**GlitchHound Games**

**Design Document for:**

# Chaos Directive

**One Liner, i.e. The Ultimate Racing Game**

“Something funny here!”™

All work Copyright ©2011-2012 by GlitchHound Games

Written by Charlie Brensinger

Version # 1.00

Tuesday, January 31, 2012Table of Contents

Chaos Directive 1

Design History 5

Game Overview 6

Philosophy 6

Philosophical point #1 6

Philosophical point #2 6

Philosophical point #3 6

Common Questions 6

What is the game? 6

Why create this game? 6

Where does the game take place? 6

What do I control? 6

How many characters do I control? 6

What is the main focus? 6

What’s different? 7

Feature Set 8

General Features 8

Multi-player Features 8

Editor 8

Game play 8

The Game World 9

Overview 9

World Feature #1 9

World Feature #2 9

The Physical World 9

Overview 9

Key Locations 9

Travel 9

Scale 9

Objects 9

Weather 9

Day and Night 9

Time 10

Rendering System 10

Overview 10

2D/3D Rendering 10

Camera 10

Overview 10

Camera Detail #1 10

Camera Detail #2 10

Game Engine 10

Overview 10

Game Engine Detail #1 10

Water 10

Collision Detection 10

Lighting Models 11

Overview 11

Lighting Model Detail #1 11

Lighting Model Detail #2 11

The World Layout 12

Overview 12

World Layout Detail #1 12

World Layout Detail #2 12

Game Characters 13

Overview 13

Creating a Character 13

Enemies and Monsters 13

User Interface 14

Overview 14

User Interface Detail #1 14

User Interface Detail #2 14

Weapons 15

Overview 15

Weapons Details #1 15

Weapons Details #2 15

Musical Scores and Sound Effects 16

Overview 16

Red Book Audio 16

3D Sound 16

Sound Design 16

Single Player Game 17

Overview 17

Single Player Game Detail #1 17

Single Player Game Detail #2 17

Story 17

Hours of Game-play 17

Victory Conditions 17

Multi-player Game 18

Overview 18

Max Players 18

Servers 18

Customization 18

Internet 18

Gaming Sites 18

Persistence 18

Saving and Loading 18

Character Rendering 19

Overview 19

Character Rendering Detail #1 19

Character Rendering Detail #2 19

World Editing 20

Overview 20

World Editing Detail #1 20

World Editing Detail #2 20

Extra Miscellaneous Stuff 21

Overview 21

Junk I am working on… 21

“XYZ Appendix” 22

“Objects Appendix” 22

“User Interface Appendix” 22

“Networking Appendix” 22

“Character Rendering and Animation Appendix” 22

“Story Appendix” 22

# Design History

This design history is to provide a written list of changes made to the design, along with the dates the changes were made and the reasons for making the change.

## Version 1.00

Version 1.00 is the first version of this document to be produced. Work began on May 6th, 2011.

# Game Overview

## Philosophy

### Sidescrolling Space Shooter

*Chaos Directive* is to be a quality addition to the ranks of Sidescrolling Space Shooters along the lines of past titles as *Thunder Force III* (the game’s primary inspiration), *Gradius*, and similar titles. It aims to bring the best of modern technology to bear with gameplay as enjoyable as the classics.

### Procedural Generation

Levels should be generated using a pseudorandom procedure. Games such as *Beat Hazard* and *Audiosurf* have driven the idea of procedural levels using seeds which can be shared and replayed. Seeds for *Chaos Directive* can be a file of any filetype or generated by the game. In the case of files, the game will create a hash (of sorts) which will enable sharing of the seed over the internet without sharing files.

Unlike *Beat Hazard* and *Audiosurf*, the content of the files chosen does not affect the pacing of the levels.

### Open for Modding

Assets used in the game will be easily added to and extended by the end user. Where possible, this practice will be enabled in such features as high scores and competition.

## Common Questions

### What is the game?

*Chaos Directive* is a modern take on classic space shooters, bringing modern features such as procedural levels, upgradeable ships and weapons, customizable avatars, and up-to-date graphics to a long-overlooked genre.

