



※ 본 아티클은 CMP MEDIA LLC와의 라이선스 계약에 의해 국문으로 제공됩니다

## 게임 설계 필수 사항: 20가지 미스터리 게임 (Game Design Essentials: 20 Mysterious Games)

John Harris

가마수트라 등록일(2008. 1. 14)

([http://www.gamasutra.com/view/feature/3485/game\\_design\\_essentials\\_20\\_.php?page=1](http://www.gamasutra.com/view/feature/3485/game_design_essentials_20_.php?page=1))

[앞서 나온 세 개의 기사, 즉 '20가지 예외적인 제어 체계(20 Unusual Control Schemes)', '20가지 오픈 월드 게임(20 Open World Games)', 그리고 '20가지 고난이도 게임(20 Difficult Games)'의 뒤를 이어, Gamasutra의 '게임 설계 필수 사항(Game Design Essentials)' 시리즈에서는 비밀을 찾는 것에서부터 알고리즘을 이용하여 생성된 콘텐츠를 해결하는 것에 이르는 '미스터리를 플레이어가 해결해야 하는' 타이틀에서 설계 교훈을 살펴 본다.]

실제로는 그렇게 오래된 일은 아니지만, 예전에 The Legend of Zelda라는 게임이 릴리스되었다. 당시에 이 게임은 혁신적이었지만 게임 세계에서 흔히 일어나듯이, 이후에 이 게임의 영감을 받은 게임들이 당연하게 쏟아져 나오는 과정에서 이 게임의 정말로 흥미로운 사항의 많은 부분이 전달되지 않았다.

이러한 특색 중 하나는 Zelda의 방대한 비밀 통로였다. 이후의 Zelda 시리즈를 포함하여 다른 게임에서는 어쩌다 한 번 서투르게 감추어진 비밀의 방이 나타나는 반면에, 첫 번째 Zelda 시리즈에서는 내내 나타났던 것이다. 실제로 이 게임의 오버월드 스크린의 거의 반에 몇 움큼의 루피에서부터 게임에 이기기 위해서 찾아야 하는 전체 던전에 이르는 보상을 제시하는 어떤 형태의 특별한 구역이 있다.

찾을 필요가 없는 것들이 이렇게 많이 게임에 감추어져 있기 때문에 게임에는 사실성으로 가장 잘 설명되는 어떤 성질이 생기는 것이다. 사실성은 비디오 게임을 설명하는 데 사용되는 유용한 단어이다. 이것은 생명이 있는 것 같다는 성질을 의미하지만, 그 안에 함축되어

있는 의미는 게임 리뷰 분야에서 그래픽의 품질로 재정의한 좁은 뜻의 "현실성(realism)"보다 더 심원하다. 적절하게 사용되었을 경우, 이 단어는 스크린 경계 밖에 플레이어가 무엇을 하든지 상관없이 나타나는 어떤 세계가 있는 것 같다는 것을 의미한다. 이것은 플렉시글라스 안에 봉할 수도 있는 타일 또는 폴리곤의 단순한 집합 이상의 것인 갖출 것을 다 갖춘 세계가 존재한다는 함축한다. 게임에서 이를 이용하면 플레이어가 자신이 하고 있는 것이 단지 게임이라는 사실을 잊게 만드는 데 도움이 된다.

물론 게임에서 이것은 성공의 정도는 다양하지만 대개 꾸며내는 것이다. 스스로 달리면서 점프하고, 도주하고, 탐사하면서 이야기를 전개할 수 있는 Grand Theft Auto 및 Dead Rising에서는 매우 잘 꾸며낸다. 강하고 직선적인 서술을 플레이어에게 강요하는 게임에서는 그렇게 잘 되지 않는다.

모든 게임에서 플레이어가 게임에 몰두하여 게임 세계의 실제 참여자인 것처럼 될 것을 요구하는 것은 아니다. 실제로 너무나 많은 게임에서 이런 상태를 원하여 그렇게 하려고 할 때 크게 실패할 수 있다. 하지만 미스터리한 요소를 게임에 주입하려고 하는 데는 다른 이유가 있다. 하지만 여기서는 독자적인 용어를 정의하고자 한다. 이 글에서 "미스터리하다"고 말할 때는, 게임이 플레이어로부터 플레이나 동작의 어떤 측면을 숨기는 것을 의미한다. 이 게임들은 감추어진 지식을 밝혀내는 것이 게임에 반드시 필요한 게임들이다. 이 게임들은 플레이어가 미스터리를 풀어야 하는 게임인 것이다. 그래서 "미스터리하다"는 단어를 사용하는 것이다.

이렇게 하는 방법에는 크게 두 가지가 있다. 가장 흔한 방법은 플레이어가 발견해야 하는 미리 제작된 콘텐츠를 많이 포함시키는 것이다. 지금까지는 이것이 가장 흔한 방법이며, 거의 모든 게임에서 어떤 방식을 이용하여 이렇게 한다. 일인칭 슈팅 게임에서는 플레이어의 카메라에서 볼 수 있는 영역의 일부만을 드러낸다. 몬스터의 동작이 알고리즘을 따르기는 하지만 게임 중에는 이해하기 어렵다는 점에서 Pac-Man 같은 간단한 게임도 미스터리하다는 말을 붙일 수 있다. 범주를 너무 넓게 정의해서 범위가 매우 넓은 게임이나 단지 새로운 콘텐츠를 밝혀내는 것 이상으로 문제를 깊게 가져가는 게임에 관여를 해야 한다면, 모든 게임을 미스터리하다고 할 수 있다.

표준 고지 사항: 이 문서는 "상위 20" 목록을 만들기 위한 것이 아니며, 이 게임들은 어떤 순위로 제시되지 않는다. 어떤 게임은 단지 주제와 지엽적으로 관련된 것으로 보일 수도 있다. 따라서 어떤 개념을 확실하게 이해하기 위해서는 명백한 경우와 명백하지 않은 경우 양쪽을 살펴보는 것이 유용하다. 이 게임들 중 상당수는 오래된 것이지만, 이는 의도한 것이다. 오래된 것이 나쁜 것은 아니다! 이 글의 목적은 다른 게임들이 수년 동안 어떻게 해왔는지를 보여주고, 바라건대 독자가 자신의 프로젝트를 진행하는 과정에서 영감을 주기 위한

것이다. 독자는 원칙이 어떻게 적용되는지를 보기 위해 이 게임들을 플레이할 수도 있다.

## 비밀을 발견하는 것을 강조하는 게임

### 1. Super Mario Bros. 3

흰색 블록

개발: Nintendo

설계: Shigeru Miyamoto, Takashi Tezuka

#### 포함 이유

시리즈의 세 번째 게임에는 단지 Super Mario 게임이기 때문에 나오는 흔한 비밀 외에도 놀랄 정도로 애매한 찾을 것이 약간 있다. 당시에 나온 Nintendo of America의 플레이어 가이드가 없었다면, 하얀색 배경 블록의 특별한 속성과 이를 이용하여 워프 휘슬(Warp Whistle)을 찾는 방법은 제쳐두고라도 매직 노트 블록(Magic Note Block), 보물선 또는 흰색 버섯 집을 찾을 수 없었을 것이다.

#### 게임 소개

Super Mario Bros. 게임에서는 비밀 파워 상승과 플레이 미캐닉 자체에 대한 통로를 찾으며, SMB3는 아마도 이 모든 것들 중에서 가장 우회적으로 감추어진 것들을 가졌을 것이다. 보이지 않는 블록을 찾는 것은 언제나 Super Mario Bros. 게임의 일부였으며, 이를 수행하기 위해서는 시행 착오를 피할 수 없다. 하지만 애매한 비밀 구역에 있어서, Super Mario Bros. 3는 매우 뛰어나다.

이 게임의 팬들은 플랫폼처럼 그 위에 올라설 수 있는 직사각형 모양 장식 블록의 일부에 특별한 속성이 있다는 것을 잘 알고 있다. 이러한 블록에는 몇 가지 색상이 있지만, 하얀색으로 된 블록만이 어떤 식으로든 예외적이다. 플레이어가 제어 패드를 아래로 누르고 있는 상태에서 몇 초 동안 하얀색 블록에 올라서 있다면, 블록 "뒤로 내려갈" 것이다. 그 다음에 NES는 스프라이트 우선 순위 기능을 이용하여 Mario가 배경 뒤에 있는 것처럼 보이게 만들 것이다. 플레이어는 어쩌면 8초 정도 거기서 물러나 있을 것이며, 배경이 되는 언덕과 같은 것 옆에 서 있다면 보이지 않을 것이다. 게임에는 이런 일이 발생할 것이라는 실마리가 없다. 설계자의 입장에서 이것은 순전히 짓궂은 일일 뿐이다.



이것은 그 자체로 매우 임의적이다. 풍경 뒤에 있을 때 플레이어는 적들을 지나쳐가기 때문에 적들은 플레이어에게 해를 끼칠 수 없다. 하지만 Super Mario Bros. 3의 모든 레벨에서 발견할 수 있는 흰색 블록은 그리 많지 않기 때문에 이 기능이 전략에 포함되는 경우는 드물다. 그래도 게임의 세 개 워프 휘슬 중의 하나를 구하기 위해서는 이 비밀 기능이 필요하다.

이렇게 하면 플레이어는 몇 레벨을 건너 뛸 수 있기 때문에 이 기능은 최초의 Super Mario Bros.에서 스크린의 맨 위를 달려가는 것과 비슷하다(이에 대해서는 Jeremy Penner가 Gamer's Quarter #7에서 훌륭히 설명했다). 하지만 그 중요성은 더 크다. 거의 보편적으로 무시되는 플랫폼 게임의 측면인 일반 배경 요소의 한 유형의 색상을 선택하여 여기에 의미를 부여함으로써, 설계자들은 그렇게 하지 않았을 경우보다 훨씬 더한 신비로움을 전체 게임 세계에 불어넣었다. 첫 번째 세계에서는 게임에서 발견되는 중요한 흰색 블록이 하나 밖에 없으며, 다른 어떤 임의적인 배경 요소도 특별한 의미를 지니지 않는다. 파이프에서 노트 블록과 도넛 리프트에 이르는 다른 모든 요소는 플레이어가 명확하게 읽을 수 있도록 설계된 것이다.

하지만 흰색 블록을 이용한 트릭은 게임 세계에 눈에 보이는 것보다 훨씬 많은 것이 있다는 의미를 부여한다. 이것은 전문가만이 알 수 있는 미스터리한 속성을 보여준다. 이것은 1990년대 초의 교정에서 일차적인 중요성을 가지는 지식이었다. 이것은 마법과 같은 느낌을 주는 것이다. 따라서 이 하나의 트릭이 전체 게임에 무게를 실어준다. 게임이 릴리스된 지 수개월 내에 모든 사람이 이 트릭을 알게 되었다는 사실이 이 트릭이 효과가 없다고 증명하지는 않는다. 오히려 효과가 있었다는 것을 증명한다. Dr. Strangelove가 언급했듯이 비밀로 유지한다면 전체 요점은 사라지는 것이다.

## 설계 교훈

어떤 면에서 이것은 궁극적인 터무니없는 트릭이다. 1-3의 어떤 지점에서 수초 동안 숙이고 있으면 게임에서 가장 귀중한 보물 중의 하나가 나오게 되리라는 것을 실제로는 알 길이 없는 것이다. 하지만 힌트가 있기 때문에 트릭이 알려지게 된다. 그리고 이것은 많은 트릭을 갖고 있는 게임에서 단지 하나의 트릭일 뿐이지만, 이것 때문에 플레이어가 게임의 나머지 부분에 관하여 궁금하게 될 만큼 이것은 애매하다.

## 2. Bubble Bobble

아이템 생성 메커니즘, 방대한 종류의 아이템, 버블 알파벳, 게임 모드 및 코드, 엔딩 조건

발표 및 개발: Taito

설계: Fukio Mitsuji

### 포함 이유

이 게임을 어떻게 포함시키지 않을 수 있겠는가? Bubble Bobble은 아케이드용으로 제작된 게임 중에서 아마도 가장 미스터리한 게임이며, 유일한 예외가 있다면 그 후속편인 Rainbow Islands가 될 것이다. 이 게임의 상당 부분은 수 년 전에 누군가가 마침내 코드를 디스어셈블했을 때 비로소 완전히 이해되었다.

### 게임 소개

Bubble Bobble은 여러 층에 걸쳐 있는 비밀 지식 때문에 고수를 위한 게임이 되었다. Bubble Bobble의 많은 시스템을 이용하기 위해서는 막대한 양의 사소하면서도 심원한 사실

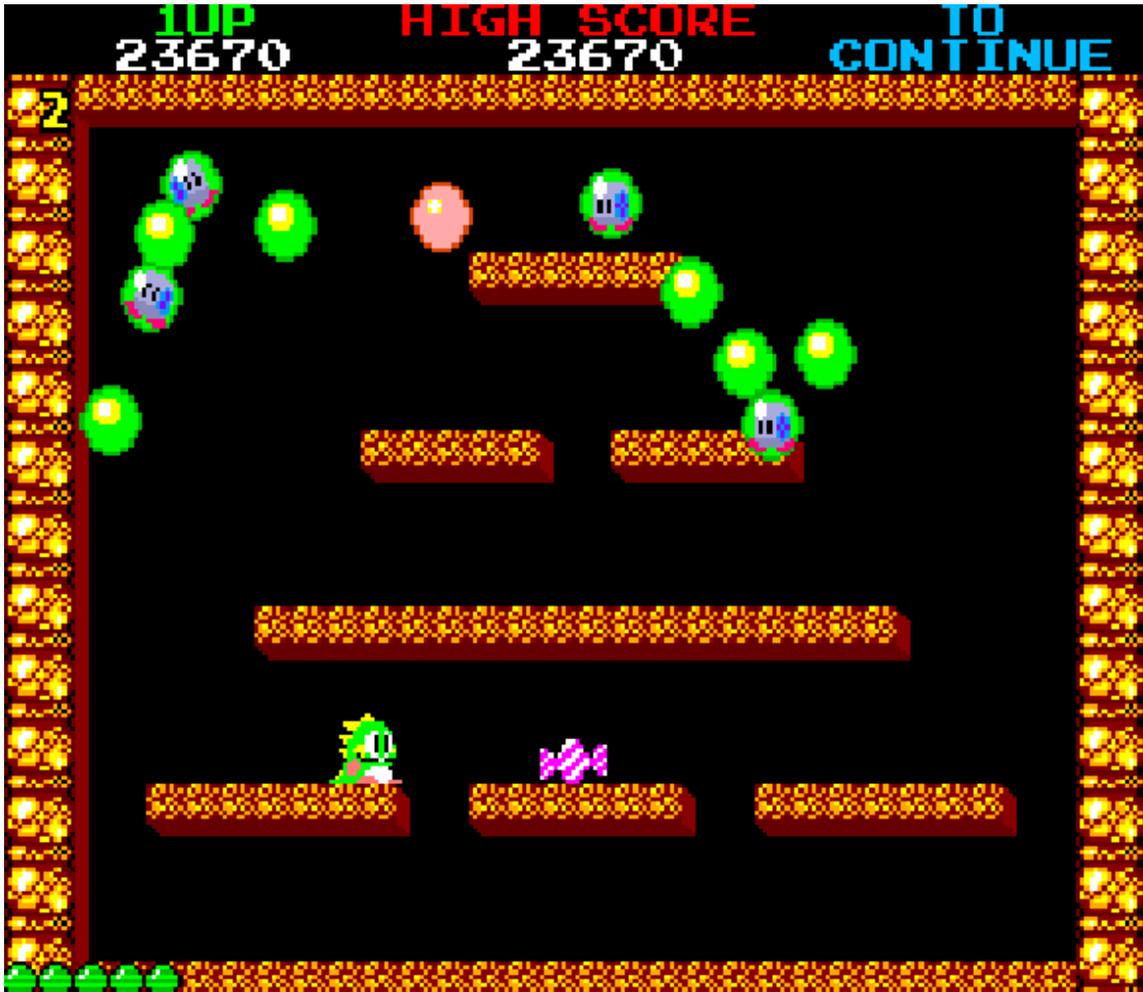
들을 알아야 한다. 그 중 일부는 이제 와서야 일반적으로 알려지고 있다.

