



HOUSTON SHOWDOWN

TowerDefence Spiel basierend auf der IrrLicht3D Engine

Dominik Hübner

dh028@hdm-stuttgart.de

Medieninformatik Semester 4

SS09

Die Mittel

Programmierung

- Sprachen
 - C++
 - XML
- Tools
 - Microsoft Visual Studio 2008
 - DirectX SDK (XInput-Treiber)
 - Irrlicht3D Engine
 - IrrXML

HOUSTON
SHOWDOWN

Die Mittel

Grafik

- Adobe Photoshop CS4
 - Texturen
 - GUI
- Autodesk 3ds Max 2008
 - Models
 - Animationen
- IrrEdit
 - Szene

HOUSTON
SHOWDOWN

Tower Defence

Herkunft

- Ursprung als Modifikation diverser Strategiespiele (StarCraft, WarCraft)
- Vielzahl an Flash-Games im Internet

Spielprinzip

- Gegner, sog. Creeps, versuchen gegebene Karte zu überqueren
- Spieler muss mittels selbstschießender Türme verhindern, dass die Gegner ihr Ziel erreichen
 - geschickte Platzierung der Türme nötig!

HOUSTON
SHOWDOWN

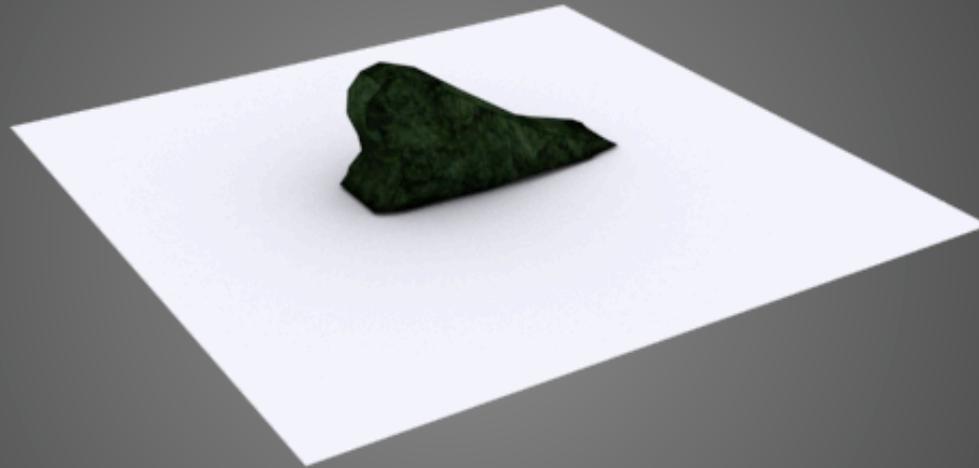
Tower Defence

Varianten

- Singleplayer
 - Standard
 - ❖ vorgebene, nicht bebaubare Wege
 - ❖ Aufwertung der Türme im Vordergrund
 - Maze
 - ❖ freie Fläche
 - ❖ Gegner suchen individuell den Weg zum Ziel
 - ❖ Schwerpunkt auf Platzierung der Türme
 - Hybrid
 - ❖ Mischform aus Standard und Maze
- Multiplayer
 - Tower Wars
 - ❖ Spieler schicken sich gegenseitig Einheiten
 - ❖ Erhöhte Schwierigkeit durch weiteren Kostenfaktor