### Why create this game?

There have been few modern entries into the space shooter field recently, and none of them as replay-friendly as a procedurally generated version. Modern amenities like customization and persistent upgrades haven’t yet graced this market, either. There is likely to be a respectable market for this game, if it can capture the essence of classics properly.

### Where does the game take place?

Deep in the vestiges of extra-dimensional space, the game revolves around a plot between an evil group known only as “Chaos” and those who bravely stand against it. Chaos has begun streaming bits and pieces of information from another dimension, but in the process inadvertently converted their entire universe into spotty 3-dimensional spacetime! (Yes, as in visually 2-dimensional.) Join the forces of good and return your universe to its prior 4-D glory, and send that extra-dimensional pollution back to where it came from.

Honestly, the game is not exactly planning to have a campaign component, but a preliminary one would look something like that above. (Credit to Jonathan Stoebe for the concept behind that.)

**If there’s no campaign, is there any sort of progression?**

Yes. The universe of Chaos Directive has been invaded by the filesystem of your PC. Galaxies have been merged with drives, sectors with directories and sub-directories, systems with archives [if possible] and bottom-level folders, and finally planets and anomalies with individual files. Chaos has established a base in a core system, and the freedom fighters have a main station in the outskirts. You can jump to any file at any time while gathering funds and equipment to finally make your way to conquer the forces of Chaos at their main base.

Between missions, you will return to your station to repair, re-configure, buy and sell equipment, and launch the next mission.

[Note that while the breakdown of the filesystem may be too large a task, other alternatives exist that will allow for a similar progression]

**Can’t that be combined with the story?**

I never really put two and two together until now. Yes, probably.

### What do I control?

The player controls their own spaceship, which is customizable (color-by-numbers, roughly), upgradeable (randomly-generated loot), and exchangeable (multiple base ships with different stats). The player will have control over what the ship shoots and where it shoots from (forward, back, above, below, for example), and will control the ship vertically and horizontally. As the ship approaches either of the four sides (or the four corners), the camera will move to show more of the surrounding environment.

### How many ships and weapons can I have?

While the exact amount is currently unknown, the player will be based out of a station which will have access to equipment, ship, and ammunition suppliers, as well as a hangar which can store currently-unused ships and equipment. Failing to complete a level can result in loss of certain pieces of equipment and/or entire vessels.

### What is the main focus?

As it is currently planned, the game is intended to be a casual, chill-out style game with internet high scores. Potential multiplayer (including both co-op and PvP) is not entirely out of the question. The hope is to have addicting gameplay without falling into the age-old trap of repetitive levels.

If a story mode is added, the individual levels will maintain their procedural generation, with cutscenes, text, and potentially voice-overs to move the story along.

Alternate game types (rather than standard shoot-the-enemy-and-collect-powerups) are in preliminary consideration.

### What’s different?

Procedural Generation – Level generation consists not only of enemies and their attack patterns, but terrain which must be navigated toward a level exit, and drops (including power-ups and equipment drops).

Customizable Ships – The player’s ship can be customized both in looks and loadout, an uncommon feature in this genre.

Internet Leaderboards – Players may upload their scores, and level seeds, to allow players to try beating them. Seeds can be peer-reviewed for fun-factor to ensure competition arises on the most enjoyable levels.

Replays – The game will enable saving replays for playback. If high scores tables are tampered with, the replays may be able to offer a defense against hacking. IE: if the computer doesn’t wind up with the same score at the end of the replay, (especially in the case of a large discrepancy) the score is probably hacked and is thrown out/sent for review before posting.

# Feature Set

## General Features

Procedural Levels

Customizable Ship with Storage for old parts

2D Graphics

32-bit color

## Multiplayer Features [If plausible]

Up to 4 Players (multiple games) or 2 players (splitscreen/shared-screen)

Co-op and PvP

Different Multiplayer-only game modes

Chat support [If plausible]

## Customization

Comes with Modding toolkit (easily-moddable even without)

Ships can be colored on multiple layers

Music can be played from the player’s library [If plausible]

Aspects of file used as seed can appear in the level (e.g. a picture appears in the background) [If plausible]

## Gameplay

Limited Twin-Stick Shooter: weapons have firing arcs, and only those weapons covering the direction indicated will fire.