첫 번째로 알아야 할 사항은 아케이드 게임에는 비밀 코드가 있다는 것이다. 당시에는 이 같은 사실이 거의 알려지지 않았다. 하지만 Bubble Bobble은 비밀 코드를 가지고 있을 뿐만 아니라 적절한 상황에서 이에 관하여 플레이어에게 알려 줄 것이며, 게임의 실제 엔딩을 보기 위해서는 하나를 사용해야 한다. 하지만 문제가 되는 상황은 거의 불쾌한 일이다. 하나를 보기 위해서는 죽지 않고 레벨 20, 30 또는 50까지 게임을 플레이해야 하며, Bubble Bobble에서는 처음 몇 단계 이후에 죽는 일이 비교적 흔하게 일어난다. 일단 적당한 레벨에 오게 되면, 해당 레벨에서 일반적으로 생성되는 보너스 아이템은 대신 많은 보너스 점수와 배경에 인쇄된 코드를 포함하고 있는 비밀의 방에 대한 문이 되겠지만, 코드 그 자체는 코드로 작성되어 있는 것이다.

처음에 해당 코드는 단지 룬 문자로 된 배경 장식처럼 보이지만, 이것은 실제로 개발자가 만든 비밀 메시지이다. 비밀은 플레이어에게 "하지만 그것은 진짜 끝이 아니었다!(But it was not a true end!)"라는 사실을 알려주는 엔딩 스크린에 나타나고, 룬 문자의 특별함이 밝혀진다. 이것은 플레이어가 2인용에서 이겼을 때 나타난다. 단지 1인용으로 게임에서 이겼을 때 플레이어는 임의의 레벨로 돌아가게 된다.

비밀 코드 중의 하나는 몇 개가 죽었던 상관없이 보너스 레벨로 가는 문이 나타나게 만드는 데, 이것은 플레이어가 이미 코드를 이해한 후에 알면 좋은 일이라고 생각된다. 기계의 어트랙트 모드(attract mode)에서 타이틀 스크린으로부터 들어가면 나타나는 또 다른 코드는 게임을 "수퍼" 모드로 만드는 데, 이 모드는 여러 레벨의 적들을 썬기 때문에 좀 더 어렵다. 하지만 2인용으로 모든 100개 레벨을 다시 통과하고 끝을 내어 게임을 진정으로 이기는 것은 수퍼 모드에서만 가능하다. (그리고 우연이지만 다행스럽게도 죽지 않고 이기는 대부분의 사람들은 실제로는 게임의 대부분을 한 명의 플레이어로 플레이하고, 마지막 보스를 물리치고 난 후에 두 번째 플레이어를 투입하여 이제 막 합류했기 때문에 두 번째 플레이어는 죽은 적이 없게 되는 것이다).

이것은 이 게임에 관하여 유일한 미스터리한 것과는 거리가 멀다. 아마도 가장 미스터리한 것은 특수 아이템이 나타나는 때를 결정하는 공식일 것이다. 대부분의 플레이어는 특수 아이템이 임의로 나타난다고 믿지만, 실제로 특수 아이템은 모두 플레이어의 동작에 의해 직접 결정된다. 게임은 발사된 거품, 점프한 횡수, 터진 물거품, 그리고 스크린 주위를 돌은 횡수와 같은 여러 가지 사소한 사항의 횡수를 세고 있다. 카운터가 한계값을 초과하면, 다음 레벨에서 아이템을 생성하도록 스케줄링하는 플래그가 설정된다.



이 횃수와 플래그는 게임이 종료될 때 재설정되지 않기 때문에, 새 플레이어가 겨우 처음 몇 단계에서 고급 아이템을 받는 경우가 종종 생긴다. 하지만 이 같은 사실은 예를 들어 누군가가 물 거품을 쏘는 것이 레벨을 건너뛰는 우산이 더 자주 나타나게 만든다는 것을 알고 레벨을 더 자주 건너뛰기 위하여 이를 이용할 수 있다는 것도 의미한다. 아이템을 나타나게 만드는 트릭의 일부가 모든 레벨에서 수행될 수 있는 것은 아니기 때문에, 어떤 아이템이 "덩어리지게" 되어 특정 보드에서 더 자주 발생하게 되는 경향도 있다.

이것은 게임 설계에서 무질서한 동작에 대한 거의 교과서적인 예이다. 플레이어의 많은 다양한 동작들은 알려지지 않은 프로세스를 통하여 독자에게 있어서는 임의적이라고 생각되는 결과를 나오게 만들지만, 자신들에게 있어서는 어떤 미스터리한 일관성을 가지는 것으로 보인다. 이것은 플레이어가 게임을 하게 만드는 흥미로운 방법이다. 즉, 거의 명령에 의해 나타나는 물체에 성공을 연결하지만, 이 명령이 어떤 것인지는 정확히 말해주지 않는 것이다.

## 설계 교훈

많은 게임에서 의사 난수를 사용하지만, Bubble Bobble은 보기에선 임의적이지만 실제로는 의사 난수를 별로 사용하지 않는다. 일단 그 원칙을 이해한 후에는 게임이 될 수 있지만, 플레이어에게는 설명되지 않는다. 그 결과는 여러 가지 플레이 스타일이 이러한 스타일에 대한 게임의 반응에서 일관된 변화를 일으키는 것이고, 이로 인하여 플레이어는 너무나 복잡해서 미신과 비슷한 반응을 보일 수도 있다.

## 링크

소스 코드 조사를 통해 밝혀진 Bubble Bobble의 미캐닉은 [Bubble Bobble 정보 페이지](#)에 설명되어 있다.

## 3. Rainbow Islands

더 많은 비밀을 가진 Bubble Bobble

개발: Taito

설계: Fukio Mitsuji

### 포함 이유

이 게임에 대한 아이템 생성 체계는 그다지 복잡하지 않지만, 어떤 계획을 세워서 사용에게 필요한 아이템을 만드는 것은 여전히 가능하다. 그 다음에는 진짜 엔딩에 도달하기 위해서 모아야 하는 보석을 얻기 위해 플레이어가 거쳐야 할 수단이 있다. 그리고 그 다음에는 보석을 색상 순서로 모아서 달성할 수 있는 상위 목표와 그렇게 하기 위한 특별한 영구적인 파워 상승이 있다! 그리고 그 다음에는 비밀 레벨이 있다! 그리고 그 다음에는 죽지 않는 경우에 대한 비밀 엔딩이 있다! 그리고 그 외에도 또 있다! 정말 놀라운 일이다!

### 게임 소개

첫 번째 게임의 플레이에 대한 후속작이기는 하지만 Rainbow Islands는 Bubble Bobble의 아이템 생성 기준의 상당 부분을 포기한다. 그 대신에 죽지 않고 많은 적을 죽이기만 하면 결국 더욱 유용한 파워 상승이 일어날 것이다. 하지만 이 게임에는 찾아야 할 비밀 구역과 엔딩이 훨씬 더 많이 있다. 이 때문에 이 게임은 아마도 Bubble Bobble 플레이 스타일의 게임 중에서 가장 뛰어날 것이다.

이 게임의 기본 미캐닉은 첫 번째 게임에서 사용된 거품의 유사물인 무지개를 만들고 그 아래에 적들을 가두는 것이다. 그 다음에 플레이어는 무지개 위에서 점프하여, 무지개가 무너져 내리도록 만들어서 그 아래에 있는 모든 적을 죽일 수 있다. 즉, 플랫폼에 올라가기 전에 적들을 아크로 날려 보내는 것이다. 이런 식으로 여러 명의 적을 한 번에 죽이면 색깔이 있는 원석이 나온다.

레벨은 세 가지 방법으로 완료할 수 있다. 주된 방법은 단지 곧장 진행하여 보스를 물리치는 것이지만, 이런 식으로 단 한 레벨이라도 마친다면 플레이어는 최고의 엔딩을 볼 수 없다. 적들은 생명을 잃게 되는 원인이며 한 레벨에서 긴 시간을 보내는 것 또한 플레이어를 죽게 만드는 것이지만, 플레이어는 그저 정상적으로 게임을 진행하고 진짜로 게임에서 승리할 수 없기 때문에 이것은 정말로 함정이다. 플레이어는 진행을 하면서 다른 사항들도 걱정을 해야 하는 것이다.



정말로 레벨을 완료하기 위해서는 보스를 만나기 전에 각 색상의 보석 하나를 얻어야 한다. 보석은 여러 명의 적을 한 번에 죽일 때 생기지만, 그 색상은 처음에는 임의적으로 나타나는 것을 보인다. 어쩌면 당연한 일이지만, 필요한 색상을 얻을 수 있는 트릭이 있다. 적이 죽은 후에 주변으로 날아갈 때, 죽은 적이 떨어질 장소의 X 위치가 그곳에 나타날 보석의

색상을 결정한다. 스크린의 왼쪽 끝은 빨간색, 스크린의 다음 7번째는 오렌지색, 그 다음은 노란색, 그리고 그 다음은 스펙트럼의 나머지 색상이 나타나며, 마지막은 스크린의 오른쪽 끝에 보라색이 나타난다.

각 레벨에는 4개의 단계가 있으며, 보석은 단계에서 단계로 전달된다. 따라서 전체 레벨을 "수퍼클리어"하기 위해서 플레이어는 보스로 가는 단계에 있는 스크린의 각 수직 조각에서 하나의 보석을 얻어야 한다. 이렇게 하면 보스는 자신이 죽을 때 거대한 보석을 플레이어에게 주며, 모두 7개의 거대한 보석으로 정상적인 마지막 레벨을 마치면 진짜 마지막 레벨을 볼 수 있다. (영리하게도 이들은 모두 Arkanoid와 Darius와 같은 다른 Taito 속성에 바탕을 두고 있다). 이 레벨들은 보석 대신에 거울을 갖지만, 얻는 방식은 동일하다. 플레이어가 이 모든 것을 가지게 되면 진짜로, 진짜로, 진짜로... 이길 수 있다. 마지막 보스를 물리칠 수 있다면, 바로 그것이다.

하지만 그 외에 또 있다. 만약에 플레이어가 빨간색에서 보라색의 색상 순으로 보석을 모았다면, 보스에 도달했을 때 특별한 문이 나타날 것이다. 이 문으로 들어가면 보스를 건너 뛰고, 어렵게 얻은 다이아몬드로 플레이어에게 보상하고, 나머지 게임 내내 지속되는 영구적인 파워 상승을 플레이어에게 부여한다. 이것은 감추어진 물체를 찾는 흔히 있는 게임플레이 장난의 맨 위에 있는 것이다.

## 설계 교훈

이 게임에는 수수께끼와 미스터리가 풍부하다. 비밀 자체만으로는 게임이 될 수 없으며, Rainbow Islands의 핵심 플레이 미캐닉이 좋기는 하지만 Bubble Bobble의 것만큼 그리 강하지는 않다. 정상적으로 플레이를 했을 때조차도 두 게임 모드 긴 게임이기는 하지만, 아케이드에서 둘 중 한 게임을 완료하고 나쁜 결말을 본 사람은 누구라도 할 일이 더 남아 있었다고 잘못 생각할 수 있는 사람은 없다.

## 링크

이 게임에 대한 가장 완전한 자원은 아마도 [Rainbow Islands 정보 페이지](#)일 것이다.

## 4. Marble Madness(홈 컴퓨터 버전)

수종 미로

Electronic Arts 사의 다양한 개발자들에 의해 개발됨(첫 번째 버전은 Atari Games가 개발)

최초 설계: Mark Cerny

## 포함 이유

지금에 이르러서도 그리 잘 이해된 것은 아니지만, Marble Madness의 컴퓨터 버전에는 수중 미로라고 하는 비밀 레벨이 포함되어 있다. 이 레벨을 통과하는 것이 어지러울 정도로 어렵고 애매하기는 하지만, 이 레벨에 이르는 방법 역시 확실하지는 않다.

## 게임 소개

(잡지 광고에 실렸는데도) 이 게임이 잘 알려져 있는 것은 아니지만, Electronic Arts에서 발표한 Marble Madness의 홈 컴퓨터 판에는 어떤 종류의 비밀 레벨이 포함되어 있다. 첫 번째 레벨에서 플레이어가 목적지로 향하는 창호의 뾰족한 쪽에서 뛰어 내려 스크린의 왼쪽에 있는 바닥에 내린다면 비밀 통로를 발견할 수 있다. 비밀 통로는 단지 여기 저기 부딪히면서 다닌다고 찾을 수 있는 것이 아니다. 플레이어는 타이머가 13 초가 남았음을 알릴 때까지 특정 장소에서 기다려야 한다. 정확하게 수행한다면, 아래로 향하는 엘리베이터가 나타날 것이며, 곧이어 플레이어는 시계에 99초가 표시된 채로 새로운 구역에 도달하게 될 것이다. 이 구역은 이 게임의 EA의 포트에서만 나타나는 수중 미로이다.



이 새 구역은 실제 게임보다 조금 더 불안스럽다. 이것은 일련의 동작을 수행하기 보다는

플레이어가 앞으로 나아가기 위해서는 무엇을 해야 할지를 풀어야 하는 일련의 퍼즐이다. 퍼즐의 상당 수는 두 개의 마블을 끌어내야 하기 때문에, 이 레벨에 한 명의 플레이어가 들어갈 수 있기는 하지만 혼자서는 실제로는 레벨을 완료할 기회를 갖지 못할 것이다.

퍼즐의 상당 수는 주 게임에는 나타나지 않는 물체인 엘리베이터나 강과 관련이 있다. 시간이 다하면 레벨이 재설정되지만 플레이어가 죽으면 게임이 즉시 종료되기 때문에 여기서는 게임 규칙도 다르다.

이것은 Marble Madness를 플레이하기 위하여 자신이 의미하는 것을 크게 변경시키고 그 외에도 미칠 정도로 어렵기 때문에, 게임의 어두운 구석에 놓여 있는 애매한 이스터 에그로 분류하는 것이 적당하다.

## 설계 교훈

수중 미로는 주 게임과 다르게 플레이되고 상당히 작으며, 게임을 마쳐도 어떤 특별한 것이 발생하지 않기 때문에 이것을 기묘한 작은 비밀로 분류하는 것이 좋다. 하지만 플레이어가 처음으로 수중 미로에 마주치게 되는 순간은 정말로 근사한 것이다.

## 5. Ogre Battle

임의적이지 않은 전투, 복잡성을 통한 혼란, 캐릭터 취득, 카오스 프레임 엔딩 스포일링 메커니즘

발표 및 개발: Quest

설계: Hiroshi Minagawa

## 포함 이유

이 게임은 어떤 시점에서 플레이하기 위해서 거의 모든 사람이 FAQ에 의존해야 하는 게임이다. 플레이어가 게임을 진행하는 중에 게임 역시 어떤 애매한 방식으로 점수를 기록하고 있다. 마지막 레벨을 마친 후에 점수는 플레이어에게 어떤 엔딩이 나타날지를 직접 나타낸다. 하지만 점수를 올리는 것은 어려우며, 점수가 어떻게 변하는지는 정확하게 설명된 적이 없다.

## 게임 소개

Ogre Battle은 여러 가지 이유로 흥미로운 게임이다. 이것은 여전히 플레이에 상당한 모사주의 접근법을 적용하는 일본의 RPG이며, 그것만으로도 조사하기에 충분한 이유가 된다. 이것은 훌륭한 도전이며, (전술과는 상대적인) 강력한 전략적 구성 요소를 갖고, 실시간으로 이동하며, 심지어는 강력한 경제적 요소도 갖추고 있다. 하지만 싸움이 시작되면, 플레이어는 대체로 여기서 벗어난다. 플레이어의 캐릭터들은 스스로 무엇을 할지를 결정한다.

캐릭터들에 대해서 설명하자면, 캐릭터들은 각각 얼라인먼트(Alignment)라고 하는 스탯을 가지며, 이 스탯은 일반적으로 해당 기간의 D&D 사용에 매핑된다. 100으로 향해 가는 높은 얼라인먼트 수치는 질서(Lawful)를 의미하고, 0으로 향해 내려가는 낮은 얼라인먼트 수치는 혼동(Chaotic)을 의미한다. 플레이어는 일반적으로 필드에 나타나자마자 많은 일행을 가지며, 각자 자신의 얼라인먼트 값을 갖는다. 뱀파이어와 같이 어두운 적을 물리치면 얼라인먼트가 상승하고 성직자와 같이 밝은 적을 물리치면 얼라인먼트가 내려간다. 얼라인먼트를 수정하는 다른 것들이 있기는 하지만, 척도의 어느 한 쪽 끝에서 한계에 다가가면 그 값을 되돌리는 것이 더 어려워진다. 캐릭터의 발달은 얼라인먼트에 달려 있는 경우가 많다.



