

# Changelog

See [progress reports](#) for a showcase of new features.

## Version 1.0.5 [2022-08-30]

### Filmmaker v0.4.6

- Added Pragma Renderer
  - Added SSAA
  - Added motion blur
  - Added directional light maps
  - Added volumetric spot-lights
  - Virtual Reality support
- Added new [transform gizmo system](#)
- Added/Improved viewport actor interaction
- Added undo-redo system
- Added [decals](#)
- Added [tutorials](#)
- Added option for importing map into project
- Added new actor context menu options
- Added support for rendering legacy Eye shader with Cycles X
- General UI improvement updates
- Significantly improve quality of baked lightmaps
- Added components:
  - pfm\_baked\_lighting
  - pfm\_cuboid\_bounds
  - pfm\_pragma\_renderer
  - pfm\_motion\_blur
  - pfm\_rt\_mover
  - pfm\_overlay\_object
  - pfm\_camera\_actor\_link
  - pfm\_selection\_wireframe
  - pfm\_cone\_wireframe

### Engine

- Added motion blur effect
- Added object BVH system for fast scene intersection tests

- Added directional lightmap baking
- Improved volumetric spot-light effect
- Fixed bloom effect appearing stretched on widescreen resolutions
- Fixed white edges around objects with masked alpha transparency
- Added entity components:
  - `renderer_pp_bloom`, `renderer_pp_dof`, `renderer_pp_fog`, `renderer_pp_fxaa`, `renderer_pp_tone_mapping`
  - `optical_camera`
  - `bvh`, `static_bvh_cache`, `static_bvh_user`
  - `light_map_data_cache`
- Fixed various crashes

## Lua

- Added `table.count`, `table.is_empty`
- Added `gui.Element:FindAncestorByClass`, `:SetAutoSizeToContents`, `:UpdateAutoSizeToContents`, `:ShouldAutoSizeToContentsX`, `:ShouldAutoSizeToContentsY`
- Added `game.Model:GetEyeball`, `.Load`
- Added `ents.EyeComponent:FindEyeballIndex`, `:GetEyeballProjectionVectors`
- Added `ents.ClickComponent.find_entities_in_kdop`
- Added `ents.BvhComponent:IntersectionTestKDop`
- Added `ents.BaseBvhComponent:RebuildBvh`, `:FindPrimitiveMeshInfo`, `.HitInfo:CalcHitNormal`
- Added `ents.DecalComponent:ApplyDecal`
- Added `util.ImageBuffer:SetPixelColor`
- Added `game.Model.Mesh.Sub:HasVertexWeights`
- Added python library
- Added `prosper.util.record_resize_image`, `.create_generic_image_descriptor_set`
- Added `Model.Mesh.Sub:MakeVerticesUnique`, `:SetVertices`, `:SetIndices`
- Added `asset.get_asset_state`
- Added `vector.calc_spherical_stereo_transform`
- Added `ents.LightMapComponent.DataCache:GetInstanceIds`, `:GetInstancePose`
- Added `ents.BaseEnvLightSpotComponent:CalcConeFalloff`, `:CalcDistanceFalloff`
- Added `ents.BaseEnvLightPointComponent:CalcDistanceFalloff`
- Added `util.ThreadPool:WaitForCompletion`
- Added `ents.citerator`, `ents.get_all_c`, `ents.IteratorFilterFunction`
- Added classes `ents.LightMapDataCacheComponent`, `ents.LightMapComponent.DataCache`

# Version 1.0.4 [2022-05-31]

## Filmmaker v0.4.5

- Replaced Cycles with Cycles X (With OptiX and OptiX denoising)