Procedural Levels for ultimate replayability (Literally billions of potential seeds, with multiple seeds used on each level)

Multiple Enemy types and Behaviors

Multiple Behaviors per Enemy type

Boss and Mini-Boss Enemies

Mazelike Levels that require exploration to proceed through

Real Penalties for mission failure (on selected game modes)

Ships can be outfitted with collected and purchased parts

Weapons and Equipment can be upgraded and configured

Randomized base stats for ships and equipment enabling a rich and dynamic loot system

Progression of a “Universe” by exploring your filesystem to collect powerful upgrades to finally combat Chaos at its source

Share high scores and replays online

Share levels without sharing the files that generated their seeds

# The Game World

## Overview

The main overview of the world will show a list of galaxies. When the player chooses a galaxy he or she will see a graphical representation of a galaxy [depending on complexity, it may be catered to what’s within, or it may be a static picture divided into grids]. Selecting a sector of the galaxy will then give you a list of systems [complexity, again, will determine if they’re selectable grid-squares or if they can be more representative], and within the system you will see a list of planets and anomalies which are represented by potential seeds of the file being used.

## Generated by the Player’s File System

The player’s main drives or partitions are to be represented as galaxies, with folders narrowing down to sectors and subsectors and finally files serving as systems. As all but the smallest files can be used to make multiple level seeds, a system can be generated using as many seeds as are available. To minimize complexity, systems will likely be limited to 10-15 points of interest, each of which will be a playable level.

## Mothership & Hangar Bay

### Overview

Between levels, the player will be situated within a mothership. This ship will have storage for player ships and equipment, and will serve as a source for story and optional missions.

### Key Locations

Canteen – Source of optional missions (speak to patrons)

Briefing Room – Source of storyline missions (speak to officers)

Command Deck – Mission selection and statistics, acts as hub, has terminals for ship/equipment

### Travel

The player can either access any of the locations from the command deck (returning there when they’re done), or jump to any location of their choice by using a quick bar.

All of the locations will consist of quality 2D backdrops with interactive items (doors/lifts representing paths to other locations when in the hub, or people to talk to, consoles to interact with in the other locations) highlighting when the cursor hovers over them.

### Objects

Command Deck:

Computer Terminals for Battlelog (stats, replays), Outfitting, Hangar Management

Doors for Briefing Room and Canteen

Main viewport to open interface for mission selection

Canteen:

Randomly generated NPCs with optional missions.

Door to Command Deck

Briefing Room:

Officer NPCs offering story-based missions.

Computer with intel for currently selected mission.

Door to Command Deck

## Rendering System

### Overview

The game will use sprite-based 2D rendering thanks to XNA, with the possibility of dynamic lighting and probable inclusion of particle systems for explosions and other effects.

## Camera

### Overview

Camera will be fixed to a side-view of the ship. There are two options, depending on which system can be brought to life: on-rails or follow.

### On-Rails

The camera will follow a set path. The player can move the ship within this viewport.

### Follow

The camera will follow the player’s ship as it is moved around the level.

## Game Engine

### 

### Overview

??

### Game Engine Detail #1

??

### Water

Probably not included as anything more than scenery.

### Collision Detection

Per-pixel preferably.

## Lighting Models

### Overview

Potentially dynamic lighting, using weapons and engines as light sources (plus a level-included one). Otherwise, no real model for lighting.

# The World Layout

## Overview

The world exists on two main levels, the overworld and the stage. The overworld consists of a galaxy broken down into sectors, systems, and ‘anomalies’ which can be planets, rogue asteroids, asteroid belts, and similar.

