

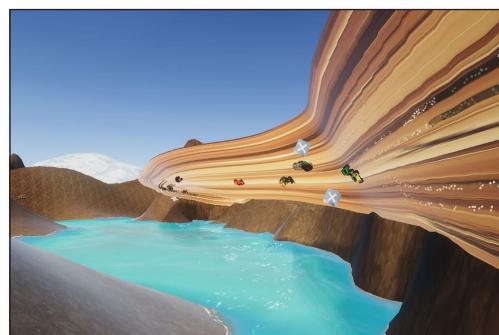
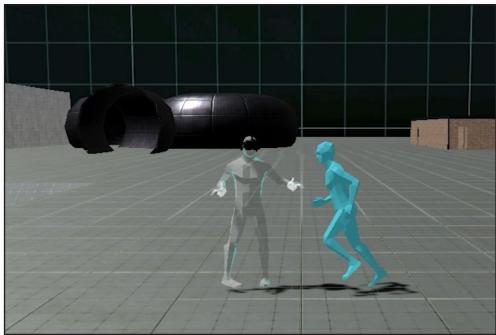
X R H C I G A M E S P O R T F O L I O

Jann Philipp Freiwald
jann.freiwald@gmail.com

Mixed Reality
Human-Computer Interaction
& Game Design

Selected Works
2021

Introduction



I'm Jann Philipp Freiwald from the city of Hamburg, and as research associate my task is to conceptualize novel and unique ideas. Promising concepts are implemented in Unity, followed by a scientific study comparing them to existing solutions for a variety of conditions. During the study data is collected, consisting of objective and subjective measures such as cybersickness, sense of presence and usability. The retrieved data is then statistically analyzed, and findings are published in a peer-reviewed scientific paper. For a complete list of my publications and their abstracts please refer to my [Google Scholar profile](#).

Besides research, my passion is game design. My favorite genre is co-op, or any game that has short rounds and can be played versus an AI. All projects in this portfolio were built and organized with Unity, Visual Studio, GitHub & Trello.

Table of Contents

Research Projects:

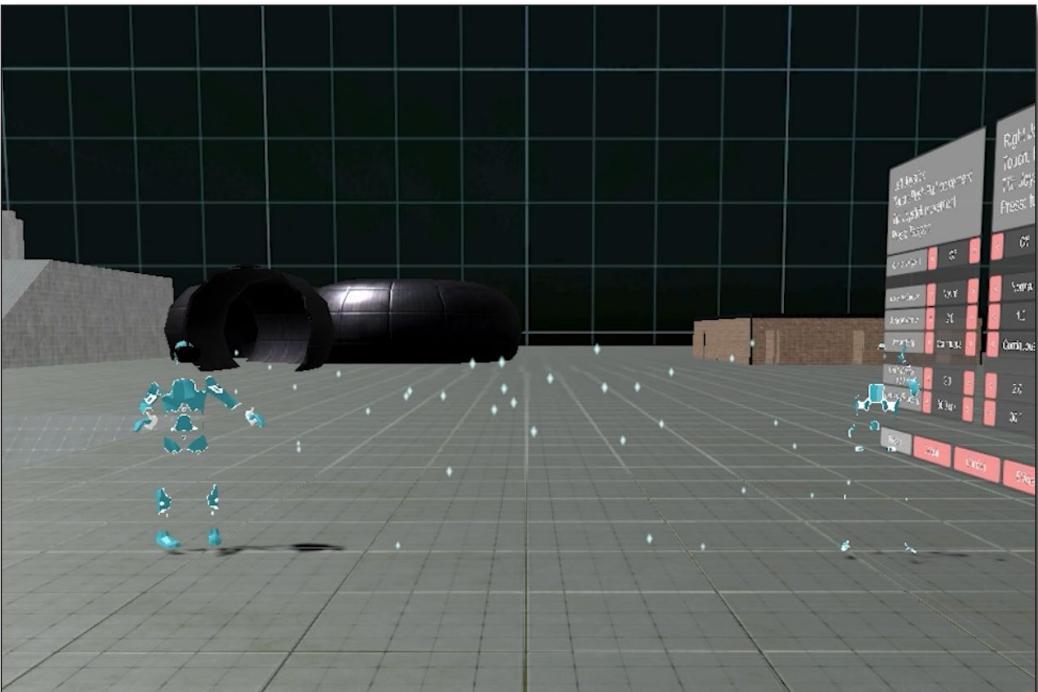
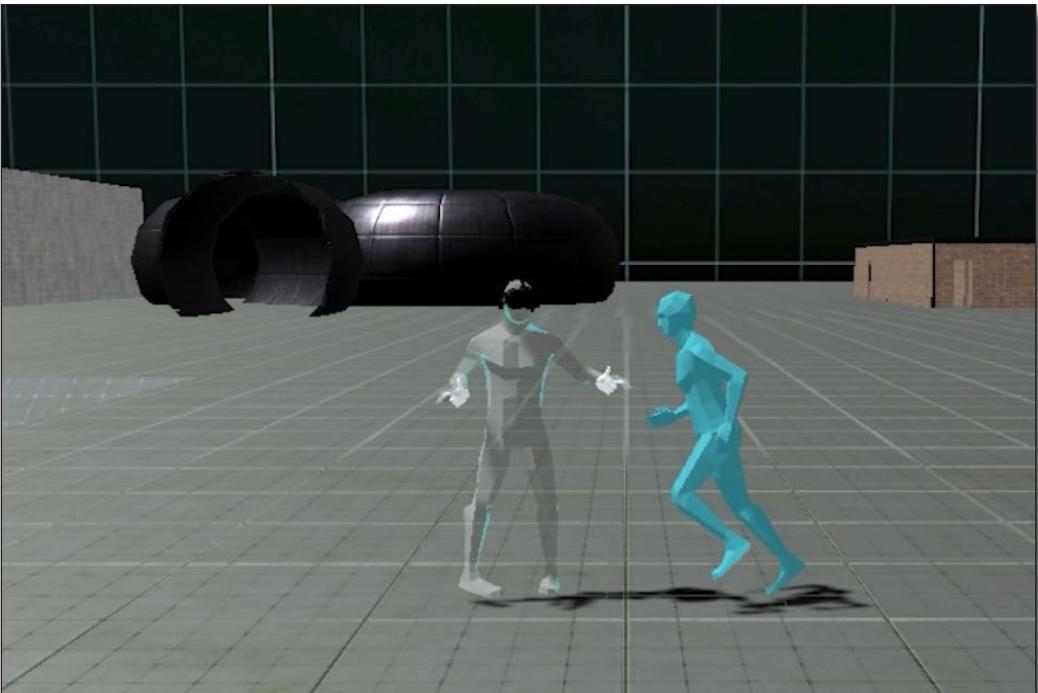
PushPull Locomotion & Smart Avatars	3
VR Strider	4
VR Invite	5
Camera Time Warp	6
Further Research	7

