James Arndt

New Smyrna Beach, FL | jamesarndt3d.com | jimmyarndt@gmail.com

---

This is not an exhaustive list of my experiences, but it details some of my most recent experiences over the last 5-10 years. The vast majority of my experience involves working within Unity, 3ds Max, Maya, Blender, Substance Painter and Designer, Photoshop and Visual Studio, building realtime applications. My strengths are in 3D modeling, materialization/texturing, lighting and rendering features, shaders and visual effects, optimization of asset content at ingestion and performance optimization of runtimes. For the rest of my work history, including game-development specific projects, please visit > https://www.jamesarndt3d.com/about.html

Hatch BHP Resolution Copper Mine Experience (PC and Mac)
Unity-based desktop application. This POC resulted in an interactive tool that gave users an overview of the Resolution Copper Mine site and the lifecycle of the mine's development. I was responsible for ingestion of dense CAD mesh data and conversion to FBX file format that could be utilized for realtime purposes in Unity. My optimization work entailed culling hidden meshes, camera proximity-based targeting for decimation and retopology, merging of redundant materials, removing hidden backfaces, utilizing instancing at the shader/material level and removing unneeded vertex data in UVW channels.

Microsoft Project Flow (PC and Mac)
Unity-based desktop application. This POC resulted in a visualization of wind over a field of turbines and was used as a digital twin of Enbridge's New Creek Wind farm in order to demonstrate how this technology can improve turbine maintenance. I was responsible for producing Unity VFX Graph visual effects to represent wind ribbons, vortices and air pockets. This VFX was data-driven by code values that modified color, intensity and direction. For example - based on a code-driven wind intensity value, the VFX would change color between green-yellow-orange or red hues.

Cameco Digital Twin (Meta Quest 2 and Vive Focus 3)
Unity-based VR application. The goal of this POC was to demonstrate a digital twin of a uranium factory with robotics, that could be used as a virtual training scenario for employees. I was responsible for authoring the virtual environment - a complete digital twin of a Cameco factory interior, with all associated props and real-world features. The process involved mesh authoring in 3ds Max, materializing and 2D texture set authoring in Substance Painter, Unity baked lighting of the environment and reflection and light probe authoring. I was also responsible for the keyframed animation of the robotic systems within the factory. This included animating a nested chain of robotic appendages around several degrees of freedom.

Vancouver International Airport Passenger Experience (iOS and Android)
Unity-based AR application. This project was a solicited POC to demonstrate a guided AR gamification with virtual ads and animated creatures on top of the real-world airport terminals. This involved creating a virtual digital twin experience of YVR airport terminal. The app included a wayfinder compass with a visualized path destination, steps, direction, estimated time AR wayfinding. The user could visit First Nations art features within a terminal and AR-based content would pop-up detailing the feature. I was responsible for authoring the digital twin virtual environment. The process involved mesh authoring in 3ds Max, materializing and 2D texture set authoring in Substance Painter, Unity baked lighting of the environment, reflection and light probe authoring. The process also included a heavy focus on optimization techniques to ensure performance overhead was low. This included occlusion culling, LODs, removing geometry not rendered to camera, atlasing of 2D textures and merging of materials. I was also responsible for visual effects, such as animated paths, highlighting of features, and ad-specific content such as a virtual hamburger.

Chanel (WebGL, PC, Mac, iOS and Android)
Unity-based web application. My responsibilities involved photorealistic materialization of fashion products - a leather handbag, a cosmetic and a fragrance. This meant authoring bespoke lighting scenarios for each product to replicate Chanel's existing studio product lighting. I was responsible for the ingestion of assets and optimizing CAD data for realtime usage, optimizing vertex data and UVW channels. I was also responsible for the conversion of working FBX files to GLB format and uploading to a digital asset management solution - Unity Asset Manager. An interesting contribution to this project was exploiting the Asset Manager tagging feature and using that meta data in the Unity runtime app to additively swap lighting scenarios based on tags. Another was extending the glTF handbag shader to composite detail normals, providing higher surface fidelity with minimal overhead (other than another texture sample in the shader). This allowed us to maintain a low resolution base texture and layer a smaller, lightweight texture over top of this to give the appearance of high detail across the asset.