## (Over)World Map

Each galaxy should have approximately 0-5 nebulae (randomly placed), which would each be contained in one of its sectors. A galaxy should have in the region of 30-35 sectors in whole. Each sector should consist of about 20 systems of interest (mostly those which contain at least one habitable planet) and each system should consist of 3-7 anomalies.

Each galaxy should also contain (hidden within randomly-selected systems) 1-3 bonus levels. These would provide the lucky (and/or persistent) player with noticeable bonuses in cash or equipment, but would have to be carefully balanced to ensure gameplay is not hurt by the bonus.

## Stage Map

A stage map is procedurally generated using the anomaly as a template. Selecting a planet will mean that that type of planet’s template is loaded, so a gas giant will produce a level where the “terrain” (to be interpreted as that part of the stage which will destroy the ship if the player runs into it) consists of acidic clouds, whereas a temperate planet will have a forested terrain, an ocean planet could have water as the terrain (or could be underwater and have the seafloor as its terrain), and asteroids or stations will have a tileset and structure to match.

# Game Characters

## Overview

The character (or, more accurately, avatar) is the player’s vessel. While vessels can be changed, indicating that there is an underlying persona (the pilot) which serves as the player’s character, the appearance of the pilot is not utilized at any point.

OPTION: Perhaps have the player personalize a “bust” which can appear in conversations. Consider a Harvest Moon 64 where the player is not a silent protagonist. The current speaker’s bust appears in the side of the text box. Also consider Rune Factory 3 (<http://www.capsulecomputers.com.au/wp-content/uploads/Rune-Factory-3-A-Fantasy-Harvest-Moon-02.jpg>) where torso shots of each of the characters conversing appear.

## Creating a Character

Every player starts with a basic ship (perhaps chosen from a few base templates, such as the fast-but-weak, strong-but-slow, and balanced stereotype) and a few pieces of basic equipment, as well as enough cash to do some customization.

Customization varies from free (and cosmetic-only) repainting of the ship, which is made up of multiple layers which can each have a color applied to it in order to allow personalization, to costly ship and equipment upgrades. Equipment upgrades will be cheaper than ship upgrades because equipment, while transferable between ships, can be damaged and even destroyed in game play much more often than ships can be.

Equipment upgrades will change the characteristics of the equipment, from power usage to fire rate to damage dealt (or absorbed). Ship upgrades will focus on engines, reactors, and structure, which will change its speed/maneuverability, power max/recharge, and health/damage reduction respectively.

See the section on Weapons and Armor for full details on what can be upgraded for equipment.

See the section on Ships for full details on what can be upgraded for ships.

## Enemies and Monsters

Enemies will consist of vessels and installations controlled (or, more accurately, brought in) by Chaos. Some of these will be ground vehicles which will follow the terrain but can move to target the player. Others will be ground defenses which are static but fire more powerful weapons (and/or have more armor) than ground vehicles. The majority, however, will be aerospace forces which can range from drones (fighters smaller than yours), vessels on a scale (and customization level) similar to the player’s, and battleships which are much larger and vary in size and function.

Compare to Thunder Force III, where drones are the enemies that come in waves and take 1-2 hits to kill, vessels are not really paralleled, and battleships are mini-bosses and generally things bigger than you.

Finally, there should hopefully be bosses. Due to the added work of creating a boss (read: asset creation more than anything), there will likely only be a few of these at launch, probably something like 5 per galaxy (maybe even 3 per galaxy, randomly selected from 5 possibilities).

Variation among enemies will be done using a combination of palette swaps and fire-type changes. While drones may have the same structure, a different color will fire a different type of weapon, and exhibit different behavior. This can be utilized even for bosses, so even 3 or 5 bosses can result in many different play styles.

# User Interface

## Overview

Most of the vital details will be shown to the player on the bottom portion of the screen. This means ship status readouts in three major panels: the Ship Status Panel, the Control Panel, and the Sensor Panel.