여기에서 주의해야 할 또 다른 사항들이 있다. 얼라인먼트 값이 낮은 캐릭터가 게임의 마을 기지 중의 하나를 점수하면, 게임은 그 마을이 점령되었다고 한다. 얼라인먼트 값이 높은 캐릭터가 그 마을을 점수하면, 마을이 해방되었다는 메시지가 나타난다. 게임은 플레이어를 설명하는 또 다른 스탯을 추적하기 때문에 이 메시지들은 매우 중요하다. 이것은 명성 (Reputation)이며, 캐릭터의 얼라인먼트가 읽기 쉬운 숫자인 반면에, 명성은 매우 작은 바를 통한 경우를 제외하면 직접 보고되지 않는다. 여기에 명성을 지배하는 기본 규칙이 있다. 마을을 점령하면 명성이 낮아지고, 마을을 해방시키면 명성이 높아지는 것이다!

다수의 다른 항목들도 명성에 영향을 주며, 이러한 사항들은 플레이어가 명성을 높게 유지하려고 할 때 게임을 더욱 어렵게 만드는 방식으로 명성에 영향을 준다. 적보다 레벨이 훨씬 더 높은 (레벨 차이가 2보다 큰) 캐릭터를 이용하여 적을 물리치면 명성이 내려가지만, 차이가 적거나 심지어는 더 낮은 레벨의 캐릭터를 이용하여 적을 물리치면 명성이 올라간다. 주어진 맵에서 게임 시간이 지나감에 따라 플레이어는 자신이 점수한 마을에서 수입을 얻는다. 돈은 맵에서 맵으로 전달되지만, 수입은 재설정된다. 이 때문에 플레이어 중에는 어떤 맵에 머무르면서 자금을 만들 생각을 할 수도 있겠지만, 한 곳에 너무 오래 머물러 있으면 명성이 내려간다.

그렇다면 명성은 왜 그토록 중요한가? 이 명성이야말로 이 게임을 훨씬 더 미스터리하게 만드는 것이기 때문이다. (일반적으로 문서화되지 않은 기회인) 플레이어의 명성에 의존하지 않거나 합류하는 캐릭터 외에도, 명성은 플레이어가 게임에 승리한 후에 받는 엔딩에 직접 영향을 준다. 명성이 너무 낮으면 플레이어는 나쁜 엔딩을 받게 될 것이다. 이러한 엔딩에서 플레이어의 캐릭터는 자신의 그룹에 속해 있는 캐릭터 중의 하나에 의해 암살되는 새로운 폭군이 된다.

물론 매뉴얼을 제외하면 이 중 어떤 것도 플레이어에게 설명되지 않으며, 매뉴얼에도 그저 모호하게만 나와 있다.

## 설계 교훈

게임 전체에 걸쳐 있는 플레이어의 행동이 알고리즘을 이용하여 혼합되고, 결국에는 (다른 무엇보다도) 이 값에 따라 스토리의 최종 결과가 선택된다. 결과는 Ogre Battle이 Pac-Man 이나 Rogue 만큼 실제로 점수화되기는 하지만 플레이어는 어느 정도의 신빙성을 가지고 그 점수를 알지 못한다는 것이다. 후속편인 N64에서는 훨씬 더 심하다. 플레이어는 명성 점수가 얼마인지를 말해주는 바조차 가지지 못하는 것이다.

## 링크

GameFAQs에는 Chaos Frame 시스템의 더욱 충실한 설명을 제공하는 Ogre Battle에 대한 멋진 [미캐닉 FAQ](#)가 있다.

## 6. Magic Sword

많은 아이템, Bubble Bobble과 비슷한 아이템 기능, 그리고 놀랄 정도로 깊은 전략

발표 및 개발: Capcom

설계: Y. Ohnishi, T. Sadamoto, Y. Okamoto.

### 포함 이유

Magic Sword는 야만인이 나오는 Bubble Bobble과 같은 것으로서, Bubble Bobble보다 훨씬 더 많은 개성과 깊이를 가진 자르고 베는 플랫폼머가 모든 권리를 가진다.

### 게임 소개

Magic Sword는 설명하기 힘들 정도로 훌륭한 작은 아케이드 게임이다. 이것은 그다지 심각하지 않은 자르고 베는 판타지 게임이며, 동시대의 Capcom 판타지 게임인 King of Dragons와 Knights of the Round보다는 Pang과 Rainbow Islands와 더 비슷한 느낌이 든다.

이 두 게임은 실제로는 D&D보다는 Final Fight에 더 가까운 난투극 게임이다. Magic Sword에는 대체로 찾아야 할 비밀 사항이 매우 다양하기 때문에 그럭저럭 비교적 순수한 모험 게임의 느낌이 든다. 이 게임의 "조력자" 캐릭터 시스템은 여기서 어떤 역할을 한다. 조력자란 플레이어를 따르면서 일반적으로 플레이어의 동작을 흉내 내는 도움을 주는 캐릭터이다. 조력자는 Gradius의 옵션과 조금 비슷하게 동작하지만, 플레이어는 한 번에 한 명의 조력자만을 가질 수 있으며, 조력자는 손상을 입을 수 있다. 플레이어는 타워의 50개 레벨을 거쳐가면서 여러 가지 종류의 자물쇠가 달려 있는 다양한 문을 보게 된다. 문을 하나 열면 열쇠 하나를 사용하게 되지만, 콘텐츠를 이용할 수 있게 된다(콘텐츠는 대개 조력자와 아이템이다). 실제로는 본질적으로 열쇠가 부족하지는 않으며 초기 레벨에서 플레이어는 본질적으로 각 유형을 무한정 공급받는다(하지만 여분의 열쇠로 레벨을 마치면 더 많은 점수와 더 많은 보너스 생명의 가치가 있다).

열쇠는 자원 관리의 요소를 보물 검색에 추가한다. 대부분의 문은 분명히 볼 수 있지만, 이 모든 문을 열기 위해서는 플레이어가 여분의 열쇠가 있을 경우 하나의 열쇠를 사용해야 한다. 그 안에는 다양한 아이템과 조력자가 있지만, 플레이어가 이전의 플레이에서 해당 아이템과 조력자를 기억하지 않는다면 그것이 무엇인지를 알지 못한다. (경우에 따라서는 적이 들어 있는 경우마저 있다.) 조력자들은 구역에 따라 적합한 여러 가지 힘을 갖고 있다. 일부는 강력한 난투 상황의 공격력을 갖고, 일부는 빠른 원거리 공격력이 있고, 또 일부는 죽지 않는 적에 강력한 마법을 가지며, 도둑은 플레이어를 위해 숨겨진 물체를 찾는다. 쌓으로 이루어진 조력자는 소수의 장소에서만 나타나거나 특별한 수단을 통해서만 나타나는 그 자체로 비밀이며, 이러한 종류의 게임에서 상당히 높은 능력의 발견물이다.



Magic Sword의 비밀들을 특히 인상적으로 만드는 것은 비밀에 사용된 코딩의 양이다. 조력자들은 각각 자신만의 서브루틴과 자신들을 찾기 위해 게임에 미치는 영향을 갖는다. 진실로 비밀스러운 비밀을 게임에 포함시키는 것에 대한 큰 트레이드오프는 왜 적은 비율의 플레이어들이 볼 콘텐츠를 포함하여 인력을 낭비하는 것인가? 이다.

한 가지 이유는 이것이 게임을 무작위화한다는 것이다. 플레이어가 유용한 객체를 나타나게 만드는 방법을 모르기 때문에 유용한 객체를 항상 찾을 수 없거나 심지어는 이것이 존재하는 것조차 모른다면, 이것은 경험의 다양성을 증가시키고 결국에는 게임을 리플레이 가능성을 증가시킨다. 플레이어마다 다른 도전 과제를 만나게 된다. 도둑보다는 리저드맨으로 보스를 물리치는 것이 더 쉽기는 하지만, 도둑을 사용하면 자신만의 이점을 제공할 수 있는 숨겨진 상자가 나타날 것이며, 이러한 이점은 보스와의 싸움에서 유용할 수 있을 것이다.

또 다른 이유는 이것이 게임을 더욱 미스터리하게 만든다는 것이다. 요즈음에는 대부분의 게임이 의미 있는 일탈에 대한 기회가 없이 선형적인 경로로 배치된다. 일탈은 Mario 게임과 같이 탐색을 위한 대체 영역을 제공하는 것에서 나올 수도 있고 플레이어가 이용할 수 있는 자원을 다양하게 만드는 것에서 나올 수도 있다.

## 설계 교훈

적과 아이템의 위치는 일반적으로 하드코딩되지만, Magic Sword에는 충분한 임의성이 있고, 플레이어가 보유하는 모든 조력자나 아이템에 의해 가능해지는 게임 상황에 대한 충분한 변화가 있으므로, 실제로는 정말로 다시 플레이할 수 있는 가능성이 높다. 이런 식으로 미리 만들어진 레벨의 영향은 플레이어의 조력자와 가지고 있는 아이템에 따라 달라지며, 여기서 플레이어는 한 번에 이 두 가지 중에 하나만을 가질 수 있다. 이것은 앞으로는 비교적 드문 게임 설계 스타일이지만 잘만 하면 플레이 가능성을 크게 증가시킬 수 있다.

## 7. Athena

블록 깨기, 많은 아이템 및 기능과 다수의 비밀

개발: SNK

### 포함 이유

이 게임은 그 안에 보물을 숨기고 있는 격파 가능한 블록의 수가 매우 많기 때문에 Super Mario Bros.의 후속작처럼 플레이한다. 블록을 흩어버리는 것이 성공의 열쇠이기는 하지만, 선택할 수 있는 무기의 수가 많이 있으며, 적을 죽이는 데 가장 좋은 무기 중의 일부는 블록을 깨는 데에는 좋지 않다.

### 게임 소개

버그, 오점이 많은 제어, 그리고 생명이 가득 찬 플레이어를 한 순간에 죽일 수 있는 잔인한 손상 모델로 가득찬 끔찍한 NES 포트의 힘을 바탕으로 한 Athena에 많은 사람들이 실망했다. 이 아케이드 게임은 비록 여전히 매우 어렵기는 하지만 다소 세련되게 되었다.

Athena는 Super Mario Bros.가 나오고 나서 얼마 후에 나왔으며, Athena의 상당 부분은 Super Mario Bros.의 직접적인 영감을 받은 것처럼 느껴진다. Mario의 블록에는 단지 가끔

씩 중요한 아이템이 포함되었던 반면에, Athena의 세계는 찾아야 할 물건을 훨씬 더 큰 비율로 감추고 있는 격파 가능한 돌로 구성되어 있다. 게임을 시작할 때 플레이어는 블록을 부술 수단이 없다. 적을 죽이면 블록을 부술 때 필요한 최초의 도구가 나타난다. 이용할 수 있는 무기에는 몇 가지 종류가 있으며, 이러한 무기의 주된 차이 중의 하나는 플레이어가 각각의 무기를 이용하여 서로 다른 방식으로 블록을 파괴할 수 있다는 것이다. 즉, 위쪽 가까운 곳, 제한된 범위 내, 멀리서, 수평으로, 바로 위, 또는 큰 폭으로 파괴하는 것이다. 일부 무기는 몬스터를 죽이는 것보다는 블록을 부수는 데 유용하다.

일단 플레이어가 블록을 열어젖힌 후에는 다양한 수준의 갑옷, 헬멧, 방어물, 무기 및 그 밖의 잡다한 아이템을 포함하여 여러 가지 종류의 찾을 수 있는 것이 나타나며, 이러한 아이템은 도처에 널려 있다. 흔치 않은 경우지만, 많은 블록이 좋은 아이템 중에 나쁜 아이템을 포함하고 있다. Super Mario Bros.: The Lost Levels에서는 독버섯이나 부비트랩이 설치된 스타맨이 가끔 나오지만, Athena의 블록에서는 갑옷과 무기의 등급을 떨어뜨리는 아이템, 독, 시간을 줄이는 아이템, 보유 아이템을 파괴하는 아이템, 심지어는 불쾌한 떠다니는 머리가 끊임없이 나온다. Castlevania의 단검 양초는 이 정도로 성가시기를 바랄 뿐이다.



하지만 이러한 아이템의 존재는 게임에 짜임새를 증가시키는 것이다. 여러 종류의 비디오

게임이 가지는 위험은 "그래서?"라는 요인이다. 게임에서 이 레벨과 다른 모든 레벨을 구분하는 것은 무엇인가? Mario에서는 여러 가지로 배열된 극복해야 할 적, 격차 및 벽, 여기 저기 흩어져 있는 숨겨진 통로, 그리고 파워 상승의 이용 가능성으로 레벨을 구분한다. 첫 번째 Super Mario Bros.에서는 그렇게 많은 여러 가지 종류의 게임 요소를 갖지는 않았지만, 이들이 배열된 방식은 예술에 가까운 것이었다. Athena에서는 획득한 파워 상승에 따라 부술 수 있는 블록의 수를 제한하여 레벨을 구분한다. 즉, 지금 가지고 있는 무기가 하나의 블록을 부수는 데 두 번을 휘둘러야 하는 노란색 칼이며 원하는 아이템은 다섯 층 아래에 묻혀 있는데 뒤에서는 말을 탄 적이 끊임없이 쏟아져 나온다면 아마도 이 아이템을 추구할 가치는 없을 것이다. 이러한 아이템에 대한 종류는 충분히 다양하기 때문에 때로는 단지 블록을 깨는 데 더 좋다는 이유로 매우 강한 아이템보다는 매우 약한 아이템이 더 유용하다.

그런데 단지 엔딩의 등급을 낮추는 대신에 조건이 만족되지 않았을 경우 마지막 보스를 실제로 패배시킬 수 없다는 점을 제외하면 이 게임도 Solomon's Key나 Mighty Bomb Jack처럼 복잡한 승리 조건을 갖고 있다. 마지막 보스와 실제로 싸우기 전에 플레이어에게 말을 알려주었다면 게임은 멋이 있었을 것이다. (게임의 모든 부분이 잘 되었다고 말하지는 않았다.)

## 설계 교훈

Athena의 핵심은 플레이어가 블록을 깰 수 있는 방법에 있지만, 플레이어가 항상 자신이 원하는 블록을 부술 수 있는 것은 아니다. 보통의 아이템이 매우 좋은 아이템을 얻는 데 도움을 주지 않을 때조차도 때때로 나쁜 아이템이 매우 좋은 아이템을 얻는 데 도움이 된다.

## 8. Mighty Bomb Jack

피라미드와 특별한 엔딩 조건의 매핑.