HOUSTON
SHOWDOWN

Die Gegner



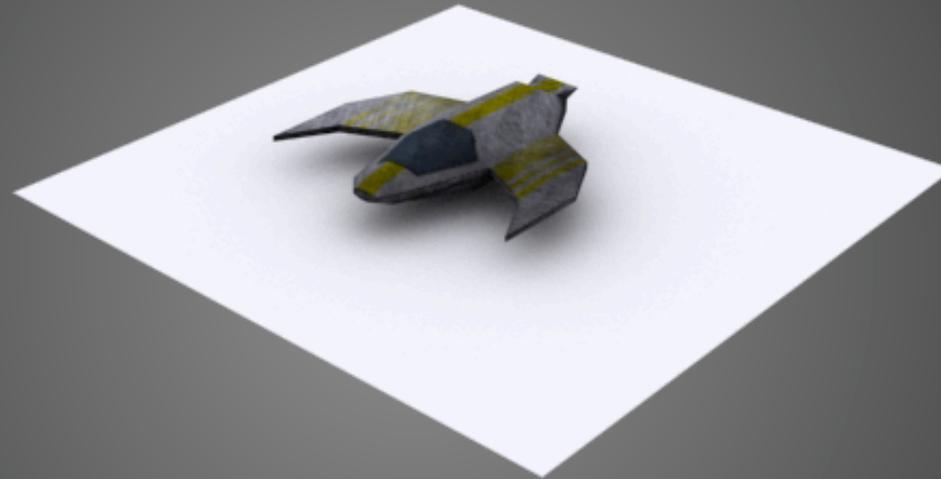
Liquids

einfachste Gegnerart

gewähren dem Spieler im späteren Verlauf Pausen

HOUSTON
SHOWDOWN

Die Gegner



Streamer

Fliegender Gegner
wird nicht von Türmen aufgehalten

HOUSTON
SHOWDOWN

Die Gegner

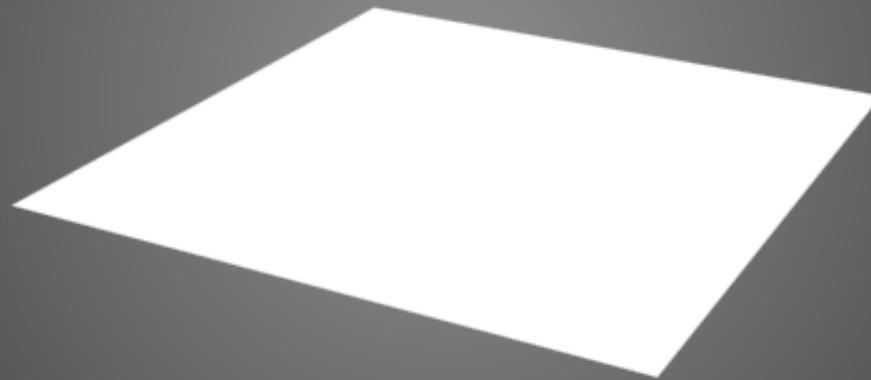


Mechanix

höchste Lebensenergie
relativ langsam

HOUSTON
SHOWDOWN

Die Gegner



Shifter

unsichtbar

wird nur von einem Tower getroffen

HOUSTON
SHOWDOWN

Die Türme



Rapid

schnell
günstig

trifft unsichtbare Einheiten

HOUSTON
SHOWDOWN

Die Türme



Cannon

starker Schaden
trifft nur Bodeneinheiten

HOUSTON
SHOWDOWN

Die Türme



AntiAir

hoher Schaden auf Flugeinheiten
trifft nur Lufteinheiten

HOUSTON
SHOWDOWN

Die Türme



Freezer

mittlerer Schaden
verlangsamt Gegner

HOUSTON
SHOWDOWN

Die Logik

Basis

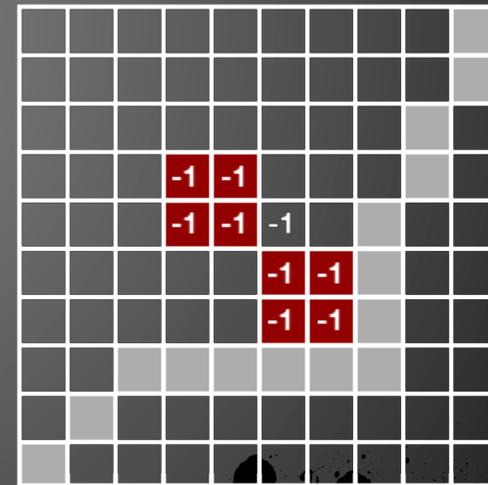
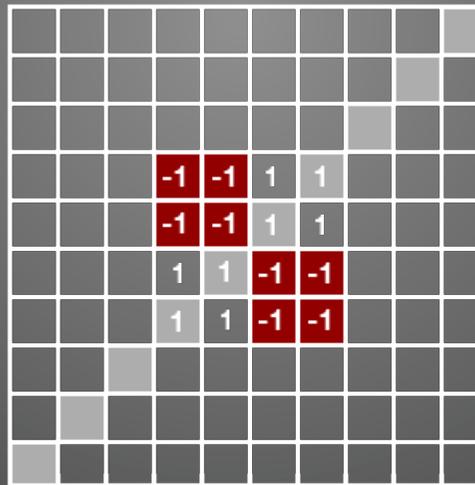
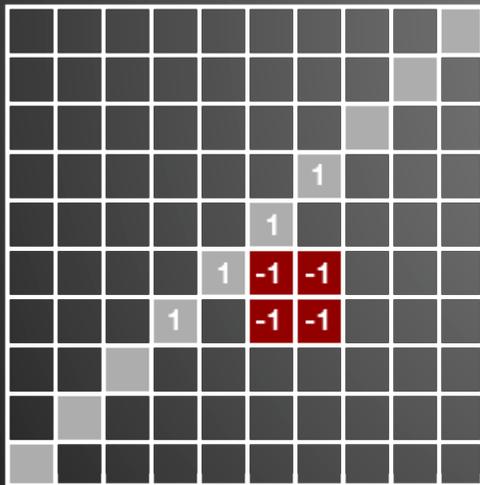
- Anzeige und Logik weitgehend getrennt
 - Engine kann mit wenigen Änderungen ausgetauscht werden
- Vorladen von Grafiken
- Laden des Gegnerskripts
 - XML
 - Timeline
 - Erstellen von Triggern
 - Gegner erscheinen in vordefinierter Abfolge
- Endlosschleife
 - Aktualisierung aller Zustände
 - Gegner
 - Türme
 - GUI
 - Input
 - Szene

