

Converge GDD

Andrew Perrault – June 20, 2015

Table of Contents

1 Overview.....	1
1.1 Setting/Theme.....	1
1.2 Platform.....	1
2 Game Rules.....	2
2.1 The Board.....	2
2.2 Your Turn.....	2
2.3 Capturing Bits.....	2
2.4 Winning.....	3
3 Controls.....	3
4 Screens.....	3
4.1 Game Screen.....	3
4.2 AI Difficulty Selection.....	3
4.3 Title Screen.....	4
4.4 Main Menu.....	4
4.5 Options.....	4
4.6 About Screen.....	4
5 Technical Design.....	4
5.1 The Game Board.....	4
6 Monetization.....	4

1 Overview

Converge is an iPhone board game. The game is similar to traditional Central Asian and African games like Mancala, Oware and Bohnenspiel but with modern styling and slightly different rules to make the game end quicker. The original game is intended for two players but in this version you will always play against an AI opponent. The game is intended to be quick to learn and quick to play but deep enough to be played many times over.

1.1 Setting/Theme

The game board will have a very simple, clean design. The game board will take up most of the screen. There will not be any numerical UI or score. You will be able to tell who is winning by the number of captured game pieces on each side of the board.

1.2 Platform

The primary platform will be iOS. It will target iPhone 6, 5 and 4 as well as iPad.

2 Game Rules

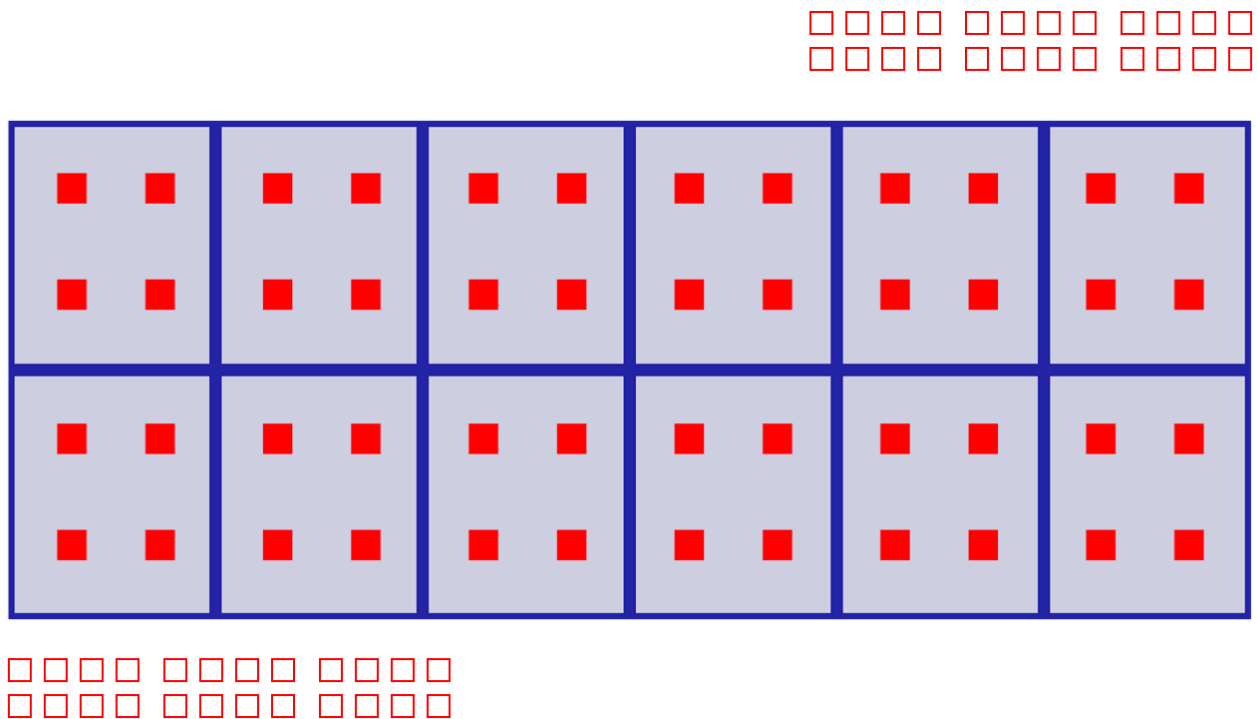


Illustration 1: The game board

2.1 The Board

The board will consist of twelve pools arranged in two rows. You control the pools at the bottom and your opponent controls the pools at the top. At the beginning of the game there are four bits in each pool.

2.2 Your Turn

Your turn consists of selecting one of the pools you control that contains at least one bit. The bits are removed and added one at a time to each pool in a counter-clockwise direction. Continue adding bits to the pools until all of the bits are distributed. If you have enough to go all around the board then you skip the original pool leaving it empty and keep adding bits to the next pool.