개발: Tecmo

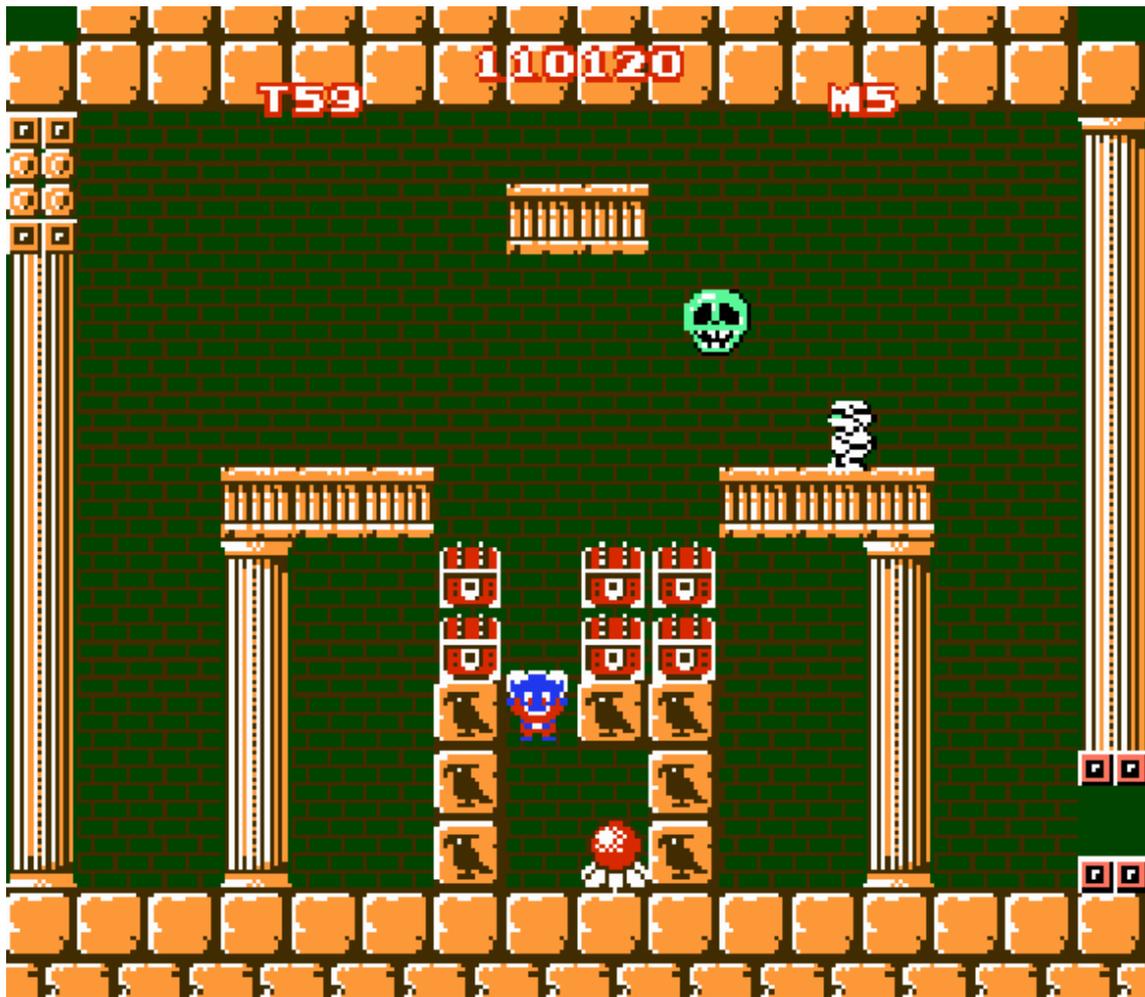
### 포함 이유

Bubble Bobble과 Solomon's Key처럼 Mighty Bomb Jack은 비밀 아이템 및 구역이 가득 차 있으며, 플레이어는 최상의 엔딩을 보기 위해 잘 감추어진 많은 아이템과 구역을 찾아야 한다.

## 게임 소개

많은 비평가들의 악평을 듣기는 했지만, Mighty Bomb Jack은 상당히 영리한 작은 게임이다. 때때로 적들이 플레이어 바로 위나 벽 속에 가득하기 때문에 이 게임은 그 구현에 있어서 조금 광대하지만 최소한 이 게임은 NES Athena보다는 적의 수가 적다. 애매함이 덜한 아케이드 게임인 Bomb Jack의 후속작인 Mighty Bomb Jack의 일차적인 플레이 방법은 Bomb Jack과 같다. 플레이어는 자신의 유일한 기술이나 다름없는 극도로 높은 점프력을 가진 캐릭터를 제어한다. 플레이어는 도약 중에 점프 버튼을 다시 눌러서 궤도를 조절할 수 있다. 점프 버튼이 눌러지면, 수직 속도가 감소하고 점프를 일찍 종료시키며 하강을 느리게 만든다. 이것은 수평 속도를 변경시킬 수 없기 때문에, 플레이어는 이 버튼을 빨리 눌러서 먼 거리를 가로지를 수 있다. 단 그렇게 가로지를 수 있는 공간이 있어야 하며, 공중에서 적이 플레이어가 가는 길 한 가운데서 생겨나지 않아야 한다. 이제 모든 것을 말했다. 이것이 의미하는 바는 플레이어가 버튼을 빠르게 눌러서 효과적으로 "미끄러질" 수 있으며 다가오는 적들 사이에서 수평으로 미끄러져서 놀라운 탈출을 수행할 수 있다는 것이다.

이것이 정확히 "Metroidvania"는 아니지만, 이 게임에서 더 깔끔한 것 중의 하나는 이것이 일련의 분기되는 레벨로 구성되어 있기는 하지만 여전히 조리 있게 구성되어 있다는 것이다. 매뉴얼에는 매뉴얼에 정밀하게 표시된 첫 번째 구역들의 윤곽을 가진 이집트 피라미드에 대한 설명이 나와 있다. 다음 레벨이 조사되면 이 레벨들의 위치도 맵에 나올 수 있다. 플레이어가 이렇게 하면 게임에 관한 일부 흥미로운 일이 분명해진다.



몇몇 장소에서 복도는 피라미드 밖으로 이어진다. 이렇게 될 경우, 그래픽은 하늘과 구름으로 바뀐다. 게임이 정밀하게 표시되면, 뭔가 있어야 할 것처럼 보이는 장소인 중요한 공간이 있는 장소가 있다는 것이 분명해지며, 이는 비밀의 방을 추적하기 위한 중요한 실마리이다. 여기에 적용된 원칙은 해당 구역이 임의로 배치된 것이 아니라 깊은 구조에 따라 배치되었다는 것이며, 이는 주의 깊은 플레이어에게 통로 위치에 관한 미묘한 힌트를 제공하여 탐사에 도움을 준다.

이러한 비밀 구역 중 두 곳에 특별한 의미가 있다. 플레이어가 게임을 정상적으로 진행하여 평범한 엔딩을 맞이할 수도 있지만, 최상의 엔딩을 얻으려면 플레이어는 깊이 묻혀 있는 특별 아이템을 찾아야 한다. 플레이어는 해당 레벨에서 뛰어 올랐을 때 사라지는 표시되지 않은 부술 수 있는 블록에 뛰어 오르고 이를 이용하여 아래로 파고 들어가서, 그 일부에 비밀문이 나타나게 만드는 아이템을 포함하고 있는 감추어진 보물 상자들을 찾아야 하는 경우가 많기 때문에 이러한 특별 아이템을 찾는 일은 매우 어렵다. 이러한 상자 중 일부는 처음에는 볼 수 없으며, 커다란 빈 방의 가운데에 감추어져 있다. 공중에서 나타나는 컬러 미라와

앵무새가 플레이어를 가차없이 쫓고 있는 중에 스크린상의 모든 사소한 장소에 뛰어 오르는 것은 골치 아픈 상황일 수 있지만 이것은 비밀 아이템을 찾기 위해서 해볼 만한 일이 분명하다.

## 설계 교훈

(역시 Tecmo에 의해 개발된) Solomon's Key처럼 이 게임도 최상의 엔딩을 얻지 않고 이길 수 있다. 오늘날에 여러 엔딩을 가지는 많은 게임들과는 달리, 이 게임에서 플레이어는 최적의 아닌 엔딩에서 속았다는 느낌을 받지 않는다. 나타난 엔딩에 무관하게 그럭저럭 승리할 수 있는 것이다. 분통 터지게 하는 비밀 방에서 크리스탈 볼을 찾아 다니는 것은 단지 만족스러운 엔딩을 더욱 만족스럽게 만들 뿐이다.

일부 개발자들은 자신들이 좋아하지 않는 방식으로 플레이하였거나(예: Ogre Battle), 일부 애매한 트릭을 풀지 않았다는 이유로(예: 모험 스타일의 Castlevania) 플레이어를 처벌하기 위한 수단으로 나쁜 엔딩을 끼워 넣는다. Castlevania II: Simon's Quest에서 Simon Belmont의 운명은 게임이 빨리 끝날수록 나빠진다.

이제 이해해야 한다. 나는 게임 설계자들이 비굴하게 따라야 하는 많은 극히 엄격한 "하지 마!"라는 규칙을 부과하는 것을 싫어한다. 지금까지 내가 들어온 거의 모든 조언에 대한 좋은 반례가 있다. 하지만 나는 (두 번째 Phoenix Wright의 마지막 경우에서와 같은 가짜 엔딩이 아니라) 진짜 여러 개의 엔딩을 게임에 집어넣으려고 한다면 플레이어가 상황에 상관없이... 특히 플레이어가 다른 어떤 것을 얻기 위하여 Mighty Bomb Jack에서 제시하는 불타오르는 고리와 같은 것을 점프해서 통과해야만 하는 경우에 자신의 성취에 관하여 좋은 느낌을 받을 수 있도록 할 것을 제안한다.

## 링크

이 게임에 대한 대부분의 자원은 일본어로 되어 있다. 이 게임에 관하여 내가 알고 있는 최고의 사이트는 [Mighty Bomb Jack 워크스루 페이지\(구글 번역\)](#)에서 찾을 수 있다. [피라미드의 맵](#)을 반드시 확인하도록 한다. 게임의 엔딩 및 이를 얻는 방법에 관한 자세한 사항은 이 게임에 대한 옛 Nintendo Game Counselor의 핸드북을 요약하는 [GameFAQ 기사](#)에서 찾을 수 있다.

## 알고리즘으로 생성된 콘텐츠에 의존하는 게임들

## 9. Starflight

Fractal과 비슷한 세계를 생성.

개발: Binary Systems

### 포함 이유

Starflight는 두 장의 플로피 디스크로 제공되지만, 그래도 대부분이 각자 자신의 독자적인 맵과 수확할 미네랄을 가지는 행성을 몇 개씩 갖는 수천 개의 스타 시스템을 포함하고 있다. 이것은 모두 해쉬와 프랙탈 맵 생성 체계를 주의 깊게 사용하여 이루어졌다. 자신의 맵에 이것과 비슷한 기법을 사용하는 또 다른 게임으로는 Elite가 있다.

### 게임 소개

Starflight는 단지 거대한 게임 세계를 만드는 것뿐만 아니라 이러한 세계를 흥미로운 것으로 만드는 점에서 개발자들에게 많은 교훈을 준다. 이 게임에서는 알고리즘에 의해 생성된 콘텐츠에 대하여 내가 "채굴(mining)" 접근법이라고 하는 것을 사용하며, 이는 Rogue와 비슷한 게임에서도 사용한다. 이것은 플레이어가 탐험하는 무작위화된 구역은 임의의 부를 위해 이용되며, 고갈이 될 수도 있기 때문에 플레이어는 더 많은 부를 얻기 위하여 자신의 영역을 확장해야 한다는 개념이다.

이 게임에서 각각의 개별 세계는 대체로 해당 세계에 포함된 미네랄 때문에 흥미를 끄는 경향이 있다. 기본적인 진전 메카닉은 행성을 찾고, 정찰선을 이용하여 행성의 표면에 착륙하고, 채굴 대상을 찾기 위해 정밀 조사를 한 다음, 더 나은 장비를 구하기 위해 채굴한 것을 팔기 위해 기지로 되돌아가는 것이다. 일부 행성은 기상 조건이 좋지 않고, 일부 행성은 (수집하고, 저장하고, 판매할 수 있는) 위험한 야생 생물이 있고, 일부 행성은 표면의 조건이 위험하지만, 플레이어가 원하지 않을 경우에는 대개 플레이어가 신경 쓸 필요가 없는 생명체가 플레이어에게 방해로 놓지 않고 기후도 보통인 행성도 충분히 있다.



많은 여러 가지 종류의 게임에 진전 루프가 존재하며, 진전 루프에는 많은 생각이 투입되지 않은 경우가 많다. 예를 들어 아케이드 게임에서 사용되는 고전적인 진전 루프는 플레이어가 순전히 실습을 통하여 발전한다는 것이다. 플레이어는 게임에서 후원하는 파워를 레벨 1에서 가졌던 것보다 레벨 50에서 더 많이 가지는 것이 아니다. 원래 RPG에서 생겼지만 그 후로 RPG를 넘어 훨씬 널리 퍼진 또 다른 일반적인 루프에서 플레이어는 몬스터를 물리쳐서 경험치를 얻고, 더 나은 장비를 구하기 위해 몬스터로부터金を 얻는 것을 통하여 향상된다.

돈을 얻는 것은 플레이어의 배를 향상시키는 수단이기 때문에 Starflight의 루프는 재무적이다. 그리고 돈은 주로 알고리즘으로 만들어진 수천 개의 행성을 모두 탐험하여 얻는다.

### 설계 교훈

무작위화된 게임이나 애매한 알고리즘을 사용하여 콘텐츠를 생성하는 게임은 자신의 진전 루프에 채굴 메타포어를 자주 사용한다. 세계는 위험이나 보상의 배치에 멋이나 이유를 많이 주지 않고 컴퓨터에 의해 자동으로 생성되며, 플레이어는 자신의 상태를 개선하기 위하여 자신이 이용할 수 있는 것을 이용하기 위해 이 세계에 뛰어 든다. 비교적 안전한 구역의 보상이 고갈됨에 따라, 플레이어는 계속해서 진전하기 위하여 더 위험한 구역으로 가지 않

을 수 없게 된다.

물론 이미 만들어진 구역을 가지는 정적인 탐험 게임도 이와 비슷하다. 차이점은 알고리즘으로 만들어진 게임은 훨씬 더 많은 구역을 가질 수 있기 때문에, 전투 또는 퍼즐 해결보다는 탐험이 보상의 획득에 대한 난관의 더 큰 부분이 될 수 있다는 것이다. 일부 동일한 개발자에 의해 동시 발생적으로 개발된 것은 아니지만 이 일반적인 플레이 스타일을 사용하는 다른 게임으로는 Starflight II와 Star Control II가 있다. 특히 Star Control II가 흥미로운데, 그 이유는 이 게임이 플레이어가 효율적으로 탐험해야 한다는 것을 의미하는 엄격한 시간 제한을 가지고 있고 어떤 행성에 더 나은 자원이 있는지에 관하여 재치 있는 실마리를 제공하기 때문이다.

## 링크

한동안 운영되었던 팬이 만든 후속작인 게임의 제작자들을 지원하는 [Starflight III: Mysteries of the Universe](#)가 있다.

Star Control II는 [The Ur-Quan Masters](#)라는 제목의 팬이 만든 개작물이 있다.

## 10. NetHack

아이템 발견, 임의적인 생성, 임의적인 아이템

발표, 개발, 설계: NetHack Dev Team

### 포함 이유

NetHack은 끝없이 놀라움을 줄 수 있는 것처럼 보이는 게임이다. 나 자신도 이 게임을 10년 넘게 해왔지만, 지금도 가끔 이전에 알지 못했던 점을 보게 된다.

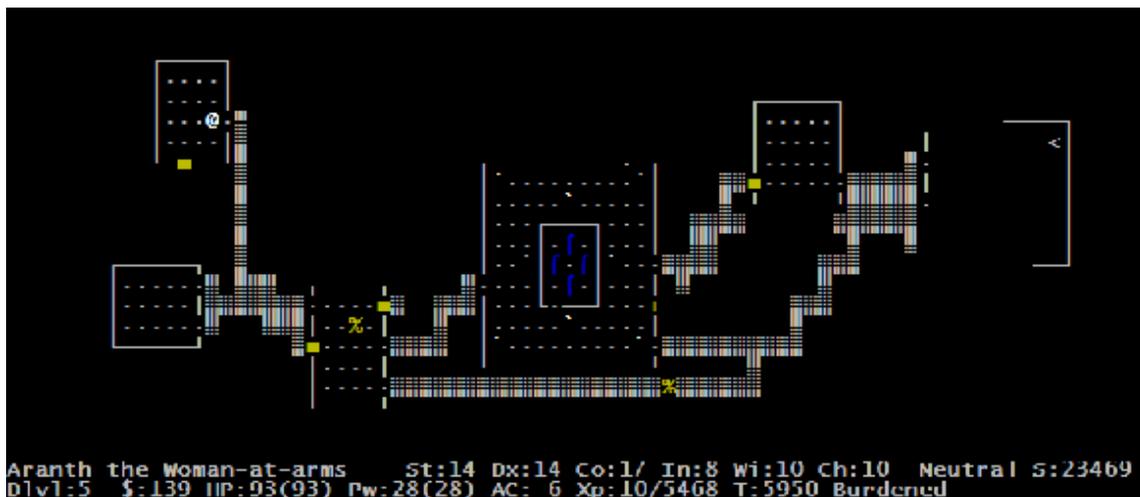
### 게임 소개

이 게임은 Rogue의 방대한 난이도를 가져오면서 난이도를 낮추기보다는 난이도를 이동시킨다. 난이도를 '임의적인 악운'이라는 열에서 '배워야 할 사항'이라는 열로 이동시키는 것이다. 많은 사람들이 난이도에 관해 불평을 하지만, 운보다는 지식이 더 신뢰성이 높기 때문에 가장 숙련된 해커들은 결국 이 게임이 너무 쉽다는 것을 알게 된다. 이렇게 된 한 가지 이유

는 이 게임이 세계에서 가장 확고한 게임 FAQ 작성자들 중 일부의 주제였기 때문이고, 반드시 말을 해야 할 사항이기도 한 또 다른 이유는 이 게임의 소스 코드가 모든 사람들이 볼 수 있도록 개방되어 있기 때문이다.

