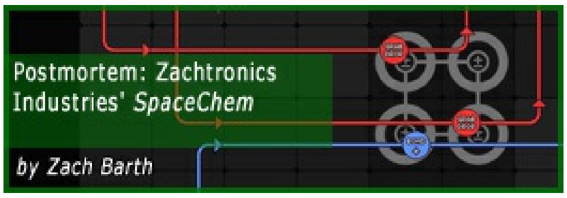
# Postmortem: Zachtronics Industries' *SpaceChem*

By Zach Barth



[*Zach Barth, developer of the cult indie puzzle hit SpaceChem explains what went into creating such a complex, nuanced game -- while still working a day job -- and also what held back the game from finding the audience it might otherwise have found.*]

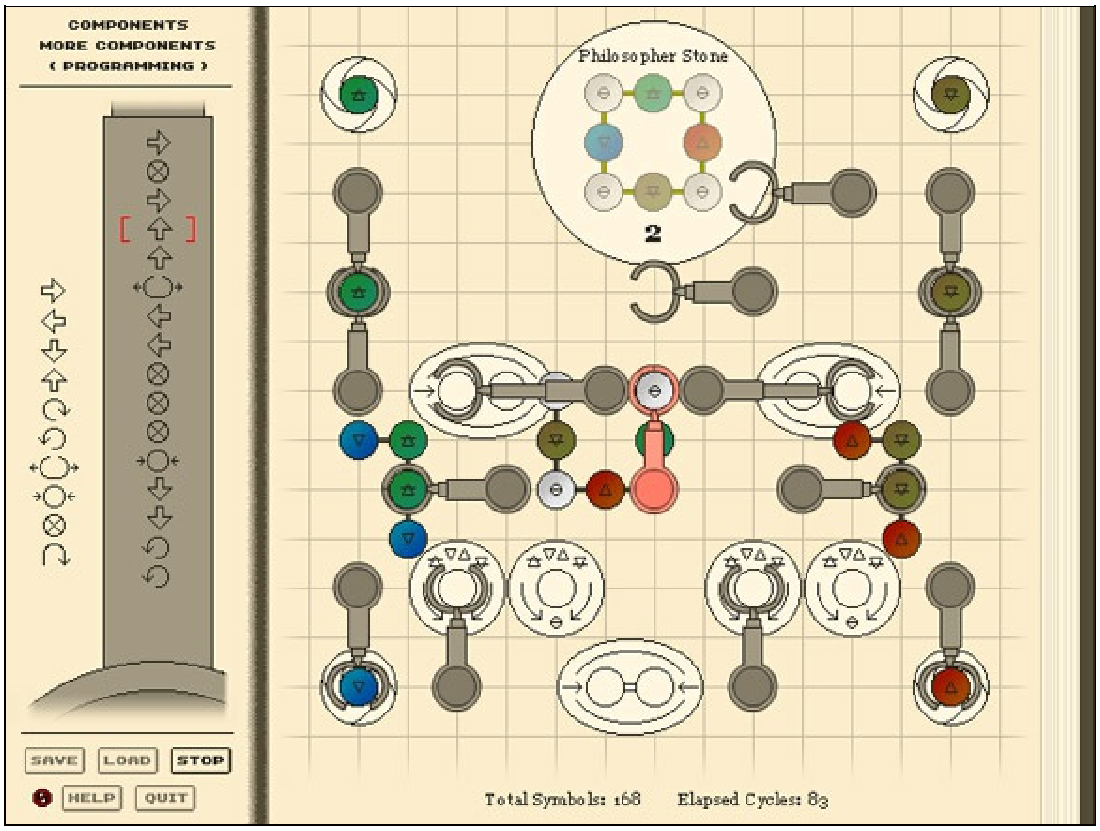
Shortly after releasing *The Codex of Alchemical Engineering*, a Flash game about building machines that create and transform alchemical compounds, I started thinking about a chemistry-themed sequel. Since *Codex* was already a simplified model of molecular bonding, expanding into chemistry proper would provide more mechanics (such as multiple bonds between atoms) and puzzles (different compounds, from simple ones like water to more complicated ones like benzene). Despite this, making immediate sequels is not in my nature, so I set the idea aside and moved on.