LEGO NASA Mars Rover Perseverance (iOS and Android)
Unity-based AR application. The goal of this app was to tie together a virtual AR experience with the company's real-world toy. I was responsible for particle-based VFX work and authoring screen space shader effects for this project. I authored various particles, such as dust clouds, swirling air patterns and a set of stylized/futuristic effects to signal landing, takeoff, low battery, etc. The screen space effects supported a core feature of the app, where the user looks through the "periscope" of the rover to examine various rock features on Mars. This shader included masking the scope in screen-space, an x-ray feature where the user could look "inside" of the rock. This was accomplished by feathering a mask while compositing two 2D textures - the base texture for any given rock and the x-ray variant of the interior. I was also responsible for 3D modeling/materialization of the interior lab components and the Mars rock formations.

Maloka VR (Meta Quest 2 and Meta Quest 3)
Unity-based VR application. Maloka VR is a guided meditation experience taking place on an interactive island. Maloka's first-time user experience needed to better showcase the titles' unique features and to improve runtime performance. I was responsible for both runtime optimization and visual effects. This work involved profiling the application to find performance bottlenecks, determining if the application was CPU or GPU bound. In our case too many draw calls were being sent to the GPU. My optimization included mesh merging, merging of materials and 2D texture atlasing, establishing more optimal culling sets and optimizing shaders by lowering the precision of data types where it did not impact visual fidelity. In a lot of cases we could convert high precision floats to medium precision half types without any noticeable change in rendering. I was also responsible for authoring otherworldly/stylized particle VFX to include swirling vortexes, highlighted zones, or natural elements like tree leaves.

Nissan HMI (PC and Mac)
Unity-based desktop application. Our team developed a functional HMI interface as a proof of concept for Nissan. I was responsible for authoring an "uber" shader for the vehicle's exterior geometry. The guidance was to produce an 'Into the Spiderverse' stylized shader with cel-shading, stepped lighting, rim/fresnel lighting, an adjustable outline and an adjustable halftone pattern within the darker shaded regions. One particular challenge was the outline feature and choosing a method to approach (i.e. render feature in screen space or at the object level). I opted for the inverted hull method of extruding geometry, inverting normals and compositing an adjustable color. This work was also interesting in that it necessitated overriding Unity's built-in lighting functions within the custom ShaderGraph, to create the stepped/cel-shaded look of the base color.

P2E Mayaa-verse (PC and Mac, Meta Quest 2)
Unity-based desktop application. P2E's goal was to create a POC of a functional metaverse, enabling and hosting multiple immersive virtual cities. My work on this project involved working with the environment and character artists to reduce performance overhead of the experience. During this project I authored a C# Unity tool that allowed our team to identify all assets referenced in a given scene with detailed information about those assets. This allowed the team to identify high value targets for optimization - overly dense meshes, material references, 2D textures that may be too large. It also allowed the team to find all GameObjects referenced by a specific material or texture. I also created a second C# tool, a runtime profiler that displayed an overlay in both the Unity editor and runtime showing current vertex/triangle count, current draw calls, current batches, average frame time, bone count, GameObject count, texture memory and the amount of textures used. One interesting aspect of this project was not only uncovering the CPU overhead of the amount of skeletal joints being processed in any given frame, but the excessive amount of keyframes for the animation set. I wrote a C# Unity tool to automate the import of skinned meshes and reduce bone-to-vertex influences without impacting the quality of the animation. This same tool would also allow removing a large amount of redundant keyframes without impacting animation curves or tangents.

BRP (Can-Am Off-Road) (PC and Mac)
Unity-based desktop application. I was tasked with authoring meshes, materializing and texturing a subset of off-road vehicles to a photoreal standard using Unity's HDRP and DX12 raytracing. My work involved authoring PBR materials, authoring the associated texture sets for use in high definition materials and balancing those materials under several lighting conditions. This included utilizing client-provided paint swatches/codes for albedo values. The biggest challenge of this project was balancing material values under several different rendering environments/lighting scenarios and post processing. I was also tasked with optimizing raytracing settings and post-processing to boost performance of the runtime. This involved balancing sample counts against visual fidelity via the render pipeline settings.