NetHack에는 임의적인 던전이 포함되어 있기 때문에, 단지 플레이어를 자신의 목적에 더 가깝게 가도록 만들기 때문만이 아니라 일단 어떤 레벨이 완전히 조사되면 찾아야 할 보물이 남아 있지 않기 때문에 새로운 던전 레벨을 찾아야만 하는 채굴 비슷한 요소가 게임에 여전히 존재한다. 그 전신인 Rogue에서는 이것을 극단적으로 취했기 때문에, 이 임의적인 아이템에는 찾아야 할 항목만이 있었으며, 그 중 일부는 생존에 필요한 식량이었다. 여기서는 식량에 대한 부담이 조금 줄어들었지만(식량이 흔한 경향이 있고, 죽은 몬스터를 먹을 수 있으며, 플레이어는 굶주림으로부터 약해졌을 때 도움을 요청할 수 있다), 던전 층에서 보물을 발견하는 일은 여전히 매우 중요한 일이다.

그 외에도 비직관적인 측면에서 몬스터 생성은 던전 레벨을 엄격하게 따르지 않는다. 실제로 처음 몇 개 레벨을 달성하고 플레이어의 히트 포인트가 위험 구역을 벗어난 후에는 또 다른 경험 레벨을 얻는 것보다는 훌륭한 아이템을 찾는 것이 플레이어에게 있어서 거의 언제나 더 이익이 될 것이다. 고차원의 마법이 적용된 갑주, 가속 부츠 또는 마법 방어용 망토는 히트 기회와 그 밖의 사소한 이익 레벨이 제공하는 추가적인 히트 포인트보다 더 큰 이익을 준다. 또한 비직관적인 관점에서 볼 때 생성된 몬스터의 최대 난이도는 던전 레벨과 플레이어 레벨 사이의 평균이기 때문에, 레벨을 얻기 위한 활동은 실제로 몬스터가 그에 대한 반응으로 더 강해지도록 만든다.



모든 임의적인 던전 게임은 어느 정도는 굴착에 이르게 된다. 플레이어가 던전 레벨을 떠난 후에는 해당 던전 레벨을 잇는 Angband와 같은 게임에서, 동일한 레벨 주변에 머무르면서 전리품을 쌓아 올리기 위하여 같은 레벨을 몇 번이고 재생성하는 관행이 너무 만연해서 '스

커밍(scumming: 찌끼를 걷어내기)'이라는 이름이 생겼다. 이 이름은 많은 수의 재미없는 레벨을 생성하다 보면 결국에는 멋진 아이템이 있는 레벨이 위쪽으로 올라오게 될 것이라는 개념에서 나온 것이다. Angband에서 스커밍은 성공적인 전략일 뿐만 아니라 게임에서 요구하는 것이기도 하다. 스커밍은 요구되는 것일 뿐 아니라 심지어는 스커밍을 중심으로 설계까지 되었다. 그래서 성공하는 플레이어들은 능력 점수를 최대한 활용할 때까지 생성된 버섯이 영구적인 보너스를 스탯에 부여하는 레벨에서 머무르는 경우가 많다.

NetHack의 가장 미스터리한 요소는 처음에 대부분의 마법 아이템의 정체를 애매하게 만드는 아이템 식별 시스템이다. 플레이어는 가속 부츠가 놓여져 있는 것을 발견할 수 있지만, 이것이 무엇인지를 알지 못한다면 신지 않을 수도 있다. 또는 알지 못하는 부츠를 신을 경우, 이 부츠가 실제로는 서투르게 만들어진 부츠처럼 나쁜 유형의 것이고 저주가 걸려 있어서 쉽게 벗을 수 없는 것으로 밝혀질 수 있다. 이것은 Rogue에 사용된 시스템과 비슷하지만, 둘 사이에는 큰 차이가 있다. Rogue에서 대부분의 아이템의 기능은 (그 자체가 임의적인 아이템으로 시작하는) 식별을 위한 두루마리를 읽거나 낭비되는 경우가 많고 심지어는 위험할 수도 있는 사용에 의해서만 파악할 수 있다. NetHack은 객체가 무엇을 하는 것인지를 이해할 수 있는 방법을 훨씬 더 많이 제공하며, 이 모든 것을 안 플레이어는 그 결과로 게임이 훨씬 더 쉬워졌다는 것을 알게 된다.

## 설계 교훈

여기에서는 두 가지 교훈을 제시한다. 첫 번째 교훈은 임의적인 게임이 아이템 식별 시스템의 도움을 받는 경향이 있지만, 이러한 게임은 이러한 시스템을 중심으로 설계되어야 한다는 것이다. 객체가 임의적으로 선택되지 않는다면, 플레이어는 앞서 한 플레이에서 이들의 유형이 무엇인지를 기억할 수 있다. 하지만 이 때문에 게임은 더욱 어려워진다. 알려지지 않은 임의적인 객체를 포함시키기로 결정한다면, 여기에 Rogue, NetHack 및 Diablo에 해당하는 세 가지 기법이 있다.

Rogue: 알려지지 않은 아이템은 사용될 수 있으며, 큰 위험이나 자원 사용 없이는 알아내기 어렵다. 어떤 나쁜 아이템도 사용되었을 때 게임을 직접적으로 끝내지는 않을 것이지만, 이러한 아이템이 많으면 몬스터의 공격에 의한 간접적인 죽음을 일으킬 수 있다.

NetHack: 플레이어가 특별한 기법을 발견했다면 아이템을 찾기가 더 쉽지만, 그렇지 않을 경우에는 Rogue의 동작 방식으로 후퇴한다.

Diablo: 알려지지 않은 아이템을 사용할 수는 있지만, 그 특별한 기능은 식별될 때까지는 활성화되지 않는다. 이것은 단지 플레이어가 때때로 타운으로 되돌아 가게 만드는 방법인

경향이 있다.

두 번째 교훈은 좀 더 애매하다. 확인 게임이 효과를 보게 만들기 위해서는, 확인 없이 사용하는 데 위험을 주기 위하여 좋은 아이템 뿐만 아니라 나쁜 아이템도 있어야 한다. 임의 항목의 모든 종류에는 그 안에 최소한 하나의 나쁜 아이템이 있어야 한다. 하지만 플레이어가 모든 나쁜 아이템이 무엇인지를 안 이후에는 임의로 아이템을 착용해보지 않는 데 대한 인센티브가 크게 줄어든다. 해당 종류의 모든 유형에서 발견된 모든 나쁜 항목은 나머지를 훨씬 더 쉽게 이해할 수 있게 만든다. 이것은 매우 긴 게임인 NetHack에서 아이템 식별 시스템은 초기 단계에서만 중요한 경향이 있으며, 부분적으로는 이 때문에 일단 대부분의 아이템이 발견된 이후에는 게임이 훨씬 쉬워진다는 것을 의미한다.

## 링크

[NetHack Wiki](#)는 이 게임에 관해서 배울 수 있는 거대한 자원이다. 또한 유즈넷 그룹인 [rec.games.roguelike.NetHack](#)도 훌륭한 자원이다.

## 11. ToeJam & Earl

임의적인 객체, 식별 위험

개발: Johnson-Voorsanger Productions (ToeJam & Earl Productions)

설계: Greg Johnson

### 포함 이유

아이템 식별 게임 및 임의적인 수준에서 Rogue와 비슷하며, 이제는 플레이어들이 더 이상 그만큼 스스로 문제를 풀어내는 데 익숙하지 않다는 단순한 이유에서 이 게임을 플레이한다는 것은 이상한 경험이다.

### 게임 소개

Rogue와 NetHack은 대학생들이 대학생들을 위하여 만든 것이다. 반면에 ToeJam & Earl은 전형적인 콘솔 게이머, 그 중에서도 대체로 어린이들을 위해 만든 것이다. 이 게임은 임의적으로 생성되는 레벨, (일부 매우 나쁜 아이템을 포함하여) 식별해야 할 객체, 이어지지 않

는 특징, 그리고 다소 높은 난이도를 가진다. 이제는 이 게임이 시장에서 얼마나 인기가 있었는지를 살펴보자.

실제로... 그리고 결국 이 게임은 매우 인기가 좋았다! "게임 닥터(The Game Doctor)"라고 칭해지는 Bill Kunkel은 이 게임을 당시에 자신이 가장 좋아하는 것이라고 했다. 이것은 평판이 매우 좋았지만, 그 전이나 그 후나 출시된 다른 어떤 것과도 너무나 달랐기 때문에 후속작을 만들 때가 되었을 때 Sega는 제작자들을 비판하고는 처음 나온 것과는 완전히 다른 게임을 만들 것을 요구했다. 그 결과로 나온 게임은 결코 나쁜 것이 아니었지만, 이제 모든 사람이 동의하는 바이지만 훨씬 더 나은 그래픽과 실제 스토리라인을 가졌는데도 불구하고 첫 번째 것만 못하다는 것이다.

ToeJam & Earl의 미스터리한 요소는 레벨 배치와 프레젠타 배치에서 나온다. 레벨은 단지 임의적으로 배열된 것이 아니라(레벨을 임의적으로 배열하는 것은 게임을 무작위화시키는 상당히 얕은 방식이다), 때로는 통과를 위해 자원을 소비해야 하는 환경적 장애물과 지원을 가진다. 프레젠타는 Rogue의 두루마리처럼 동작한다. 하나를 사용하면 바로 효과가 나타나는 것이다.



ToeJam & Earl에는 일부 추가적인 미묘한 차이가 있다. 이 게임은 2인용으로 플레이할 수

있으며, 플레이어들이 서로 멀리 떨어지게 되는 경우에는 게임 화면이 분할되어 서로에게 속박되지 않고 자신이 원하는 곳을 탐험할 수 있다. 하지만 플레이어가 둘 다 같은 스크린에 있을 때 프레젠티가 열리면 프레젠티의 좋고 나쁨에 상관 없이 두 플레이어 모두에게 영향이 생긴다. 어떤 형태의 프레젠티를 열면 생명이 즉시 손실되며, 그 영향은 두 플레이어가 서로 가까이 있을 경우 추가 생명(Extra Life) 프레젠티와 같은 다른 프레젠티들처럼 두 플레이어 모두에게 발생할 것이다.

이 게임의 최선의 설계 선택 중의 하나는 랜더마이저(Randomizer) 프레젠티를 포함시켜 Rogue와 비슷한 설계의 결정 중의 하나를 개선한다는 것이다. 이 프레젠티를 열면 게임의 다른 모든 프레젠티들이 다시 뒤섞이는 것이다! 플레이어에게 있어서 이것은 막대한 후퇴이며, 이것이 진행되는 과정 중에 자기 자신을 무작위화시키기 때문에 적지 않은 영향이 생긴다. 하지만 일단 어떤 아이템이 알려지면 이것은 영원히 알려지고 다시 확인할 필요가 없다는 것은 사실이 아니기 때문에, 설계에서 식별 게임에 의존하여 이 큰 문제를 수정한다. 랜더마이저가 확인될 때까지 플레이어는 확인하는 프레젠티가 늘어남에 따라 주의를 더 높여야 하며, 줄여서는 안 된다.

## 설계 교훈

식별 게임은 계속 배울 것이 있을 때에만 흥미롭다. Rogue에서는 비교적 작은 던전을 만들어서 이 문제를 해결하기 때문에 승리한 게임에서도 모든 아이템이 나타날 가능성은 매우 낮다. ToeJam & Earl에서는 아이템이 다시 무작위화될 수 있는 기회를 제시한다. 하지만 플레이어가 어떤 실수를 저지른 결과로 뒤섞이는 일이 항상 일어나야 한다는 "이유만으로" 알려진 아이템을 단지 재설정하는 것은 공정하지 않다. 예를 들어 이후의 Xbox용 ToeJam & Earl III에서는 적들 중 하나가 공격을 하면 아이템이 뒤섞일 수 있다.

링크: 설계자 Greg Johnson과의 [인터뷰](#).

## 12. Dungeon Hack

D&D 공식 규칙의 지원과 함께 임의의 던전과 아이템 식별

개발: DreamForge Intertainment

### 포함 이유

이 게임은 또 다른 임의의 던전 게임이며 매우 애매하지만, 게임에 흥미로운 제한을 부여하는 D&D 라이선스를 사용하기 때문에 흥미롭다.

## 게임 소개

Dungeon Hack은 임의적인 던전 게임이며 무작위화된 아이템을 갖지만, 전통적으로 Rogue와 비슷한 게임으로 간주되지는 않는다. 이 게임은 도움이 되는 자동맵이 포함되어 있기는 하지만, 생성된 마법 스타일의 미로에서 설정된다. 또한 이 게임의 던전은 Rogue에 있는 던전에 비해 자유로운 형태가 훨씬 덜하다. 맞는 키로만 열 수 있는 잠겨진 문이 있기 때문에 경험을 훨씬 더 선형적으로 만든다. (다행히도 미로 생성 알고리즘 덕분에 각 레벨을 해결할 수는 있다.)

Dungeon Hack은 Dungeons & Dragons 컴퓨터 게임의 쿼플레이로의 복귀를 나타내는 것이었다(AD&D Dungeon Master's Guide의 초판의 뒤에 있는 임의 결정표 참조). 그 이후로 게임들은 해당 지역의 적에 대항하는 일행을 위해 특별한 이익 또는 불이익을 제공하기 위한 특별한 보물을 이용하여 섬세하게 계획하여 배치하는 쪽으로 점차 기울어졌다. Dungeons & Dragons의 골드 박스 게임들은 이 패턴을 따른다.



Dungeon Hack은 그 차이점과 Eye of the Beholder 스타일의 게임처럼 보이려고 하는 시도에도 불구하고 NetHack에서 어떤 아이디어를 가져온 것으로 보인다. NetHack의 던전이 가

지는 특징과 똑같이, 플레이 중에 그 기능을 발견해야 하는 다양한 장치들이 각 미로의 벽을 따라 놓여 있다. 아이템은 플레이 중에 찾아야 하고 그 외관은 기능과 일치해야 하지만, 대부분의 아이템은 Advanced Dungeons & Dragons Guides 제2판에 있는 전리품과 정확히 일치한다. Rogue와 비슷한 종류의 게임의 이상은 그저 어떤 오래된 것을 포함시키는 것이 아니라, 너무 큰 위험을 초래하지 않으면서 사용 중에 대상을 확인하는 데 대한 적절한 위험을 제공하는 아이템을 제공하는 것이다. 이 분야에서 Dungeon Hack은 부족하다.

하지만 나중에 놀라운 장소에서 재사용된 게임 요소가 하나 있다. Dungeon Hack에는 많은 인위적인 아이템이 포함되어 있다. 이 인위적인 아이템은 3개의 칼 대 거대 파충류보다는 하나의 링에 더 가까운 것이 되는 경향이 있는 "진짜" D&D 인공물처럼 동작하지는 않지만, 그럼에도 불구하고 특별한 힘을 가진 고유한 아이템이다. 하지만 이 아이템 각각은 다른 모든 아이템과 일치하고 동시에 착용되었을 때 상당한 보너스를 제공하는 세트의 일부이다. Diablo II의 제작자들이 메모하고 있었다고 생각하지 않을 수 없다.

