

게이머의 성향을 반영한 맵 시뮬레이터 구현에 관한 연구 (한국형 MMORPG게임을 중심으로)

김정현⁰, 김경식

호서대학교 대학원 컴퓨터 공학과 게임전공
megaray@empal.com⁰, kskim@office.hoseo.ac.kr

A Study on the Realization of a Map Simulator Reflecting
Propensities of Gamers (Focused on Korean Style MMORPG)

Jung-Hyun Kim⁰, Kyung-Sik Kim

Dept. of Computer Game Engineering in Graduate School of Hoseo Univ

요약

RPG게임에서의 맵은 단순한 캐릭터들의 이동이나 활동을 위한 무대를 제공하는 것뿐만 아니라 몬스터와 NPC등 게임의 모든 요소가 혼합된 월드로서 작용을 한다. 그러나 맵과 맵에 따른 몬스터의 설계는 게임이 어느 정도 개발된 후에 테스트를 통해서 밸런스를 맞춰가야 하는데 이를 위해서는 개발기간이 추가되는 문제가 발생하게 된다.

본 논문에서는 게임기획초기에 맵과 맵에 따른 몬스터의 배치를 게이머의 성향에 따라 설정함으로써 특정 맵만을 사용함에 따르는 비효율적인 맵 사용을 방지할 수 있는 맵 시뮬레이터를 설계 구현하면서 연구한 결과를 기술한다.

Abstract

The maps in the world of the role playing games provide not only areas for motions and actions of player characters, but also worlds of totally included monsters and NPC(Non Player Character). However, the design of maps and monsters in them could not be tested for their balances until the games had been developed quite a bit. So they need additional expense of extension period of developments.

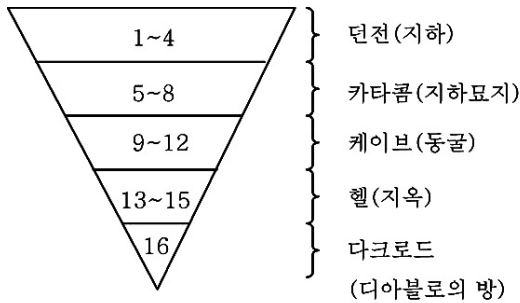
In this paper, we have designed and developed a map simulator to expect the popularities of the maps in the early stage of the game planning in order to prevent inefficient concentration of users on some maps by expecting the popularities of the maps in the early stage of game planning and by setting the parameters of monster arrangements in the map.

Keyword : MMORPG, 맵 시뮬레이터, NPC

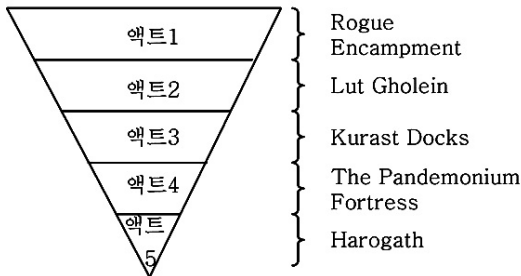
1. 서론

RPG게임에서의 맵의 역할은 단순하게 PC(Player Character, 이하 PC)가 활동하는 무대를 제공하는 것 이외에 몬스터, NPC등의 PC를 제외한 캐릭터들의 무대이기도 하며 기타 다른 게임요소가 혼합된 하나의 월드로서 작용을 하게 된다. RPG게임의 맵은 분할 맵과 단일 맵으로 크게 나뉘는데 이는 보통 서버에게 주어지는 부담과 게이머들에게 주어지는 편리성 사이에서 적절하게 설정되어 만들어진다.

RPG게임에서의 맵의 구조를 보면 기존의 패키지 RPG게임의 경우는 디아블로1의 경우 층으로 구분된 던전(Dungeon)으로서 층을 더할수록 레벨이 높아지는 구성을 채택하고 있으며, 디아블로 2의 경우는 액트(Act)를 중심으로 액트1에서 액트5로 갈수록 레벨이 높아지는 구조를 가지고 있다.



