



포스트 모템: Zachtronics Industries의 스페이스캠

Postmortem: Zachtronics Industries' SpaceChem

작성자: 잭 바스(Zach Barth)

작성일: 2012년 6월 13일

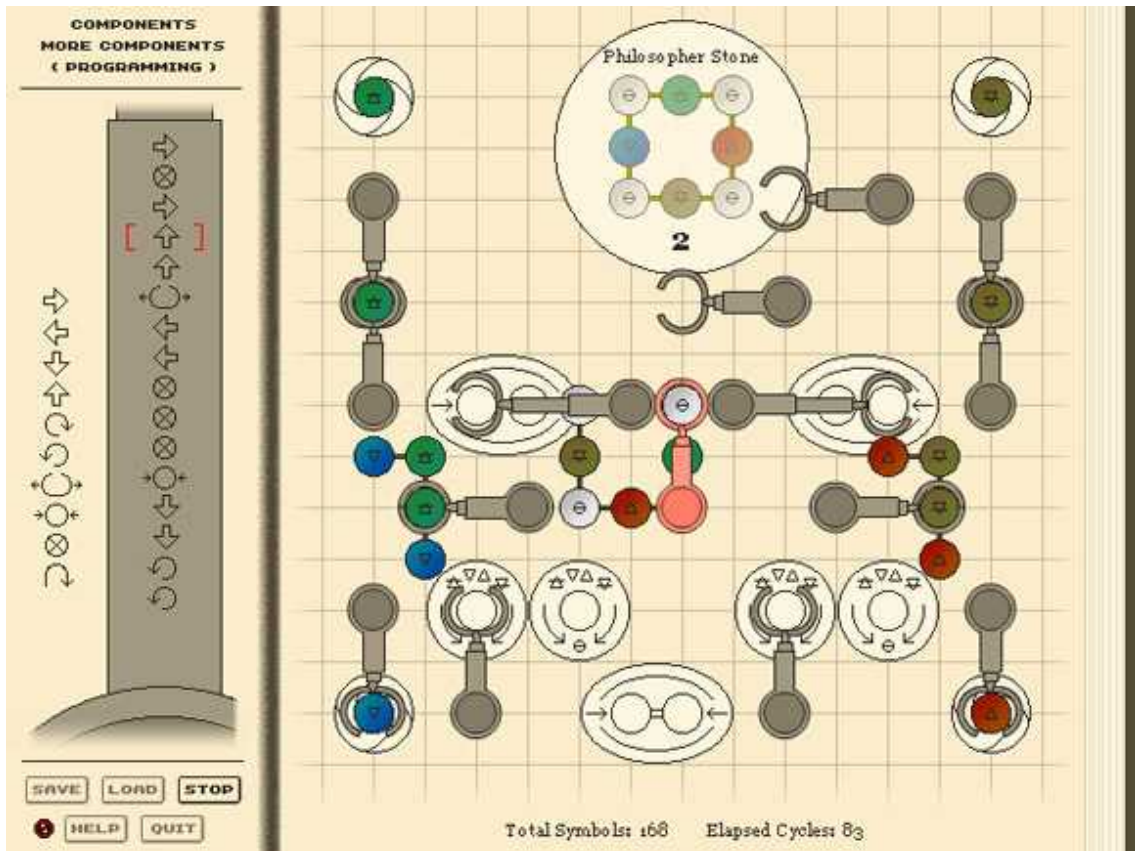
[매니아 층에서 커다란 인기를 얻은 인디 퍼즐 게임 <스페이스캠(SpaceChem)>의 개발자 잭 바스가 - 다른 본업과 병행하며 - 어떻게 이 복잡미묘한 게임을 개발하게 되었으며, 어떤 이유로 게임이 더 많은 사람들에게 받아들여지지 못했는지를 말한다.]

합성화합물을 생성 및 변형하는 기계를 제작하는 플래시게임인 <The Codex of Alchemical Engineering>을 출시한 직후 나는 화학을 소재로 한 후속작을 구상하기 시작했다. <Codex>가 이미 단순화된 형태의 분자 결합 모델이었므로 이를 화학 분야로 적절히 확장하면 (원자 간의 다중결합처럼) 더 많은 역학과 퍼즐(물처럼 단순한 화합물에서 벤젠처럼 복잡한 화합물에 이르기까지 수많은 화합물들)을 만들어낼 수 있을 것이었다. 그렇지만 나는 후속작을 곧바로 내놓는 성격이 아니었기 때문에 이런 생각을 한 권으로 제쳐두고 다른 일을 했다.

약 1년 뒤 나는 시애틀에 있는 가스 워크 공원(Gas Works Park)을 방문할 기회가 있었고 그곳에 방치된 화학공정용 파이프라인에서 영감을 얻었다. 화학에 기초한 <Codex>의 후속작을 다시 구상하면서 나는 낮은 수준의 <Codex> 조작을 고급 수준의 파이프라인 건설역학과 결합해보자는 생각을 떠올렸다. <스페이스캠>은 이렇게 시작되었다!

그 뒤로 6개월간 구상을 발전시키면서 코스믹 호러(cosmic horror) 장르를 채택해 게임에 보스 전투(boss battle)도 포함시켰다. 본업을 병행하면서 여유 시간을 이용해 동료 직원 하나와 함께 게임 개발을 시작했는데, <스페이스캠>이 출시될 무렵에

는 개발팀이 일곱 명까지 늘어났다.



