



# KI IN GAMES

AUS SICHT EINES  
SPIELEENTWICKLERS

# INHALT

- Kurze Vorstellung über mich
- Überblick SimAirport/SimCasino
- Begriff KI in der Spieleentwicklung
- Die komplizierte Welt der KIs
- KI: State Machine
- KI: Needs Based AI

# ÜBER MICH

- Student der Wirtschaftsinformatik an der Uni Hildesheim
- Etwa 3 Jahre tätig in der Spieleentwicklung
- Entwickler und Community Manager bei LV Game Dev
- Beteiligt an SimAirport und SimCasino



# SIMAIRPORT

- Thema: Flughafen
- 2D Grafik; Sicht von Oben
- Agentensimulation mit Fokus auf die „Needs Based AI“
- Erlaubt das Bauen und Managen eines Flughafens



# SimAirport



# SIMCASINO

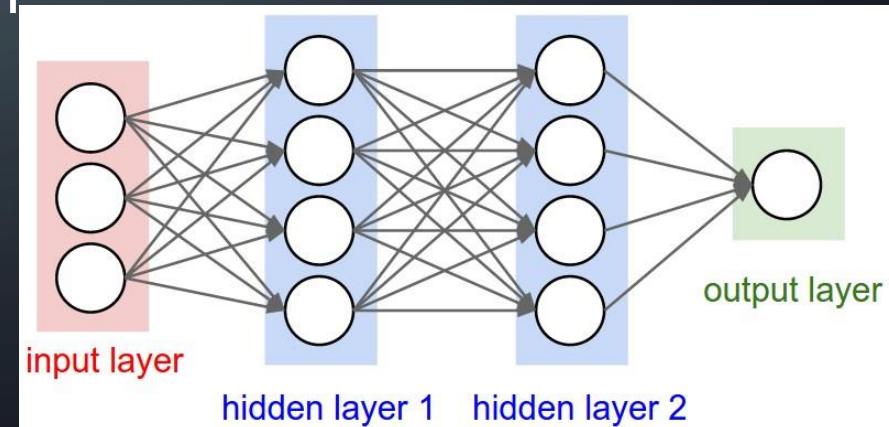
- Thema: Casino, Hotel, Resort
- 3D Grafik; Sicht aus dritter Person im Baumodus, erste Person im Erkundungsmodus
- Agentensimulation mit Fokus auf die „Needs Based AI“
- Erlaubt das Bauen und Managen eines Casinos inklusive Hotel, Kino und ähnlichen Einrichtungen



# BEGRIFF KI



- Genutzt für alle Simulationssysteme die "Agenten" steuern
- Agenten sind z.B. simulierte Menschen, aber auch sonstige Gegner
- Maschinelles Lernen eher weniger in Verwendung
  - ML basierte KIs könnten zu stark werden
  - Manche KIs sollen berechenbar/dumm sein





# DIE KOMPLIZIERTE WELT DER KI

- Schwer zu entwickeln ohne auf Probleme zu stoßen
- Komplexeres Verhalten führt gerne zu außergewöhnlichen Fehlern
- Unerwartete Ausprägungen
- KIs müssen auf ein Spiel angepasst sein und sollten nicht zu stark sein
  - Wenig Wiederverwendung zwischen Spielen
  - Spaß meist wichtiger als perfektes Verhalten

# UNERWARTETE KI - GHANDI

- Sid Meier's Civilization (das Original aus 1991)
- Gandhi bekannt als Pazifist
- Bei Bedrohung antwortet er mit Nuklearen Waffen
- Tiefes Designproblem