- Significantly improved performance of live raytracing render
- Live raytracing render now automatically updates on certain scene changes (e.g. lighting)
- Graph Editor is now mostly functional and can be used for animating
- Added internal [Web-Browser](#) which automatically detects and imports downloaded assets
- Added new project file format (old format is no longer supported)
- Added reflection probe and lightmap baking
- Added option to copy/paste and share actors with animation data between projects
- Added secondary and tertiary viewports
- Significantly reduce real-time performance impact of retargeted actors
- Sky settings have been moved to "sky" actor component to allow animating sky properties
- Sky overrides and angles can now be previewed in the game viewport
- Actors and actor components can now be removed
- Improved asset explorer filter
- "emission\_factor" entry field in material editor is now a color field instead of a simple slider
- General load and save performance improvements
- Added quicksave and autosave
- Added new rebindable hotkey system and default hotkeys
- Bones can now be animated with transform widgets in 3D viewport
- Support for volumetric spotlights when rendering with Cycles
- Generated model asset icons now zoom in and focus on the character head for character models
- Added IK configs, IK posing and animating with IK
- Actors can now be right-clicked in game viewport to edit materials directly
- Fixed timeline zoom issues and improved zooming behavior
- Fixed incorrect camera FOV values when importing a SFM session
- Lots of crash, interface, general bug fixes, improvements and QOL changes

## Engine

- Added support for meshes with more than 65535 vertices
- Significantly improved asset loading times
- Added model asset import support for formats: blend, fbx, dae, x3d, obj, abc, usd, pmx (MMD)
- Assets can now be automatically detected and imported from archive files
- Added support for Source Engine ["Eyes" shader](#)
- Added support for "\$alpha" VMT parameter
- Added console commands:
  - asset\_multithreading\_enabled
  - asset\_file\_cache\_enabled
  - render\_enable\_verbose\_output
  - debug\_dump\_font\_glyph\_map
  - locale\_localize
  - util\_convert\_cubemap\_to\_equirectangular\_image

- util\_convert\_equirectangular\_image\_to\_cubemap
- Added "-console" launch parameter arguments: "gui", "guid"
- Console command launch parameters that appear after "+map" are now executed after the map has been loaded
- Added entity components:
  - game\_animation\_player, game\_animation\_recorder
  - func\_portal, portal
  - liquid\_surface, liquid\_volume, buoyancy, surface, liquid
  - weapon\_phycannon
  - wireframe\_camera
- Added maps: test\_3d\_skybox, test\_ai\_nav, test\_fog, test\_mirror, test\_physics, test\_portal, test\_water, test\_particles
- Added gravity gun script
- Added support for rotating GUI elements
- Added support for UTF8 fonts and localization
- Added "local view target factor" for controlling character eyes
- Updated particle system to new renderer
  - Various blob particle improvements
- Fixed water buoyancy physics
- Fixed fog not rendering
- Fixed 3D skybox not rendering
- Fixed water not rendering
- Fixed issue where light source could appear as turned on despite being turned off or removed previously
- Fixed crash when loading models with custom activity names
- Added Fortress Forever to default mount list
- Added support for equirectangular skyboxes and sky angles
- Fixed crash when attempting to export model asset with invalid flex rules
- Fixed some model meshes not being rendered
- Maps in new-game menu are now split into native maps and external maps
- General performance improvements
- Spot-light inner cone angle has been replaced with blend fraction
- Spot-light outer cone angle has been changed from half-angle to full-angle
- prop\_dynamic entities no longer spawn with physics
- General crash fixes and improvements

## Lua

- Lua-defined component members can now be animated
- Add "prepared render command buffer" system for issuing multi-threaded render commands
- Component member flags can now be specified as strings
- Added Component member meta data information
- Added function bindings:
  - asset.find, asset.clear\_unused, asset.normalize\_asset\_name, asset.load, asset.reload, asset.precache, asset.is\_supported\_extension, asset.wait\_until\_loaded,