<스페이스캠>의 전작 <The Codex of Alchemical Engineering>

잘된 점

1. 개방형 퍼즐로 만든 것

개발 과정에서 의문스러운 결정도 많이 했지만, 저마다의 해법이 가능한 개방형 퍼즐(open-ended puzzle)만큼은 우리가 제대로 내린 최고의 결정이었음이 분명하다. 개방형 퍼즐 없는 <스페이스캠>은 <스페이스캠>이 아니었을 것이다!

<스페이스캠>의 표준 게임플레이 “공식(formula)”은 플레이어에게 일련의 도구(지시문과 원자로)와 (X, Y, Z 분자를 생성하라’는 식의) 분명한 목표가 있는 문제, 그리고 해법을 만들어낼 빈 공간을 제공하는 것이다. 이 때문에 우리는 거의 대부분의 퍼즐을 설계할 때 모범답안을 생각하지 않고 할 수 있었다. 대신 각각의 문제가 논리

적으로 다른 문제와 중복되지 않고, 앞서의 해법을 반복하는 것으로는 해결할 수 없도록 만드는 데에 집중했다.

시험플레이로 난이도를 테스트한 뒤, 우리는 레벨을 재배열하고 논리적으로 중복되는 레벨을 제거하고, 게임의 난이도가 (급격하게 높아지기는 해도) 순서대로 구성될 수 있도록 레벨간 격차를 조절했다. 어떻게 보면 이런 작업을 통해 퍼즐의 설계가 한층 쉬워졌다고 할 수 있는데, 퍼즐과 해법을 동시에 설계할 때 나타나는 '닭이 먼저냐 달걀이 먼저냐'의 문제를 피할 수 있었기 때문이다.

이런 방식으로 퍼즐을 만들면서 생긴 재미난 결과 하나는, 퍼즐이 더욱 개방형이 되면서 엔지니어와 기획자들이 실생활에서 마주하는 종류의 문제들과 유사해지기 시작한 것이다. 이를 부담스럽게 여기는 플레이어도 있었지만, 대다수 플레이어들은 커다란 보람과 동시에 다른 게임에서 찾을 수 없는 자신만의 해법에 자부심을 느꼈다. 그리고 <스페이스캠>을 순식간에 그들의 '가장 좋아하는 게임' 목록에 올려놓았다.



버려진 파이프라인에서 영감을 얻은 <스페이스캠>

2. 리스크를 낮게 유지한 것

인디 게임 개발을 본업으로 하는 것만큼 멋지지는 않지만, <스페이스캠>의 개발자 대부분은 게임과 관계없는 본업을 유지하면서 여유시간을 이용해 게임을 개발했다. 그 덕분에 우리는 개발에 대한 보상으로 이익의 배분에만 신경 쓸 수 있었다.

내부적으로 부족한 역량을 메우기 위해 해외의 하청업체들과 협업하고 무료 툴과 소프트웨어를 최대한 활용함으로써 시간 외의 투자를 소규모로 유지해 리스크를 최소화하면서 <스페이스캠>을 출시할 수 있었다. 본격적인 게임 개발자들에게는 분명 통하지 않는 방식이겠지만, 이 덕분에 우리는 실패의 두려움 없이 게임업계에 뛰어 들 수 있었다.

3. PC라는 훌륭한 플랫폼을 선택한 것

PC는 <스페이스캠> 같은 게임에는 최고의 플랫폼이었다. 애플 앱스토어나 엑스박스 인디게임즈(Xbox Live Indie Games) 같은 소위 “개방형(open)” 플랫폼과 비교해 PC의 진입장벽은 매우 낮았다. 컴퓨터에서 게임 소개를 읽은 사람은 누구든지 몇 번의 클릭만으로 게임을 사서 플레이할 수 있다. 우리는 이를 이용해 다음의 몇 단계로 <스페이스캠>을 선보일 수 있었다.

1. <스페이스캠>을 처음 출시했을 때는 오직 Zachtronics Industries 스토어에서만 구입이 가능했는데, 이 웹사이트는 <스페이스캠>의 판매만을 위해 만든 사이트였다. 우리가 자체 스토어를 만들기로 한 이유는 윈도우, 맥, 리눅스까지 3종의 모든 PC 플랫폼을 대상으로 판매하기 위해서이고,페이팔 수수료를 제외한 매출을 100% 완전하게 확보하기 위해서였으며, 또 온라인 게임 유통 플랫폼인 ‘스팀(Steam)’에 오르지 못했기 때문이었다. 우리의 고객기반이 다른 온라인 스토어에 비해 훨씬 작기는 했어도 <Codex>나 <Infiniminer> 같은 이전의 무료 게임으로 다져진 팬 층이 있었던 덕분에 <스페이스캠>도 꽤 순조로운 출발을 보였다.
2. 게임이 출시되기도 전에 PC게임 웹사이트 Rock, Paper, Shotgun이 2011년 인디게임 페스티벌(IGF) 참가목록에서 <스페이스캠>을 찾아내어 우호적인 기사를 써 주었다. 출시 뒤에 게임 리뷰와 인터뷰가 올라가면서 게임에 관심을 갖는 사람들이 많아졌고, 덕분에 다른 게임 뉴스 사이트에도 관련 기사와 인터

뷰가 나오게 되었다. 여기에서 얻을 수 있는 교훈 하나. 마케팅 예산이 거의 없는 인디 게임개발자들에게 게임 기자들은 최고의 친구이다!