Game Projects:

Flame Grilled Murder	9
Ignition HD	10
G-Zero	11
Micross	12
Game Dev Teaching	13

About Me

14



PushPull & Smart Avatars

Time Frame: 1 year

PushPull is a hand-based locomotion technique that lets the user drag the world around them, either for movement or turning. While active, the player's triggering hand functions as a fixpoint. When that hand is moved, the entire player will either translate or rotate, such that the hand stays in place relative to the world. Push-Pull, stutter-stepped teleportation and targeted teleportation are all mapped to the touch-sensitive joysticks of common VR controllers.

This technique is paired with 'Smart Avatars', which decouples the avatar from their player. This approach allows for natural looking animations during non-continuous movement like teleportation. Instead of a one-to-one mapping, the avatar is controlled by an AI agent that will follow the player. When close by, the avatar animation switches to inverse kinematics. In the depicted ghost mode a transparent agent is mapped one-to-one, while a second opaque agent will follow at a natural speed. Other visualization types like dissolving avatars have been explored as well.

I conceptualized and implemented the techniques, and will be responsible for conducting the user study, data analysis, as well as writing the research paper.

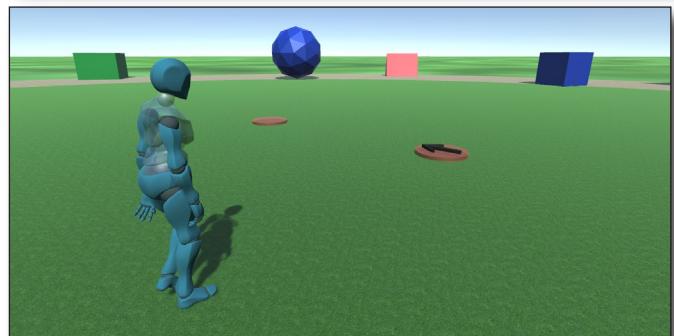
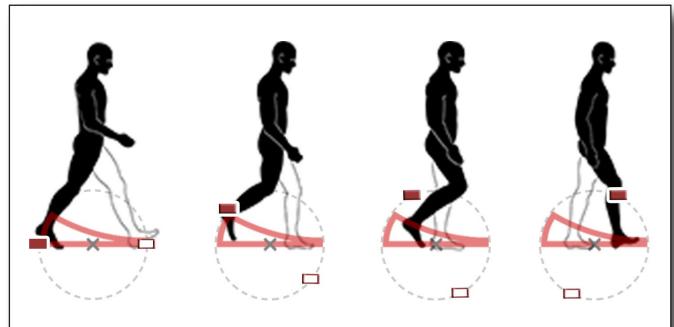
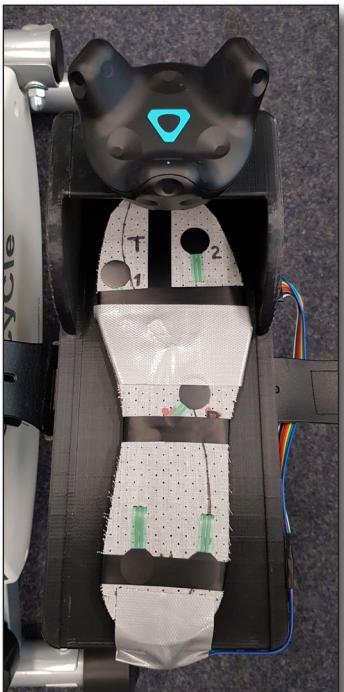
[Watch the teaser on Youtube](#)