- asset.wait\_until\_all\_pending\_jobs\_complete, asset.poll, asset.poll\_all
- o ents.Entity:GetUri, :GetMemberValue, :SetMemberValue, :DebugPrintComponentProperties
- o ents.EntityComponent:Log, :GetUri, :GetMemberUri, :GetDynamicMemberIndices, :GetStaticMemberCount, :GetMemberIndices
- o ents.ModelComponent:ReloadRenderBufferList, :IsDepthPrepassEnabled, :SetDepthPrepassEnabled, :IsAutoLodEnabled, :SetAutoLodEnabled
- o ents.BaseEnvLightSpotComponent:GetBlendFractionProperty, :GetOuterConeAngleProperty, :GetConeStartOffsetProperty
- o ents.CameraComponent:GetFocalDistance, :SetFocalDistance, :GetFocalDistanceProperty
- o ents.PanimaComponent:DebugPrint, :GetCurrentTime, :SetCurrentTime, :GetCurrentTimeFraction, :SetCurrentTimeFraction, :ReloadAnimation
- o ents.LightComponent:SetLightIntensityType, :SetLightIntensity
- o ents.ReflectionComponent:GetLocationIdentifier, :GenerateFromEquirectangularImage
- o ents.LightComponent:IsBaked, :SetBaked
- o ents.add\_component\_creation\_listener
- o ents.find\_installed\_custom\_components, ents.get\_registered\_entity\_types
- o ents.RenderComponent:IsInPvs
- o ents.TransformComponent:Teleport
- o ents.WorldComponent:RebuildRenderQueues
- o ents.ModelComponent:UpdateRenderMeshes
- o file.is\_empty
- o debug.render\_mesh
- o game.get\_number\_of\_scenes\_queued\_for\_rendering, game.get\_queued\_scene\_render\_info, game.set\_gameplay\_control\_camera, game.reset\_gameplay\_control\_camera, game.get\_gameplay\_control\_camera, game.clear\_gameplay\_control\_camera, game.get\_primary\_camera\_render\_mask
- o geometry.calc\_rect\_circle\_touching\_position
- o gui.create\_render\_target, gui.create\_color\_image, gui.register\_default\_skin, gui.set\_focus\_enabled, gui.is\_focus\_enabled
- o game.DrawSceneInfo:AddSubPass
- o gui.Base:CheckPosInBounds
- o gui.DropDownMenu:FindOptionIndex
- o gui.Element:GetRelativePos, :SetLocalAlpha, :GetLocalAlpha, :IsRemovalScheduled, :ResetRotation, :GetRotationMatrix, :SetRotation, :SetStencilEnabled, :IsStencilEnabled, :SetLocalRenderTransform, :ClearLocalRenderTransform, :DrawToTexture, :GetLocalRenderTransform, :AnchorWithMargin, :ClearFocus, :SetParentAndUpdateWindow
- o gui.Text:GetTextHeight, :UpdateSubLines, :SetMaxLineCount, :GetMaxLineCount
- o gui.Shape:ClearBuffer, :SetShape, :GetBuffer, :SetBuffer, :GetVertexCount
- o gui.WIContextMenu:IsPopulated
- o gui.ProgressBar:SetLabelVisible
- o input.InputBindingLayer, input.add\_input\_binding\_layer, input.get\_input\_binding\_layers, input.get\_input\_binding\_layer,

