HBO

WESTWORLD

REAL-TIME ENVIRONMENT BUILDOUT FOR WESTWORLD

MAGNOPUS

AT A GLANCE

We built a virtual environment for use in an LED volume while shooting parts of Westworld Season 4 Episodes 7 & 8. We worked closely with the Westworld Art Department to build a stylized Times Square environment set 30 years in the future.

Our main objective was to create a virtual environment buildout that served as a seamless digital extension of the physical set built within the LED volume. Part of that challenge was to ensure the digital set matched the color of the physical set pieces as they transitioned from physical to virtual.

This environment was created and displayed in Unreal Engine 4, and took advantage of the immersive nature of LED volumes to provide 360° imagery and interactive lighting on the actors inside the space. The environment consisted of various iconic Times Square building facades and recognizable pedestrian plazas, all of which provided the actors points of reference they could react to.

Resourcing

- 2 x Engineers
- 9 x Real-time VAD Artists

Schedule

- Concept Art & Design – 4 weeks
- VAD Construction – 12 weeks
- Volume Integration & Testing – 2 weeks
- Principal Photography – 2 days

CHALLENGE

We were asked to build a virtual environment for use in an LED volume while shooting parts of Westworld Season 4.

Our main objective was to create a virtual environment buildout that served as a seamless digital extension of the physical set built within the LED volume.

Part of that challenge was to ensure the digital set matched the color of the physical set pieces as they transitioned from physical to virtual. That included the colors and materials of the tiled walkway, concrete planters, and foliage inside the planters.

APPROACH

We worked closely with the Westworld Art Department to build a stylized Times Square environment set 30 years in the future.

We were presented with a piece of concept art by the client: their vision of future Times Square as it is depicted in Episode 7 and 8. 7th Ave, Broadway.

We started out using a purchased model of Times Square, along with all the immediate buildings and evaluated what aspect of the model we needed to keep, what to modify, and what to delete.

SOLUTION

This environment was created and displayed in Unreal Engine 4, and took advantage of the immersive nature of LED volumes to provide 360° imagery and interactive lighting on the actors inside the space.

The environment consisted of various iconic Times Square building facades and recognizable pedestrian plazas, all of which provided the actors points of reference they could react to.

TECHNICAL DETAILS

The environments for Westworld, Season 4 Episodes 7 and 8, were created and displayed in Unreal Engine 4, and took advantage of the immersive nature of LED volumes to provide 360° imagery and interactive lighting on the actors inside the space.

In this section, we zoom in on the technical requirements to build such an immersive environment, while exploring various challenges and solutions we encountered along the way.

SCENE LAYOUT

ARCHITECTURAL DESIGN

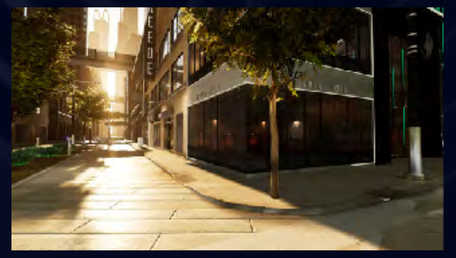
FOLIAGE

SHADERS / BLUEPRINTS

LIGHTING

ANIMATED BACKGROUND CHARACTERS

STATIC BACKGROUND CHARACTERS



UNREAL BLUEPRINTS

SCENE LAYOUT

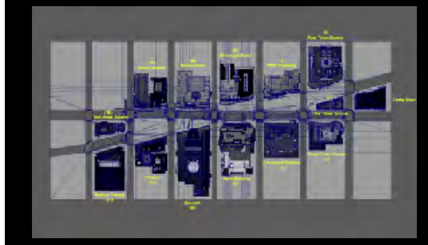
From the concept art of the client's vision of future Times Square, we started from a purchased model and evaluated what to keep, to modify, or delete.

What we did:

- Create additional background buildings.
- Create 2D building cards, in the far distance.
- To tie back to past Westworld episodes, we placed taller high-rise buildings recognizable from previous episodes.
- Created textures with a combination of photo projections with hand paint-over and tile-able textures.

The main theme of the new Times Square layout was the planters which ran down the main corridor, with two small walkways bisecting it at different intervals. The physical set was placed at one of those bisected intersections.

- We removed 45th street and replace it with a second pedestrian plaza.
- We created elevated covered walkways and pedestrian bridges connecting one building to another.