Technology

- Oculus Integration
- Mecanim & Inverse Kinematics
- NavMesh
- Scriptable Objects (Event System, Settings, Manager States)
- Shader Graph

VR Strider

Time Frame: 2 years



[Watch the teaser on Youtube](#)

Technology

- Steam VR with Vive Standalone Trackers
- Mecanim & Inverse Kinematics
- Blender
- 3D Printing
- Custom integrated circuit board
- Pressure sensors and linear resonant actuators
- Embedded microcode & Bluetooth device manager in C++



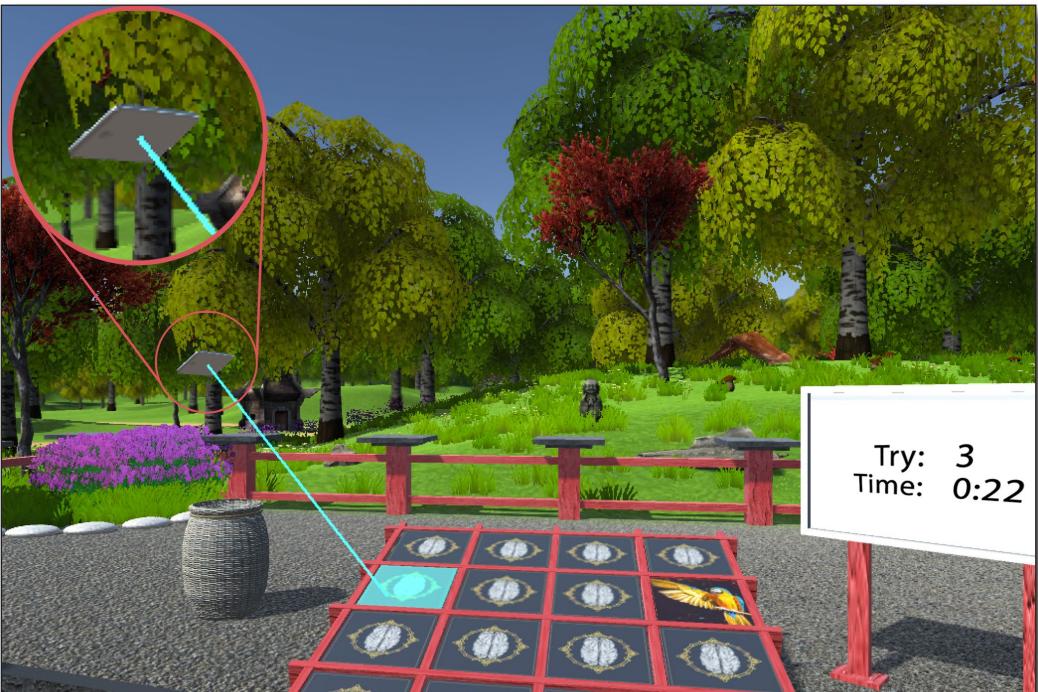
VR Invite

Time Frame: 1 year

Current immersive experiences are often limited to those users wearing a VR headset. VR Invite is a project-independent smartphone app, which allows multiple non-immersive bystanders to observe and interact with the virtual environment and the VR user. The system renders multiple view ports of the scene on a host computer and transmits the image data via wireless local network to the tracked mobile devices.

The user study showed that the opportunity for direct interaction positively influences the bystander's sense of presence in the virtual environment and the reported usability of the Smartphone app. However, social presence was rated higher in passive conditions in which the real person was the center of attention, as opposed to the avatar on the screen.

I conceptualized the app, implemented the network streaming, and was responsible for data analysis as well as writing the research paper.