[그림 1] 디아블로 1의 던전 구조[8]



[그림 2] 디아블로 2의 액트(Act)구조[9]

이외의 다른 패키지 RPG게임 역시 비슷한 구조를 가지고 있다. 하지만 이러한 두 경우 모두 맵의 확장성이라던가, 몬스터의 확장성은 확장팩을 제공하기 이전에는 거의 없기 때문에 일정시간의 플레이타임이 지나게 되면 맵의 구조와 맵에 배치된 요소를 모두 파악할 수 있는 장점과 함께 같은

곳을 반복해야 하는 단점을 동시에 가지게 된다.

그러나 패키지RPG게임에서 MMORPG로 변화되면서 가장 큰 변화는 맵이 계속해서 확장되어진다는 점과, 그에 따른 게임요소(몬스터, NPC)가 추가되어진다는 점이다. 또한 MMORPG게임의 경우는 특별한 지역 및 던전을 제외하고는 모두다 캐릭터가 이동할 수 있는 맵으로 이루어져 있기 때문에 각 지역에 맞는 레벨을 가진 사냥터를 제공하는 것이 무엇보다 중요하며 각 사냥터마다의 밸런스를 설정, 유지하는 일 역시 매우 중요한 요소이다[1, 4].

특히 한국형 RPG게임의 경우 서구형 RPG게임과 비교해서 맵과 몬스터의 배치가 더욱 중요한 역할을 담당하고 있으며, 게임의 재미에 직접적인 영향을 미치는 요소로 작용한다. 그러나 게임설계에 있어서 맵과 몬스터의 설계는 일반적으로 게임의 베타테스트 과정에서 직접 플레이 해보면서 각 지역의 몬스터의 배치, 몬스터의 레벨 등을 테스트 해가면서 설정하는 것이 일반적이기 때문에 게임기획 초기단계에서 테스트해보고 설정하는 것에 비해서 게임제작 과정을 거친 후에 테스트를 해야 한다는 문제점이 있었다.

본 논문에서는 게이머들의 성향을 반영한 한국형 RPG게임의 맵 설정 시뮬레이터를 연구하였다. 이를 통해서 기획 초기단계부터 기획서에 따른 맵과 몬스터의 설정을 통해서 게이머들의 게임진행을 미리 살펴보고 게이머들의 성향을 최대한 만족시킬 수 있는 맵과 몬스터의 구성 등을 추출해 낼 수 있다. 빌드오더(Build order)와 같은 공식의 게임플레이를 통한 특정 맵 편중 문제의 해결을 통해서 최대한 많은 맵이 사용될 수 있을 것으로 기대한다.

2. 한국형 MMORPG게임의 특징

한국형 MMORPG게임과 서구형 MMORPG게임의 경우 외관은 크게 차이가 없어 보인다. 특히 최근의 트렌드인 무협이나 현대물을 주제로 한 게임을 제외하고는 많은 게임들이 북유럽의 신화나 중세의 분위기를 차용해서 사용하고 있으며 몬스터 역시 드래곤, 고블린, 오크 등의 이름만 차이가 있을 뿐 비슷한 몬스터를 사용하고 있다. 또한 직업군 역시 전사, 마법사, 도적, 궁사를 기본으로 한 세부직업으로 나뉘고 있다.

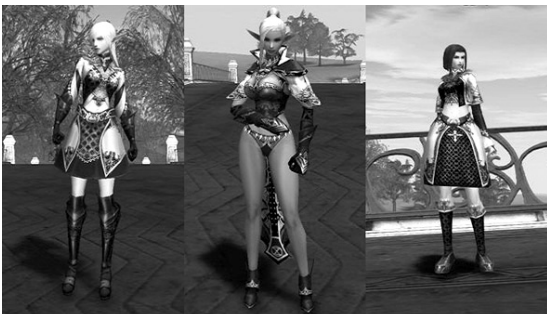
그러나 이러한 비슷한 특징을 갖고 있는 것으로 보이는

두 게임의 본질에는 큰 차이가 있는데 서구형 MMORPG 게임이 PC게임과 콘솔 게임을 중심으로 발달해온 게임으로 미션과 퀘스트가 게임의 근간을 이루고 있으며 이를 설계하는 과정이 매우 중요한 기획자의 역할인 반면에

한국형 MMORPG게임의 경우는 미션과 퀘스트는 그다지 중요한 역할을 담당하지 않으며 오히려 게이머들 간의 관계를 극한으로 이끌어 낼 수 있는 공간을 만드는 데 치중하고 있다[2].