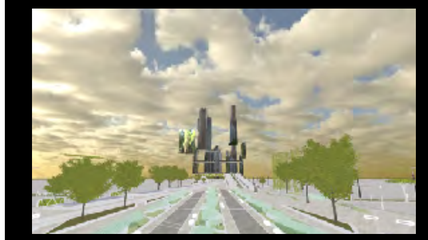


Purchased model

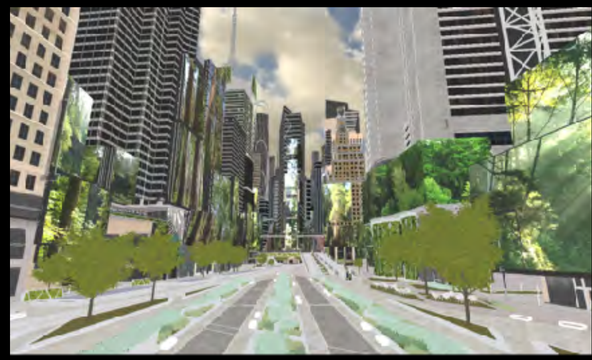
Generic background buildings

Hero background buildings

Background building clusters



2D background building cards



Background buildings together with foreground buildings

Sketchup blockout model from client showing the planters layout

Overhead plan showing alternate LED volume location

Elevated walkways with lighting

All buildings together with lighting

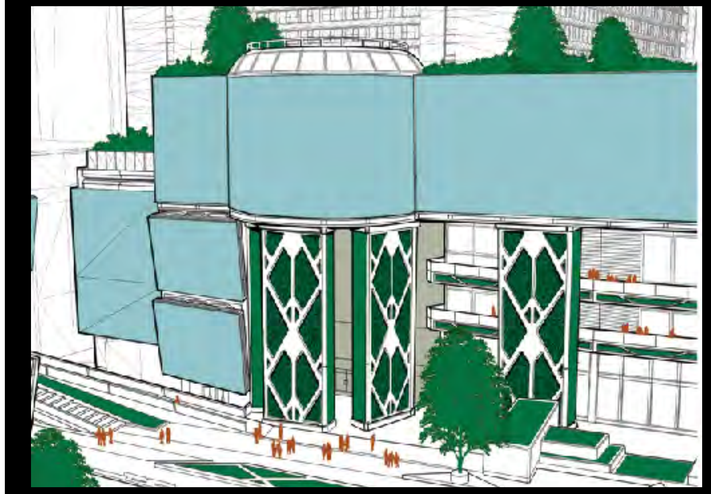
CASE STUDY: WESTWORLD SEASON 4

ARCHITECTURAL DESIGN

The main buildings surrounding the shooting location were our hero/foreground assets and were kept similar to the current-day NYC buildings with some slight façade changes.

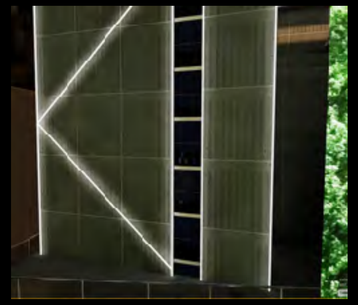
What we did:

- Remove billboards from street level and replaced them by green elements.
- Made the remaining billboards look more futuristic and streamlined.
- Textures on buildings:
  - We used tile-able textures/materials. These materials used greyscale base color maps that could be tinted and adjusted per material instance for different assets. We achieved customization by placing dirt and grunge decals on top of the tile-able materials.
  - Cubemap shaders were used to simulate interiors to all of the buildings immediately surrounding the shooting location. Cards were placed behind the transparent glass facades and then customizable cubemap materials applied to give the appearance of depth in the interior. These could be tinted and the brightness adjusted to suit each space.
  - Buildings with emissive lighting features used a vertex color material to customize the emissive color of each light strip.

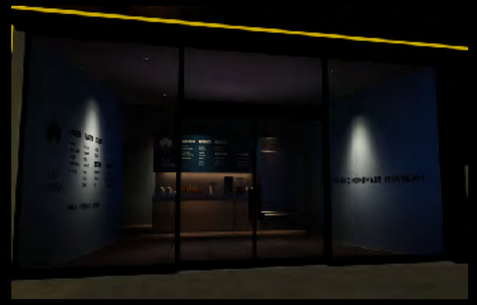


Magnopus concept for Marriott Hotel – Refined billboard placement and emphasized green space details

Sleeker billboard models to give a future-tech feeling



Placing dirt and grunge decals on top of the tile-able materials



Customizable cubemap materials give the appearance of depth in the interior

Cubemap shaders

Buildings with emissive lighting features

FOLIAGE

Trees

We used Speedtree which is free for use by licensed Unreal Engine users to create the background trees, European Ash. We incorporated fine morphological details such as smooth juvenile bark and bifurcated limb pattern. Several LODs were made for the trees, including a card version for instances far from the camera.