About a year later I visited Gas Works Park in Seattle and was inspired by its derelict chemical processing pipeline.

Thinking back to the idea for a chemistry-inspired *Codex* sequel, it occurred to me to combine the low-level manipulations of the *Codex* with a high-level pipeline construction mechanic. The idea for *SpaceChem* was born!

The idea evolved over the next six months, picking up a cosmic horror story with boss battles in the process.

I started developing the game in my spare time with a coworker from my day job, eventually growing the team to seven people before shipping *SpaceChem*.



***The Codex of Alchemical Engineering*, the predecessor to *SpaceChem*. Believe it or not, it's actually harder!**

## What Went Right

### 1. We Created Open-Ended Puzzles

Although we made a lot of questionable development decisions, the game's open-ended puzzles are unquestionably the biggest thing we did right; without them, *SpaceChem* would not be *SpaceChem*!

The standard gameplay "formula" for *SpaceChem* is to give the player a set of tools (instructions and reactors), a challenge with a clear end condition (create molecules X, Y, and Z), and an empty area in which to create a solution. Because of this, we were able to design almost all of the puzzles without knowing how they might be solved, focusing instead on making sure that each challenge was logically unique and could not be solved by repeating a previous solution.

After playtesting the levels, we reordered them, removed logical "duplicates", and filled "gaps" to create a fairly linear (albeit steep) difficulty curve. In some ways this makes puzzle design easier, as it avoids the chicken-and-egg problems that arise when designing a puzzle and its solution simultaneously.

An exciting side-effect of creating puzzles this way is that they end up being more open-ended and start to resemble the kind of problems that engineers and designers face in real life. Although some players find this intimidating, others find it intensely rewarding and discover a sense of ownership in their solutions not found in other games, quickly propelling *SpaceChem* into "favorite game" territory.

**Superfund? More like *super-fun*!**

**2. We Kept Our Risks Low**

Although it's not as sexy as full-time independent game development, most of the people who worked on *SpaceChem* did so in their spare time while holding full-time jobs outside of the games industry. This allowed us to rely almost entirely on profit sharing for compensation.

By working with international contractors to fill our talent gaps and using free tools and software wherever possible, the non-time investments to bring *SpaceChem* to market were small and low-risk. Although this clearly wouldn't work for "serious" game developers, it allowed us to break into the games industry with a good deal of momentum and no fear of failure!

**3. We Picked a Great Platform -- PC!**

The PC was the best possible platform for a game like *SpaceChem*. The barrier to entry is extremely low compared to similarly "open" platforms like the Apple App Store and Xbox Live Indie Games. Almost every person who reads about your game on a computer is capable of buying and playing the game within a few clicks. Leveraging this, we were able to roll out *SpaceChem* in a few phases:

1. When *SpaceChem* launched, it was only available for purchase on the Zachtronics Industries store, a website created solely for the purpose of selling *SpaceChem*. We chose to build our own store because it let us sell for all three PC platforms (Windows, Mac, and Linux), because it let us keep 100 percent of the revenue (minus PayPal fees), and because we were unable to get on Steam. Despite the fact that our customer base was much smaller than other online stores, the Zachtronics Industries fan base, built on previous free games like the *Codex* and *Infiniminer*, quickly embraced *SpaceChem* and set us off to a strong start.
2. Before the game had even been released, PC game website Rock, Paper, Shotgun found *SpaceChem* in the 2011 IGF entry list and wrote a flattering post about the game. After launch it posted a review and an interview, drawing additional and much-needed attention

to the game, helping to secure additional posts and interviews on other gaming news sites. The lesson here: as an independent game developer with little to no marketing budget, games journalists are your best friends!
3. Shortly after the word got out about how awesome *SpaceChem* was, we were able to reconnect with Valve about distribution on Steam. After a bit of work to integrate with the Steam platform for achievements and friend leaderboards, *SpaceChem* was launched on Steam, giving us access to their tremendous audience, epic seasonal sales, and painless update process. Although we've launched *SpaceChem* on many distribution channels in the past year, none have been as successful for us as Steam.
4. Toward the end of 2011, we partnered with the Humble Bundle to include *SpaceChem* in the Humble Frozen Synapse Bundle, something that was only possible because of our choice to target Windows, Mac, and Linux.