3. <스페이스캠>의 명성이 높아지면서 우리는 '스팀'을 통해 배급하는 문제를 놓고 '밸브(Valve)'사와 다시 연락을 취할 수 있었다. 업적(achievement)과 리더보드(랭킹)과 관련해 스팀의 플랫폼과 약간의 통합작업을 거친 뒤, <스페이스캠>은 스팀에 등재되었다. 이로써 방대한 규모의 이용자들을 접하게 되는 한편, 업데이트 과정도 간단해졌다. 지난 해 여러 유통경로를 통해 <스페이스캠>을 출시 했었는데, '스팀'만큼 성공적인 유통경로는 없었다.
4. 2011년 말에 우리는 '험블번들'(Humble Bundle: 인디게임 여러 개를 묶어 저렴하게 제공하는 행사)과 제휴해 '험블 프로즌 시냅스 번들'에 <스페이스캠>을 포함시켰다. 이는 우리가 윈도우와 맥, 리눅스를 타겟으로 했기에 가능했던 일이었다.

4. 개발언어로 C#을 채택한 것

C#을 이용해 <스페이스캠>을 개발하면서 매 개발과정마다 이득이 있었지만, 출시 후 전혀 새로운 플랫폼 - 아이패드-용으로 변환할 때는 특히 그랬다. 나 자신을 비롯해 우리 개발자들이 모두 C#에 능숙했기에 C#은 당연한 선택이었다. 지원되는 툴도 탁월하고 컴파일 시간도 빠른 덕분에 우리는 새로운 아이디어를 시도할 때 드는 시간을 크게 줄일 수 있었다.

마찬가지로 이러한 특징 덕분에 나는 게임의 기획자로서 개발자의 도움 없이도 게임을 거의 모두 쉽게 훑어보고 수정할 수 있었다. 또, C#이었기 때문에 게임 엔진과 게임 내 스크립트, 도구까지 모든 것을 같은 언어로 만들 수 있었다.

우리도 처음에는 관리형 언어를 이용할 경우 게임 성능 문제가 생길 수 있다고 걱정했지만 실제로는 문제가 되지 않았다.

또, C#을 썼기 때문에 <스페이스캠>을 거의 모든 플랫폼으로 변환할 수도 있게 되었다. 처음에는 윈도우와 X박스를 모두 목표로 하기 위해 XNA를 이용할 계획이었다. 이후 우리는 X박스 대신에 맥 OS X와 리눅스를 목표로하기로 결정했고, OpenGL과 SDL로 전환해 세 종류의 플랫폼 모두를 대상으로 쉽게 개발할 수 있었다. 비 윈도우 플랫폼에는 Mono를 사용했다.

나중에 아이패드 용으로 변환하기로 했을 때에는 MonoTouch를 이용할 수 있었고 플랫폼의 영향을 받는 부분만 다시 만들기만 하면 되었다. 변환 작업의 대부분은 마우스와 키보드 대신 터치 인터페이스로 작동하도록 UI를 업데이트하는 일이었다.

5. 커뮤니티 기능을 혁신한 것

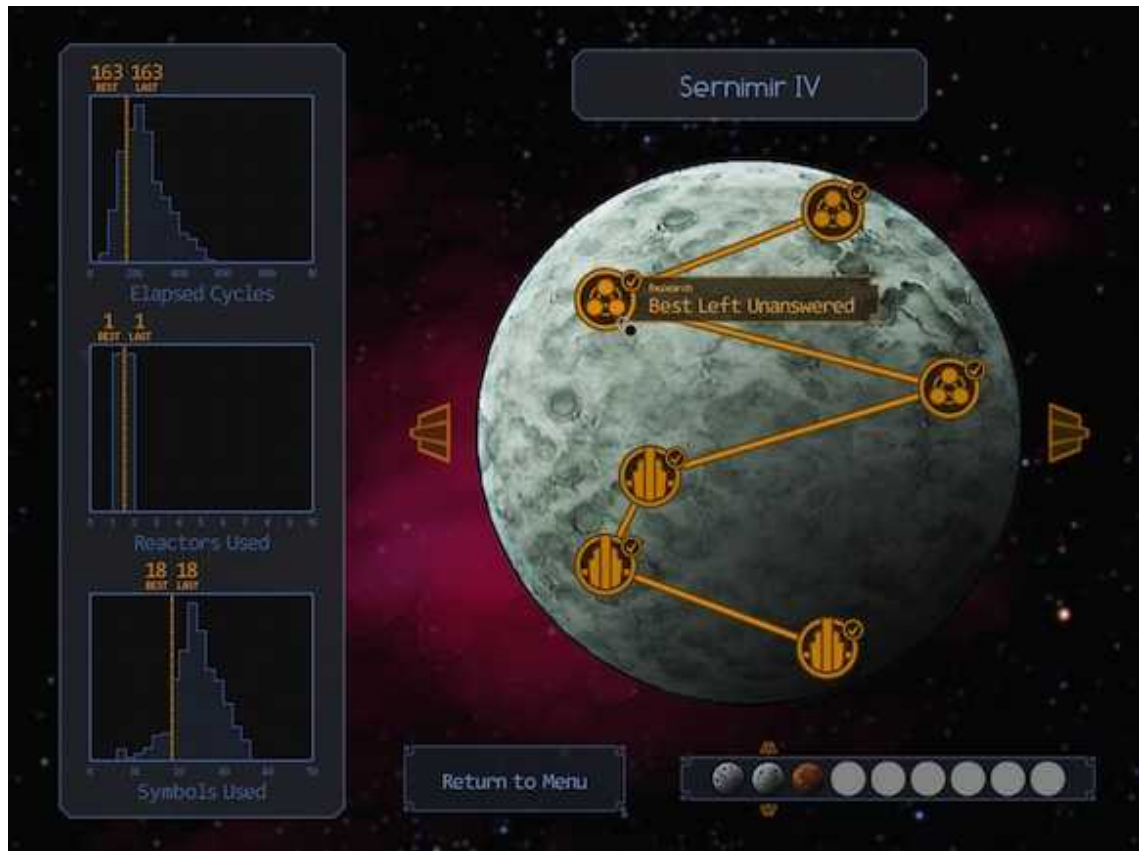
내 전략 중에 플래시 기반의 개방형 “공학 게임(engineering game)”인 <Codex>와 <КОДСТРЯКТОР>의 특징은 플레이어들이 자신들의 해법을 텍스트 블록으로 저장하고 불러올 수 있다는 것이었다. 저장 및 불러오기 기능이 그다지 원활치 않았음에도 불구하고 이 기능은 예상 밖의 결과를 가져왔다. 플레이어들이 게임 사이트인 콩그리게이트(Kongregate)와 Zachtronics Industries 웹사이트에서 가장 효율적인 해법을 경쟁하면서 저장/불러오기 텍스트를 증거로 삼았던 것이다. <스페이스캠>도 유사한 개방형 게임을 지향했기 때문에, 플레이어들이 자신들의 해법을 평가하고 경쟁할 수 있는 방법이 필요할 것임을 미리부터 알고 있었다.