Ground Cover

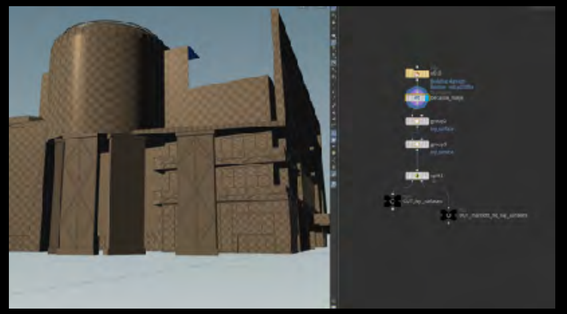
We create digital foliage that matched the species and color of the physical plants on stage. We sourced some models from Quixel Megascans, Turbosquid and comparable sites. From the high-poly assets we created low-poly versions. Finally, we placed the digital plants into the planters in our scene.

Ivy

Ivy covered architectural details in one of the shots. We created entirely within Houdini. *Creating ivy sprites*: Single grid (or plane) was created and UV mapped to correspond with the location of a desired detail in a Megascans texture set. In Masha, we used four different planes, each with different UV mappings. This gave us a wide variety of frond shapes and combinations for a very low tricount and call count.

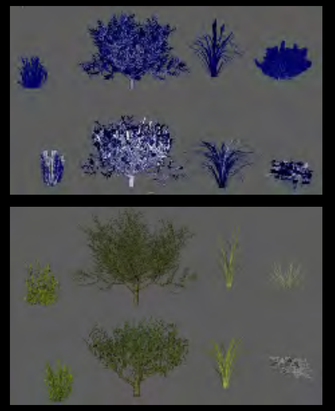
The trees were positioned in a variety of orientations and at a range of non-uniform scales in order to give them a more randomized feel in the scene. In practice, we actually only used one tree most of the time.

We achieved a look very similar to the client-provided concept art.

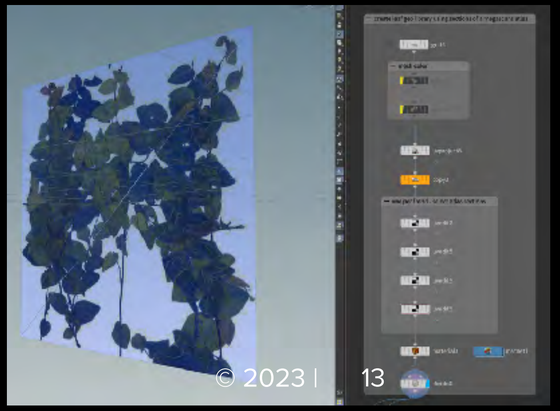


Preparing the ivy surfaces

Client's plant layout drawing



Purchased assets



Diffuse color, normal, and opacity textures sampling

FOLIAGE

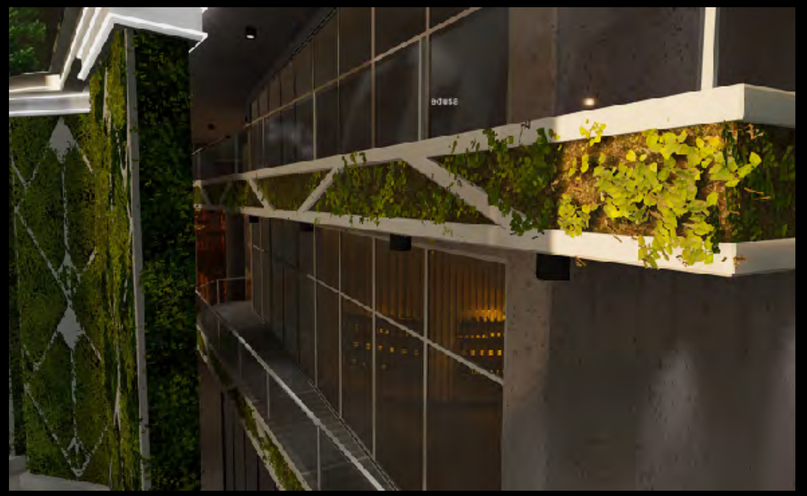
Ivy (continued)

Placing the sprites: Each point was the center of an ivy sprite. To add variation we used attributes like point normal, which UV mapping to use, and scale variations were added to each point and randomized within parameters. *Building the ivy material*: We use Unreal to build material using sections from the Megascans textures. *Adding support materials*: We added simple ivy material to surfaces below ivy sprites to give the impression of fullness.