이를 통해 게이머들은 게임을 단순한 게임으로 즐기는 것이 아니라 게임을 하나의 사회로 받아들이며 사회생활을 통한 간접경험에 더 치중한다는 것을 알 수 있다. 일례로 외국에서는 굉장히 많은 매니아층을 거느리며 큰 성공을 거둔 게임인 다크에이지오브카멜롯[10] 이나 에버퀘스트2[11] 등의 게임이 한국에서는 전혀 인기를 얻지 못하고 있는 것이 현실이며, 최근 개발된 길드워[13] 역시 기존의 MMORPG게임이 가지는 문제점을 보완하려 새로운 시스템으로 제작되었지만 해외에서는 크게 호평을 받았으나 정작 한국에서는 큰 인기를 얻지 못하고 있다.

한국형 MMORPG의 특징은 게임을 플레이 하는 게이머들에게서도 그 특징이 나타나는데, 한국의 MMORPG 게이머의 경우는 게임안의 캐릭터(아바타)에 대한 부와 명예에 굉장히 집중한다는 점이 특징이다(그림 3). 특히 캐릭터를 나타낼 수 있는 레벨과 외관상으로 보이는 아이템(갑옷, 무기, 장신구, 기타)에 강한 집착을 보이며 이를 위해서 많은 시간과 노력을 아끼지 않는다[2,3,7].



[그림3] 리니지2 A급 디크크리стал 로브 종족별 착용 이미지[12]

2.1 RPG게임을 플레이 하는 유저들의 성향

RPG게임(특히 한국형 RPG게임)을 플레이 하는 유저들은 가능하면 최대한 빠른 시간에 레벨 업을 통해서 부와 명

예를 얻기를 바란다. 또한 게임을 플레이하는 성향은 기본적으로 3가지로 압축될 수 있는데 보통 골드, 레벨, 전투를 목적으로 플레이를 하게 된다[3,5,6].

2.1.1 골드 추구형

골드의 경우는 일반적인 RPG게임 내에서의 화폐를 말하는 것뿐만 아니라 아이템까지 통합하는 재화의 모든 것을 말한다. 골드에 치중하는 게이머는 보통 다른 캐릭터에 비해서 골드를 더 많이 얻을 수 있는 캐릭터를 채택하거나 일반적인 캐릭터를 가지고도 더 많은 골드를 얻을 수 있는 사냥터를 찾아다니면서 사냥하는 것이 일반적이다.

2.1.2 레벨업 추구형

레벨업 추구형의 경우는 가장 빠른 시간 안에 레벨업을 통한 게임내의 지위와 명예를 얻고자 하는 유형의 플레이를 말한다. 레벨업에 치중하는 게이머의 경우는 가장 빠른 시간안에 레벨업을 할 수 있도록 경험치를 많이 주는 사냥터를 집중적으로 선택하여 플레이를 하며, 파티플레이와 각종 버프(캐릭터에게 도움을 주는 마법)를 통해서 자신의 공격력 및 방어력을 극대화해서 자신의 캐릭터보다 레벨이 높은 캐릭터를 집중 공략해서 보다 많은 경험치를 얻기 위해 노력한다.

2.1.3 전투 추구형

전투를 추구하는 플레이는 크게 PC와 몬스터의 전투를 즐기는 것부터 PC와 PC간의 전투인 PvP 뿐만 아니라 단독 사냥, 파티플레이 사냥, 레이드 사냥 등 다양한 몬스터와의 전투를 추구하는 플레이를 말한다. 이러한 플레이를 추구하는 게이머의 경우는 맵의 정보를 최대한 활용하여 자신과 맞는 레벨의 몬스터가 있는 사냥터를 찾아 나서고, 결투장이나 PK(Player Kill)를 통해서 전투를 즐기기도 한다.