<스페이스캠>의 막대그래프는 글로벌 리더보드를 대체하기 위해 만들어졌다. 이 막대그래프는 흔히 발생하는 두 가지의 문제를 해결했다.

1. 리더보드 최상위에 이름을 올리기 위해 속임수를 쓰게 된다.
2. 대부분의 플레이어들에게 글로벌 리더보드의 존재는 다만 자신이 얼마나 영망 인지를 알려주는 수단일 뿐이다.

글로벌 리더보드와 달리 <스페이스캠>의 막대그래프는 플레이어의 해법이 통계집단과 비교해 어떠한 지를 신속하고도 객관적으로 제시한다. 대부분의 플레이어들은 자신의 해법이 형편없다는 것을 깨닫지만, 막대그래프를 본 뒤에는 곧 도전의식을 느끼고 점수를 향상시키기 위해 퍼즐을 다시 플레이한다. 막대그래프가 세 개의 서로 대립되는 매트릭스(주기의 횟수, 기호의 개수, 원자로 개수)를 나타내기 때문에, 하나의 기준에 최적화한 플레이어들은 대개 다른 기준에서는 좋은 성적을 거두지 못한다. 덕분에 한 기준의 카테고리에서는 낮은 점수의 막대그래프를 계속 메워감으로써 평균을 깨기가 더욱 쉬워졌다.

<스페이스캠>의 모든 특징 중에서 점수 막대그래프가 아마 가장 인기가 있는 항목일 것이며, 내가 개인적으로 제일 마음에 들어 하는 부분이기도 하다. 막대그래프를 만드는게 리더보드보다 별로 더 어렵지도 않다는 점을 고려하면, 게임이나 커뮤니티 플랫폼 - 스팀! 너 말야 - 에 이를 포함시키지 못할 이유가 없다.



게임에 수학이 들어가다니?

플레이어들이 서로 경쟁할 수 있게 하기 위해 우리는 플레이어의 해법이 스크린샷 및 점수 정보와 함께 우리의 웹사이트로 전송되고, 그 다음 포럼이나 트위터, 기타 인터넷 어디에라도 포스팅될 수 있는 기능을 설계했다. 플레이어들이 서로 경쟁하는 것 외에, 이는 게임에 대한 입소문을 퍼뜨릴 수 있는 광고 기능까지 하게 된 것이었다.

그러나 이 기능을 만들면서 우리는 <스페이스캠>의 스크린샷을 해석하기가 거의 불가능에 가깝다는 것을 알게 되었다. 그래서 동영상을 기록하는 것으로 전략을 바꿨고, 유튜브를 "호스팅 솔루션"으로 활용해 커뮤니티 기능을 포함시켰다.

게임 안에 동영상 녹화 기능을 집어넣으면서 우리는 기술적인 관점에서나 라이선스 관점에서 C#으로 크로스플랫폼 동영상 녹화를 수행하기가 얼마나 어려운지를 알게 되었다. 우리는 결국 (플랫폼마다 컴파일한) 소형 네이티브 툴을 만들어서, (OpenGL 프레임버퍼를 이용해 캡처한) raw RGB 정보를 Ogg Theora 동영상 파일로 스트리밍

할 수 있게 했다. 이 동영상 파일은 플레이어의 데스크톱에 저장하거나 플레이어의 인증을 받아 유튜브에 올릴 수 있다.

잘못된 점

1. 구매층을 오판했다

<스페이스캠>에 앞서 나는 이미 '공학' 게임들로 소규모지만 강력한 팬층을 보유하고 있었다. 게임 사이트 Kongregate에서 <Codex>는 40만 번 이상 플레이 되었다. <스페이스캠>이 20달러에 출시되었다는 사실을 감안하면, 우리의 개발비용을 충당하고 약간의 이익을 내기 위해서는 기존 팬들을 <스페이스캠>의 고객으로 끌어들이는 것만으로는 부족하다고 생각되었다.

그래서 우리는 게임의 구매층이 누구였으며 어떻게 하는 것이 적절하지를 생각하다가, 일반적인 Zachtronics의 이용자 층, 즉 더 길고 세련된 '공학' 퍼즐게임을 원하는 사람들을 구매층으로 가정했다. 특기할 점은 이들은 통합 회로설계와 레이아웃에 관한 소비에트 테마의 퍼즐 게임인 <КОИСТРЯКТОР>의 이용자 층이기도 했다는 것이다.