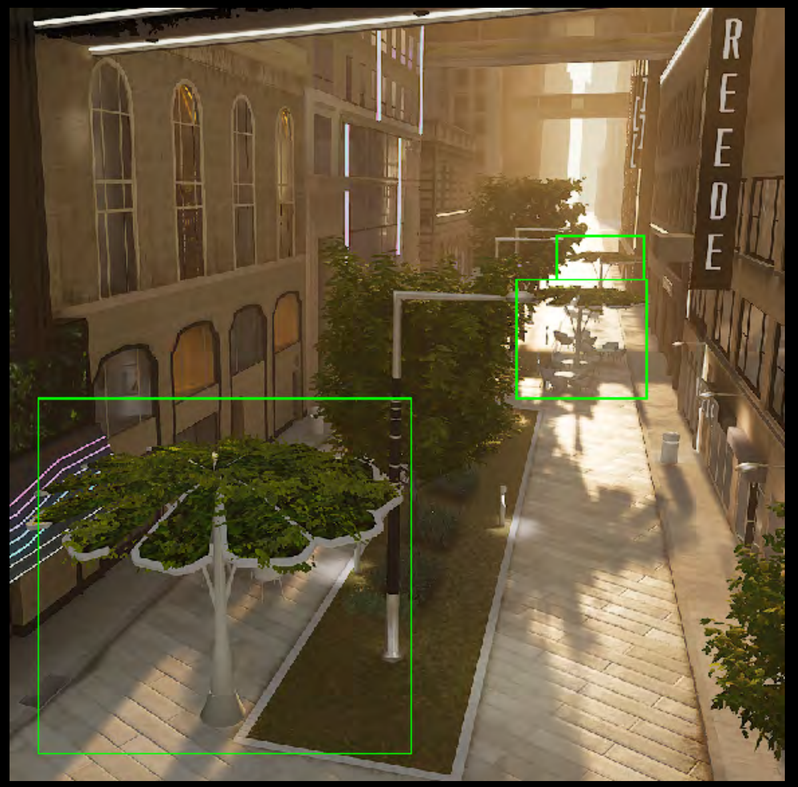
Canopies

Using the procedural ivy system previously detailed, we created cover for the top of the canopies scattered around Times Square.

Human's-eye view



Detail on upper floor, showing sprites in the sunlight. 'Growing matrix' support texture clearly visible behind sprites



Canopies

Front view

SHADERS

Billboard Shader Tech

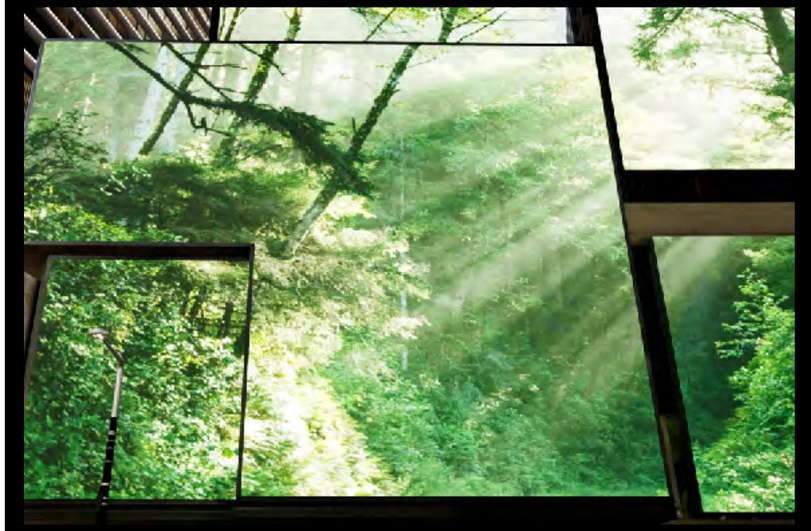
Those massively sized billboards showcased a beautiful scenery of a forest and in the later episode erupt into chaos. Many are broken and are glitching.

What we did:

- We designed a plan of action for how the billboards were actually going to be laid out and constructed.
- We used shaders logic to accurately display the angle a user would view a Billboard
- For the glitching, the billboards were given a new Texture Atlas. The models were then UV'd to correctly match and line up with the new image. The glitch animation was controlled by a Flipbook. The glitch effect was made in After Effects. The frames were then converted to a flipbook with controls for speed and tiling.