또한 대부분의 임의적인 던전 게임과는 것으로서, 초기 생성 후에 추가되는 "돌아다니는" 몬스터가 없다. 각 레벨마다 초기 개체수가 있고 그것이 전부이다. 실제로 각 레벨이 정해진 수의 경험치만을 가지고 있다고 생각할 수 있을 것이다. 그리고 이 게임에서는 플레이어가 히트 포인트와 주문을 다시 얻기 위해서는 해당 구역에서 몬스터들을 모두 물리쳐야 하는 D&D 스타일의 휴식 미캐닉을 사용한다. 그리고 제한된 식량 체계가 탐험 시간이 아닌 치료와 주문을 제한하도록 하기 위하여 식량은 휴식 중에 소비된다.

## 설계 교훈

레벨 생성기는 항상 해결이 가능한 잠겨진 문에 대한 퍼즐을 만드는 방식에 있어서 매우 교묘하며, 임의 던전 게임에서는 충분히 사용되지 않는 특징인 특별 던전 구역을 만든다. 식량을 사용하는 방식 또한 시간이 아닌 휴식을 제한하는 혁신적인 것이다.

## 13. Diablo 시리즈

임의 던전, 적의 배치 및 보물, 멀티플레이어 탐험

개발: Blizzard Entertainment

선임 설계자: Erich Schaefer와 David Brevik

## 포함 이유

Diablo 시리즈는 세계에서 가장 인기 있는 상용의 임의적인 던전 게임이며, 10년이 지난 후에도 여전히 팬이 있다. 플레이어의 미스터리 수준은 NetHack보다는 덜하지만, 흥미로운 점이 있다.

## 게임 소개

Diablo의 임의적인 던전과 모습, 그리고 다시 증식하지 않는 몬스터는 다른 종류의 채굴 게임을 만든다. 여기서 논의한 다른 임의적인 던전 게임과는 달리, 플레이어에게 형편이 좋을 때 던전을 탐험하는 데 방해가 되는 것은 몬스터 뿐이다. 즉, 환경의 장애물이 방해물이 되는 경우는 많지 않다.

솔직히 말해서 Diablo 게임은 임의적인 던전 탐험 게임에 관한 많은 관측은 사항들을 축소한다. 전체적인 유형들이 외양은 비슷하지만 기능은 임의적이기 때문에 알려지지 않은 아이템은 존재하지 않는다. 대체 왜 이 게임에 식별이 있는지 궁금해하는 사람도 있을 것이다. 알려지지 않은 아이템을 사용할 수는 있지만, 이 알려지지 않은 아이템은 기본적인 영향을 갖기 때문이다. 그 결과는 플레이어가 대상을 확인하기 위하여 더 자주 기지로 되돌아가야 하는 것 뿐이다. 사용이나 실험을 통하여 아이템을 확인할 기회가 없으며, 알려지지 않은 물약이나 스크롤 타입도 없다(Rogue 게임 같은 스타일로는 없다).

개발자들을 옹호하는 입장에서 볼 때, 진정한 Rogue와 같은 객체 식별을 구현하기 위해서는 영구적인 죽음이 지나치게 많이 다시 로드되는 것을 피해야 하기 때문에 개발자들이 이렇게 했을 가능성 있지만, 이것은 많은 주류 플레이어가 즐기지 않는 변화이다. 그리고 플레이어로 붐비는 멀티플레이어 게임에서는 도움을 줄 다른 플레이어들이 있기 때문에 나쁜 마법 아이템을 사용하는 데 따른 페널티가 그렇게 나쁘지는 않다. 그리고 Rogue의 식량 체제를 없앤 것은 잘한 일이다. Rogue에서는 그렇게 나쁘지 않았지만, 이러한 게임의 구조에서 식량은 자리가 없다.



Diablo의 던전은 다른 임의적인 게임의 던전에 비해 비교적 간단하며, 바로 이 부분이 설명이 필요한 곳이라고 생각된다. 아키텍처에 있어서 Diablo의 던전은 실제로 매우 복잡해서, Rogue, NetHack, Dungeon Hack 또는 그 밖의 다른 많은 게임들보다 훨씬 더 복잡하다. 그리고 이러한 던전에는 탐험의 목표와 고유한 몬스터들 외에 몬스터의 방과 신전이 흩어져 있다. 하지만 이러한 사항 외에도 이들에게는 일종의 동일성이 있다. 레벨은 게임의 측면에서 몬스터와 전리품으로 정의된다. 임의적인 던전의 벽의 배치는 그다지 흥미롭지 않다. 결국 이 모든 게임에는 자동맵이 있기 때문에 길을 잃어버릴 위험은 없다. Diablo에서는 복도가 직선이든, 구부러진 곳이 있든, 갈림길이 있든, 실제로 방이든 간에 모두 그다지 흥미롭지 않다. 그것은 단지 공간일 뿐이다.

그것은 모두 공간일 뿐이며, 그 형태는 크게 문제가 되지 않는다. 플레이어가 포위되는 것을 피하기 위하여 이것을 고려하고 필요한 경우 달아날 경로를 개방된 상태로 유지해야 한다는 것은 사실이다. 하지만 Rogue의 설계에 있는 많은 단도를 버렸기 때문에 던전 자체의 취지는 줄어들었다. 진짜 Rogue 스타일의 게임에서 플레이어는 식량은 가장 적게 사용하고 위험은 가장 적게 겪으면서 가장 많은 전리품을 얻기 위하여 효율적으로 탐험해야 한다. Diablo는 Rogue의 이러한 측면을 포기한 유일한 임의적인 던전 게임이 결코 아니지만, 맵의 구조가 플레이어에게 미치는 실제 영향이 너무 작다면 왜 임의의 맵을 게임에 집어 넣는 수고를 하는지 가끔 궁금해진다.

## 설계 교훈

Diablo 게임을 플레이하는 데 따른 인센티브는 주로 임의적인 전리품을 찾는 데서 나온다.

탐험 행위에 대한 결과가 부족하기 때문에 이 게임의 던전은 전리품 사냥에 비해서는 재미가 조금 덜하다. 바닥에는 함정이 없고, 식량이 없으며, 임의로 나타나는 몬스터도 없다. 싱글플레이어 게임이라면 재미가 덜하겠지만, 멀티플레이어 게임에서는 이것이 더 낫다. 아마도 이런 이유에서 Diablo 시스템이 기본적으로 많은 MMORPG가 사용하는 템플릿이 된 것이다.

## 14. Barcode Battler

해쉬 기반의 캐릭터 생성

개발: Epoch

### 포함 이유

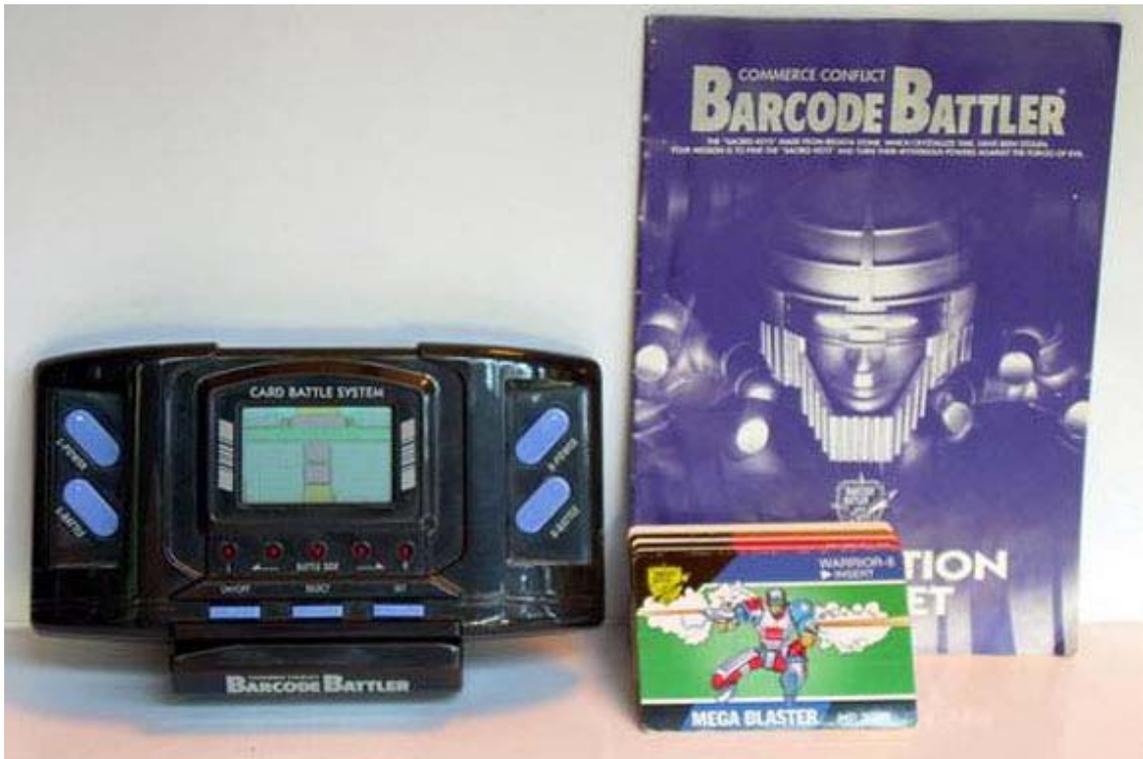
고유한 방식으로 몬스터 스탯을 생성하는 Pokemon 스타일의 게임. 이 게임에서는 UPC 스타일의 바코드에 포함된 데이터에 적용되는 해쉬 함수를 사용한다.

### 게임 소개

이것은 정말로 굉장한 아이디어이다.

이 게임은 Barcode Battler라고 하며, 한때 일본에서 큰 인기를 끌었다. 이 게임의 동작 방식은 각 플레이어가 보통의 UPC 타입 코드를 읽고, 플레이어의 집에 있는 모든 제품에서 코드를 선택하고, 기계에 부착된 바코드 스캐너에 집어넣는 것이다. 보통 소매 목적으로 사용되는 이 데이터는 기계에 의해 읽혀질 것이며, (해쉬 함수의 적용이나 의사 난수 발생기로 사용하는 것과 같은) 혼란화 과정을 거친 후에 "몬스터"에 대한 필수적인 통계를 정의하는 데 사용될 것이다. 그 다음에 몬스터는 과정은 같지만 어쩌면 다른 코드를 사용하여 생성된 다른 플레이어의 몬스터와 싸우게 될 수 있다.

이 결과는 몬스터 생성 메커니즘에는 진짜 임의성이 없기 때문에 동일한 코드로 스캐닝을 하면 항상 동일한 몬스터가 나오는 것이다. 그 결과는 게임의 캐릭터를 정의하는 데이터가 어떤 의미에서 이 캐릭터와 동일한 것처럼 "동물 친구" 같은 종류의 캐릭터에 대한 통계를 생성하기 위하여 물리적 바코드를 사용하면 코드와 시각적 표현에 일종의 공통의 동일성이 생긴다.



또한 이것은 코드 주위에 실세계의 Magic: the Gathering 스타일의 경제를 수립한다. 이야기는 Barcode Battler의 팬들이 훌륭한 몬스터를 만든 코드를 찾았을 때 많은 플레이어들이 자신의 사본을 구하기 위하여 쇄도하기 때문에 해당 코드의 제품이 매진되는 것으로 끝나는 쪽으로 흘러간다. 하지만 CCG와는 달리 Barcode Battler 시스템의 제작자들은 이러한 게임을 운영하기 위한 강력한 인센티브인, 자신의 시스템에 대한 어떠한 재무적 보상도 얻지 않았다.

그리고 카드 게임과는 달리, 인기 있는 제품이 가능한 최상의 몬스터를 만들지 않도록 할 좋은 방법이 없기 때문에, 희소성은 의미 있는 객체의 이용 가능성을 제한하는 데에서 나오는 것이 아니라 단지 강력한 특성이 나올 가능성을 줄이는 데서 나온다. 20 온스짜리 콜라병에서 슈퍼 메가 드래곤이 나올 수도 있지만, 설계자들은 이 드래곤에 대한 중-공간의 훨씬 더 작은 비율을 사용하여 이러한 상황이 발생하는 가능성을 줄일 수 있다.

### 설계 교훈

혼동 시스템은 이 모든 것의 중요한 부분이다. 어떤 바코드에서 좋은 몬스터가 만들어졌는지를 알기가 쉬웠다면, 플레이어들은 실제로 바코드를 스캐닝할 필요가 없거나 심지어는 손수 만든 바코드를 사용하여 자신의 몬스터를 만들어낼 필요도 없게 될 것이다. 대부분의 플레이어가 코드와 몬스터 사이의 관련성을 알지 못한다는 사실 때문에 과정이 미스터리해지

며, 그렇기 때문에 더욱 사실처럼 보인다.

비슷한 종류의 시스템을 사용하는 미국 게임인 Scannerz가 이 게임을 모방했다. 이 시스템을 사용하는 또 다른 시리즈는 바코드 대신에 콘솔의 디스크 트레이에 삽입된 보통의 CD에서 나온 데이터를 사용하는 Monster Rancher의 PlayStation 버전이다.

링크

이 게임에 관한 [위키피디아의 기사](#)가 도움이 된다.

이 장치의 영국 버전에 대한 [팬 사이트](#).

## 중요한 플레이 정보를 숨기는 게임

### 15. Street Fighter II

스스로 플레이하는 방법을 발견하기 위해 플레이어에게 의존하는 문서화되지 않은 동작 집합

개발: Capcom

계획: Akira Nishitani

#### 포함 이유

Street Fighter II는 일련의 파이팅 게임의 상징으로 선택되었으며, 이러한 유형의 최초의 공전의 히트를 친 작품이기 때문에 대부분의 게임이 이 게임처럼 끝을 낸다. (기술적으로 볼 때 최초의 Street Fighter도 명령어 기반의 특별한 동작을 가졌지만, 이 게임을 그렇게 좋아하는 사람은 이제 없다.)

#### 게임 소개

모든 파이팅 게임에서 능숙해지는 첫 단계는 동작을 익히는 것이다. 물론 요즈음에 대부분의 사람들은 FAQ를 읽을 뿐이지만, 이러한 FAQ는 어떻게 작성되는가? 예전에 대부분의 아

케이드 게임은 자신의 방식을 벗어나서 바로 제어판 위에 정보를 인쇄하여 게임을 제어하는 방법을 보여주었다. 바로 그 첫 번째 하우도켄(Haudoken)이 일어나게 하기 위해서는 어떤 일이 있어야 했는지를 상상해 보라. 특히 첫 번째 쇼류켄(Shoryuken)의 경우, 정확히 무엇이 직관적인 입력이 아닌가? 명령어 입력에 대한 직접적인 제어의 아이디어에서 벗어나는 것은 중대한 일이며, 많은 파이팅 게임에서 동작은 플레이어를 위해 미리 인쇄되지 않는다.

특별한 동작을 숨기는 데에는 어떤 목적이 있는 것인가? 이것은 마스터하는 것을 더 힘들게 만들면서 초기의 접근 가능성을 증가시킨다. 마스터하는 것은 어려운 것으로 가정되기 때문에 이것은 파이팅 게임에게 있어서 반드시 불리한 것이 아니다. 최소한 타이틀의 수명의 초기에서 플레이어는 게임의 옵션을 완전히 이해하지 않고 수행해야 하기 때문에 이것은 시합을 더욱 흥미롭게 만드는 데 도움이 된다.

그리고 실제로 파이팅 게임은 내용이 풍부하지 않다는 것이 매우 잘 지켜지는 비밀이 아니라고 생각되기 때문에 파이팅 게임을 마스터하는 것은 어려워야 한다. 탐험할 맵이 없고, 찾아야 할 파워 상승이 없고, 왼쪽에서 오른쪽까지의 라인은 조정성을 위한 많은 공간을 주지 않고, 비밀은 특정한 방식으로 플레이하는 것으로 제한되거나(추가 엔딩을 위해 시합에서 지지 말 것) 제어기 코드를 입력하는 것에서("서브-제로 승리... 사망") 제한되는 경향이 있다. 파이팅 게임의 플레이 모델은 설계자가 중요하다고 보는 게임 요소에 집중할 수 있도록 하기 위해 의도적으로 단순화되지만, 간단하기 때문에 게임을 탐구할 수 있는 것으로 만들기 위해서는 다른 수단이 추가되어야 한다. 이 경우에 탐구되는 것은 캐릭터의 능력과 캐릭터들이 서로를 비교하는 방법이다.