2011년 3월로 다시 돌아가보자면, 출시한지 세 달이 된 <스페이스캠>은 스팀을 통해 훨씬 광범위한 이용자를 접하게 되었고, 우리는 상상 이상으로 훨씬 많은 돈을 벌 가능성을 앞두고 있었다. 일반 대중이 상당히 화학과 밀접해 보이는 게임을 할 용의가 있다고 확신할 수만 있다면 말이다.

우리의 첫 번째 틀린 가정은 모든 사람이 과학을 좋아한다는 생각이었다. <포털 (Portal)>¹ 같은 게임 덕분에 인터넷 세계에서 "과학"은 사랑 받을지라도, 실제 화학처럼 보이는 우리 게임은 대다수 사람들에게 화학 수업을 연상시켰다.

나는 <스페이스캠>과 관련해서 "하지만 난 화학을 잘하지 못한다"라는 말을 수없이 보고 들었다. 게임을 같이 설계한 한 동료는, 우리가 게임의 이름을 <SpaceGems>라고 부르고 연금술에 관한 게임을 만들었다면 적어도 두 배는 더 많이 팔았을 거라고 여러 번 주장했다. 내가 비록 실생활의 지식과 연관된 게임 만드

¹ 밸브 코퍼레이션(Valve Corporation)이 개발한 1인칭 슈팅/퍼즐 비디오 게임

는 것을 좋아하지만, 그의 주장을 부인할 자신은 없다.

우리의 두 번째 틀린 가정은, 훌륭한 퍼즐 게임은 응당 어렵고도 길어야 한다고 생각했던 것이다. 퍼즐 게임을 제대로 만들려면 도전은 필수적이지만 <스페이스캠>의 메카닉에서 드러나는 어려움은 당황스러울 정도였다. 바로 그 때문에 퍼즐이 개방형으로 느껴지기는 해도 말이다. 이 본질적인 어려움과 방대한 퍼즐의 결합으로, <스페이스캠>의 엔딩을 보기 위해서는 40시간 이상을 투자해야 했으며, 이는 단 2%의 플레이어만이 다다를 수 있는 수준이었다.

2. 매트릭스를 엉망으로 만들었다

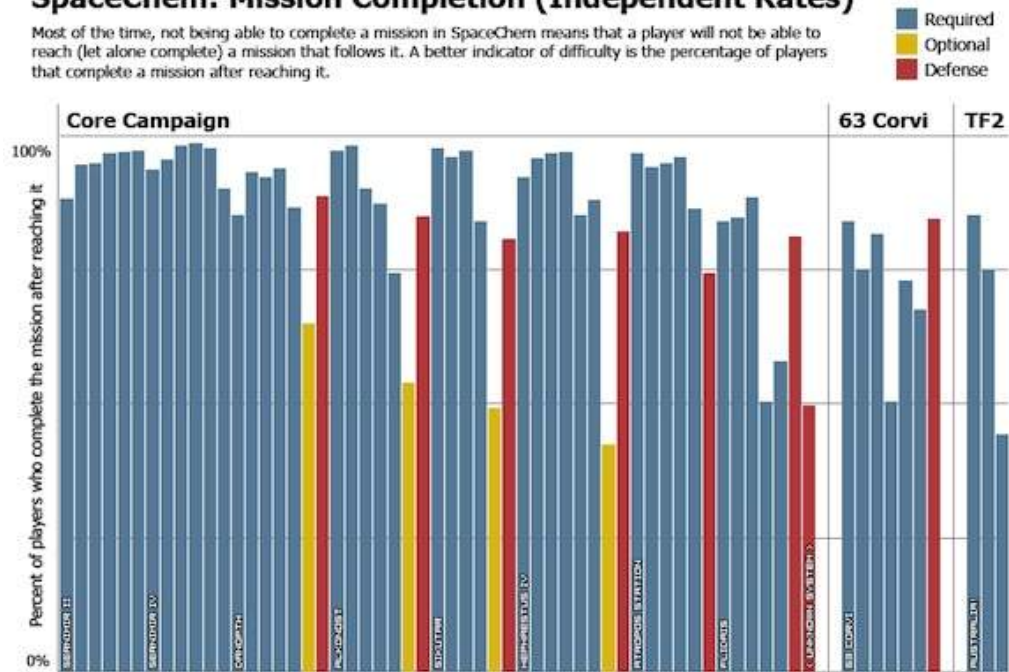
최초의 비공개 베타테스트에서 우리의 매트릭스 캡처시스템은 어떤 데이터도 캡처하지 못했다. 지나고 나서 생각하니, 이는 <스페이스캠>에 대한 우리의 통계 수집 전략이 미비했음을 보여주는 사건이었다. 출시 후 버그를 수정하느라 바쁜 나머지 게임 데모가 매트릭스 데이터를 불러오지 못한다는 것을 알아채는데 몇 주가 걸렸다.

나중에 우리는 게임 시작 시에만 매트릭스를 업로드했기 때문에 고객의 정보를 전혀 캡처하지 못했다는 사실을 깨달았다. 그 중에는 튜토리얼만 보고 질려서 다시는 게임을 해볼 생각도 하지 않은 플레이어도 많이 있었을 것이다. 이 문제를 해결했을 때, 우리는 <스페이스캠>이 명칭할 만큼 어렵다는 것 같은 명백한 사실만을 알아낼 수 있을 따름이었다.