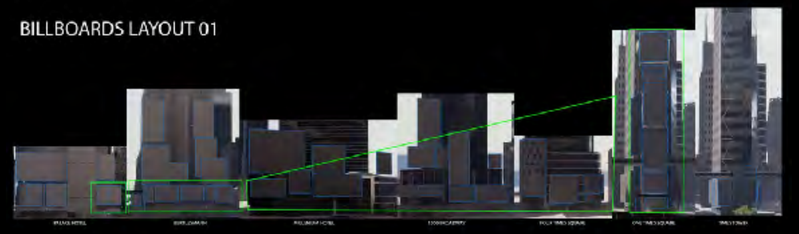
Fake Storefronts Shader

Due to time constraints, we used pre-rendered 3D cubemaps and shader logic to create the illusion of 3D storefronts in Unreal Engine. We could then quickly craft a hundred different storefronts in Photoshop and apply those directly into the material to swap out dozens of rooms in a matter of minutes.



Massively sized billboards showcasing a beautiful scenery of a forest

Fake storefronts shader



Billboards glitching

LIGHTING

The artistic treatment we adopted was:

- Back center as focal point in order to draw the eyes of the viewer down the street.
- Characters in the same scenes generally backlit, even if that meant flipping the sun.
- Three lighting scenarios based on cardinal directions: North, West, and South to allow the filmmakers to always have the sun at their backs if they wanted to backlight the scene in the LED Volume.
- Soft lighting coming from a dominant direction
- Terragen 4 generated HDRI bitmaps for background sky extensions.

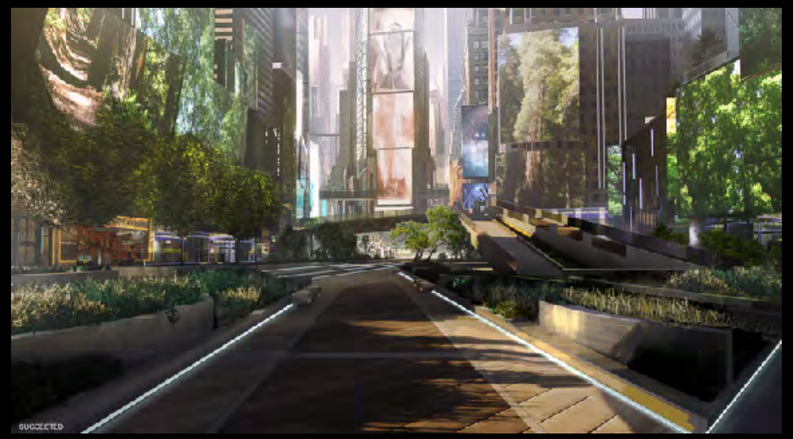
Lighting in Season 4, Episode 7

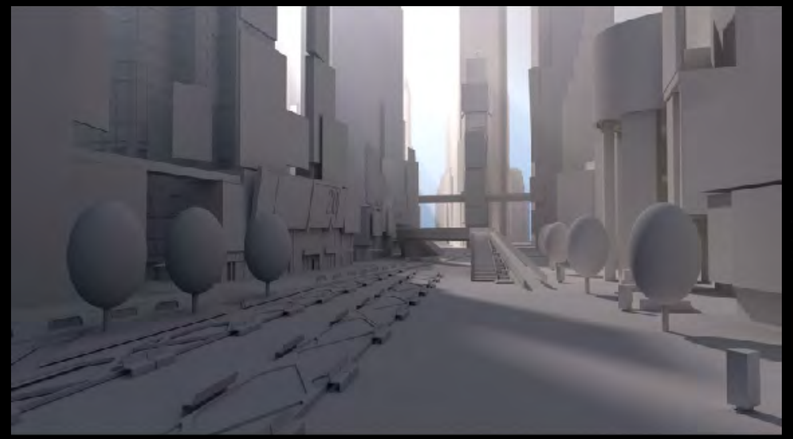
The lighting in this episode can be broken down into three parts:

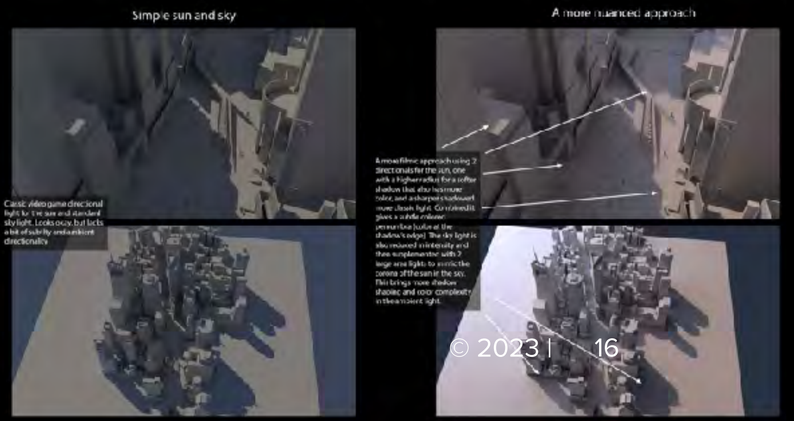
- Directional sunlight.
- Ambient skylight.
- Localized lighting per shot, the bulk of which was done on set, mostly around the trees and foliage in the foreground.