Bild aus Civilization 5

# HELFER A IST VON EINEM OBJEKT BLOCKIERT

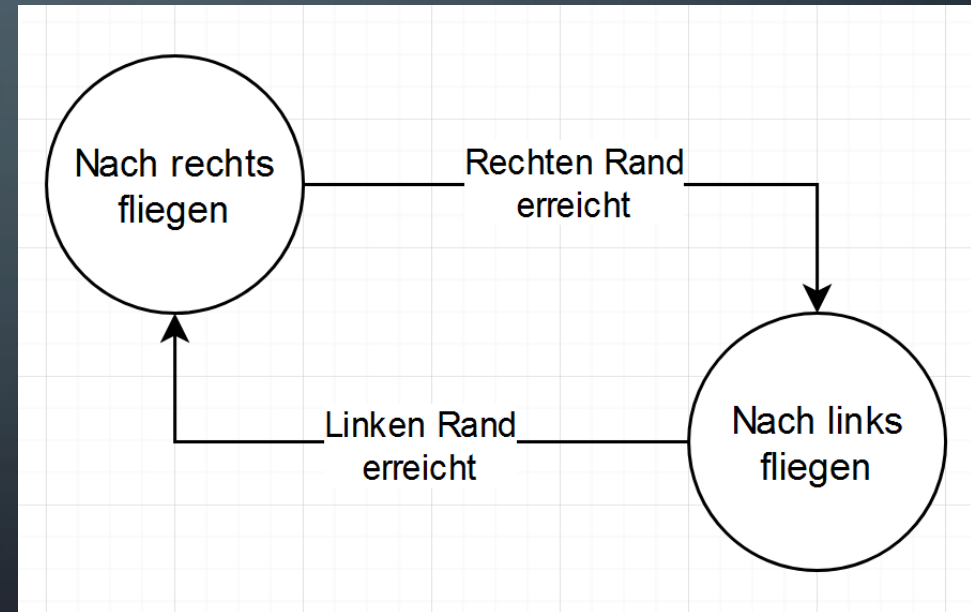
- Landwirtschaftssimulator (alt, neuere sind schlauer)
- Helfer sind „hilfreich“
- Fehler sieht einfach aus, ist aber schwer zu lösen



# KI: STATE MACHINE



- Simplere KI die zwischen „States“ wechselt
- Ein State ist eine Aktion oder eine Aufgabe
- Wird durch Interaktionen oder Limitierungen in nächsten State versetzt



# KI: STATE MACHINE

- Können Komplexe Verhalten in einfache Blöcke aufteilen
- In „Stealth“ Spielen gerne genutzt
- Beispiel: Hitman



# KI: STATE MACHINE – DIE PROBLEME

- Starre KI die nie „ausbrechen“ kann
  - Kann nur Aktionen ausführen, die einprogrammiert sind
- Sehr einfach durch den Spieler zu berechnen
  - Dies kann auch ein Vorteil sein, falls es designtechnisch vorgesehen ist
- Lässt sich nicht gut für organisches Verhalten nutzen
  - Agenten fühlen sich an, als würden sie gesteuert und führen immer das gleiche aus

# „NEEDS BASED AI“

- Definiert Bedürfnisse, die erfüllt werden sollen
- Benötigt Objekte bzw. Aktionspunkte, die Bedürfnisse bedienen können
- Entscheidet anhand von aktuellen Bedürfnissen und Objekten welche Aktion ausgeführt werden soll
- Gut nutzbare KI für organisches Verhalten (z.B. bei Menschen)

# EINBLICK IN DIE INFORMATIONEN

- Olaf wird uns begleiten
- Bedürfnisse werden gelistet
- Höhere Werte => Stärkeres Bedürfnis
  - Farbe hilft bei der Unterscheidung
- „LC“ für wichtige Aktionen die unbedingt durchgeführt werden müssen





# NEEDS

- Menschliche Bedürfnisse
  - Hunger/Durst
  - Erholung
- Spiel-bedingte Bedürfnisse
  - Check-In
  - Sicherheitskontrolle
  - Boarding



# NEEDS IN SIMAIRPORT

- **Menschliche Bedürfnisse**

- Hunger, Durst
- Langeweile, Entspannung
- Frustration
- Information
- Toilette, Sauberkeit

- **Ziele**

- Check-In, Sicherheit, Boarding
- Airport verlassen
- Gepäck abholen
- Büroarbeit
  
- Umwelt/Dekoration

# OBJEKTE FINDEN

- Objekte „strahlen“ aus welche Bedürfnisse sie erfüllen können
- Agenten evaluieren Objekte und wählen eins aus
- Beispiel: Olaf hat viel Hunger, ist aber auch durstig



# PROBLEME DER „NEEDS BASED“ KI

- Hoher Rechenaufwand bei vielen Objekten
- Agenten können in Zyklen enden
  - Beispiel: Durst ↔ Toilette