---

**4. We Used C#**

Using C# in *SpaceChem* paid off in every phase of the development process, especially when we made the post-release decision to port to a completely new platform -- the iPad.

All of our developers (as well as myself) were proficient in C#, which made it an obvious choice. The excellent tool support and fast compilation times greatly reduced the time it took to try new ideas.

These same features made it easy for me, as a designer, to casually browse and tweak almost anything in the game without a developer's assistance. Our choice of language also meant that we were able write everything -- the game engine, in-game scripting, and tools -- in the same language.

Early on, we were concerned that using a managed language could lead to performance problems, but it was not an issue in practice.

Using C# also gave the option of porting *SpaceChem* to almost any platform. Initially, we planned to use XNA to target both Windows and Xbox. When we later decided to target Mac OS X and Linux instead of Xbox, we switched to OpenGL and SDL and were able to easily develop for all three of those platforms, using Mono on the non-Windows platforms.

When we later decided to attempt a port to the iPad, we were able to leverage MonoTouch and only had to rewrite platform-specific aspects of the game. Most of the porting effort was in updating the UI to work with a touch interface instead of a mouse and keyboard.

**5. We Were Innovative With Our "Community Features"**

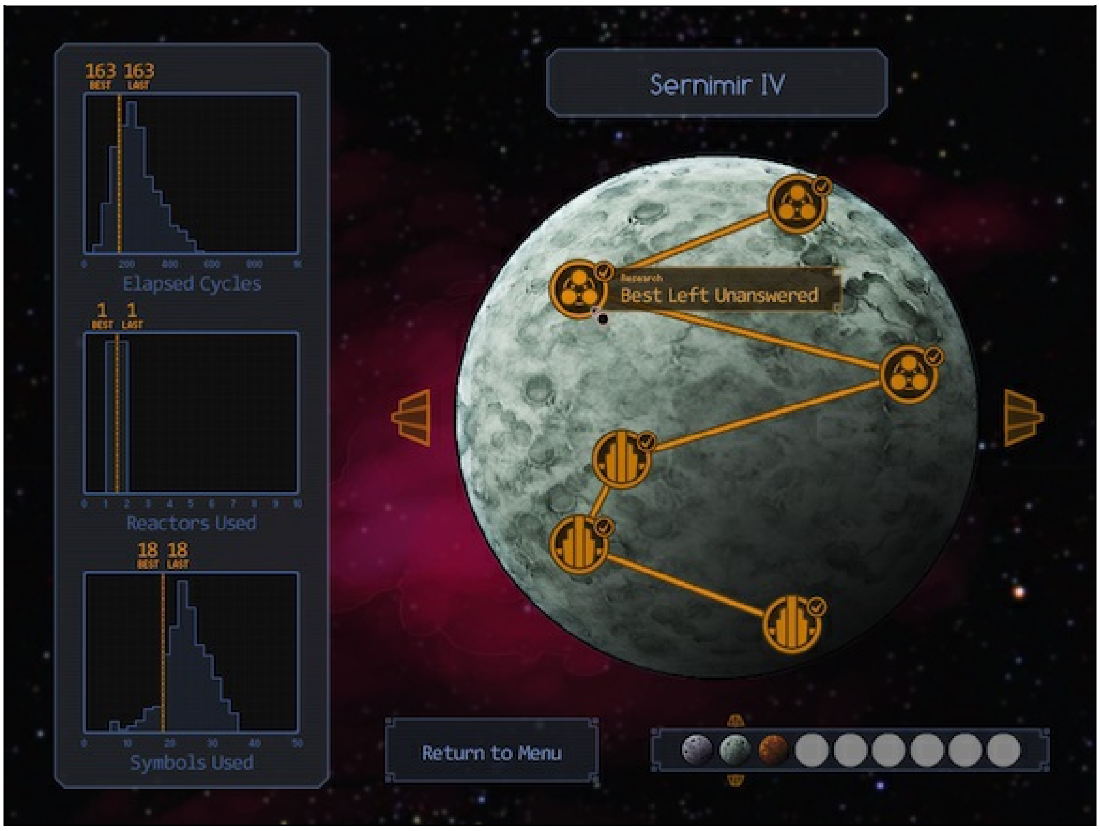
Two of my previous Flash-based open-ended "engineering games", *Codex* and *KOHCTPYKTOP*, included a feature allowing players to save and load their solutions as blocks of text. Although it made for a terrible save/load experience, it had the unintended consequence of allowing players to compete in the comments on Kongregate and the Zachtronics Industries website to build the most efficient solutions, using the save/load text as proof. Since *SpaceChem* was going to be a similarly open-ended game, we knew early on that we needed ways for players to evaluate their solutions and compete in private.