또한 이러한 성향의 게이머들은 레벨업과 아이템에도 관심이 있지만 골드 추구형 게이머나 레벨업 추구형 게이머에 비해 추구하는 정도가 덜한 편이며 골드와 아이템을 전투에 필요한 도구로서 활용하는 경우가 많다. 또한 아이템과 레벨이 부족해도 전투를 하는 등 다양한 전투경험을 체험하는 것이 목표인 형태이다.

3. 맵 설정 시뮬레이터 구현

3.1 맵 설정 시뮬레이터 기본 설정

본 논문에서 제안하는 맵 설정 시뮬레이터는 RPG게임의 기획에 있어서 일정한 개발단계에 이르러 맵과 몬스터를 배치한 후에 테스트를 통한 레벨 조절이 아닌 기획초기 단계에서부터 맵과 몬스터를 설정한 후에 테스트를 통해서 레벨을 조절할 수 있도록 한 것이다. 그 기본 사항은 다음과 같다.

총 맵수	1~12	맵의 수를 규정
맵당 몬스터수	1~12	하나의 맵 당 출몰하는 몬스터 수
맵당 몬스터 드랍 골드	1~100000	몬스터가 드랍 할 수 있는 골드
맵당 몬스터 제공 경험치	1~100000	몬스터가 제공하는 경험치
맵당 몬스터 레벨	1~100	몬스터의 레벨 규정
맵당 몬스터 골드 드랍률	0~100%	몬스터가 드랍 하는 골드에 대한 드랍 확률 규정
레벨업 조건	100~	이전 레벨업 경험치×2

[표 1] 맵 설정 시뮬레이터 기본 설정 사항

일반적인 RPG게임의 경우는 맵의 수가 12개를 넘아가는 것이 일반적이지만 테스트용도로써 사용하기에 12개를 사용하는 것이 크게 문제가 없다고 판단하여 12개로 사용하며, 각 맵 당 출몰하는 몬스터의 수를 12마리로 규정하며 총 12개 맵에서 144마리의 몬스터에 대해서 테스트를 할 수 있도록 규정하였다. 또한 몬스터의 레벨을 1~100레벨까지 사용할 수 있게 함으로써 맵의 수가 12개로 제한되는 문제점을 몬스터의 레벨을 높여가면서 사냥터를 구분할 수 있도록 하였다. 또한 제안하는 맵 시뮬레이터의 경우는 PC와 몬스터간의 레벨이 2이상 차이가 나면 공격을 하지 못하도록 되어 있는데 이는 기본적으로 한국형 MMORPG의 경우 레벨이 2이상 차이가 나게 되면 사냥의 효율이 극도로 떨어지기 때문에 이점을 고려하여 설정하였다. (각종 물약과 버프를 통해서 극복할 수 있지만 이점은 어느 정도 게임의 서비스 기간이 지나야만 가능하기 때문에 기획초기의 테스트과정에서는 제외했다.)

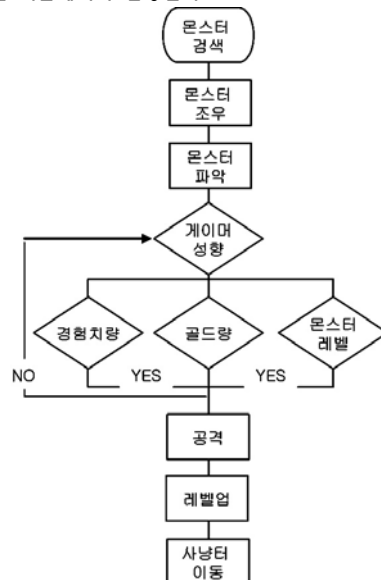
3.2 맵 설정 시뮬레이터의 흐름

맵 설정 시뮬레이터의 흐름은 일반적인 RPG게임에서의 사냥의 순서와 동일한데 다만 다른 점이라면 게이머의 성

향에 따라서 경험치와 골드, 그리고 전투를 위한 몬스터의 레벨을 참고해서 사냥할 몬스터를 선별한다는데 있다. 그리고 사냥과정을 통한 레벨업이 이루어지면 레벨업된 레벨과 캐릭터가 추구하는 성향에 맞는 사냥터를 검색하여 다시 사냥을 반복하게 된다.