Technology

- Steam VR
- Google AR Core
- C# sockets

Camera Time Warp

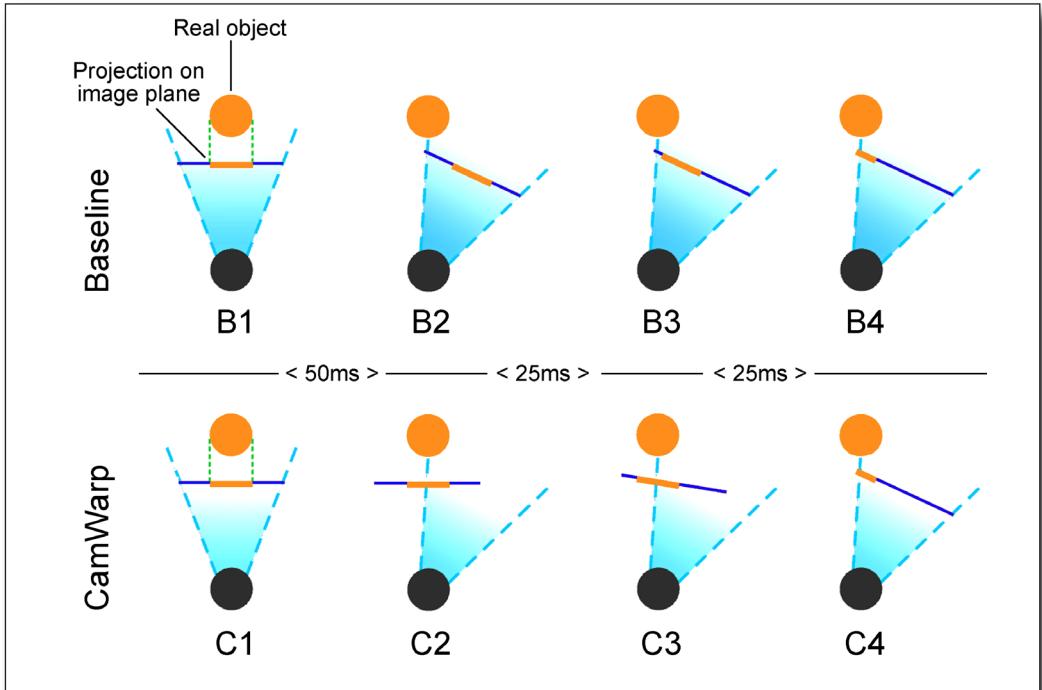
Time Frame: 1 year



Camera Time Warp is a reprojection technique for video-seen-through augmented reality, which reduces the registration error between captured real-world videos and rendered virtual images. This is achieved by inducing a delay to the movement of the virtual image plane which displays the camera feed. When the camera feed delay and the movement delay match, registration errors are reduced significantly.

A Fitts' Law task experiment was conducted, where participants reported significantly reduced levels of discomfort and cybersickness compared to the baseline. A second experiment revealed a positive effect of Camera Time Warp on task performance and accuracy in an AR sorting task.

I conceptualized and implemented the technique, conducted both studies and data analyses, and wrote the research paper.



[Watch the teaser on Youtube](#)