그렇다면 동작에 관한 소식은 처음에 어떻게 나오는가? 요즈음에는 게임의 모든 흥미로운 사항에 대한 의견상 완전한 카탈로그 같은 것과 전략 가이드가 있기 때문에 플레이어들은 자연스럽게 완전한 동작 집합을 갖게 된다. 파이팅 게임의 홈 버전에는 매뉴얼에 선택된 동작이 포함되는 경우가 많을 것이며, Soulcalibur 게임과 같은 일부 게임에는 모든 동작을 의도적으로 드러내는 훈련 모드가 있다. 그리고 요즈음에 대부분의 게임은 결국 이 모든 출처로부터 모든 정보를 담은 GameFAQs와 같은 사이트를 갖게 된다.

초기에는 이러한 옵션들을 이용할 수 없었지만, 일부 게임은 몇 개의 동작을 캐비닛 아트에 인쇄하곤 했다. 컴퓨터 플레이어는 가능한 모든 동작에 관한 완전한 지식을 갖고 있기 때문에, 사람들은 상대방이 어떻게 동작하는 지를 살펴 보고 최소한 어떤 종류의 동작을 이용할 수 있는지를 알아낼 수 있었다. 파이팅 게임은 조이스틱과 버튼의 동작을 이용하여 해석적인 태도로 동작 애니메이션 대전을 수행하여 발견 가능성에 도움을 주는 경우가 많기 때문에 동작의 애니메이션은 종종 동작을 수행하는 방법에 대한 실마리가 된다. Street Fighter II에는 디지털 조이스틱과 6개의 버튼을 이용하여 새로운 동작을 찾기 위해 검색해야 할 막대한 양의 제어기 입력이 있기 때문에 최소한 어느 정도의 발견 가능성이 중요하다.

## 설계 교훈

Street Fighter II와 같은 파이팅 게임은 나타나기 전에 중요하기는 했지만 핵심은 아니었던 비디오 게임 플레이의 요소에 집중하기 위하여 게임 세계를 단순화시킨다. 또한 이 게임은 더욱 강력한 능력에 대한 접근을 숨김으로써 게임에 신비로움을 더하고 플레이어가 이를 마스터하기 위해 투입해야 할 노력을 증가시킨다. 하지만 SFII보다 카리스마가 떨어지는 게임에서 이런 식으로 옵션에 대한 접근을 숨긴다면 해당 게임에 무엇이 생길지에 관하여 생각해 보는 것도 흥미로운 일이다.

## 링크

David Sirlin은 [자신의 블로그](#)에서 파이팅 게임 설계에 대한 흥미로운 글을 제시하고 있다.

## 16. Pokemon

숨겨진 스탯.

개발: Game Freak

설계는 여러 명이 했으며, Satoshi Tajiri가 주요 인물 중의 한 명으로 보인다. 최근의 DS 게임들에는 Shigeru Ohmori가 게임 설계 리더라고 나와 있다.

## 포함 이유

전문가의 전략을 지배하는 캐주얼 플레이어 사이에서의 Pokemon 게임의 잘 알려지지 않은 요소는 각 몬스터의 전투 이력을 예상되지 않은 방향으로 이끄는 감추어진 변수의 집합이며 레벨을 획득할 때 스탯 성장에 큰 영향을 준다.

## 게임 소개

Pokemon은 시리즈를 5차례 구현하면서 그들 사이의 트레이드, 각 게임마다 추가된 새로운 기능, 그리고 이들과 상호 작용하는 수십 개의 방법으로 너무나 복잡해졌기 때문에 그 자체만으로 어떤 부류에 속해 있는 것이다. Pokemon은 무심한 관찰자가 의심할 수 있는 것보다 초등 학생들을 훨씬 넘어 서는 팬을 요구한다. 일반적으로 이 프랜차이즈를 조소하는 것이 유행이며, 많은 경우에 정당화되고 있지만, 게임은 비난을 넘어서는 것이다. 그리고 Pokemon의 매력의 상당 부분은 게임을 가득 채우고 있는 미스터리에서 나온다. 이 게임을 둘러 싸고 있는 지식은 당시의 Street Fighter II와 Mortal Kombat의 특별한 동작을 아는 현재의 13세 미만의 버전이다.

나는 Pokemon 시리즈에 관한 것만으로도 이 전체 기사를 쉽게 채울 수 있었다. 이들은 완전히 이해하기 위해서는 메가바이트 크기의 FAQ를 쏟아 부어야 하는 게임들이다. 비슷한 복잡성을 가진다고 생각할 수 있는 유일한 다른 게임은 NetHack이다. 사육, 진화 경로, 시간 및 일의 차이, 베리, 액세서리, 트레이딩, 일상적인 사건, 콘테스트, 일회성의 조우, Nintendo만의 몬스터와 그 밖의 많은 사항들 사이에서, 이것은 심각한 노력 없이 이해하기 힘들다. 이것은 거의 캐릭터를 만들기 위해 설계된 것처럼 보인다. 이 모든 것에도 불구하고 여전히 일반적으로 잘 알려져 있지 않은 한 가지 요소가 있다. 이것은 Pokemon 아마추어와 슈퍼 아마추어 사이를 구분하는 선이다. 즉, EV라고도 알려져 있는 노력치(Effort Value)를 아느냐 이다.

노력치란 팬이 지은 이름이다. 공식적인 이름은 알려져 있지 않다. 노력치는 이 게임에 책임이 있는 어떤 회사도 아직까지 공식적으로 인정하지 않은 Pokemon의 한 요소이다. 이와 관련된 모든 게임 기능은 부정확한 용어로 말해진다. 우리가 어떤 하드 넘버를 EV에 넣을 수 있는 것은 세이브를 해킹하고 다른 추가적인 게임 수단을 사용하는 사람들의 작업을 통해서만 가능하다.



EV는 흑백의 Game Boy 원형에서 캡처된 것 이후로 모든 포켓몬에 연결된 숨겨진 스탯의 집합이다. HP, 공격, 방어, 특수 공격, 특수 방어 및 속도의 기본 스탯 외에도 각각은 숨겨진 EV 값을 갖는다. EV는 경험 점수처럼 포켓몬이 싸울 때 올라가지만, 싸운 종류에 따라 달라진다. 몬스터의 레벨과는 관련이 없다. 종류만이 관련이 있다. Bidoof와 싸우면 플레이어의 군대는 HP EV를 얻지만, Golduck과 싸우면 특수 공격 EV를 얻게 된다. 많은 낮은 레벨의 몬스터와 싸우면 조금 도움이 되겠지만, 각 스탯에 대해 작성할 수 있고 전체적으로 처리할 수 있는 EV의 양에 제한이 있으므로 주어진 포켓몬을 완전히 최적화시키려면 플레이어는 관리된 전투 프로그램에 따라 해당 포켓몬을 인도해야 한다.

EV 점수는 전투 승리 메시지에서 보고되지 않으며 포켓몬의 스탯 디스플레이에도 나타나지 않지만, 보이지 않는 곳에서 포켓몬 전투로 축적된다. 어떤 몬스터의 레벨이 상승하면 기본 스탯의 상승 및 임의적인 인자 외에도 자신의 대전 스탯의 추가 점수에 대한 교환으로 EV 점수의 일부를 받는다. 이런 방식으로 어떤 포켓몬의 전투 이력은 단지 레벨을 얻는 것 이

외의 방식으로 그 발전에 영향을 주게 된다.

포켓몬에게 "자유" 경험을 부여하는 방법이 있다. 포켓몬에게 진기한 캔디(Rare Candy)를 주면 자유 경험 레벨을 주는 것이다. 포켓몬이 스탯에 대한 자신의 레벨 보너스를 얻는 중에 EV를 축적하기 위하여 어떤 식으로든 싸우지 않았다면 스탯 상승은 그렇게 될 수 있는 것보다 작을 것이기 때문에 이러한 경험 점수는 그저 빈 칼로리일 뿐이다. 이러한 보너스는 발전 레벨 100 이상으로 올라가며, EV 훈련을 받은 포켓몬은 되는 대로 만들어진 것보다 항상 우세한 결과가 나온다.

실제 포켓몬의 방식에서는 시스템의 동작이 게임에서는 결코 설명되지 않지만 EV를 만드는 다른 방법들이 존재한다. 포켓몬에게 비타민을 주면 특정 EV 값이 조금 상승할 것이다. Diamond와 Pearl에서 포켓몬에게 Macho Brace를 갖추면 두 배의 EV를 얻을 수 있게 될 것이며, 이는 상당한 보너스이다. 또한 포켓몬이 "포케러스(Pokerus)"에 걸려 약해지는 만드는, 임의적으로 발생하는 매우 드문 영향이 게임에 존재한다. 아프게 되면 벌어들인 EV가 두 배가 되고, 보너스는 Macho Brace의 보너스와 함께 축적되고, [오염된 피\(Corrupted Blood\)](#) 스타일이라는 우연한 접촉을 통하여 플레이어의 컬렉션 중에 퍼질 수 있기 때문에 아프게 되는 것은 실제로는 매우 좋은 일이다.

## 설계 교훈

EV는 게임의 가장 미스터리한 요소일 수 있다... 이것은 너무나 많은 미스터리한 요소를 가질 수도 있다. 이에 관한 한 가지 흥미로운 사실은 이것에 처벌의 요소가 있다는 것이다. 이것은 많은 진기한 캔디를 사용하는 플레이어가 이에 대한 대가를 치르게 하기 위해 존재하는 것처럼 보인다. 하지만 이것은 포켓몬의 전투 이력이 경험 점수 이외의 방식으로 진전하게 만들어서 숫자가 붙어있는 픽셀의 집합보다 조금 더 나은 것처럼 보이도록 돕는 역할도 한다.

## 링크

Pokemon Diamond에 대한 GameFAQs [EV FAQ](#).

## 17. Gauntlet II

비밀의 방

개발: Atari Games

설계: Ed Logg

## 포함 이유

Gauntlet II는 더욱 다양한 플레이, 전략에 대한 더 많은 기회, 그리고 일반적으로 내용이 더 많기 때문에 전체적으로 원작보다 더 좋은 게임이다. 이 게임에서 가장 귀찮은 특징은 가끔 레벨을 완료한 이후에 들어가며 심지어는 성공할 경우 플레이어가 T 셔츠 콘테스트에 들어가도록 허용하기도 하는 잘 알려지지 않은 특별한 레벨인 비밀의 방이 될 수 있다. 하지만 이 방을 찾는 방법은 어느 누구도 잘 모른다.

## 게임 소개

Gauntlet 원작에는 분명히 탐사 영역이 충분히 있었지만, 일단 플레이어가 모든 맵에 익숙해지면 맵 각각에서 자신의 전략을 고안하는 임무로 정착할 수 있다. 하지만 후속작은 원작 이상으로 정말로 이상하고 놀라운 특징을 갖고 있다.

너무 자주 나타나는 일이지만, 어떤 레벨이 완료될 때마다 다음 레벨로 가기 전에 "비밀의 방 찾기(To find a secret room:)"라는 메시지가 어떤 지시와 함께 나타날 것이다. 전형적인 지시로는 "맞지 말 것(Don't get shot)", "속지 말 것(Don't be fooled)", 그리고 "적극적일 것(Be pushy)" 등이 있다. 이러한 지시가 방에 들어가는 데 필요한 요구 사항과 실제로는 관련이 없지 않다고 생각되지만, 나타난 메시지가 당시에는 반드시 좋은 것으로 보이지는 않는다.



알려져 있는 것은 올바른 레벨에서 올바른 트릭을 사용하면 해당 트릭을 수행한 플레이어가 비밀의 방(SECRET ROOM)이라는 제목이 불길하게 붙어 있는 특별한 레벨에 홀로 모습을 드러낼 것이라는 것이다. 나타날 수 있는 방에는 여러 개가 있으며, 매우 엄격한 한계가 적용되는 보물의 방처럼 시간이 설정된다. 여기에는 여러 개의 영구 능력 물약이 있을 수 있으며, 시간 내에 완료하기가 매우 어렵다. 시간 내에 출구에 도달하지 못해도 플레이어에게 부정적인 결과가 생기는 것은 아니지만, 플레이어가 성공하여 시간 내에 완료하면 특별한 코드, 플레이어가 글을 보내고 콘테스트에 참여할 수 있는 Atari Games의 주소, 그리고 무료 T-셔츠가 제시된다.

T-셔츠 제공은 오래 전에 기간이 만료되었지만, 이 방의 유혹은 계속된다. 능력을 향상시키는 물약은 모든 캐릭터를 강력한 존재로 만들 수 있고, 플레이어는 멀티플레이어 게임에서조차 이에 대한 경쟁을 하지 않기 때문에 플레이어가 해야 하는 것은 단지 물약에 도달하는 것 뿐이다. 하지만 그것 이상으로 비밀의 방은 특별하다. Gauntlet II를 여러 차례 플레이하면서도 첫 번째 비밀의 방을 보지 못하는 경우가 있을 수 있다.

요즈음에 게임이 하는 것에 이르면 남아 있는 미스터리는 별로 없다. Aeria가 죽는 것을 막을 수 있는 방법은 없다는 것을 알고 있고, Bubble Bobble의 "해피 엔드"에 도달하는 방법을 알고 있으며, 모든 워프 존이 어디에 있는지 알고 있는 것이다. 하지만 여전히 비밀이

방에 들어가는 방법은 모르고 있다. 누군가가 소스 코드를 디스어셈블하여 알아낼 수 있다는 생각이 든다. 아마 언젠가는 그렇게 될 것이다. 오래지 않아 그렇게 되기를 바란다.

## 설계 교훈

오늘날에도 (내 생각에는 게임 개발에 참여한 사람들 자신을 제외한) 어느 누구도 Gauntlet II의 비밀의 방에 들어가는 방법을 잘 알지 못한다. 이것은 그렇지 않았다면 간단했을 아케이드 게임에 상위 목적을 추가하는 재미있는 실험이다. Atari는 이를 이용한 게임을 여러 차례 만들었다. 즉, Toobin'과 Vindicators도 T-셔츠 콘테스트 진입 화면이 있다.

## 18. King's Bounty와 Heroes of Might & Magic

맵 짜맞추기.

개발: New World Computing/3DO

설계: Jon Van Caneghem

## 포함 이유

Heroes of Might & Magic 시리즈에 영감을 준 게임은 HoM&M 때까지 자신보다 더욱 유명한 자매품들과는 관련성이 별로 없는 King's Bounty라고 하는 New World Computing에서 나온 비교적 잘 알려지지 않은 게임이다. 두 게임 모두 탐험, 경제, 약간의 기회, 마법 및 전략, 차례 기반의 전투 등이 관련된 게임이지만, 또한 원작 및 일부 후속작 양쪽에 어떤 발견을 하는 관찰은 오버 퍼즐과 훌륭한 기억력도 관련이 있으며, 아마도 기억력은 플레이어의 모든 기술 중에서 가장 중요한 기술일 것이다.

## 게임 소개

Might & Magic 게임은 결말에서 플레이어에게 흥미로운 도전 과제를 주는 일이 가끔 있다. Might & Magic II는 시간 제한이 있는 무작위화된 암호문으로 마치며, World of Xeen의 퀘스트를 마친 이후에 플레이어는 십자말 풀이이기도 한 던전을 탐험하기 위하여 되돌아갈 수 있다.

King's Bounty는 Might & Magic I과 II의 시대에 New World Computing이 만든 또 다른 게

임이다. 원래 이것은 M&M 게임과 관련이 별로 없었지만, 제작자들이 이 프랜차이즈를 다양화시키기로 결정했을 때 파생적으로 발생한 Heroes 시리즈에 대한 기본으로 사용되었다. 그리고 HoM&M의 일부 버전에 포함되기도 한 King's Bounty에서 가장 좋은 점은 오버 퍼즐이다. King's Bounty에서 퍼즐이 어떻게 동작하는지를 가장 잘 알고 있기 때문에 앞으로 이 게임에 대해 설명하고자 한다.

이 게임의 목적은 왕국의 왕이 도둑맞은 흠을 찾고 이것을 왕에게 돌려주어서 흠을 도둑맞았기 때문에 생긴 반란을 종식시키는 것이다. 흠을 훔친 사악한 드래곤 보스가 왕국의 어딘가에 흠을 감추었고, 이것을 찾는 것이 플레이어의 임무이다. 플레이어는 그 위치를 보여주는 맵을 얻고, 맵을 5행 5열의 조각으로 나누고, 이 조각들을 자신의 24명의 부관에게 나누어 주었으며, 자신은 중앙의 조각을 보관한다.