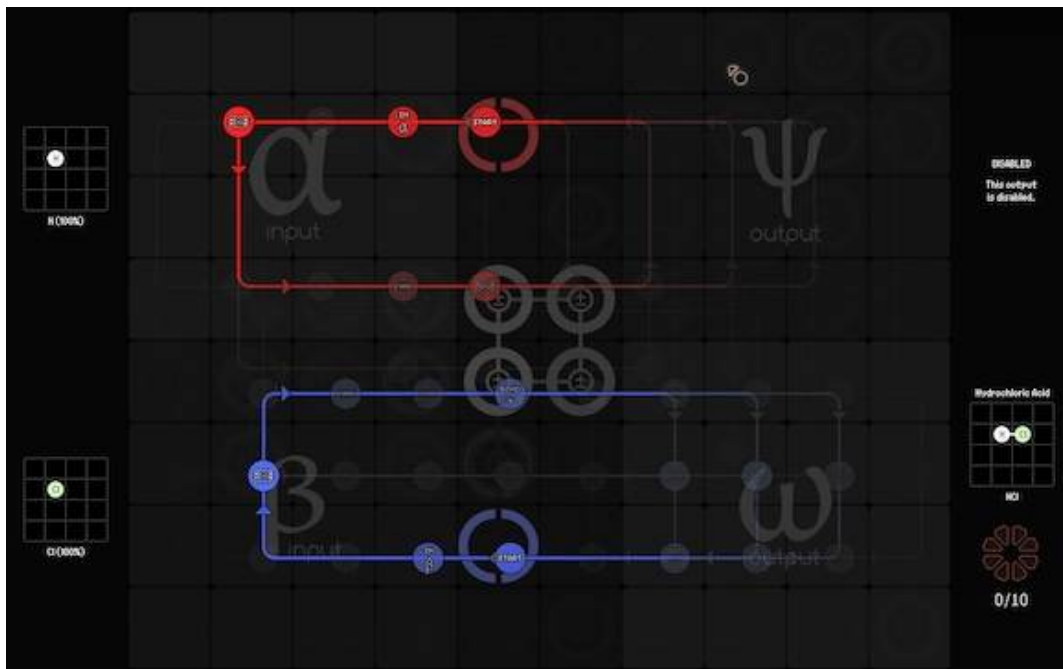
긍정적인 면을 보자면 나는 깔끔한 인포그래픽을 만들게 되었다!

SpaceChem: Mission Completion (Independent Rates)

Most of the time, not being able to complete a mission in SpaceChem means that a player will not be able to reach (let alone complete) a mission that follows it. A better indicator of difficulty is the percentage of players that complete a mission after reaching it.



<스페이스캠>에 있는 퍼즐의 완료율



<스페이스캠> 특정 퍼즐에 대한 "평균" 500개의 해법

3. 게임을 너무 길게 만들었다

나는 게임을 만들 때는 가능한 많은 콘텐츠를 제공하는 것이 중요하다고 생각했다.

<스페이스캠>과 내 전작들 간의 가장 큰 차이는 콘텐츠에 있었다. 보통 스토리 없이 5~15개의 퍼즐을 제공했던 것과 달리 50개의 퍼즐에 완전한 스토리를 포함시켰던 것이다. 게임에 나오는 초기의 퍼즐 몇 개는 몇 분 만에 해결할 수 있지만 퍼즐의 대부분은 몇 시간씩 걸렸다. 게임의 최종 보스 전투 같은 경우는 해결하는 데 보통 며칠이 걸렸다. 그 단계까지 갈 수만 있다면 말이다.

퍼즐 게임에 어려운 퍼즐을 넣는 것은 잘못된 일이 아니다. 플레이어들의 능력과 관심, 인내심은 언제나 광범위하며, 따라서 어려운 퍼즐은 뛰어난 플레이어들에게 도전의식을 불어넣고 그 밖의 다른 플레이어들에게도 게임에 대한 의욕을 불어일으킨다. 그러나 게임이 지나치게 어려워서 게임플레이의 진도를 나가지 못해 스토리의 전개가 가로막히면, 뛰어난 플레이어를 제외한 모든 사람들은 스토리를 완성시키지 못하고 게임에서 얻는 만족과 즐거움 역시 사라진다. 우리가 이 점을 깨달았을 때는 너무 늦은 뒤였다. 우리의 스토리와 퍼즐이 긴밀하게 연결되어 있기 때문에 두 가지를 분리해서 스토리가 먼저 끝나게 만드는 것은 게임의 전체 캠페인을 다시 작업해야 한다는 말과도 같았다.

<스페이스캠>에서 제공되는 퍼즐은 두 가지의 카테고리로 나뉜다. 캠페인 모드의 퍼즐, 그리고 이용자가 만든 "ResearchNet" 퍼즐이다. 나는 플레이어들이 캠페인 모드에서 좀처럼 나아갈 수 없다며 "게임이 끝나지 않는다고" 말하는 것을 수없이 많이 들었다. 하지만 각각의 ResearchNet 퍼즐을 해결하지 못했다고 해서 끝나지 않는 일을 하는 것 같다고 얘기하는 플레이어는 본 적이 없다. 이는 플레이어들이 자연스럽게 스토리의 엔딩을 "완료" 지점으로 인식한다는 뜻이다. 따라서 갈수록 난이도가 높아지는 스토리 중심의 퍼즐 게임을 만들 생각이라면, 이야기의 엔딩 뒤에 가장 어려운 콘텐츠를 집어 넣어야 할 것이다!

