

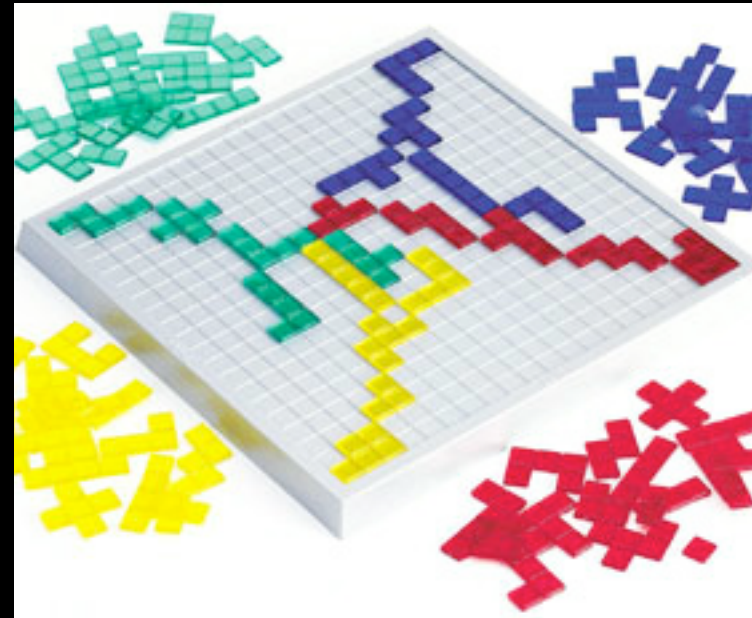


Blackbird Blokus

Alex Hills, Dan Kimmel, Varun Singh, Greg Young

Normal Blokus

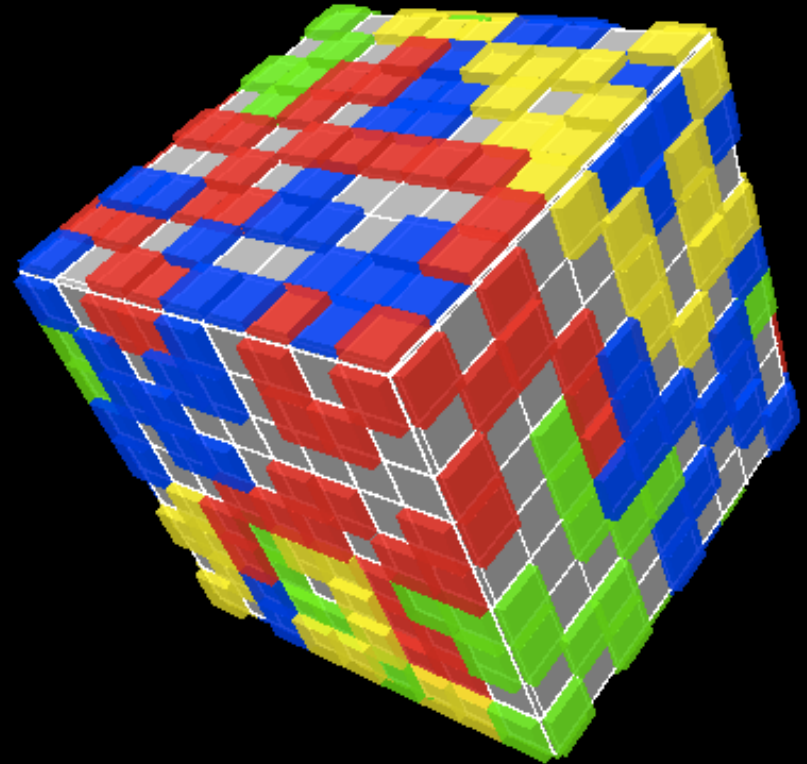
- You play on a 20x20 grid
- Your first move must cover your corner square
- Each subsequent move must touch at least one of your previous moves at a corner and can't touch any of your previous moves on a side