*SpaceChem*'s histograms were developed as a replacement for global leaderboards. They solve two common problems:

1. Getting your name at the top of the leaderboards is a fantastic incentive for cheating.
2. For most players, the **only** thing a global leaderboard manages to tell you is that you suck (and not even by how much).

Unlike global leaderboards, *SpaceChem*'s histograms allow you to quickly and impersonally see how your solution stacks up against the aggregate. We have found that most players discover that their solution is terrible, but quickly formulate a personal challenge after looking at the histogram and replay the puzzle to improve their score. Because we include three antagonistic metrics (number of cycles, number of symbols, and number of reactors), players optimizing for one criterion often do poorly in the others, padding the graphs with low scores that make it easier to beat the average in a single category.

Of all the features in *SpaceChem*, the score histograms are probably one of the most popular, and one of my personal favorites. Considering that they're not much more difficult than a leaderboard to implement, there's no reason not to include them in your game and/or community platform -- we're looking at you, Steam!



**Who put math in my game?!**

To enable players to compete in private, we designed a feature where a valid solution could be exported to our website as a set of navigable screenshots with baked-in score information, which could then be easily posted to a forum, Twitter, or anywhere else on the internet. In addition to allowing players to compete, it would include advertising for the game, helping to increase the "viral potential".

However, while implementing this feature we discovered that*SpaceChem* screenshots were absolutely impossible to follow. We switched to recording videos and leveraged YouTube as a "hosting solution" that happened to have built-in community features.

When we started implementing the in-game video recording features, we learned just how hard it is to perform cross-platform video recording in C#, both from technical and licensing standpoints. We ended up writing a small, native tool (compiled for each platform) that allowed us to stream raw RGB information (captured using OpenGL framebuffers) into Ogg Theora video files. These video files are then either saved to the user's desktop or uploaded to YouTube with credentials provided by the player.

---

## What Went Wrong

**1. We Misjudged Our Audience**

Prior to *SpaceChem*, I had already established a small but strong fan-base with my previous "engineering" games. On Kongregate,*Codex* had over 400,000 plays; it didn't seem like a stretch to convert enough of them into *SpaceChem* customers to cover our costs of development and make a little money, given that the game launched at $20.

So, when we started thinking about who our audience was for the game and what would be appropriate, we assumed the standard Zachtronics audience, who seemingly wanted a longer, more polished "engineering" puzzle game. It should be noted that this is the same audience that enjoyed *KOHCTPYKTOP*, a Soviet-themed puzzle game about integrated circuit design and layout.

Fast forward to March 2011: three months after release,*SpaceChem* was available on Steam to a much larger audience and we were looking at the possibility of making far more money that we had ever imagined, if only we could convince the general public that they wanted to play a game that very much appeared to be about chemistry.

The first incorrect assumption we made was thinking that everyone likes science. Although the internet may love "Science!" thanks to games like *Portal*, games that look like actual chemistry remind most people of chemistry class.

The number of times I've read and heard the words "but I'm not good at chemistry" in connection with*SpaceChem* is staggering. A particular game design colleague has asserted many times that had we called the game "*SpaceGems*" and made it about alchemy, we would have sold twice as many copies. Although I like building games around real-life knowledge, I'm not sure if I disagree with his assessment.