우리는 데모를 어디에서 끝낼 지를 결정할 때도 비슷한 딜레마에 부딪혔다. <스페이스캠>이 다른 게임과는 매우 다르기 때문에 나는 플레이어들이 구매를 결정하기 전에 게임을 충분히 경험할 수 있도록 하는 것이 옳다고 생각했다. <스페이스캠>의 퍼즐은 행성 별로 나누어져 있는데(각 행성 당 6개의 퍼즐), 처음 두 개의 행성은 전부 튜토리얼용 퍼즐로 구성되었다.

세 번째 행성에서 데모를 끝냄으로써 플레이어들은 실제 퍼즐이 얼마나 어려운 지 충분히 느낄 수 있었고, 또한 세 번째 행성의 끝에 등장하는 첫 번째 보스 전투도 접할 수 있었다. 안타깝게도 그로 인해 데모의 길이가 네 시간 정도로 길어져 버렸기 때문에 사람들에게 구입 충동을 일으키는 데는 도움이 되지 못했다.

나는 여러 경로를 통해 (특히 모바일 공간에서) 데모를 짧게 만드는 편이 언제나 판매에 도움이 된다는 주장을 들었다. 그게 사실이라 하더라도 나는 우리가 내린 결정을 완전히 후회하지는 않는다. 세 번째 행성까지 도달하는 동안 이 게임이 자신에게 맞지 않는다고 느낀 이용자가 만일 게임을 산 뒤에 그 사실을 알았더라면 크게 실망할 것이기 때문이다. 마찬가지로 세 번째 행성까지 통과한 플레이어가 더 플레이를 하고 싶어한다면, 게임의 나머지도 좋아할 것이 분명하다. 우리 비즈니스의 목적은 고객을 행복하게 만드는 것이다. 여기에 어긋나는 일을 뒤흔어 하겠는가?

4. 게임을 접하기가 너무 어렵게 만들었다

<스페이스캠>과 같은 게임은 흔치 않다. 그 때문에 특히 “주류” 타이틀에만 익숙한 사람들에게 게임을 설명하기란 대단히 어렵다. 더 슬픈 일은, 게임이 겉보기에 진짜 화학으로 착각하는 경우가 많을 정도로 화학처럼 보인다는 것이었다. 화학은 <포털> 같은 중독성 있는 문제 해결 게임이라는 실제 특징과는 전혀 일치하지 않았다. 더욱 나쁜 것은 유사 화학처럼 보이는 이 게임의 겉모습이 사람들의 호기심을 전혀 끌어낼 수 없었다는 것이다. 결과적으로 이 게임은 찾아내고 해보고 구입하기가 거의 불가능한 게임이 되었다.

우리에게 다행스럽게도 <스페이스캠>은 극도로 중독성이 있었다! 그리고 대부분의 중독성 물질이 그렇듯이 게임에 한 번 빠진 사람들은 주변의 친구들을 끌어들이게 된다. 얼마간은 우리가 계획한 “커뮤니티 기능” 덕분에, 또 얼마간은 게임의 충성 팬들 덕분에 <스페이스캠>의 얼리어답터들이 친구들에게 이 게임을 해야만 한다고 설득하고 나섰다.

<스페이스캠>이 어떤 게임인지 설명하고 게임에 대한 흥미를 불어넣을 수 있는 사람이 있었을 지는 의문이지만(“너 대학 전공이 컴퓨터 과학 맞지? 이 게임은 너한테 딱이야!”), 팬들이 친구들에게 해보라고 조르는 게임을 만들었다는 점에서는 우리는 꽤 성공적이었다.

5. 튜토리얼을 제대로 만들지 못했다

이제껏 나는 복잡한 게임 만들기를 좋아했지만 그 게임들을 적절히 설명할 능력은 부족했다. <스페이스캠>도 예외가 아니었다. 게임을 출시했을 때 <스페이스캠>의 튜토리얼은 투입과 산출, 결합과 분리, 원자로(reactor)와 파이프라인 등의 기본 요소들을 포괄하는 12개의 퍼즐로 구성되어 있었다. 12개 퍼즐 가운데 두 개는 세부 설명이 담긴 퍼즐로, 해법을 만드는 방법에 대한 분명한 지침과 함께 각각의 부분이 어떤 역할을 하는지를 단계별로 설명하고 있었다. 동시에 게임 전체에 걸쳐서 새로운 고차원 개념(가령 원자간 결합에 따른 분자 생성 및 수비 미션에서 해야 할 행동 등)을 마주하게 될 때 이를 설명하는 13개의 "정보 스크린(info screens)"도 있었다.

튜토리얼을 거친 플레이어들의 반응은 즉각적인 이해에서 완전한 이해불능(게임의 기획이나 플레이어들의 게임 해독 능력이 좋아진 요즘 게임에서는 이런 일이 참 드물 것이다)까지 다양하게 나타났다. 내가 추정한 이유는 다음과 같다.

퍼즐의 목적을 명확하게 보여주지 못했다. <스페이스캠> 퍼즐의 핵심 목적은 "투입"을 가지고 특정한 요구에 따라 "산출"로 변환하는 것이다. 소프트웨어 업계에 있는 사람들에게는 아무런 문제가 없었지만, 일반 대중들로서는 특히 유사 화학의 맥락에서 이런 목적을 이해하기는 쉽지 않았다.

처음 제시되는 퍼즐에서 투입과 산출을 분명하게 보여주었지만 대다수의 플레이어들은 자신이 하는 일의 목적을 모르는 상태에서 지시사항을 따랐을 뿐이다. 우리는 첫 번째 세부설명 단계를 마친 플레이어들에게 투입을 산출로 변환하는 <스페이스캠>의 목적에 초점을 맞춘 동영상 튜토리얼을 제공하는 것으로 이 문제를 일부 해결했다.