Objective: cover as much of the board with your pieces as possible

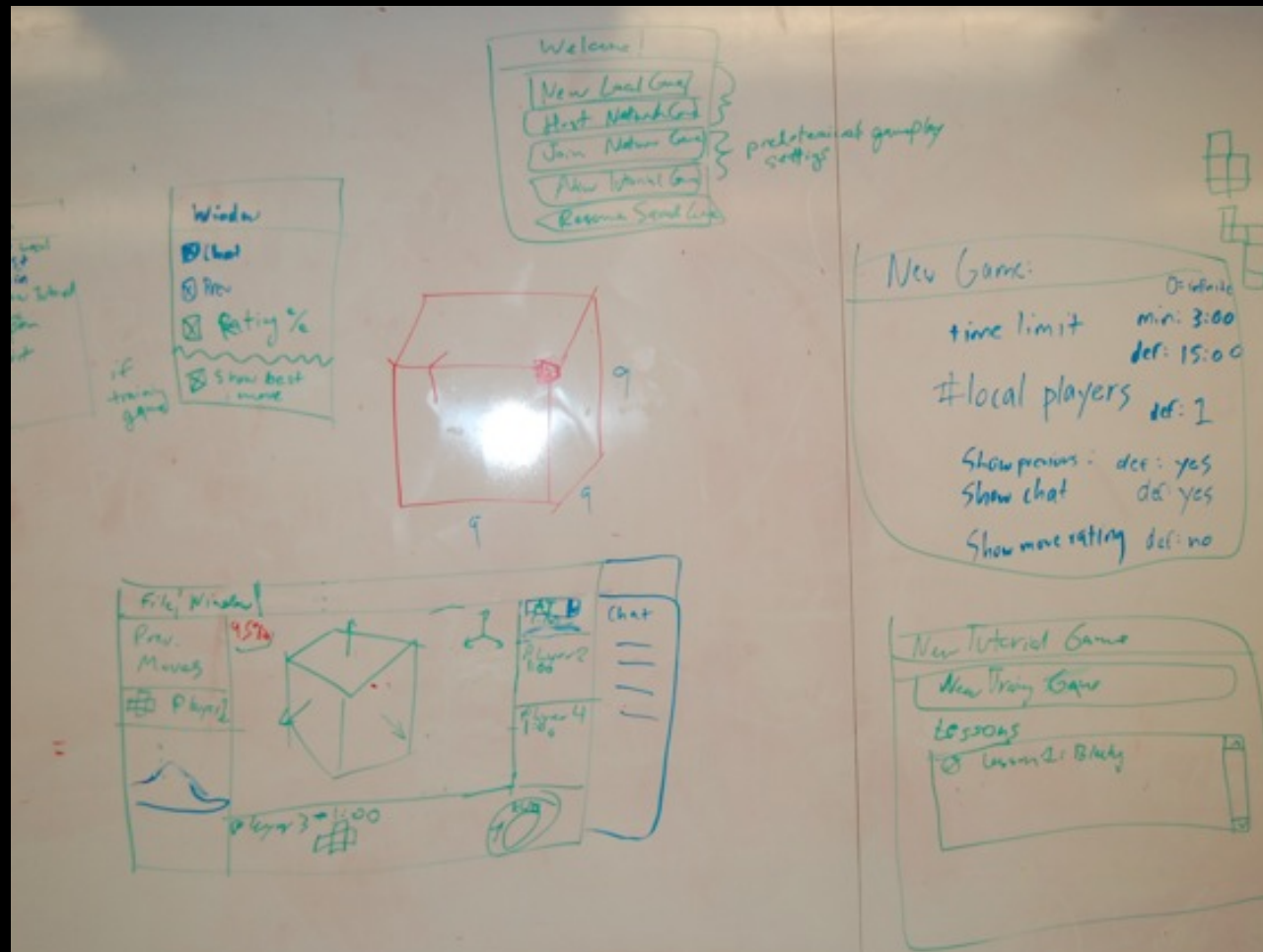
Our Version

- Played on a 9x9x9 cube
- Your starting squares are already filled in
- Otherwise, the same rules about moving apply, and you have the same objective



Requirements

- Fun to play
- Single and multiplayer
 - AI players
 - Networked gameplay
 - Hotseat gameplay
- Simple interface



Specs

Design Decisions

- Player abstraction
 - HumanPlayer, AIPlayer, NetworkPlayer
- GameTree
- Network Design
- GUI Design

abstract class Player

- The key abstraction in the project
- Move `getMove()` throws Exception
- `void removePiece(Shape s)`
- `getColor, getName, getScore, getPieces`

GameTree<M,S ext...>

- M = Move, S = State
 - States are vertices in the search tree (they must be able to calculate possible moves)
 - Moves are edges
- M getBestMove(int minDepth)

Network Design

- Server, multiple clients
- Multithreaded
- NetworkComponent

Live Demo!

Questions?