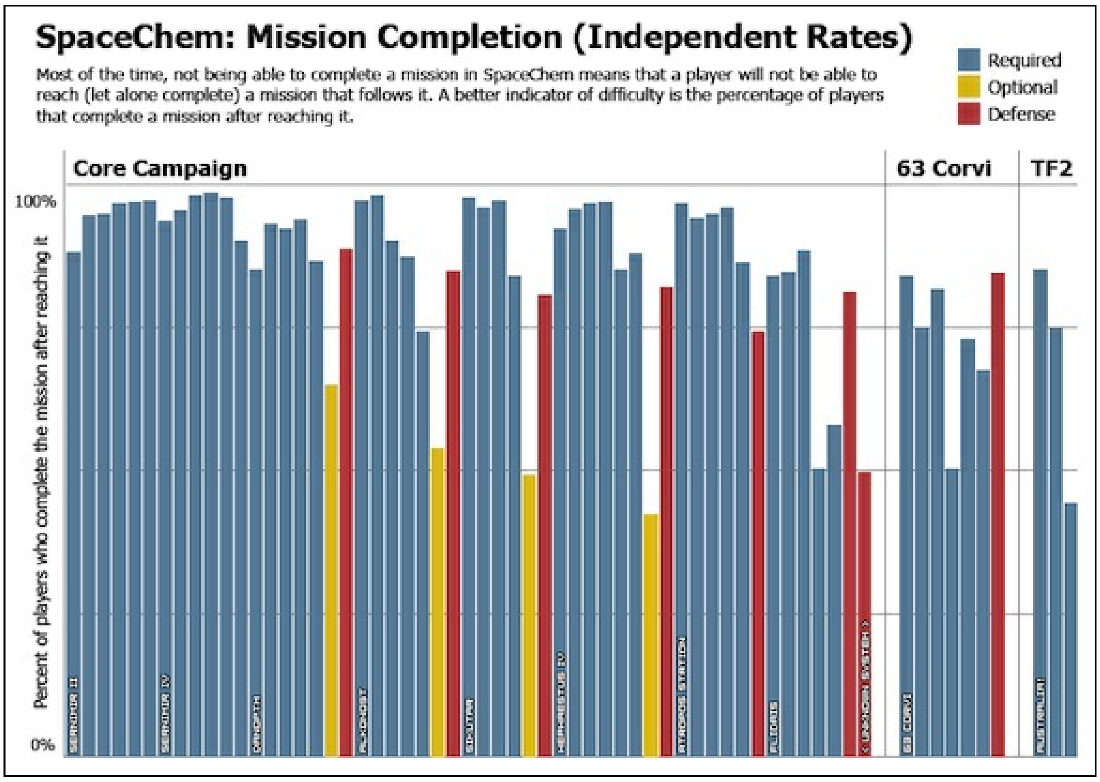
The second incorrect assumption we made was thinking that puzzle games need to be difficult and long to be good. Although challenge is essential to making puzzle games "work", the difficulty that emerges from the mechanics of *SpaceChem* can be bewildering, even if it's what makes the puzzles feel so open-ended. Combining this intrinsic difficulty with an abundance of puzzles that must be beaten to "complete" the game gives *SpaceChem* an oppressive 40+ hour difficulty curve that only 2 percent of players reach the end of.