Technology

- Oculus Integration
- OVRvision
- Stereoscopic cameras

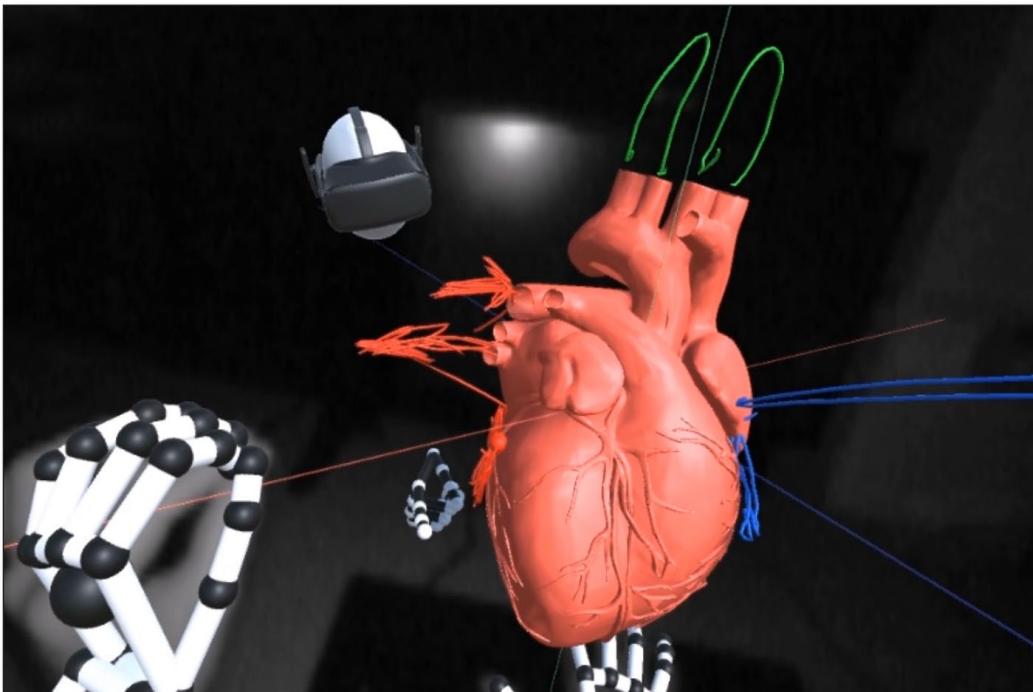
Further Research ||||

Time Frame: each 1 year

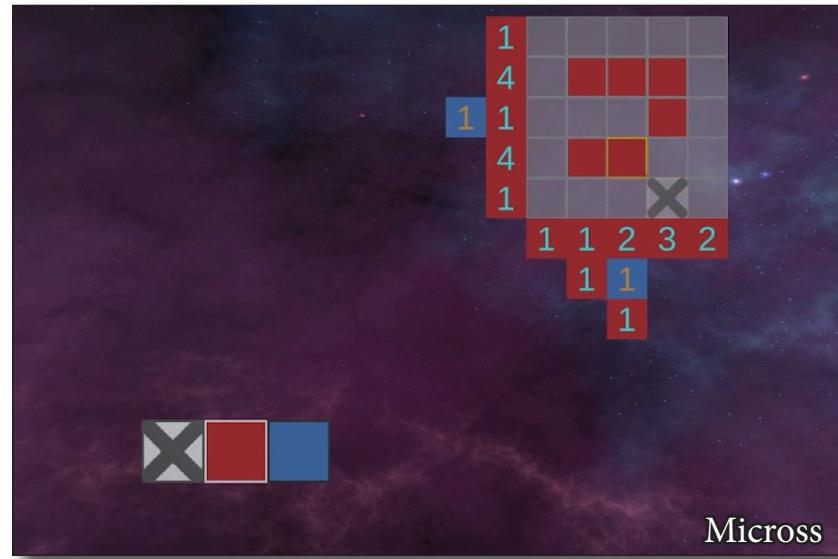
I was part of several other research projects in the realm of VR, implementing techniques and performing studies. This includes wearable haptic devices such as the vibrotactile Haptic Glove, designing a cybersickness susceptibility questionnaire, and studies on virtual collaboration. Here, I analyzed the perception of other participants in VR meetings based on their appearance and the visualization of their locomotion, as well as techniques to effectively communicate a user's view direction to observers.

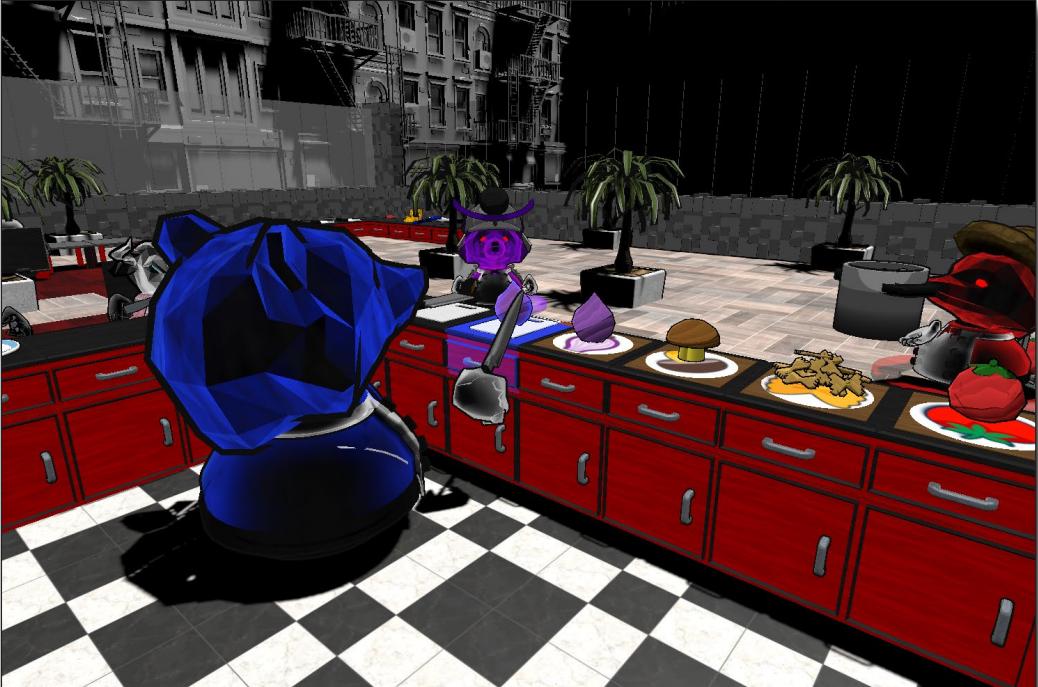
I conducted further avatar-centric research on finger tracking in shared mixed reality spaces, as well as realtime 3D scans of people through multi-camera point clouds. These 3D scans were achieved by filtering images in a heavily multithreaded GPU compute shader, transmitting the results over network and assembling the meshes in a vertex shader.

For a complete list of my publications and their abstracts please refer to my [Google Scholar profile](#).



Game Projects||||





Flame Grilled Murder

Time Frame: 3+ years

Overcooked meets serial killer simulator. One VR head chef and up to 4 controller sous chefs cook dishes according to the wishes of their customers. However, some customers have an unhealthy appetite for meat... customer meat. It's up to the players to decide who will be served food, and who will be served *as* food. Witnesses of a murder will attempt to flee the scene and alarm the police, so be ready to hide all the evidence before the inspector arrives! Or worse, survive the police raid when you get caught to clear your name. No witnesses, no crime.