스텝 1	캐릭터 설정(그림 5 참조) 게임 캐릭터에 대한 설정을 한다. 골드/경험치/PvP 특성의 합이 100%가 되게 설정한다.
스텝 2	맵의개수 설정(그림 6 참조) 전체 게임 맵의 수를 설정한다. (최대 12개까지 설정가능)
스텝 3	맵 당 몬스터 수 설정(그림 7 참조) 스텝 2에서 설정한 맵의 수에서 하나의 맵에 들어갈 몬스터의 수를 설정한다 (최대 12마리까지 설정가능)
스텝 4	각 맵의 몬스터 설정(그림 8 참조) 스텝 1에서의 캐릭터의 설정처럼 맵을 구성하고 있는 몬스터에 대해서 각 몬스터에 대한 특성치를 설정한다.
스텝 5	시뮬레이션 실행(그림 4 참조) 시뮬레이션의 사냥은 반복적으로 사냥을 하며 그림 5의 캐릭터 설정에 따라서 자신의 성향치를 가장 우선적으로 만족시킬 수 있는 몬스터를 먼저 사냥하며 사냥을 더 이상 할 수 없는 단계(레벨업, 몬스터와 레벨차이가 2이상)에 이르면 다른 맵으로 이동한다.

[표 2] 맵 시뮬레이터 실행절차



[그림 4] 맵 설정 시뮬레이터의 흐름도

```

void CRPGUser::Hunting()
{
while(!bReSet) {
int move_level_up = MOVE_LEVEL_UP;
int move_mon_down = MOVE_MON_DOWN;
//맵 선택
int nSelectMap = 0;
int nHuntingMonSerial = 0;
if((nSelectMap = SelectMap())== -1) {
if(!bReSet) bReSet = TRUE;
return;}
int temp = nSelectMap + 1;
//맵 선택 로그 기록
if(!CirQueuePut(&quUserMapLog, &temp)) return;
//몹이 없거나 맵이 선택되지 않을 때
if(environment.GetAllMonNum() == 0 || nSelectMap == -
1) break;
//몹이 없거나 레벨 올랐을때
while((nHuntingMonSerial =
environment.GetMonSerial(nSelectMap, nLevel)) != -1) {
//3. 사냥을 하되 몹을 다 잡거나 레벨이 올라가면 1,2를 다시 수행
한다.
//1마리 제거하되 렘이되는 몹을 사냥
environment.SetMapMonNum(nSelectMap,
(environment.GetMapMonNum(nSelectMap) - 1));
environment.SetMonDeadLive(nSelectMap,
nHuntingMonSerial, 2);
environment.nAllMonNum--;
//경험치는 100% 획득
nExp += environment.GetMapMonDropExp(nSelectMap,
nHuntingMonSerial);
//경험치 로그
if(!CirQueuePut(&quUserMapExpLog, &nExp)) return;
//4. 아이템은 드랍률에 의거 먹는다.
//아이템 획득
if(GetRand(0, 101) <
environment.GetMapMonDropGoldProb(nSelectMap,
nHuntingMonSerial))
{ nGold +=
environment.GetMapMonDropGold(nSelectMap,
nHuntingMonSerial); }
//레벨업 이동
if(pow(2, nLevel) * LEVEL_START <= nExp) {
nLevel++;
//맵이동 로그
if(!CirQueuePut(&quUserMapMoveLog,
&move_level_up)) return;
//레벨 로그

```

```

if(!CirQueuePut(&quUserMapLevelLog, &nLevel))
return;
//골드 로그
if(!CirQueuePut(&quUserMapGoldLog, &nGold))
return; break; }
//몹이 없을시 로그 기록
if(environment.GetMapMonNum(nSelectMap) == 0) {
//맵이동 로그
if(!CirQueuePut(&quUserMapMoveLog,
&move_mon_down)) return;
//레벨 로그
if(!CirQueuePut(&quUserMapLevelLog, &nLevel))
return;
//골드 로그
if(!CirQueuePut(&quUserMapGoldLog, &nGold)) return;
}
//5. 시간은 무시한다.
//6. 골드로 포션, 아이템 구매는 없는 것으로 본다.
//몹 모두 잡았을 때, 사용자가 설정한 한계치에 달했을 때
}
if(!bReSet) bReSet = TRUE;
}
}

```

[표3] 시뮬레이터 흐름의 Pseudo Code

3.3 맵 설정 시뮬레이터의 구현

맵 설정 시뮬레이터는 표 2의 스텝 1의 실행절차에 따라 서 그림 5와 같이 캐릭터의 이름, 직업, 성향치를 입력한다.