**2. We Flubbed Our Metrics**

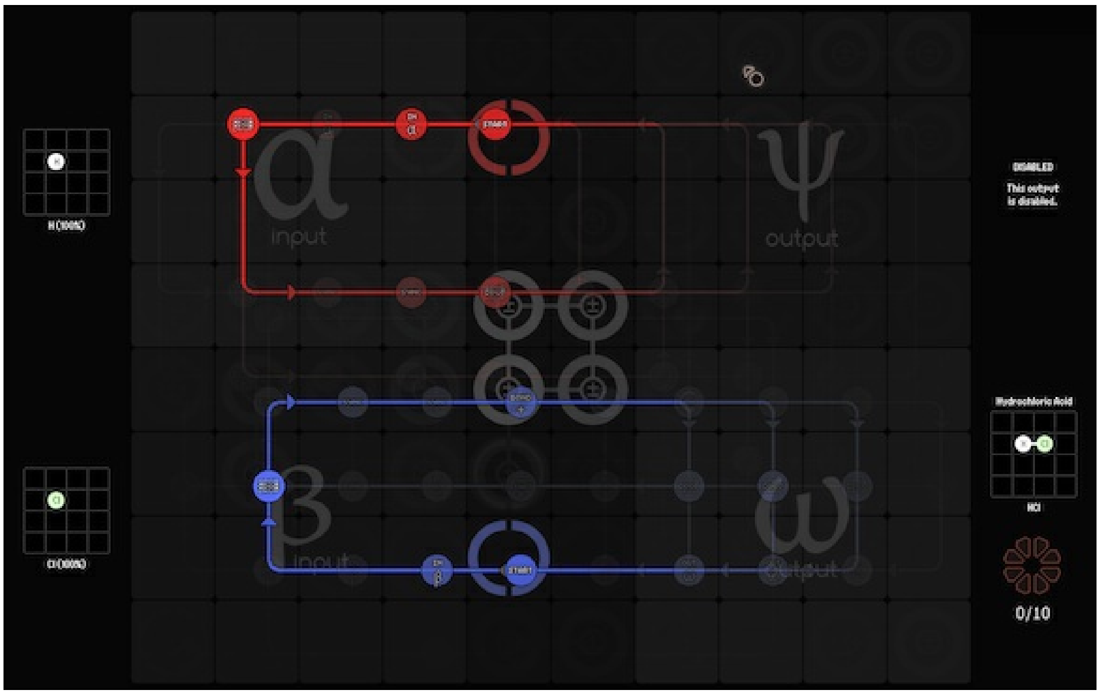
In our first private beta playtest, our metrics capturing system failed to capture any data, an event that in hindsight describes much of our statistics-gathering strategy for *SpaceChem*. Preoccupied with fixing post-launch bugs, it took us weeks to realize that our demo couldn't upload metrics data.

Later we discovered that, because we only uploaded metrics at the start of the game, we never captured any information from single-launch customers, which likely included many players who got stuck in a tutorial and then never bothered to try the game again. By the time we got things straightened out, we only managed to learn the magnitude of obvious things, such as the fact that *SpaceChem* is stupidly difficult.

On the plus side, I got to make some neat infographics!



**Completion rates for the puzzles in the *SpaceChem*. (*Click for larger image*)**



**The "average" of 500 solutions for a particular puzzle in *SpaceChem*.**

**3. We Made the Game Too Long**

When making a game, it's important to provide as much content as possible -- right?

The biggest difference between *SpaceChem* and my previous games is that it was a commercial endeavor. The second biggest difference was the amount of content -- 50 puzzles, plus a full story, instead of the five to 15 story-free puzzles I would usually aim for. Although some of the earlier puzzles in the game can be solved in minutes, many of the puzzles take hours. Some, such as the game's final boss battle, routinely take players days to solve -- if they even get that far.

There's nothing wrong with having difficult puzzles in a puzzle game; the capabilities, interest, and patience of your players will always span a huge range, so difficult puzzles can keep the best players challenged and give everyone else something to aspire to. However, when progression through a story is blocked by progression through the gameplay, making the game too difficult denies all but your best players completion of the story and the satisfaction and enjoyment that goes with it. By the time we realized this, it was too late. Since our story and our puzzles were tightly coupled, separating them so that the story ended earlier would have required reworking the entire campaign.

The puzzles available in *SpaceChem* fall into two categories: campaign puzzles and user-created "ResearchNet" puzzles. Although I've heard many players describe their lack of progress in the campaign as "not finishing the game", I've never heard anyone claim that, by not beating every single ResearchNet puzzle, they felt as if they had unfinished business. This seems to indicate that players naturally tag the end of the story as the "completion" point. So, if you're going to make a story-driven puzzle game that gets progressively harder, you really ought to put the hardest content after the end of the story!

We ran into a similar dilemma when deciding where to end the demo. Since*SpaceChem* is so unlike other games, it seemed fair to make sure that players got a good taste of the game before being forced to make the purchase. The puzzles in *SpaceChem* are divided into planets (about six puzzles each), with the first two planets consisting entirely of tutorial puzzles.

Ending the demo on the third planet had the benefit of giving players a good feel for how difficult the non-tutorial puzzles were and demonstrated the game's boss battles, the first of which did not show up until the end of the third planet. Unfortunately, it also made the demo about four hours long, which isn't good for encouraging impulse buys.