This game features elements centered around comfort in VR, such as PushPull locomotion and gravity-gloves (as seen in Half-Life: Alyx). The multiplayer level editor can be used to generate new individual levels, or as part of 'Endless Mode'. Here, players serve customers at night, and extend their establishment during the day with ingredient boxes, appliances, tools, weapons and murder contraptions.

[Watch the teaser on Youtube](#)

Technology

- Steam VR & Steam Integration with Remote Play
- NavMesh (agents and VR locomotion constraints)
- Blender & ProBuilder
- Addressables
- Shader Graph & Custom Renderer Features
- Runtime Mesh slicing and combining
- Scriptable Objects (Manager States, Recipes)
- DoTween
- Cinemachine
- Post Processing

Ignition HD

Time Frame: 1+ years

Ignition HD is a reimagining of 1997's 'Ignition' or 'Bleifuß Fun' as it was called in german. Ignition was the game that sparked my interest in game engines. "How could they have prepared every possible image?" I had wondered as an eight year old kid. This project aims to recreate the feeling of the physics, cars and tracks of the original, and incorporate elements from other fan favorite arcade racing games such as items from Mario Kart. I started working on Ignition HD as a side project mainly to get an understanding of Mirror networking, which I intend to use in Flame Grilled Murder.

Currently the game supports up to 100 AI cars or concurrent online players thanks to Steam's matchmaking and lobby services. There are 10 starting positions distributed over each track. The player who finishes their own 3 laps first gets the highest number of points. Additional points can be earned by hitting other cars with items, and the player with the highest total points wins the cup. Tracks can be prototyped and iterated on quickly with custom editor tools for mesh generation based on bezier curves. In the future online play will transition to Microsoft Playfab for dedicated servers and crossplay.

[Watch the teaser on Youtube](#)

Technology

- Steam Integration with Remote Play
- Mirror Networking
- Assembly Definitions
- Custom Editor Tools for Bezier Curves
- DoTween
- Cinemachine
- Post Processing
- Unity Test Framework (Unit testing)
- Scene Transitions Based on Photoshop's 'Soft Light' Implementation

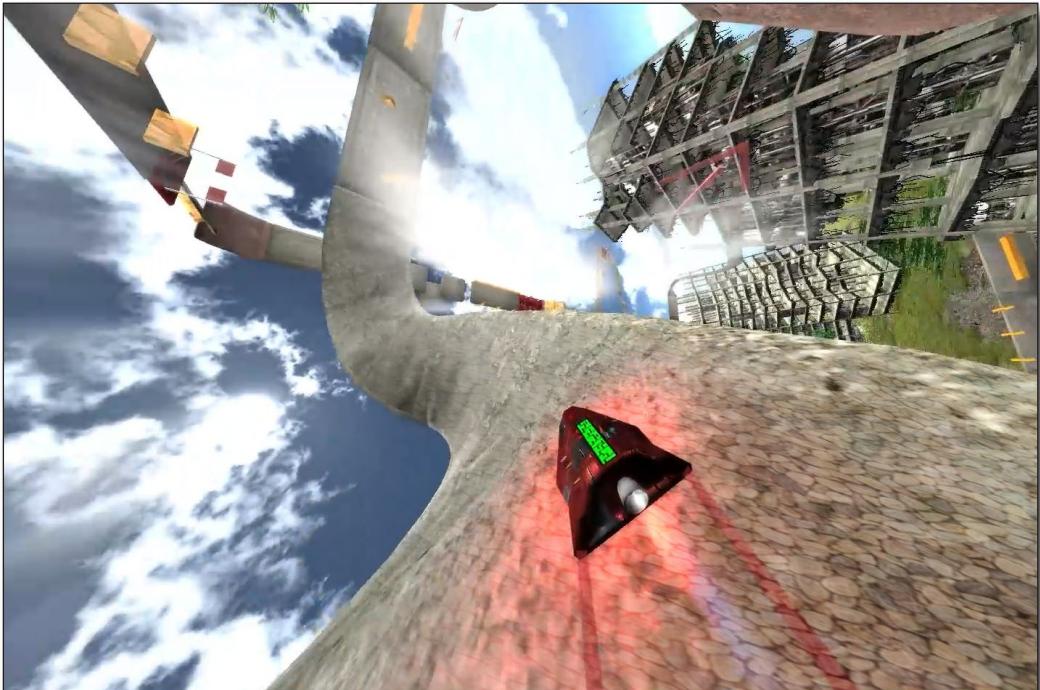


G-Zero

Time Frame: 1 year

G-Zero is F-Zero in anti-gravity, paired with Velocibox, an obstacle avoidance game. Where boost is usually the limiting factor in racing games, here it is the brake. The car will always travel at a certain minimum speed, and only by boosting the brake-meter fills, so that you can use it at just the right moment to dodge an incoming obstacle. Dying to an obstacle resets the car to the last checkpoint, costing valuable time.