We used atmospheric fog for the distant city and matched physical lighting and brick patterns on set with the LED wall's physical floor. Real foliage grounded the scene in reality and concealed the lower portion of the LED wall to minimize pattern matching.

Concept art







LIGHTING

Lighting in Season 4, Episode 8
The major lighting design for this scene was to incorporate the glitching billboards as the largest source of light followed by the bright light at the end of the street.

The lighting in this episode can be broken down into the same three parts as in Episode 7. This scene had the addition of the large LED billboards, lamp posts, and LED strips around buildings. So we baked those onto the virtual environment from the emissive texture alone and supplement it with Unreal Light sources.

In this episode, we brought the atmospheric fog down closer to ground level and graduated it upwards towards the sky. Because this scene happens in a moment of chaos, we also created custom volumetric fog volumes spread around the street and down the alleys.

We also brought a fog machine to the stage to supplement the haze effect in the background. In Unreal Engine, we corrected the panels and colors to balance out the challenges introduced by this physical fog (washed out blacks and lower contrast).






LIGHTING

GPU Lightmass and Dynamic Lights

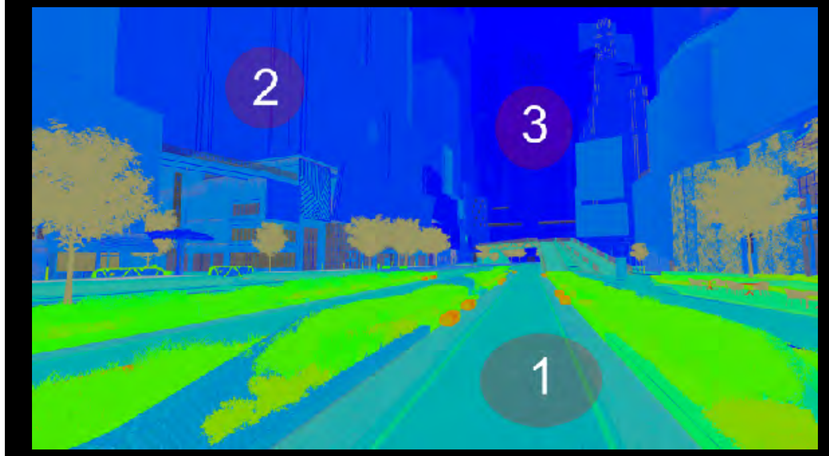
The virtual environments for both episodes were half real-time and half baked. We wanted to get the most realistic lighting by baking in our ambient sky light, but we also needed some dynamic lights because we had moving background crowds in the scene. For this, in Unreal Engine, we used GPU Lightmass to take advantage of our RTX enabled hardware. This provided the best quality lighting solution for baking in the engine.

Light flicker

Throughout episode 8 there are lamp posts that flicker on and off. This was done through a Light Function Material, a shader that uses logic to have the lights flicker on/off based on a cine curve.

Post Process & Color Grading

We didn't use much in the post process volume for either episode. Mostly, we turned off features such as Auto Exposure, Lens Flare, Vignette, and Ambient Occlusion. Color Correct Regions were utilized during live shoot days. These regions act as localized post process volumes and are useful for color matching the physical set to the virtual environment and much more.



Area near number '1 are immediate area to be the highest quality density for light maps. As distance from the camera increased, the light map density started to decrease in order to save time on rendering.

All of the regions had a yellow highlight where they were located in the scene. Many of them were layered in per shot.

All additional lights (around foliage and certain area highlights) were real-time.