왕국은 거대한 네 개의 대륙으로 구성되어 있으며, 각 대륙에는 싸워야 할 부관들의 수가 충분하다. 각 부관은 해당 보스의 개성을 주제로 한 군대가 있는 해당 대륙의 수 많은 성 중의 하나에서 거주한다. 플레이어의 역할은 대륙의 많은 마을과 은신처에서 용병을 모집하고, 주기적인 수입을 제공하기 위하여 성을 점수한 다음, 밖으로 나가서 악당을 잡는 것이다. 일부 패배한 악당은 플레이어의 발전에 도움이 되는 인공물을 제공하지만, 가장 중요한 것은 게임에서 제거된 각 악당이 맵의 조각을 제공한다는 것이다.



빅 보스는 흠을 숨길 때 자신이 찾을 수 있는 가장 깊은 던전이나 가장 강한 요새에 숨기는 표준 마왕 규약을 따르지 않았다. 대신에 빅 보스는 네 개 대륙 중 한 곳에 있는 임의의 구역에 흠을 보관했다. 흠은 잘 숨겨져 있기 때문에 플레이어는 걸어 다니면서 흠을 보지 못한다. 흠은 마지막 대륙의 접근 가능성이 가장 낮은 구석에 숨겨질 수도 있고 플레이어가 게임 중에 흠이 숨겨진 곳을 백 번은 걸어서 지나갈 가능성도 있다.

플레이어가 흠을 찾을 수 있는 유일한 방법은 흠이 숨겨진 구역을 찾는 것이다. 하지만 그저 모든 곳을 시험하기에는 공간이 너무 많을 뿐만 아니라 게임에 대한 전체적인 시간 제한도 있다. 플레이어에게 날짜가 부족하면 퀘스트는 실패로 끝이 나며, (지금까지 가장 강력한 악당인 빅 보스 Arech Dragonbreath 자신이 가지고 있는) 맵의 중앙 조각 없이 찾으려면 찾는 데 1일이 아닌 10일이 걸린다.

플레이어가 정확한 장소에서 찾으면 흠을 발견하고 즉시 게임에서 승리할 것이다. 이것은 즉각적인 승리일 뿐만 아니라 이기기 위한 유일한 방법이기도 하다. 플레이어가 맵의 모든 조각을 모으지만 게임의 대륙들 중 하나에 있는 장소에 그림을 일치시킬 수 없을 때에도 플레이어는 지게 된다. 하지만 플레이어가 단지 가장 약한 악당들로부터 맵의 조각을 얻은 후에 흠이 숨겨져 있는 곳을 발견할 수 있다면, 피를 많이 흘리지 않고 전체 경쟁에서 이길 수 있다. 전투, 경제 게임, 탐험, 군대를 모집하고 주문을 모집하는 일 등은 모두 수단으로 서만 존재한다.

King's Bounty에 대한 실제 게임은 그림 맞추기 퍼즐이다. 다른 모든 것은 단지 그것을 쉽게 만들 뿐이다. 하지만 최소한 하나의 맵 조각을 발견하지 않고 정확한 장소를 발견하는 것은 사실상 불가능한 일이며, 범위를 유용하게 줄이기 위해서는 몇 개의 조각이 필요하다. 또한 비교적 어려운 악당은 중앙에 맵 조각을 가지고 있는 악당이다. 따라서 이 게임은 정보를 수집하는 것에 관한 게임이다. 또는 다른 방식으로 설명하지만, 이것은 미스터리를 해결하는 것이다. 이것은 어떠한가?

## 설계 교훈

King's Bounty는 게임의 모든 전투 미캐닉이 일종의 따분한 설계의 서비스에 만들어진 일부 영리한 잘못된 지시와 관련이 있다. 실제로 플레이어들은 보통 어찌되었건 간에 대부분의 악당들을 물리쳐야 하겠지만, 싸우고 싶지 않은 플레이어는 믿음직하게 다른 방식으로 이길 수 있다. 게임은 탐색 이외의 무엇인가에 대하여 이 기술을 사용하도록 허용함으로써 탐험을 잘 하는 플레이어에게 보상을 한다. "역할 수행 게임"은 거의 언제나 효과적으로 "판타지 전투 시뮬레이션"에 대한 비슷한 말이기 때문에 탐험의 미캐닉이 큰 역할을 수행하는 게임을 살펴보는 것은 멋진 일이다.

## 19. Baten Kaitos

덱, 덱의 유지, 무승부의 운

개발: Monolith Soft

전투 설계: Hiroya Hatsushiba, Yoshiharu Kuwabara

### 포함 이유

Magic: The Gathering이 큰 히트를 친 이후에 실세계 또는 RPG 설정에서 카드 기반의 전투 시스템은 흔한 일이 되었지만, 전투 모드의 난수 발생기를 근본적으로 대체하는 플레이어가 사용자 지정할 수 있는 덱을 사용하는 Baten Kaitos와 그 후속작 만큼 잘한 것은 많지 않다.

### 게임 소개

Baten Kaitos는 여러 가지 면에서 드문 것이다. 모든 캐릭터는 평범한 인간 대신에 약간 펑크 스타일의 어떤 종류의 날개를 갖고 있다. 모든 영역에 미리 렌더링된 그림을 사용하는 PlayStation Final Fantasy 기법을 위하여 폴리곤을 이용한 배경은 빠졌다. 그리고 그 다음에는 전투가 있다.

전투는 기묘하다. 플레이어가 참여하는 모든 싸움은 카드 게임으로 수행된다. 캐릭터의 차례가 되면, 플레이어는 자신의 손으로 카드를 플레이하여 공격한다. 플레이어는 하나의 카드를 플레이하여 어느 정도의 손상을 줄 수도 있고, 카드의 구석에 있는 숫자에 따라 연속적으로 플레이할 수도 있다. 이렇게 하면 상당히 더 큰 손상을 줄 수 있지만, 이렇게 하려면 플레이어가 덱과 손 양쪽에 올바른 카드를 가지고 있어야 한다. 뛰기 위해서는 넓은 범위의 값에서 나온 카드가 있어야 하며, 게임이 진행되면서 더 높은 숫자가 조금씩 나온다.



나는 실제로 카드 대전 시스템을 충분히 강조할 수 없다. 카드가 전체 게임이기 때문에 캐릭터는 통계에 거의 영향을 주지 않는다. "매그너스(magnus)"라고 하는 이 카드는 플레이어가 수집하여 덱으로 정리되고, 덱은 CCG 플레이어라면 제대로 평가할 수 있는 방식으로 전투 외부에서 언제라도 재정렬될 수 있다. 하지만 실제로는 전투 자체가 어떤 실제 CCG 처럼 플레이되는 것은 아니다. 전투는 여전히 실시간으로 발생하기 때문에, 이 게임은 런을 찾는 러미와 스넵의 조합에 더 가깝다. 플레이어가 더 이상 카드를 플레이할 수 없거나 매우 엄격한 시계가 플레이어의 차례에서 시간을 알리면 차례를 끝낸다.

아마도 카드 전투 시스템의 가장 큰 영향은 싸움이 시작될 때 서투르게 행동하면 완전히 당할 수 있다는 것일 것이다. 플레이된 카드는 덱에서 자동으로 보충되지만, 플레이어가 어떤 카드도 플레이할 수 없다면, 다음 차례가 되기 전에 맞바꿔야만 한다. 플레이어가 흔치 않게 나쁜 무승부 운을 가진다면, 기본적인 조우마저도 어렵게 될 수 있다. 게임을 통하여 얻은 손 크기 확장이 이러한 현상의 발생 가능성을 낮출 수 있지만, 이를 완전히 배제하기는 어렵다.

### 설계 교훈

이것은 매우 이상한 게임이다. 이 게임이 현재의 시장에서 큰 히트를 칠 수 있었는지를 상상하기는 어려운 일이지만, 이 게임은 자신의 기이함을 세상에서 가장 자연스러운 것처럼 이용하여 발전을 이루었다. 그리고 전투를 카드로 교체할 수 있는 쉬어 볼(sheer ball)을 가진다는 점에서 싫어하기 어렵다. 실제로 Gamecube보다 더 인기 있는 시스템에서 게임이

만들어질 수 있었다면 나를 놀라게 하기에 충분하다. 작은 자주색 상자에 배타적인 것으로서, 게임이 이익을 낼 수 없을 것이라고 이미 결론이 내려졌을 수도 있었기 때문에 이들은 단지 재무적 성공은 무시하고 자신들이 만들려고 했던 것을 만들었다. 금제를 느슨하게 하는 것 만큼 확실한 실패는 없다!

## 20. ADOM

수년이 지난 후에도 여전히 미스터리한 게임.

개발 및 설계: Thomas Biskup

### 포함 이유

Rogue와 비슷한 근원과 NetHack의 최고의 특징의 상당수를 복사하는 능력, 자기 자신의 다수의 비슷한 특징과 자리를 잡고 있는 전체 "울트라 빅토리" 시스템 외에도, 오늘날까지 이 "기묘한 책"의 목적을 Biskup 외에는 아무도 이해하지 못하는 아이템이 Thomas Biskup의 유명한 Rogue 스타일의 게임에 존재한다. 이것이 어떻게 그토록 오랫동안 미스터리로 남아있을까?

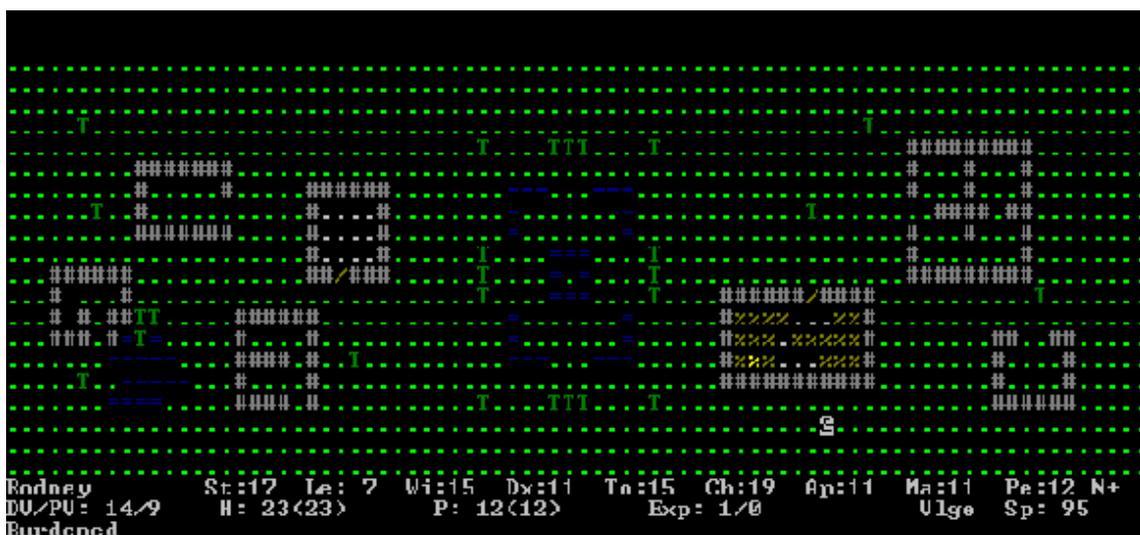
### 게임 소개

이것은 또 하나의 Rogue 스타일의 게임이다. 그리고 아마도 이 모든 것 중에서 NetHack으로부터 가장 큰 영감을 가져온 게임일 것이다. 하지만 이 게임은 임의적인 보물과 식별 게임을 포함하고 있다는 이유로 여기에 있는 것이 아니기 때문에 이것에 관해서는 내 말을 끝까지 들어야 한다.

NetHack은 수십 명이 만든 제품이고, 패치 기여자까지 포함시키면 수백 명이 되는 반면에, ADOM은 대부분이 한 사람의 작품이다. 그럼에도 불구하고, 이 게임은 놀라운 수준의 정교함을 보여준다. 한 사람의 설계자가 작업했다는 것을 고려하면, 그는 천재이다. Thomas Biskup이 어떤 종류의 신적 존재가 틀림없다는 인식을 하게 되는 때는 던전 바닥의 허브가 콘웨이의 생명(Conway's Life)의 규칙에 따라 자라는 것을 발견하는 때이다. 약간의 스타일 문제를 용서한다면(NPC 대화의 많이 읽으면 위축이 된다), 이것은 극히 미스터리한 게임이 된다.

아직도 우리를 놀라게 만들 수 있는 힘을 가진 컴퓨터 게임은 거의 없으며, 대체로 개발자들은 그렇게 하기를 포기했다. 우리들 중 상당수가 비디오 게임 플레이를 시작했을 때, 우

리는 여전히 학생이었다. Super Mario Bros.의 워프 존에 관하여 처음으로 들었을 때 어땠는지 기억이 나는가? 이런 말하면 유감이기는 하지만, 오늘날 퍼블리셔들은 자신의 게임을 둘러쌀 수 있는 미스터리의 기운을 의도적으로 쫓아서 수입을 약간 늘릴 수 있다는 것을 깨달았다. Nintendo는 스스로 Official Nintendo Player's Guide의 발표로 NES 게임의 비밀을 판매하는 데 앞장을 섰으며, 이제는 거의 모든 주요 게임과 함께 게임 전략 가이드가 판매될 것이라고 예상된다.



ADOM의 경우에는 공식적인 치트북이 없고 NetHack과는 달리 검색할 수 있는 소스 코드가 없기 때문에, 수 많은 플레이어들의 집중적인 연구와 엄청나게 헌신적인 FAQ 저자들의 노력에도 불구하고 이 게임에 관해서는 여전히 미스터리한 것이 남아 있다. 이 게임의 깊은 미스터리를 디코딩하는 데 많은 진전이 이루어지지 않았다는 것은 아니다. 오랫동안 플레이어가 단지 세계를 구하는 것 대신 신의 경지에 올라갈 수 있도록 하는 특별한 승리의 가능성인 "울트라" 엔딩의 세부 사항은 단지 루머였다. 실행 파일을 조사한 사람들은 Thomas Biskup이 사람들의 주의를 돌리기 위하여 코드에 사용되지도 않는 관계가 없는 것을 포함시켰다는 것을 밝혀냈다.

남아 있는 미스터리 중의 하나는 "기묘한 책(weird tome)"이라고 하는 아이템과 관련이 있다. 이 책의 기술된 목적은 단지 귀중한 아이템으로 교환하기 위하여 이것을 다른 캐릭터에게 전달하는 가져오기 퀘스트의 패키지일 뿐이지만, 플레이어는 이것을 읽을 수 있다. 플레이어가 관련된 기술에서 (보통을 한참 초과하는) 극도로 높은 스탯과 완벽한 점수를 가지지 않고 겉보기에는 마법 음악의 페이지로 구성되어 있는 이 책을 읽으려고 시도하면 오랫동안 혼동만 될 것이다. 현재로서는 이 책의 콘텐츠는 그저 커다란 주의 전환용 사물이라고 일반적으로 동의하고 있기는 하지만, 게임의 다른 몇몇 캐릭터는 책을 받았을 때 재미있는 말을

한다.

## 설계 교훈

압도적인 플레이어의 관심 앞에서 게임의 비밀을 숨기는 것은 매우 어려운 일이다. 콘솔의 경우에 개발자들은 비교적 안전한 하드웨어라는 이점을 가지지만, 컴퓨터에서는 흔치 않은 슬랙에 의존하지 않고는 훨씬 더 어렵다. 여기서 게임의 인기는 이에 반하여 작용한다. 사람들이 게임의 미스터리에 더 많이 매혹될수록 사람들은 더 열심히 미스터리를 풀어내려고 노력할 것이다.

## 링크

ADOM의 나머지 미스터리는 커뮤니티에서 작성한 가이드북의 한 섹션에 [요약되어 있다](#).

ADOM 플레이어 커뮤니티의 멤버들이 이 책의 목적을 발견하기 위한 노력을 논의하는 Google 그룹의 스레드는 다음과 같다.

[소스 코드 디스어셈블링 \(2004\)](#)

[최근의 세이브 파일 해킹과 최대 스탯으로 캐릭터 작성](#)

[2003년 이후의 일부 오래된 실행 파일 조사](#)