- input.remove\_input\_binding\_layer, input.get\_core\_input\_binding\_layers,
- input.update\_effective\_input\_bindings, input.get\_effective\_input\_bindings,
- input.set\_binding\_layer\_enabled, input.is\_binding\_layer\_enabled,
- input.is\_binding\_layer\_enabled, input.InputBindingLayer:FindBoundKeys
- o locale.localize
- o math.snap\_to\_gridf, math.get\_power\_of\_2\_values, math.calc\_bezier\_point
- o math.Mat4.create\_reflection
- o math.Quaternion:MirrorAxis
- o math.Transform:ToPlane, math.Transform:Reflect
- o math.Vector2:GetAngle, math.Vector:Reflect
- o Model:GetExtensionData, :GetSubMesh, :FindSubMeshId
- o Model.Mesh.Sub:GetTriangle, :GetIndexType, :SetIndexType, :AddIndex, :SetIndex, :GetIndex, :HasUvSet, :GetExtensionData, :ReverseIndices, IndexType and MaxIndex enums
- o Renamed Model.Mesh.Sub:GetTriangles to :GetIndices
- o panima.Animation.load, :RemoveChannel
- o panima.Channel:Resize, :GetSize, :SetTime, :SetValue, :ClearValueExpression, :GetValueExpression, :RemoveValue, :FindInterpolationIndices, :FindIndex, :RemoveValueRange, :AddValueRange, :SortValues, :GetInterpolatedValue, :FindIndexRangeInTimeRange
- o prosper.blur\_texture
- o prosper.RenderTarget:GetColorAttachmentTexture, :GetDepthStencilAttachmentTexture
- o prosper.Window:GetMonitorBounds, :Maximize, :IsMaximized
- o shader.BaseTexturedLit3D:IsDepthPrepassEnabled, :SetDepthPrepassEnabled
- o table.is\_empty, table.table\_to\_map
- o util.File:GetSize, util.import\_assets, util.generate\_hair\_file, util.generate\_hair\_data
- o udm.is\_numeric\_type, is\_generic\_type, is\_non\_trivial\_type, type\_to\_string, get\_numeric\_component, lerp, get\_numeric\_component\_count, is\_vector\_type, is\_matrix\_type, get\_matrix\_row\_count, get\_matrix\_column\_count, is\_array\_type, get\_class\_type, parse
- o input.is\_ctrl\_key\_down, is\_alt\_key\_down, is\_shift\_key\_down
- o udm.Array:ToTable, udm.PropertyWrapper:ToTable, :ToAscii
- o udm.Property:AddValueRange, :RemoveValueRange, :SetValueType, :RemoveValue, :Copy
- o Material:SetLoaded
- Added classes/enums:
  - o util.ThreadPool
  - o util.ZipFile
  - o gui.DrawToTextureInfo
  - o asset.AssetLoadFlags
  - o shader.BindState
- Fixed EntityComponent:GetMemberValue, :SetMemberValue not working correctly for some member types
- Fixed util.remove not removing all objects in table correctly in some cases
- Fixed Lua-defined component members of type string not working correctly

# Version 1.0.2 [2021-09-13]

## Filmmaker v0.4.3

- Added options for animating actor properties
- `util_export_asset` command can now export retargeted models
- Fixed depth buffer issues if camera nearZ or farZ is set to 0
- Fixed some meshes not being rendered when rendering with Cycles/LuxCoreRender

## LuxCoreRender

- Added volumetric lighting options
- Added subdivision options for materials
- Added option for enabling [PhotonGI cache](#)

## Engine

- Added new animation system
  - Added animation math expression system
  - Added animation driver system
- Added entity component member system
- Fixed translucent objects causing depth rendering issues
- Fixed eyes\_updown / eyes\_rightleft flex controllers not working correctly
- Fixed crash instance when trying to import some Source Engine materials with invalid texture references
- Fixed crash instance when importing some Source Engine maps
- Fixed issue where some meshes would appear invisible if the actor has a skin > 0
- Fixed textures with format ABGR8888 getting imported with swapped color channels

## Lua

- Added new [Lua API documentation](#)
- Added **CompositeComponent:GetEntities, :GetRootGroup, :AddChildGroup, :GetChildGroups**
- Added **panima** library
- Added **ents.EntityComponent:GetMemberIndex, :GetMemberInfo**
- Added **util.get\_object\_hash**

- Added **ents.UniversalEntityReference**, **ents.UniversalComponentReference** and **ents.UniversalMemberReference**
- Added **ents.AnimatedComponent.parse\_component\_channel\_path**, **:ClearPreviousAnimation**, **:ReloadAnimation**, **:AdvanceAnimations**
- Added **ents.EntityComponent:GetMemberInfos**
- Added **udm.Array:GetArrayType**, **udm.Element:GetArrayValues**, **:SetArrayValues**, **udm.to\_json**
- Added type aliases:
  - Entity components can be used in place of entity arguments on C++ function calls
  - EulerAngles and Quaternions can be used interchangeably
  - Vector types can be used interchangeably where applicable

## Version 1.0.1 [2021-06-22]

### Filmmaker v0.4.2

- Improve retargeting performance
- Fix various retargeting issues
- General retargeting improvements
- Update default material hair settings