HOUSTON
SHOWDOWN

Die Logik

Karte

- 2-dimensionales Array
 - 1: Feld bebau- und begehbar
 - -1: Feld belegt
- begrenzt die Bewegungsfreiheit der Gegner selbstständig



HOUSTON
SHOWDOWN

Die Logik

Gegner

- Künstliche Intelligenz
 - Pfadsuche mittels A*-Algorithmus
 - Pfad wird mittels Liste von Wegpunkten gespeichert
- Aktualisierung des Zustandes in jedem Frame
 - Bewegung
 - Wegpunkte werden nach und nach abgelaufen
 - Letzter Wegpunkt ist das Ziel
 - Überprüfung ob Gegner getötet wurde
 - durch Erreichen des Ziels
 - ❖ Spieler verliert 1 Leben
 - durch Schaden
 - ❖ Spieler gewinnt Geld

HOUSTON
SHOWDOWN

Die Logik

Türme

- Künstliche Intelligenz
 - Zulässigkeit
 - ❖ Werden Gegner eingeschlossen?
 - ❖ Start- oder Zielpunkt blockiert?
 - Auswahl der Ziele
 - ❖ Gegner mit kleinster Entfernung zum Ziel werden priorisiert
- Schussverhalten
 - Schussrate und Schaden abhängig von Typ und Ausbaustufe
- Überprüfung ob sich das Ziel noch in Reichweite befindet
 - Falls nicht: Entfernung sämtlicher Einflüsse auf den Gegner

HOUSTON
SHOWDOWN

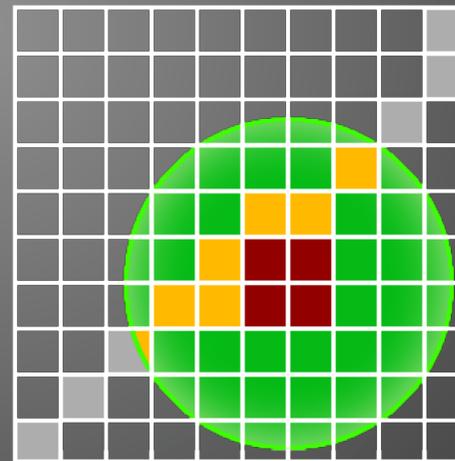
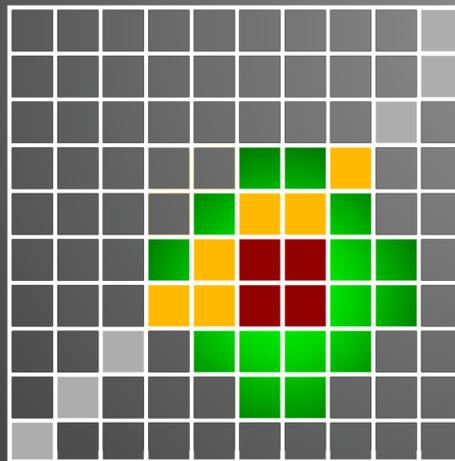
Die Probleme

- Berechnung der KI nicht im Zeitrahmen eines Frames möglich
 - Scheduling
 - ❖ Berechnungen bekommen Rechenzeit zugeteilt
 - ❖ Unterbrechen diese nach Ablauf
 - ❖ Ergebnisse werden erst nach Vollendung verwertet
 - Optimierung der Datenstrukturen
 - Pfadsuche der Gegner
 - ❖ Bewegung immer zu dem Feld mit den geringsten (geschätzten) Kosten zum Ziel
 - Problem: alle bisher überprüften Felder müssten mit jedem Schritt erneut untersucht werden
 - Lösung: Sortierung mittels Binary Heap Priority Queue

HOUSTON
SHOWDOWN

Die Probleme

- Verhalten der Türme
 - Auswahl der Ziele über die (x,y)-Koordinaten der Map-Datenstruktur wirkt auffällig ungenau
 - Gegner werden genau auf Position innerhalb der Schussreichweite ausgewählt



HOUSTON
SHOWDOWN

Die Probleme

- Balancing
 - hoher Zeitaufwand
 - nicht durch einzelne Person ausführbar
- Engine
 - Teilweise keine vollständige Unterstützung versprochener Features
 - ❖ Fehlerhafte Positionierung der Objekte
 - ❖ Fehlerhafte Texturkoordinaten
 - Veraltete Datenformate
 - ❖ Datenformate wie *.md2 (Quake2, 1997) werden wesentlich besser unterstützt als aktuelle

HOUSTON
SHOWDOWN