[그림5] 캐릭터 설정

그림 5의 캐릭터는 골드 성향치 90%의 골드 추구형 마법사 캐릭터이다. 맵 시뮬레이터에서 특정 직업이나 이름은 가중치가 들어가는 등의 변화요인이 없기 때문에 직접 입력을 받는다.



[그림 6] 맵의 개수 설정

그림 6은 스텝 2의 과정으로 맵의 수를 설정을 하게 되는데 현재는 5개의 맵을 통해서 캐릭터가 사냥하는 경로를 살펴보고 그림 7, 그림 8과 같이 몬스터의 수와 몬스터의 특성에 관해서 설정한다.



[그림 7] 맵 당 몬스터 수 설정

그림 7은 스텝 3의 과정으로 스텝 2에서 설정한 맵에 몬스터의 수를 설정한다.



[그림 8] 각 맵의 몬스터 설정

그림 8은 스텝 4의 과정으로 그림 7에서 설정한 맵 당 몬스터에 대한 특성을 설정한다. 몬스터에 대한 설정을 자유롭게 설정이 가능하나 시뮬레이터가 가지는 기본 설정값인 레벨 2가 차이난다면 공격을 못하는 룰을 염두에 두고 설정을 해야 한다.

3.4 설정값 적용 및 시뮬레이션 결과

시뮬레이터의 실행은 스텝 5의 과정으로 그림 4의 흐름도에 따라 각 설정값에 의거하여 시뮬레이션을 수행하는 과정이다.

시뮬레이션을 위해 설정한 설정값은 표 3과 같다. 표 2의 실행절차의 스텝 4에서 몬스터의 설정에 이 값을 입력하고 시뮬레이션을 실행하면 그림 9, 그림 10의 결과를 얻게 된다.

	몬스터 수	드랍 골드	드랍 경험치	몬스터 레벨	골드 드랍률
1번 맵	1	50	10	1	100%
	2	60	20	1	100%
	3	70	15	1	100%
	4	60	20	2	80%
	5	70	25	2	80%
2번 맵	1	70	15	1	100%
	2	60	20	1	100%
	3	70	25	2	100%
	4	80	25	2	100%
	5	80	20	2	100%
	6	90	30	3	80%
	7	100	35	3	80%
3번 맵	1	80	25	2	100%
	2	90	25	2	100%
	3	80	20	2	100%
	4	80	20	2	100%
	5	100	35	3	80%
	6	110	35	3	80%
	7	100	30	3	80%
4번 맵	1	90	25	2	100%
	2	80	20	2	100%
	3	100	30	3	100%
	4	90	35	3	100%
	5	110	35	3	100%
	6	130	45	4	80%
	7	140	40	4	80%
	8	120	50	4	80%
5번 맵	1	100	35	3	100%
	2	90	30	3	100%
	3	100	30	3	100%
	4	120	35	3	100%
	5	130	45	4	80%
	6	130	45	4	80%
	7	120	50	4	80%
	8	140	55	4	80%

[표 4] 시뮬레이팅 설정 값

1번맵의 2번과 3번 몬스터의 경우는 같은 레벨1의 몬스터이지만 각각 60과 70의 골드를 드랍하여 3번 몬스터가 더 10의 골드를 더 많이 드랍하는 반면에 경험치는 2번 몬스터가 5를 더 제공하는 차이점을 가진다. 이처럼 각 맵의 몬스

터의 특성을 다르게 설정함으로써 캐릭터가 추구하는 성향치에 따라서 구분하여 사냥할 수 있도록 몬스터의 특성치를 설정한다.