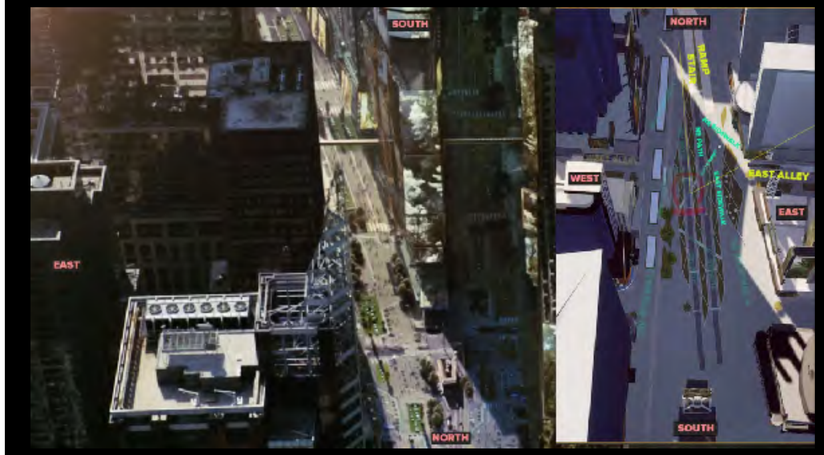
Fog – Behind the scene

ANIMATED BACKGROUND CHARACTERS

Background characters in Westworld presented the unique challenge of having realistic human motion, on an LED wall. We drew on our previous experience with CG humans to fill Times Square with a crowd: volumetric capture for humans in VR and AR experiences; crowd simulations using rigged 3D models and output to alembic caches; and stereoscopic videos of humans projected onto cards.

What we did:

- We focused on determining the appropriate placement for the crowd characters all around the scene. To ensure the characters moved in a way that followed the logic of the layout, we carefully considered their placement.
- Made sure the crowd felt integrated into the environment. We planned for crowd characters to use each one of the architectural elements such as elevated walkways, ramps, sitting and dining areas, storefronts, kiosks, etc., in a natural way. Pedestrians crossed bridges, folks would stop and look at kiosks, gaze into storefronts, and sit and chat at the tables.
- Crowd density is a major determining factor of scope when dealing with crowds. We needed to create a makeup that was appropriate for the story, rather than throwing as much as we possibly could at the scene.



We used an image like the one above to figure out the pathing system. Every virtual human looks like they're going somewhere on purpose and not trudging across the fountain or slamming into walls.

Determined directional priority for the scene by using the previz cameras as a heat map. In this heat map, the brighter the cone, the more often we planned to shoot in that direction. Therefore, we could direct our efforts to prioritize those areas for the camera.

ANIMATED BACKGROUND CHARACTERS

- CG methodology – Cards: Foreground crowd characters were played by real-life extras on set. We determined that any crowd character close enough to have their face distinguishable in a 150mm lens ought to be a real photographed human, not a low poly animated character. Using about 150 extras, we set up a blue screen and captured a grab bag of elements that we could composite onto cards in the scene.
- 3D characters: Distant crowd characters didn't need to have their faces visible, and their costumes were non-specific. We used Anima to make sure these 'blobs of colours' moved with some sort of human topography. In Anima, we drew walking paths and static crowds on street level and on balconies, ensuring that all paths started and ended out of view of the camera and that characters never approached too close to the camera's view. Unreal then pulled those scenes creating actors for each character, importing animations, textures, meshes, and materials.

This volume of crowd characters allowed us to plug gaps in the horizon when looking down alleyways and side streets. As a result, we were able to shoot our scene from any direction that struck the filmmaker's fancy. This was a great exercise to leverage the 360° nature of the Nant stage!

We included a variety of walking, standing, gawking, and talking variations to compose the whole scene and captured the extras at a variety of angles.

Ultimately, we were able to create a scene with over 1,500 crowd characters to give the scene a sense of life.

STATIC BACKGROUND CHARACTERS

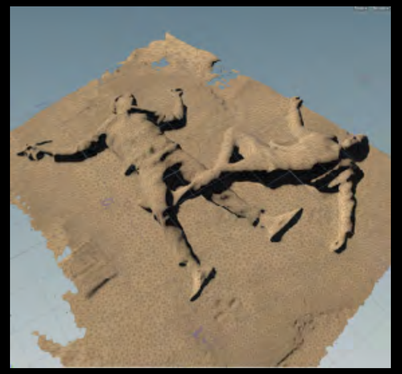
Times Square in Episode 8 was filled with dead bodies. They had to be photorealistic while remaining performant at runtime.