The tracks are randomly generated in 3D space by assembling several predefined track sections, such as turns, straights, obstacles and checkpoints. This is done via a depth-first search, where a path is discarded if it intersects with a previously built section or world geometry. Many sections have obstacles only on the top or bottom side, and the car can switch between those sides at will. The game can be played with one or two players, and an optional AI for an additional time challenge. G-Zero was my first finished game project in Unity. The anti-gravity tech was partially reused in Ignition HD.



Technology

- Depth-first search



Micross

Time Frame: 1 year

Micross is a recreation of the Picross puzzle game. My fiancé is solving puzzles quicker than I can buy her new puzzle games, so I made one that uses procedural generation for an almost unlimited supply of levels. The numbers on the side of the playing field indicate groups of uninterrupted blocks of a certain color. If two numbers with the same color are next to each other, there has to be at least one empty space between the groups.

Almost all levels can be produced in such a way that solving them is deterministic. The game reads any number of image files and generates puzzles in sizes between 5x5 to 20x20 for each of them through quantization. Using Pokémon sprites alone, there currently are more than 40.000 individual puzzles available. A redesign of the UI is currently being discussed.

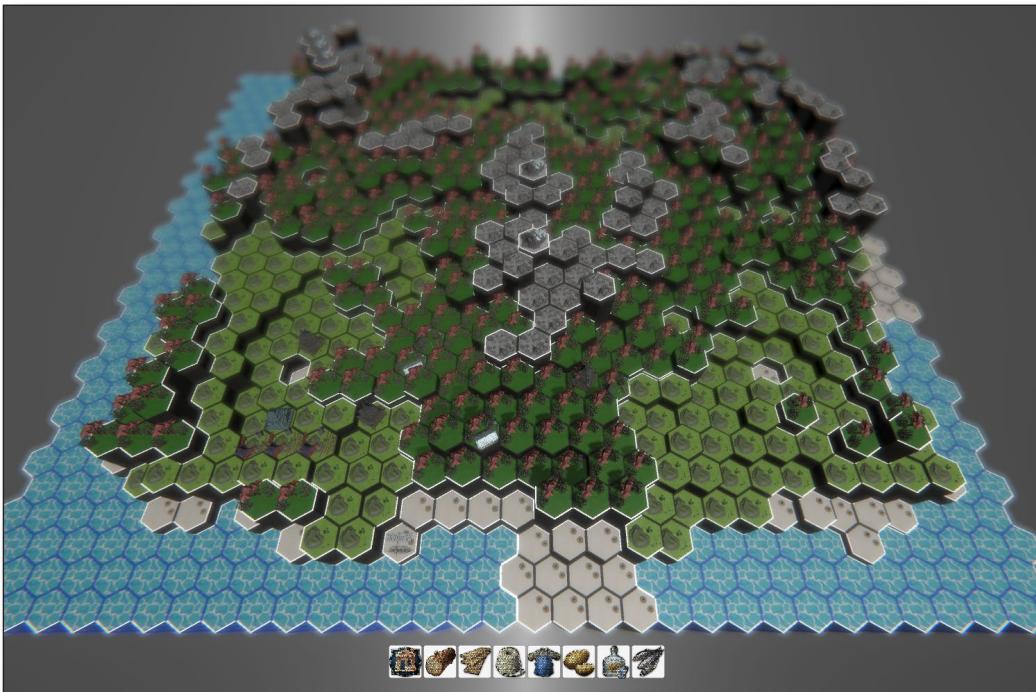
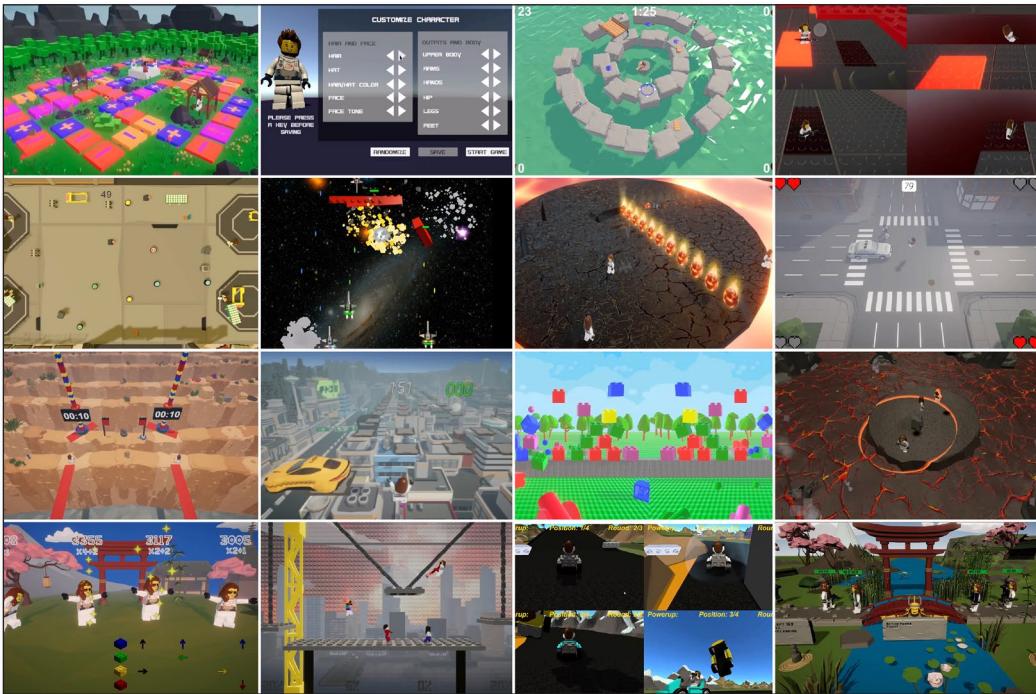


