

The background is a dark, cosmic scene. In the center, a large, bright, glowing ringed planet, similar to Saturn, is partially obscured by a large, dark, circular shadow. To the right, a smaller, bright celestial body is visible. The overall color palette is dominated by deep blacks, greys, and vibrant oranges and reds from the planet's rings and the surrounding nebulae.

CONVECTION

ADAM ANDERSON
TONY IANACONE
REECE RENSINK
CHRIS MOLINI

FRACTURE

GAME

OVERVIEW

Convection is a 2D, "bullet hell" shooter game where the player controls a space fighter ship combating waves of vicious enemies. The action is intense and frenzied, and pushes the Game Boy Advance to its limits. As the player battles enemies, he can also pick up power-ups that heal, boost his attack, or give him missiles.

As the player fights their way through the game, towards the inevitable confrontation with the powerful Boss, they rack up points by defeating enemies, and increase their score multiplier by avoiding damage.

CONTROLS

D-PAD: Move ship. The ship has velocity and acceleration, and you must control the ship appropriately.

A: Fire. Hold down for a continuous attack.

B: Release missile. The missile homes in on the closest enemy, and releases a large explosion on contact. This explosion devastates enemies in the area. (This attack requires ammo.)

L-R: Switches main attack between the standard "stream" of weak bullets, or to the shotgun attack if ammo is available.

START: The "play game" button. Begins the game on the starting menu. When the player dies, they can hit continue to continue playing (with only half their score). Also, it pauses the game.

SELECT: The "quit game" button. Returns the player to the main menu when killed, and from credits.

POWER-UPS

There are a series of power-ups in the game. They are gained from defeating enemies.

HEALTH



Restores some health to the player. This is a heal-over-time effect.

SPREAD



Gives the player shotgun ammo. Max ammo is 50.

MISSILE



Gives the player two additional missiles. There is a max capacity of 6.

ENEMIES

The different types of enemies are presented here, along with a description of their attacks.

BEE



Flies around the top of the screen, reaching a point. Upon selecting a position, they pause, and then fly quickly down the screen.

PIERCER



Swerves around the screen, occasionally firing a projectile. Neither sturdy nor damaging, but can gather in large swarms.

SPINNER



Bounces around the top of the screen, occasionally releasing a trio of bullets that slowly move down the screen, spreading out as they go. The bullets are neither fast nor hugely damaging, but in swarms they fill the screen and are difficult to avoid.

SHIELD



Randomly moves about the top of the screen, occasionally firing a series of bullets that are aimed directly at the player. Each bullet does significant damage, and these projectiles can be difficult to avoid.

BOSS



A vast, incredibly tough foe that has a dizzying number of attacks. To find out what they are and how to defeat him, you gotta play the game!

CREDITS

Art: Christopher Molini

Sound:

Newgrounds

For reuse and distribution please see

<http://creativecommons.org/licenses/by-nc-sa/3.0/>

Music list

Fire Aura.wav	by Kid2Will
Short_Bass_Preview.wav	by Kid2will
DeathHunt_5_21.wav	by Kid2will

freesound.org

For reuse and distribution please see

<http://creativecommons.org/licenses/sampling+/1.0/>

Music list

Prophets_Last_Words.wav	by Flick3r
Tranceporter_2.wav	by Flick3r

Sound Effect list

Coachgun_Fire2_dv.wav	by PhreaKsAccount
plascanh.wav	by inferno
smallas.wav	by inferno
hvyilas.wav	by inferno
largex.wav	by inferno
Hardkick_23.wav	by Flick3r
Health.wav	by Flick3r
PierceShot.wav	by Flick3r
feebleefforts.wav	by james duckett
flamewind.wav	by scarbelly25
k9dhhPulseKick.wav	by starpause
060102swaveEnsPadNoteC2.wav	by starpause
pulsing_increase.wav	by FreqMan
DSRXPLOD.wav	by wildweasel
DSGETPOW.wav	by wildweasel
Gunshot_and_reload.wav	by gcmax
Gunshot_and_reloadV2.wav	by gcmax
small_rocket_flybys_and_explosion	by belloq
FireRocket.wav	by belloq
rocketexpl.wav	by nthompson
glock_reloadsequence.wav	by Vosvoy

DEV LOG

GENERAL

When choosing our game, we decided on the idea of a space shooter for a few reasons.

- 1: We did not need complex terrain or hitmaps, which would both free up processor time and make coding easier.*
- 2: We would be able to be very efficient with our graphics.* Space ships are mostly static, and so they don't need very complex animations. As it worked out, we can have all of our sprite graphics and GUI loaded at once.
- 3: Our AI would be simpler yet more interesting.* When controlling characters from a top-down, abstracted view like we have, we can make the attacks far more pattern-based and get more interesting and challenging attacks with simpler code.
- 4: Bullet hells are fun!* We wanted to make a game we would enjoy playing. (We succeeded, too, which is nice.)

All in all, we chose the game concept because it seemed like the right mix of fun product and (relatively) simple development process.

We divided up work as following:

- Reece:** Sound effects and music.
- Adam / Tony:** General game logic.
Projectiles, sprite handlers, Waves, power-ups, etc.
- Chris Molini:** Art, AI and some enemy design.

We all worked well on our assigned positions, and so that was a good thing.

DIFFICULTIES

In any large programming undertaking, there are always frustrations, difficult bugs, and other difficulties and disappointments. In general, we had trouble in these areas:

- 1: VisualHAM.** In hindsight, VHAM was a real struggle to work with. Troubles compiling on Windows 7, and seemingly inconstant performance and compilation results made working with the SDK a pain. Compounded with the outright bugs in the program was the limited control system for the SDK - no refactoring, declaration checking, etc. These didn't cause problems per se, but hampered productivity.
- 2: The GBA.** As we moved forward with the game, we started running into performance and memory problems - i.e., we didn't have any of either to spare. There was only so much we could do about performance issues, and so we ended up just letting the framerate drop when many objects were onscreen, but we began running into problems with memory. This was a frustration for the last two weeks.
- 3: C.** Programming a game in a non-object oriented language is difficult, especially with strict processing and memory constraints. Of all the issues we ran into, this issue - figuring out how to have a dynamic, flexible setup with the limitations of the language we were using - was resolved the most completely. Our end setup is flexible, dynamic, and scales as well as can be expected.

MISC

A few other notes on our development:

SOUND

Sound proved troublesome. We got a single song looping fairly early in the development process, and then we were able to get a longer one in shortly thereafter. Unfortunately, we seemed to run into a brick wall - we couldn't figure out how to have two sounds playing at once. However, literally a few days before the due date, we had a breakthrough and fitted in all of the sound effects we needed.

GRAPHICS (And why they're simple)

As mentioned, one of the reasons we settled on Convection for our game was the simplicity of the technical details. An area where this shows is in our graphics - not the number of them (which has already been mentioned), but rather the technical simplicity.

For example, the game features a scrolling starfield, as if the player is flying through space at an enormous rate (which is, in fact, what we were going for). This effect, which loops throughout the entire game, is simply three very simple backgrounds scrolling down the screen. This not only looks good, but gives us a relatively plain field to place our sprites, which we made VERY brightly colored (to offset the simple background and put the focus on the action).

And, then there's our sprites. We actually don't use any scaling or rotating - we don't need to (though we did figure out the code to do so). Every sprite in the game is straight up and down, and solely changes its look via animation. This both looks better and simplifies our lives.

FINAL THOUGHTS

It was fun - challenging, but fun.