[그림 9] 골드성향 90% 캐릭터의 움직임



[그림 10] 경험치 성향 90% 캐릭터의 움직임

골드성향 90% 캐릭터의 경우 총 35마리의 몹을 잡으며 1-2-1-2-2-3-3-3-4-5번 맵을 움직이며 사냥을 했다. 그 결과 레벨 4 까지 성장시켰으며 각 맵당 250, 360, 430, 740, 800, 1390의 골드를 습득했다. 하지만 경험치 성향 90% 캐릭터의 경우는 골드성향 90%의 캐릭터와 같은 35마리의 몬스터를 사냥했으나 골드의 양에 관계없이 각 맵 당 한번씩의 레벨을 하면서 빠르게 레벨업에 치중했음을 알 수 있다.

따라서 이를 바탕으로 경험치를 중시하는 캐릭터에 비해서 골드를 중시하는 성향의 캐릭터들은 1번, 2번, 3번 맵을 반복적으로 사용하며 골드를 습득하는 것에 더 추구했음을 알 수 있다. 따라서 게임 디자이너가 자신의 의도한 디자인 방향과 시뮬레이터에서 나온 결과를 비교해서 게임이 갖는 특성에 맞도록 수정을 할 수 있다.

4. 결론

한국형 MMORPG의 경우는 맵에 따른 몬스터의 설정이 서구형 MMORPG게임과 비교해서 게임의 재미와 유저들의 게임플레이에 강한 영향을 미치고 있다. 특히 이른바 공식으로 불리는 성향에 따른 맵의 이용은 방대한 맵과 몬스터의 수가 존재하는 MMORPG게임에서는 낭비라고 할 수 있다.

본 논문에서는 맵 시뮬레이터를 통해서 유저들의 성향을 반영한 몬스터의 배치에 관해서 연구하였다. 이를 통해서 기획초기단계에서 유저들의 성향을 반영한 맵과 몬스터 배치를 시뮬레이터 해보고 게임을 설계할 수 있는 장점이 있다.

차후 연구과제로는 본 논문에서 제안한 시뮬레이터의 미약한 부분인 유저들의 성향변화와 다양한 레벨디자인 요소를 감안하여 테스트해볼 수 있는 시뮬레이터의 개발이다.

참고문헌

- [1] Markus Friedl, 엄태선 역, 온라인 게임기획&인터랙티브, 정보문화사, pp62-72, 2003
- [2] 이인화, 한국형 디지털 스토리텔링, 살림출판, pp30-46, 2005
- [3] Neak Hallford, Jana Hallford, 한 쿨 임 역, Sword&Circuitry, 제우미디어, pp21-30, 2002
- [4] 전경란, MMORPG의 장르적 특징 및 함의, 제48회 전국 국어국문학 학술대회 발표논문집, 국어국문학회, 2005.5.29, p.500
- [5] Chris Crawford, 오동일 역, The Art of Computer Game Design, 북스엔피플, pp29-39, 2005
- [6] <http://www.mud.co.uk/richard/hcdis.htm>
- [7] Jessica Mulligan, Bridgtte Patrovsky, Developing Online Games: An Insider's Guide, 송기범역, 제우미디어, pp343-354, 2003
- [8] Diablo1, Blizzard Entertainment, 1996
- [9] Diablo2, Blizzard Entertainment, 2000
- [10] Dark age of Camelot, Mythic Entertainment, 2001
- [11] EverQuest2, Sony Online Entertainment, 2004

[12] Lineage2, NC Soft, 2003

[13] GuildWar, Arenanet, NC Soft, 2005



김정현 (Jung-Hyun Kim)

2001년 호서대학교 게임공학과(학사)
 2003년 호서대학교 게임공학과(석사)
 2006년 호서대학교 컴퓨터공학과 게임전공 박사과정(수료)
 관심분야: 게임 플레이 설계, 레벨 디자인



김경식 (Kyung-Sik Kim)

1982년 서울대학교 전산기공학과(학사)
 1990년 서울대학교 컴퓨터공학과(석,박사)
 1984년~1991년 한국전자통신연구원
 1991년~현재 호서대학교 게임공학전공교수
 관심분야 : 게임교육, 게임프로그래밍