Cartier (WebGL, Desktop, iOS and Android)
Unity-based WebGL application. I was responsible primarily for the ingestion of extremely dense CAD mesh data and converting that to FBX file format that could be used for real-time purposes in a Unity application. My approaches included mesh decimation, mesh merging, vertex welding, mesh culling and retopologizing of surfaces. Each asset consisted of thousands of nested child mesh objects, and prior to mesh merging I authored a Python script to identify the most vertex-dense meshes. Once I identified target objects for decimation, I needed to discern high value targets for decimation vs. low value targets. This included factors such as the mesh's visibility and proximity to the camera, and also included factors such as the size of the mesh in any given frame. After decimation, meshes that would never be animated via code or keyframes or meshes that would not be accessed via code for setting visibility were combined into single mesh objects. The goal was to reduce both vertex density and draw calls to the gpu.

Microsoft Blade (Apple Vision Pro)
Unity-based Apple Vision Pro application. A digital twin of a wind farm. I was responsible for authoring a high fidelity wind turbine mesh, materialization and producing associated 2D textures. These included textures for both a "clean" pass and a "dirty" pass to show the turbine in a worn state. I was also responsible for creating a reusable prefab in Unity, with script-animated rotor blades and anemometer. I was also responsible for procedural placement of various foliage within our virtual environment utilizing Unity's spline framework. Lastly I was tasked with authoring a bespoke terrain shader that allows for distance-based detail compositing, increasing grain/noise fidelity based on camera proximity to obscure the low resolution of geospatial terrain textures mapped over such a large topography. I also extended our ocean shader to mask the ocean underneath our island terrain to reduce or eliminate the potential of z-fighting or render priority issues.

Audi Nürburgring Digital Twin (PC and Mac)
Unity-based desktop application. A virtual recreation of the Nürburgring racetrack used by Audi in the testing of vehicles and the training of drivers within a virtual simulation. My primary responsibilities were content optimization and authoring of 3D content around the track. Nürburgring is a 13 mile long track and as such was a technical challenge to work with in the Unity runtime. My work involved optimizing and converting CAD assets to FBX format and ingesting them into Unity in a form suitable to runtime performance. This included the authoring of various buildings and trackside signage and producing LODs for each asset. I was responsible for the modeling, materializing and texturing of these assets. Much of trackside safety signage required precise placement spatially, based on real-world location and the final work was the result of iteration and feedback from the team at Audi.

Bosch Rexroth (Apple Vision Pro)
Unity-based Apple Vision Pro application. The goal of this project was to demo a virtual Capgemini booth with a virtual Bosch machine at Hannover Messe. The digital twin of the booth and machine would be interacted with in both table top view and immersive view. I was responsible for authoring the Capgemini booth, authoring meshes, materialization and authoring associated 2D texture sets. I was also responsible for asset optimization of the Bosch robotic machine that was populated within the booth. This included culling hidden mesh objects, decimation and retopology, merging redundant materials, removing hidden backfaces, simplifying shaders where it did not impact the asset's visual fidelity. Lastly, I was responsible for the keyframed animated sequences of the Bosch machine, as it cycled through different processes - a robotic arm would pick up a set of batteries, and deposit them into a container.

Samsung Product Viewer (PC and Mac)
Unity-based desktop application. My primary responsibility was to convert Keyshot material data to data that could feed into HDRP materials in Unity. In essence, set material properties via Unity's importer based on data extracted from equivalent Keyshot materials. A simple example of this - extracting Keyshot's "Diffuse" material property value for albedo color and importing and applying that data into the "Color" property of an equivalent Unity material. I authored a Python tool that facilitated moving this material data between Keyshot and Unity. I was also responsible for "baking" Keyshot procedural textures to 2D texture maps that can be used within Unity. These 2D textures needed to be properly mapped back into their associated material in Unity.