## Ship Status Panel

The ship status panel is a visual overview of the ship’s current state. It consists, primarily, of a rectangular display with an alert bar at top (positioned above the display to keep it closer to the main viewport and therefore the player’s field of view). The main display shows the ship as it is currently loaded. The background will be colored in using a gradient from light blue to pure black, denoting full to depleted shields, respectively. The ship itself will be colored using a gradient from green to red, indicating the hull integrity. Individual components will be colored the same way, with the addition of black in the case of a destroyed component.

The alert bar will be divided into three sections, which, from left to right, are the Shield, Component, and Hull alert segments. These will be gray sections, stylized to look as if they are separate panels, with dark gray shapes within them (by default). These shapes will be a shield, a gear, and a triangle, respectively. The triangle needs some explanation: the only recognizable symbol for “armor” would be a shield, which in this context would be confusing as it could represent either armor or shields. By using the shield for the shields and referring to hull integrity rather than armor, we can alleviate most of this confusion. The hull is therefore represented by a standardized warning symbol, the triangle. This also works because the hull is the most critical of the three – only when the hull is completely destroyed is the ship lost.

Each segment’s shape will remain gray until the integrity of that item (all instances of that item, in the case of the Component section) drops to 50% or lower. When the integrity is at a level L 25% < L <= 50%, the icon will change to yellow. When 0 < L <= 25%, the icon will change to orange. Finally, when the item is at 0 (except for the hull, as the level will end) the icon will turn red.

The Component icon has two special features. Firstly, it will also indicate the number of components of concern (at or below 50% health), displayed in the center of the gear. Finally, if at least one component has been destroyed, a red alert triangle will appear behind the gear icon.

## Control Panel

The control panel displays the current settings of a number of components which drain energy. It consists of one bar, two dials, and 4 subpanels. If limited-ammunition weapons are implemented (along the lines of megabombs), the current quantity of these will also be displayed in a small readout.

The bar, located at the top of the panel, represents the maximum stored energy of the ship, with the filled section representing the current percentage of that currently in storage. If no energy is stored, no equipment can do anything, except in the case of already-charged weapons (as these hold enough charge for a single shot).

The dials represent the current settings of shield recharge and maneuvering jet power. The shields have three settings – off means that the shields will not recharge when depleted, balance means that the shields will recharge using only excess energy being produced by the reactor (in this mode, not firing your weapons will result in faster shield recharge), and top will always attempt to maintain shields at full capacity regardless of the energy balance. The jets also have three settings, low, medium, and high. These affect how quickly the ship moves, with faster movement draining more energy.

The four subpanels correspond to weapons systems. At the top of each panel is the corresponding weapon’s name. In the middle is a toggle switch, which can be either on or off – on top of providing the visual element of a switch, this will also turn on a green ‘on’ light or red ‘off’ light to indicate the weapon’s activation status. A weapon which is deactivated will not fire. If excess power is available, deactivated weapons will charge, but when a deactivated weapon is charged its energy will go back into the main capacitor the next time it falls below full (at a maximum rate of 25% of the weapon’s power per second). At the bottom is a charge status bar. When the bar reaches about 80% filled, the weapon can be fired. Anything in the remaining portion represents an overcharge – it takes twice as long to fill, but is significantly more powerful. The bar will remain red until it reaches the 80%, at which point it will turn green, and when it is completely overcharged the bar will fade between green and white.

## Sensor Panel (?)

The sensor panel, if implemented, is really just a map panel. The map is revealed as the player visits it, but many enemies are visible on the map regardless of that area’s exploration status. This also serves the purpose of alerting the player to enemy movements, as some enemies will roam rather than wait for the player to come to them.

# Ship Components

## Overview

Ship Components are a very important aspect of Chaos Directive. As in most other aspects of the game, the components are procedurally generated. Each component has multiple stats, many of which can be upgraded by the player between missions.

Components are created with base stats falling randomly within a given range; upgrading improves this base stat by a percentage, but is cheaper for stats which were generated lower in the range (incremental, e.g. numbers from 0-10% of the max are upgraded at the same cost.)

Components include weapons, shields, and maneuvering jets.