What we did:

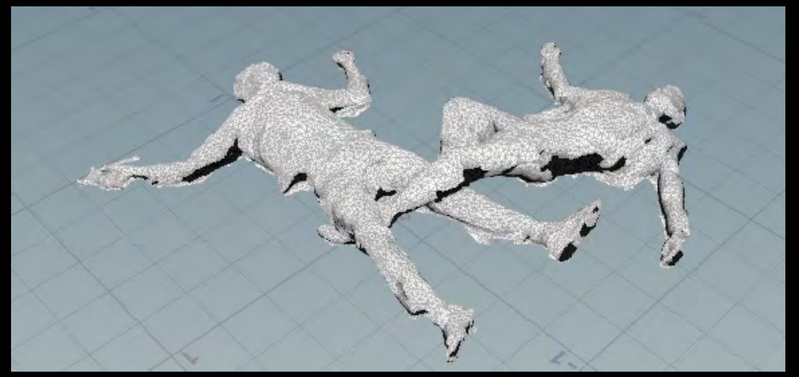
- Photogrammetry of the background actors at the same time as they were being photographed for use as VFX plates. We used Polycam installed on several iPads in order to do both the photo-capturing and processing.
- Mesh Creation: Polycam uploads to cloud servers running Apple's RealityKit solving software.
- Post-Production: We used Houdini to process the resulting meshes and textures into game-ready assets.
- Create 'Area of Interest': We keyed the diffuse color texture and extracted the tracking markers and scale markers. We deleted all geometry outside. We created bounding boxes around all items of interest (bodies and weapons) and deleted everything outside.
- Removed floor, keeping actors.
- Optimized mesh.
- Created the UVs.
- Baked textures for single mesh.
- Combining meshes.
- Baked texture Atlas.

~60 second scanning of actors in a corpse-like pose, with weapons in their hands

High-density mesh (100k tris) resulting from scan processing



Volume of interest (all contained points are shown in yellow)



Resulting mesh; almost all body and very little floor

Final result in the scene

UNREAL BLUEPRINTS

Room Changer Blueprint Overview

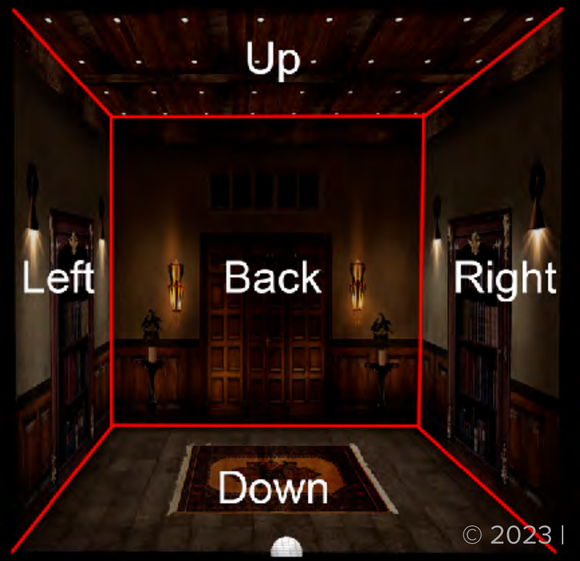
The windows and storefronts of the city mostly used a special cubemap shader to display a room with parallax. However, when a building had windows on two different sides, we needed a way to show the same room from different perspectives. We created a Blueprint that was geometry based that would apply a special material and logic to apply the correct room textures to be seen from the correct perspectives. As the rooms were made of textures, we could have a large number of artists create tens of rooms per day, which is more efficient than constructing it all through 3D methods. The blueprint also allowed adjustment of the room size.

Construction Script Overview

The bulk of the Blueprint came from the Construction Script inside the Blueprint. We started with a very simple cube with five faces. The “front” face was deleted as it acted as our window. The other walls were then categorized as Up, Down, Left, Right, and Back. Each wall of the geometry was assigned a Material Instance to show the correct wall for that side of the room. All sides of the wall sourced the same Parent Material and same Parent Texture, but there was shader logic in place that dictated where the texture went.

In the above image, notice how each window shows a different room, even though it’s the same room.

The rooms were designed as a texture in a special manner as shown above



If the room format changed to be a left or right side corner then the blueprint removed that particular wall so it could be seen through in the environment.

UNREAL BLUEPRINTS

Construction Script Setup
The setup consisted of an Enumerato, and a single variable which is the texture input (the desired room texture we wanted to see.) The geometry of the room was also set here.

Material Setup
The Blueprint has already gone through all the steps necessary to create a 3D room with 2D textures. The Blueprint still relies on a single parent material to ensure all of this is successful.

This blueprint took about a week of development and refinement and allowed the Art Department to quickly crank out numerous unique rooms far faster than 3D modeling, texturing, and lighting the rooms.

It also allowed incredible ease of use to change the rooms all around the level with global color grading controls to adjust them on the fly. If the Director on set wanted to change the room during a shot, we had the ability to within seconds. If the call was to change the color or brighten/dim the room, we could accommodate that.

Final results