[Watch the teaser on Youtube](#)

Technology

- Android
- Touch input
- Quantization
- Asset Streaming

Game Dev Teaching

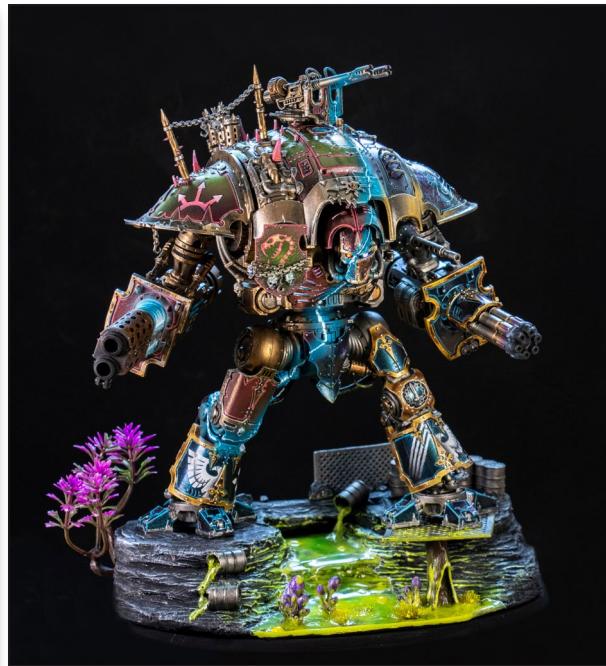


Games developed in the scope of my teaching position are designed to be made within one semester, or 6 month. The majority of my master-students have never worked with Unity, version control or kanban boards before, so each semester starts with a crash course in practical game dev. Usually, groups of 4-6 students make a game of their choice with a set of guidelines. For the 2021 course however I had grand ambitions: I wanted to make a single game with all of my 50 students, remotely. Risk management was the main concern of this project. Seeing as any group could leave at any time, each contribution had to be dispensable. Additionally, we decided to use github forks instead of branches, so that I could handle merges and conflicts. The merges happened at the end of each sprint, every 3 weeks in the form of predefined milestones. We held weekly status updates in Discord, so that I could direct them if required. Text channels were provided for each group to enable asynchronous feedback.

The game we made was 'Lego Party', our version of Mario Party with Lego aesthetics. Every team had to create one mini game, with the exception of the groups responsible for the main board game and the one handling character customization and scene loading. These were critical to the overall success of the project, so I focused my supervision on them. At the end of the semester we delivered the finished product with 15 fully functional minigames.

The prior year's course theme was "Anno". This was the first year of online teaching, so to be safe I programmed a game myself and gave the students 'fill in the blanks' tasks, where they had to recreate what I presented to them at each milestone. Here, I built a simplified version of Anno 1800 with a tile-based hexagonal map. The map is generated from a greyscale image, where the height of a tile is based on the value of a pixel, and the type of tile is based on its height. The game features two citizen levels and a selection of production lines, as well as a population simulation.

About Me



In my free time I enjoy relaxing and artistic hobbies. For my research I bought a 3D printer, and just recently learned how to model my own figurines in Blender. I print the models in solid grey and then paint them with acrylic paint. This hobby is called 'Mini painting' and I picked it up as quarantine past time in 2020. Recently I started adding electronic components like glowing LED eyes for extra visual interest. I'm also an amateur photographer, which comes in handy when I want to present my paint jobs. I've been experimenting with videography as well, and would like to learn more about it in the future. I enjoy a variety of music genres, such as progressive metal, electronic music and acoustic guitar, and have been playing the guitar myself since I was seventeen.

I love playing parlour games with friends, such as the Exit game series, which is similar to escape room puzzles. I also like to come up with my own games and prototype them. For example, I made a card game called 'Deliverance' that was meant to be used as combat system in pen-&-paper games, but turned into its own thing over time. I designed all the cards and had it professionally printed. When friends are over, we also frequently compete in local multi-player games such as Mario Kart and Super Smash Bros. Instead of online competitive games I much prefer co-op games with friends, or games where I or a team play versus an AI.

My favorite gadgets are any kind of 3D displays. Stereoscopy has always fascinated me, which is why I love tech such as the Nintendo 3DS, Nvidia Shutter Glasses, and VR headsets.