증 문자열-->
<ELEMENT AuthHash (PCDATA)>
<!--멀티플레이를 위한 프로토타입 타입-->
<ELEMENT ProType (PCDATA)>
<!--클라이언트 버전 정보-->
<ELEMENT ClientVer (PCDATA)>
<!--게임 시작시 필요정보-->
<ELEMENT GameInfo (PlayerList,PlayerSequence,FieldList,StartHoleID,HoleNum,Mode,GameContinueInfo)>
<!--플레이어 정보-->
<ELEMENT PlayerInfo (PlayerName,PlayerNum?,CustomSetting?,Score?,PlayerStatus?)>
<!--플레이어 이름-->
<ELEMENT PlayerName (PCDATA)>
<!--사용자 임의 설정값-->
<ELEMENT CustomSetting (Tee,Diff?,Index?,Enable?,Score?,TShotDistance?,TShotFairWay?,GreenShot?,Put?,InCount?,BunkerShot?,ShotAverage?,ShotMax?,FairWayRate?,GreenShotRate?,SandSaveRate?,StartHole?,Stats?,ForceDrop?,HoleID?,ClubName?,ClubNo?,UseTee?)>
<ELEMENT InstansKey (PCDATA)>
```

그림 5. XML 프로토콜 DTD
Fig. 5. XML Protocol DTD

<그림 5>는 본 시스템에 적용될 XML 프로토콜 DTD의 일부를 나타낸다. 본 시스템은 게임 환경 데이터의 동기화에 의한 원활한 지속적인 싱글 플레이 지원에 있다. 이는 데이터 동기화라는 측면에서 OMA DS의 일반적인 목적에 부합 하지만, OMA DS를 그대로 활용하기에는 SYNCML의 복잡성과 데이터 처리 방식이 본 시스템과 상이하기 때문에 본 시스템에 적합한 XML 프로토콜을 설계하였다.

본 시스템의 XML 프로토콜은 헤더(Header)와 바디(Body)로 구성된다.

- 헤더 : 사용자 인증 데이터와 인증 데이터, 세션키 등의 네트워크 및 인증 데이터를 적재한다.
- 바디 : 프록시 서버에 의해 발급된 세션키로 암호화된 사용자 환경 데이터를 적재한다.

IV. 결론

본 논문에서는 지속적인 싱글 골프 게임 플레이를 지원하기 위한 네트워크 시스템의 설계를 제안하였다. 본 시스템은 OMA DS와 유사하게 현재 게임 상태를 서버와 동기화를 수행함으로써, 물리적 공간의 제약을 받지 않고, 콘솔기기가 존재하는 곳이라면 어디든 종전의 게임 환경을 재구축 하여 게임을 재개하는데 그 목적

이 있다. 또한 프로토콜 확장에 유리한 XML기반의 프로토콜을 사용함으로써, 프로그램의 확장 및 업데이트에 유리하도록 설계하였다. 본 시스템의 PC콘솔, 프록시 서버, 사용자 데이터서버로 구성되며, 프록시 서버를 사용함으로써 비인가 사용자의 불법적인 접근 및 네트워크 스트레스를 최소화 하려고 노력하였다.

현재, 본 연구팀은 본 논문에서 제시한 설계를 기반으로 윈도우 운영체제에서 동작하는 시스템을 구현중이며, 구현이 완료될 경우, 실제 프록시 서버 및 사용자 데이터 서버에 가해지는 스트레스를 테스트 하고, 네트워크 및 XML 기반 프로토콜에 의한 병목 지점을 탐색하고 이를 해결하기 위한 연구를 수행할 예정이다.

V. Acknowledgement

본 논문은 중소기업청에서 지원하는 2012년도 산학협력력 기업 부설연구소 지원사업(No. C0002985)의 연구수행으로 인한 결과물임을 밝힙니다.

참고문헌

- [1] Su-jik Lee, "The effect of motivation of virtual reality sports and service factors on leisure satisfaction," Sejong University, Master's Thesis, 2011.
- [2] Sang-Hyuk Ahn, "Development of a simulator for Interactive 3D Golf Game" Hallym University, Master's Thesis, 2007.
- [3] Byeon-min Moon, "The Study of Screen Golf's fun factors on exercise immersion experience, participating satisfaction and exercise continuation behavior," Silla University, Master's Thesis, 2010.
- [4] Cho Kyu-kwon, Kim Yong-Gun, Heo Chon, "The Relationship between Life-Style and Quality of Lives to Screen Golf Participants", The Korean Journal of Sport, Vol. 9 No. 3, pp. 289-297, 2011.
- [5] OMA Forum, <http://www.openmobilealliance.org>
- [6] Uwe Hansmann, Riku Mettala, Apratim Purakayastha, Peter Thompson, "SyncML Synchronizing and Managing Your Mobile Data", Prentice Hall Pub. Co., pp. 21-24, 2003.
- [7] Ju-Geon Pak, Keehyun Park, "Construction and Validation of a Data Synchronization Server supporting OMA DS Standards," Journal of the Korea society of computer and information, Vol. 16, No. 5, pp. 79-91, May. 2011.
- [8] KeeHyun Park, Ju-Geon Pak, "Efficient Transmission Method for Mobile Data Synchronization Based on Data Characteristics", Lecture Notes in Electrical Engineering, Vol. 120, pp. 253-266, Dec. 2011.
- [9] Extensible Markup Language, <http://www.w3.org/XML/>
- [10] Dong-Guy Kawk, Ho-Byung Park, Chae-Woo Yoo, "A Transformation of XML Documents with Semantic Constraints using XML Schema," The Korean Insitute of Infomation Scientists and Engineers, Vol. 32 No. 2, pp. 592-594, 2005.