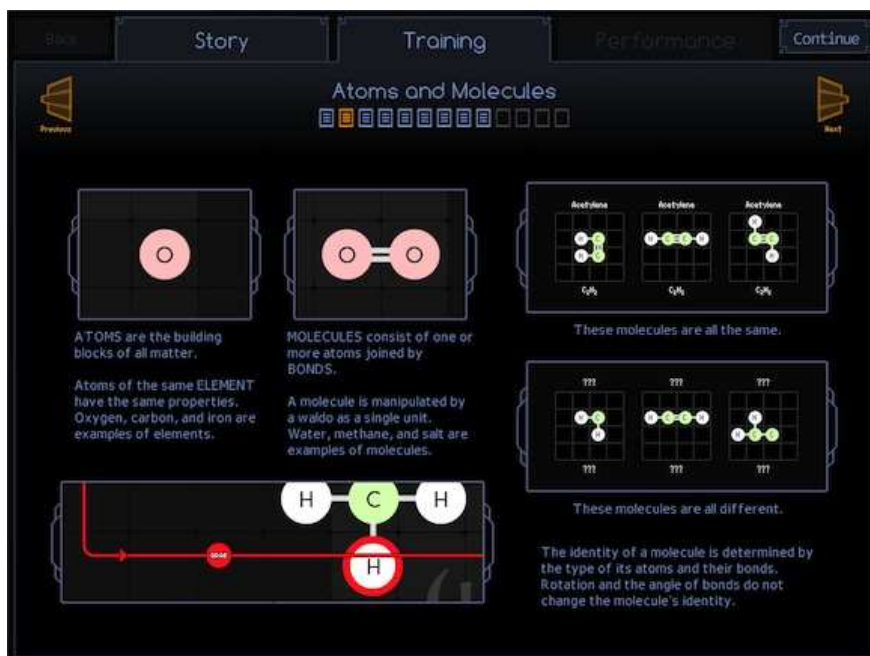
게임을 충분히 단순하게 만들지 못했다. <스페이스캠>에서 레벨을 쌓기 위해서는 투입, 잡기, 화살표, 놓기, 산출과 같은 최소한의 요소들을 가지고 전체 '루프(loop)'를 만들어야 한다. 대부분의 플레이어들이 게임의 목적을 제대로 이해하지 못했다는 점을 고려하면, 처음부터 최소 5개의 각각 다른 요소들을 다루는 일은 플레이어들에게 감당하기 어려운 요구였다. 이 문제는 "최소한의 가능한 해법"을 더 작게 만들도록 설계공간을 바꾸면 해결될 수도 있었지만, 출시 이후에는 가능한 선택이 아

니었다.

세부요소를 너무 많이, 너무 빨리 공개했다. <스페이스캠>은 무엇을 옮길 수 있고 무엇을 예측할 수 있으며, 퍼즐을 풀기 위해 무엇이 필요한지와 같은 수많은 세부 요소와 규칙으로 가득 차 있다. 이런 규칙들의 상호작용 덕분에 <스페이스캠>이 개방형의 참신한 게임이 될 수 있었지만, 플레이어들이 이를 잘못 해석하는 상황에서는 혼란을 더하는 요소이기도 했다.

대표적인 사례가 <스페이스캠>의 조종기(programmable manipulators)인 “왈도(waldo)’이다. 원자로의 기본 설정은 상단의 붉은색 왈도와 하단의 푸른 색 왈도로 구성된다. 기본 설정이 이렇게 되어 있기 때문에 대부분의 플레이어들은 붉은 색 왈도는 상단에서만 사용할 수 있고, 푸른색 왈도는 하단에서만 사용할 수 있다고 생각했다. 우리는 이 문제와 기타 비슷한 문제들을 완화시키기 위해 반대 사례를 포함하는 것으로 튜토리얼 퍼즐을 바꾸었지만, 그렇다고 해서 문제가 완전히 해결되지는 않았다.

설명을 위한 글을 지나치게 많이 집어넣었다. 우리는 게임의 규칙과 특징을 설명하기 위해 지나치게 많은 텍스트를 집어넣었다. 돌이켜보면 이는 게임을 역지로 설명하지 말고 게임을 바꿔야 한다는 분명한 신호였다.



분자에 관해 얘기를 늘어놓을 때는 이미 사용자를 잃은 것이다.

결론

전체적으로 볼 때, <스페이스캠>의 개발과 출시는 우리 팀 어느 누구의 예상보다도 훨씬 순조롭게 이루어졌다. 원래 내 바람은 개발비용을 충당하고 약간의 돈을 버는 것이었지만, 결과적으로 나는 직업을 바꿔 게임스튜디오를 차리고 새로운 타이틀을 개발 중에 있다. 덧붙이자면 새로운 게임은 <스페이스캠>보다 훨씬 대중적인 게임이 될 것이다!

Data Box

개발자: Zachtronics Industries

퍼블리셔: Zachtronics Industries

출시일: 2011년 1월 1일

플랫폼: PC(윈도우/맥/리눅스), iPad, OnLive

개발자 수: 7명

개발기간: 1년

예산: 4천 달러와 '상당히 많은' 자유시간

코드 행수: 17,000행(게임) + 5000행(유틸리티)

개발도구: Visual Studio(C#), Subversion, MonoTouch

게임을 위해 개발한 가짜 원소: 4개