I've heard claims from various sources (particularly in the mobile space) that making your demo shorter is always better for sales. Even if that's true, I don't entirely regret our decision. Any player who realized the game was not for them when they reached the third planet would have been strongly disappointed if they had to buy the game to learn this. Likewise, any player who made it past the third planet and wanted more would be likely to love the rest of the game. Our business exists to make our customers happy -- why would we act against that?

---

**4. We Made a Game That Was Too Hard to "Get"**

There aren't many games like *SpaceChem*, which makes it spectacularly hard to explain, especially to people who are only familiar with more "mainstream" titles. To make matters worse, the surface appearance of the game -- fake chemistry, often mistaken for real chemistry -- completely belies the game's addictive *Portal*-like problem solving. To complicate things further, our presentation of fake chemistry lacks the appeal (i.e. sexy scientists) to draw curious viewers in! The end result is a game that is nearly impossible to discover, try, and buy.

Fortunately for us, *SpaceChem* is extremely addictive! And, as is the case with most addictive substances, it is the responsibility of your questionable friends to get you hooked. Partially because of our planned "community features" and partially because it's just what great fans do, *SpaceChem*'s early adopters took the time to tell their friends that they*had* to try it.

Although I doubt anyone was able to explain*SpaceChem* and make it sound like fun ("You went to university for computer science, right? Well, I have the game for you!"), making a game that our fans *demanded* their friends try got us pretty far.

**5. We Never Got the Tutorial Right**

Historically, my love for making complicated games has outpaced my ability to explain them properly.*SpaceChem* is no exception to this trend.

When we launched, the tutorial for*SpaceChem* consisted of 12 puzzles that covered the basics -- inputs, outputs, bonding, unbonding, reactors, and pipelines. Two of the 12 puzzles were "walkthroughs", which included explicit instructions for building a solution and explained, step by step, what each piece was responsible for. Additionally, there are also 13 "info screens" spread throughout the entire game explaining new, high-level concepts (such as atoms + bonds = molecules, and what to do on defense missions) as they are encountered.

The responses of players who played through the tutorials ranged from immediate understanding to complete incomprehension (something that I think is actually quite uncommon for modern games, given the combination of better design practices and better "game literacy" among players). Some of the reasons I suspect for this are:

**We failed to clearly show the objective of a puzzle.**The primary goal of a *SpaceChem* puzzle is to take "inputs" and convert them into the specified required "outputs". For anyone in the software industry this is an obvious endeavor, but for the general population it's not a given, especially in the context of fake chemistry.

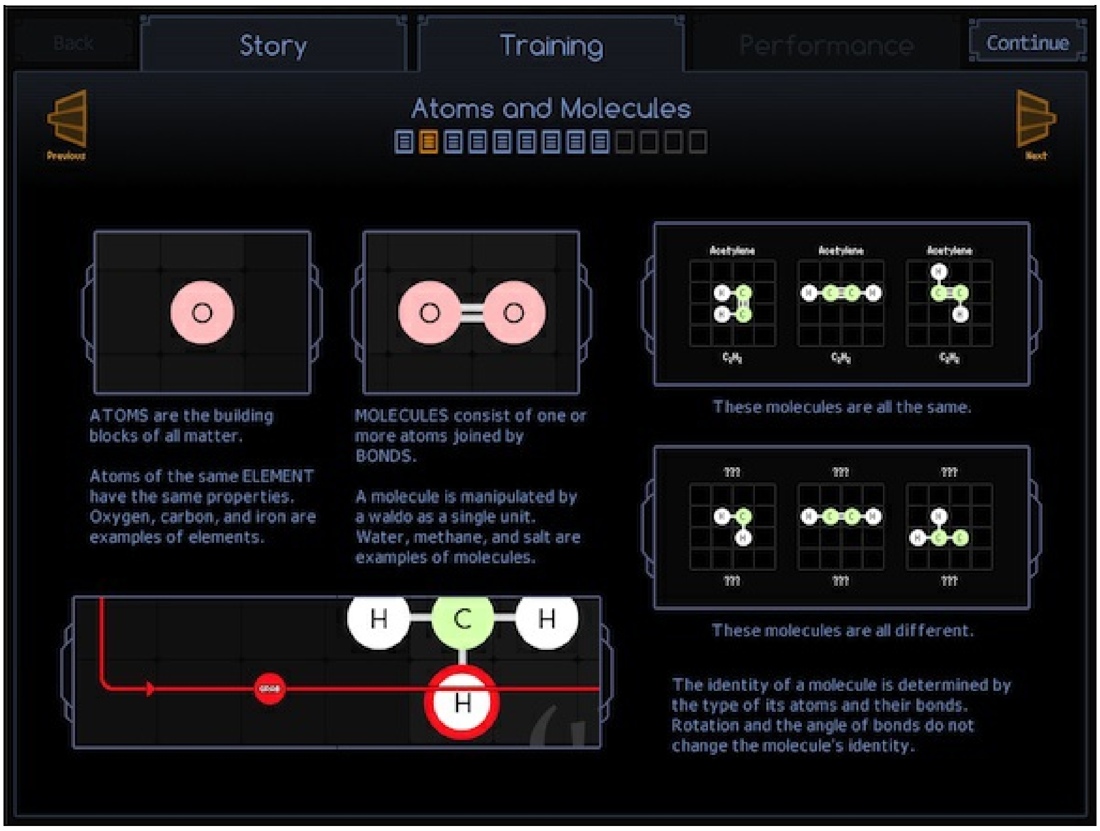
Although the first puzzle (a "walkthrough") clearly demonstrated inputs and outputs, most players follow the instructions without recognizing the purpose of what they're doing. We partially resolved this by including a video tutorial that focuses on the high-level goal of *SpaceChem* -- transforming inputs to outputs -- giving the player some context when they reach the first walkthrough.