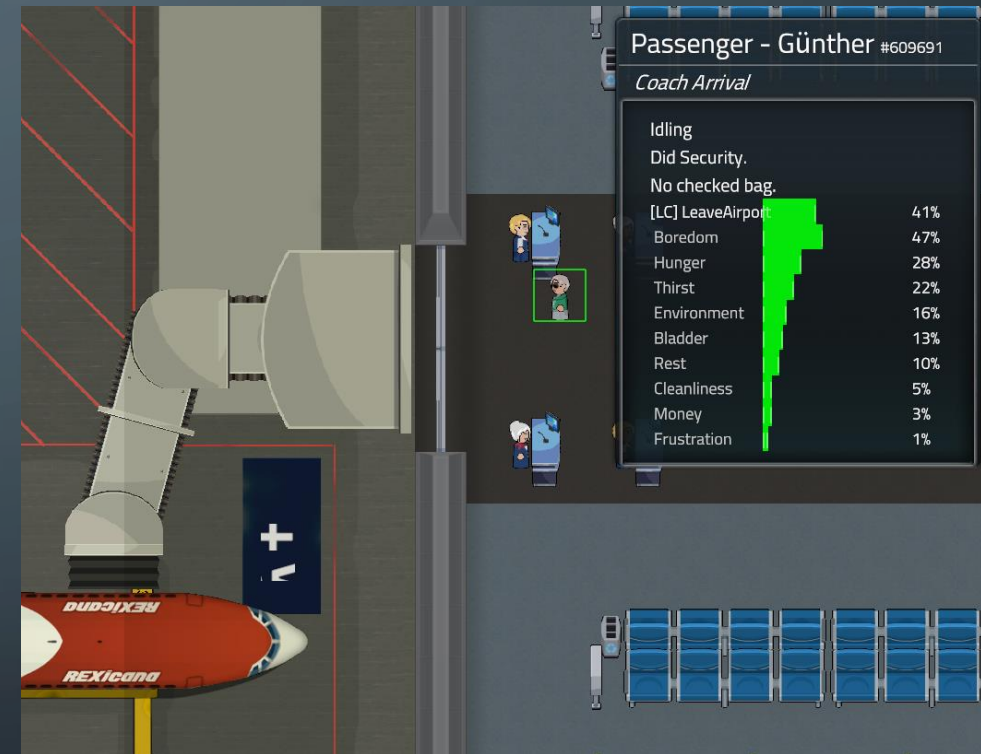
# PROBLEME DER „NEEDS BASED AI“

- Agenten können nur zum Ziel gestoßen werden
- Zeitbasierte Aktionen sind schwerer umzusetzen
- Lösung: Bedürfnisse mit hohem Bedarf um Agenten zu führen



# WARUM DER AUFWAND

- Needs Based AI führt zu Organischem Verhalten
- Macht die Simulation menschlicher
- Imitiert das Verhalten echter Menschen
- Am Ende ist Günther glücklich und bekommt seinen Flug



The image features a dark blue background with white, stylized circuit board traces in the corners. These traces consist of straight lines and right-angle turns, ending in small white circles that represent components or connection points. The traces are located in the top-left, top-right, bottom-left, and bottom-right corners, framing the central text.

**Das Ende?**

# LÖSUNG FÜR LS HELFER I

- Modifikation: „Courseplay“
- Ansatz: Schlaue Generierung eines Fahrwegs bevor die Arbeit beginnt
- Erlaubt Einstellungen spezifisch für ein Feld

**COURSE GENERATOR SETTINGS OF (942 VARIO)**

Section	Setting	Value	Description
BASIC	Work width	15.6 m	Work width. Value: 0 - 50
	Multiple tools	1	Number of vehicles on the same course.
	Number of headlands	2	Number of headlands. Value: 0 - 40
HEADLAND	Start work on	headland	Start working on the headland (harvest) or center of the field (other works)
	Headland corners	sharp	Headland corners
	Headland direction	clockwise	Should the vehicle drive clockwise or counterclockwise on the headland
	Headland overlap	7%	Overlap headland tracks as percent of working width. Value: 0 - 25
CENTER	Field center	up/down	Field center

The interface also features a preview window on the right showing a field with a generated course and a red 'STOP' sign. The preview window is labeled with numbers 41, 42, 43, and 44.