Microsoft Project GeForce ESRI (Microsoft Hololens 2)
Unity-based Hololens 2 application. The goal of this effort was to develop a geospatial data pipeline to import and visualize geospatial data in an AR/MR headset, in this case the Hololens 2. I was responsible for authoring shaders for this project. A few of them were a generalized object highlighting shader with an adjustable animated pulsing, a mask shader to mask the edges of the virtual world as the user motioned through the environment. Much of this work was built on extending the existing Mixed Reality Toolkit package content. I was also responsible for authoring 3D mesh objects from our existing 2D user interface textures. In essence all of our 2D UI content would be turned to 3D objects that could rest in 3D space and have depth to them.

EG+ Nissan Configurator (WebGL, PC, Mac, iOS and Android)
Unity-based WebGL application. The goal of this PC was to create a vehicle configurator showing the power of Unity ray tracing rendering tech. My primary work on this project involved improving the runtime performance of the DX12 application. This involved a multi-faceted approach of adjustments to the render pipeline raytracing settings, as well as directly optimizing the art content. Our project utilized server-side rendering via host to stream raytraced content via the web to any platform. My work involved working directly on the Nissan vehicle asset, culling hidden meshes, camera proximity-based targeting for decimation and retopology, merging of redundant materials, removing

hidden backfaces, utilizing instancing at the shader/material level and removing unneeded vertex data.

YVR Digital Twin (PC and Mac)
Unity-based desktop application. The goal of this project was the real-time data visualization of Concourse D of the Vancouver International Airport, to assist with situational awareness of operations and security. My work involved the ingestion of client-provided, dense CAD data and its optimization. This generally involved mesh decimation, hidden object removal and mesh backface removal. I was responsible for authoring the Concourse D environment of the YVR airport, to include authoring of meshes in 3ds Max, producing texture coordinates, retopologizing existing CAD data, materializing and texturing the environment and props in Substance Painter and Photoshop. I was also responsible for authoring lightmaps using Unity's baked lighting tools as well as authoring reflection probes and light probes for the concourse.

Sword Coast Legends (Xbox, Playstation, Steam)
In this role I served as the team's Technical Artist for Sword Coast Legends. I was tasked in the creation of 2D texture masks for a coloring system utilized in the customization of creature/character skins and armors. I was responsible for material balancing and coloring of characters, weapons and creatures in both the menu front end and under in-game lighting scenarios. I ensured albedo, normal and specular texture level continuity across hundreds of pieces of armor and characters and maintained a consistent measured PBR pipeline. I was also tasked with the illustration of numerous 2D icons used within the game's menus and in-game user interface.

JAWS: Ultimate Predator (Nintendo 3DS)
In this role I served as a 3D generalist and worked an 8 month on-site contract position. I was responsible for 3D modeling, texturing, lighting, rigging and animating a variety of real time assets. These assets included static props, marine and aviation vessels, NPC characters and ocean wildlife, environmental set pieces and various bits of ocean fauna. My primary tasks were to model in 3ds Max, produce 2D textures with Photoshop and export to our proprietary game engine.

Firebrand Games (Nintendo 3DS, 2DS and Wii)
In this role I served as 3D Artist tasked in the creation of 3d assets for the Nintendo Wii and DS platforms. These objects included various vehicle, prop and environmental models, textures, particle effects and animations. I was tasked with modeling virtual worlds, hand-painting textures combined with photo sourcing. I was also responsible for implementing lighting using vertex color baking techniques and hand vertex painting of 3D models. Vehicle and prop artist tasked with modeling, texturing and lighting of real time assets. Skilled in the manipulation of LUA scripts to alter particle effects, character physics, model placements in the world and the triggering of environment animations. Technical setup artist tasked with creating a volume system for each environment that determined what was culled and what was drawn for any given frame. PVS system implementations per level. I also served as a rigging artist tasked with creating simple rigs for environment geometry and creating world animations using joint systems. I created texture animations for usage in game, utilizing UV animations with textures. Shipped titles within this time frame are: Need for Speed Undercover (Nintendo DS), Need for Speed Nitro (Nintendo DS), Dirt 2 (Nintendo DS), Cars 2 (Nintendo DS), Spongebob Boating Bash (Nintendo DS), Hot Wheels: Track Attack (Nintendo Wii and DS) as well as Octane Demo for the Nintendo Wii.

These are only a few examples of projects I've worked on over the last 15 years. I have worked in the gaming/realtime industry for 16 years and so this document only highlights a snippet of my work experiences.