**We failed to make the game start simply enough.** To build a level in *SpaceChem*, it's necessary to build an entire "loop" with the minimum following components: input, grab, arrows, drop, and output. Considering most players barely understood their goal, forcing a minimum of five different tasks on them from the beginning is overwhelming. This problem could have been fixed by changing the design space to have a smaller "minimum possible solution", although it wasn't much of an option post-release.

**We exposed too many details too quickly.** *SpaceChem* is filled with lots of details and rules, such as what can be moved, what can be predicted, and what is required to solve a puzzle. Although the interaction of these rules makes *SpaceChem* open-ended and emergent, it also makes it confusing, especially when players misinterpret what they see.

A common example of this concerns "waldos", *SpaceChem*'s programmable manipulators. The default configuration for a reactor consists of a red waldo at the top and a blue waldo at the bottom; because this is the default setup, many players assume that the red waldo can only be used on the top and the blue waldo can only be used on the bottom. We mitigated this and similar cases by tweaking the tutorial puzzles to include counter-examples, although the problem is by no means fixed.

**We used too much text to explain things.** There were many situations where we were forced to use text to explain rules and nuances of the game. In retrospect, this is a clear sign that we needed to change the game, not explain it more forcefully.



Atoms and Molecules

ATOMS are the building blocks of all matter.

Atoms of the same ELEMENT have the same properties. Oxygen, carbon, and iron are examples of elements.

MOLECULES consist of one or more atoms joined by BONDS.

A molecule is manipulated by a waldo as a single unit. Water, methane, and salt are examples of molecules.

These molecules are all the same.

These molecules are all different.

The identity of a molecule is determined by the type of its atoms and their bonds. Rotation and the angle of bonds do not change the molecule's identity.

**You lost me when you started talking about molecules.**

## Conclusion

All things considered, the development and release of *SpaceChem* went much better than anyone on the team anticipated. I had originally hoped to cover our costs and make a little money, but ended up leaving my job and starting a game studio to work on new titles. New titles, I might add, that are hopefully more accessible than *SpaceChem*!

## Data Box

- **Developer:** Zachtronics Industries
- **Publisher:** Zachtronics Industries
- **Release Date:** January 1, 2011
- **Platforms:** PC (Windows/Mac/Linux), iPad, OnLive
- **Number of Developers:** 7
- **Length of Development:** 1 year

- **Budget:** $4,000 and *lots* of free time
- **Lines of Code:** 17k (game) + 5k (utility)
- **Development Tools:** Visual Studio (C#), Subversion, MonoTouch
- **Fake Elements Invented:** 4