2.3 Capturing Bits

When all of the bits are distributed if the last pool to receive a bit is on your opponents side and now contains 2, 4, 8 or 16 bits then you capture those bits and remove them from play. If the next to last pool you added bits to also has 2, 4, 8 or 16 bits you can also capture those bits. You can continue capturing to the right and capturing as long as you only capture your opponents bits and the pool contains exactly 2, 4, 8 or 16 bits. If the last distributed bit is on you side of the board then you capture

nothing. If the last distributed bit is in a pool that finishes with any number other than 2, 4, 8 or 16 then you capture nothing even if other pool contains those numbers.

2.4 Winning

As soon as one player captures at least half of the bits – 24 – then the game is over and that player wins. If at any time all of your pools are empty at the beginning of your turn then your opponent captures all the remaining bits and the game ends.

3 Controls

The game controls are very simple. When it is your turn tap the pool you want to distribute.

4 Screens

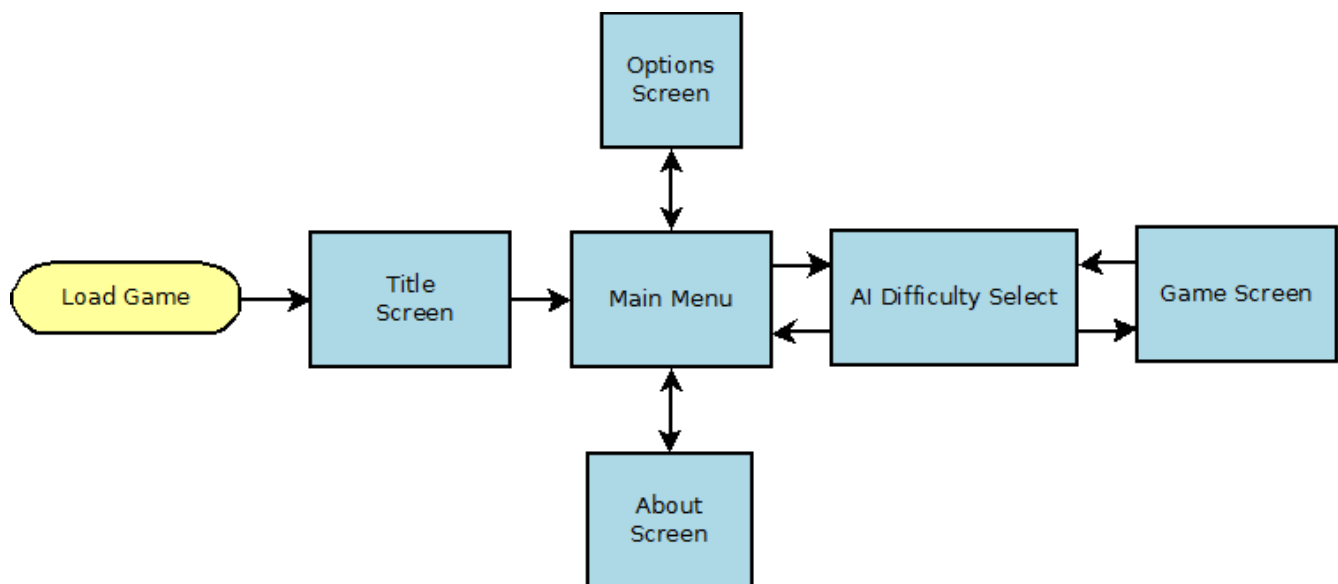


Illustration 2: Game Screens

4.1 Game Screen

The game screen will mostly be filled with the game board. There will be a small icon at the top to return to the main menu.

4.2 AI Difficulty Selection

This screen will let you select easy, normal or hard AI. This screen depend on making the AI difficult enough to make easier levels relevant. Multilevel AI will be low priority for the prototype and this screen may get cut.

4.3 Title Screen

Title screen will be displayed as the game is loading.

4.4 Main Menu

Main menu will have options to play the game, view the options menu or view the about screen.

4.5 Options

Options for sound on/off and music on/off.

4.6 About Screen

This lists information about me and about the licenced assets used in the game.

5 Technical Design

The game will be written in Swift using the Sprite Kit framework. The main menu and options screen will use UIKit.

5.1 The Game Board

Each pool will be represented by a class. The game class will have a doubly linked list shaped like a ring linking the last element to the first. Distributing the bits will be as simple as iterating through the list starting at `selectedPool.Next()`.

6 Monetization

To be competitive this game should be free. The simplest way to monetize the game would be ads. If ads on the game screen seem too obtrusive then they could be added to a loading screen before the game starts. Also the main menu could provide a link to a paid version of the game with no ads.