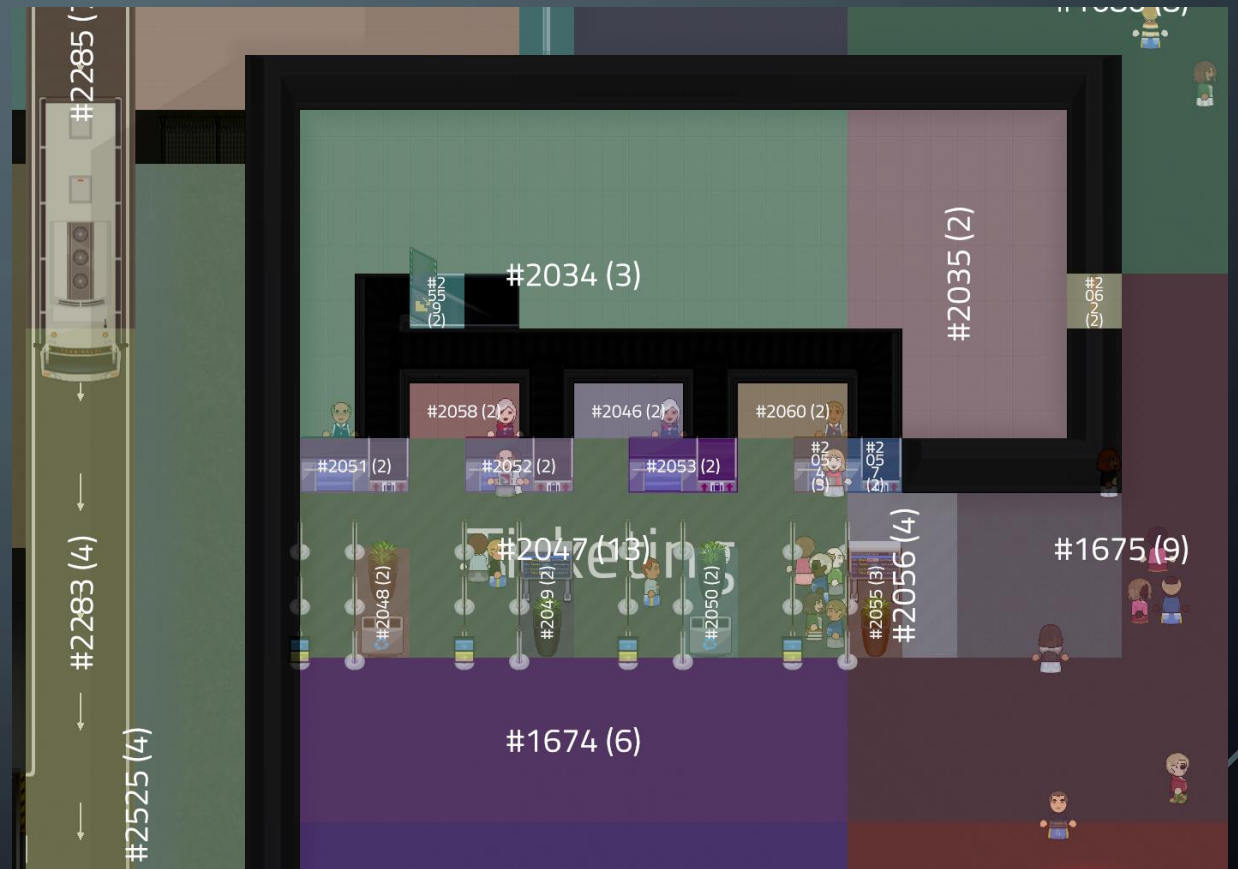
# LÖSUNG FÜR LS HELFER II

- Versteht was ein Loch im Feld ist
- Besseres System um nicht auf Zäune zu fahren
- Kann Kurven fahren



# PATHFINDING IN SIMAIRPORT

- Welt ist in Blöcke aufgeteilt
- Weg wird über Blöcke gesucht
- Path Following kümmert sich um die Bewegung innerhalb der Blöcke



# PATH FOLLOWING IN SIMAIRPORT

- System: Flow Fields
  - Vectorfeld
  - Steuert die Beschleunigung von Agenten
- Sehr performant
- Agenten folgen dem Weg sehr organisch