## Generic Stats (Available for all Components)

* Durability [upgradeable]
* Efficiency (power used per charge) [upgradeable]

## Shield Stats

* Maximum Capacity [upgradeable]
* Full Charge Time (seconds to recharge from 0) [upgradeable]
* Primary Type [cannot be changed – extra damage reduction versus these weapons]
* Depletion Effect [cannot be changed – special effect triggered upon depletion]

## Weapon Stats

* Fire Rate (effectively charges / second) [upgradeable]
* Damage [upgradeable]
* Number of Shots per Charge [upgradable]
* Firing Arc [upgradable – straight (20o), 45o, 90o, 180o]
* Homing [cannot be changed]
* Dual Type [IF present, can be toggled – double strength or double firing arc]
* Overcharge Effect [cannot be changed – special effect upon overcharge]

## Jet Stats

* Move Rate [upgradeable]

# Musical Scores and Sound Effects

## Overview

Hopefully there will be time/budget/connections enough to have a custom score and sound effects. Score is, in this designer’s opinion, a higher priority.

## Music Design

Music will hopefully have an electronic/chiptune feel to it, reminiscent of the games which inspired it. Fast-paced and frantic for gameplay with a nice and balanced main theme for between times.

## Sound Design

Sounds will also take a retro feel, possibly using sfxr to create simple but identifiable effects.

# Single-Player Game

## Overview

The player plays a leading member of an elite mercenary force based in a capital ship that roams the galaxy. His forces have dedicated themselves to eradicating the forces of Chaos by collecting payment from those being suppressed by them. By talking to the player’s superior officers (NPCs), the player can gain new missions to progress the story along. By talking to patrons of the ship’s bar the player can pick up side-missions for extra money and equipment.

## Single Player Game Detail #1

## Single Player Game Detail #2

## Story

Describe your story idea here and then refer them to an appendix or separate document which provides all the details on the story if it is really big.

## Hours of Gameplay

Talk about how long the single-player game experience is supposed to last or what your thoughts are at this point.

## Victory Conditions

How does the player win the single-player game?

# Multiplayer Game

## Overview

Describe how the multiplayer game will work in a few sentences and then go into details below.

## Max Players

Describe how many players can play at once or whatever.

## Servers

Is your game client-server or peer-to-peer or whatever.

## Customization

Describe how the players can customize the multiplayer experience.

## Internet

Describe how your game will work over the internet.

## Gaming Sites

Describe what gaming sites you want to support and what technology you intend to use to achieve this. Perhaps Dplay or TCP/IP or whatever. It is probably a good idea to break the tech stuff out into a separate area, you decide.

## Persistence

Describe if your world is persistent or not.

## Saving and Loading

Explain how you can save a multiplayer game and then reload it. If you can or why this is not possible.

# Character Rendering

## Overview

Provide an overview as to how your characters will be rendered. You may have decided to include this elsewhere or break it out to provide more detail to a specific reader.

## Character Rendering Detail #1

## Character Rendering Detail #2

# World Editing

## Overview

Provide an overview about the world editor.

## World Editing Detail #1

## World Editing Detail #2

# Extra Miscellaneous Stuff

## Overview

Drop anything you are working on and don’t have a good home for here.

## Junk I am working on…

Crazy idea #1

Crazy idea #2

# “XYZ Appendix”

Provide a brief description of what this appendix is for and then get down to business and provide data to the reader.

Here are a few examples of some of the appendices in my latest design…

# “Objects Appendix”

# “User Interface Appendix”

# “Networking Appendix”

# “Character Rendering and Animation Appendix”

# “Story Appendix”

Okay, that’s it. I wanted to spend more time on this and really make it a great roadmap for putting a game design together. Unfortunately it would take a ton of time and that is something that we don’t have enough of in this business. I think you get the idea anyhow. Also, don’t get the impression that I think a design should provide the information in any particular order, this just happened to be the way it fell out of my head when I sat down. Change this template any way you want and if you feel you have improved on it, send it back to me and I can pass it out as an alternative to anyone that asks me in the future.

Good luck and all that!

Chris Taylor