### Engine

- Added updated Lua documentation and ZeroBrane configuration files
- Updated "lua\_help" documentation database
- Added "TF2" and "HL1" demo gamemodes
- Fixed random crashes on some hardware
- Fixed issue where game could freeze permanently during shutdown
- Fixed Lua JIT-Compiler not being enabled
- Fixed importing glTF models saving material files in wrong location
- Fixed exporting maps not working in some cases when the map name has a "." in the name
- Fixed crash on startup with certain launch parameters
- Fixed automatic asset reloading on asset file change not working
- Fixed issue where Steam locations wouldn't get mounted properly on new Steam installations
- Added additional crash dump debug information
- Transitioned AI-nav file-format to UDM system
- Transitioned Lua documentation file-format to UDM system

## Lua

- Added **Skeleton:IsRootBone** string parameter overload
- Added **Skeleton:GetBoneHierarchy**
- Added **phys.Transform.IDENTITY, :GetAngles, :SetAngles, :GetForward, :GetRight, :GetUp, .x, .y, .z, .pitch, .yaw, .roll, phys.ScaledTransform.IDENTITY**
- Added **CompositeComponent:ClearEntities**
- Add **EVENT\_UPDATE\_BONE\_POSES** and **EVENT\_ON\_BONE\_POSES\_FINALIZED** animated component events
- Added **table.copy**
- Added **engine.library\_exists**
- Added **ents.PlayerComponent:SetFlashlightEnabled, :ToggleFlashlight, :IsFlashlightEnabled**
- General LuaDoc improvements

## Version 0.9.15 [2021-05-30]

### Filmmaker v0.4.1

- Added console commands: **pfm\_max\_fps, pfm\_asset\_icon\_size**
- Added bone visualization to retarget editor
- Retarget rigs can now be auto-generated from existing rigs if a match can be found
- Various retargeting improvements
- Fixed "hair" settings in material editor not working
- Fixed PFM not showing up in new game menu
- Fixed material editor and explorer not working properly

### Engine

- Added console commands: **udm\_validate, asset\_clear\_unused, asset\_clear\_unused\_textures, asset\_clear\_unused\_models, asset\_clear\_unused\_materials, cache\_version\_target, loc\_reload**
- Added launch option: **-audio\_api <audioApi>**
- Added game mount priority options
- Physics engine can now be toggled between **PhysX** and **Bullet**
- **Bullet** is now the default physics engine
- Audio engine is now implemented as a module (can be changed in options)
- Assets are now automatically cleared from memory when not in use
- Transition particle system format to UDM
- Fixed entities appearing invisible in some cases
- Fixed various character animation issues

- Fixed character eyes not working correctly if entity is scaled
- Fixed issue where models with a "." in the name could not get imported properly
- Fixed maps not being rendered if map was compiled without BSP
- Fixed crash when generating reflection probes
- Fixed crash when loading map with invalid world model
- Fixed various crash instances
- Fixed various shader/rendering issues

## Lua

- Added **debug.beep** and **debug.print**
- Added **phys.CollisionObject:SetAlwaysAwake, :IsAlwaysAwake**
- Added **math.map\_value\_to\_fraction** and **math.map\_value\_to\_range**
- Added **locale.set\_text**
- Added **composite** and **animated2** entity components
- Added **util.DirectoryChangeListener** class
- Added **animation.Channel, .Slice, .Animation2, .Pose, .Player** classes
- Added **ents.Entity:SetUuid**
- Added **ents.RenderComponent:GetRenderModeProperty**
- Added **ents.EyeComponent:GetEyePose**
- Added **udm.PropertyWrapper:GetChildCount, :HasValue** and **:ToAscii**
- Added **ents.AnimatedComponent:GetLayeredAnimations, :ApplyLayeredAnimations**
- Added **game.Model.Animation:GetBoneId**
- Added **ents.IteratorFilterModel**
- Added **asset.delete, .determine\_type\_from\_extension, .get\_legacy\_extension, .get\_binary\_udm\_extension, .get\_ascii\_udm\_extension, .determine\_format\_from\_data, .determine\_format\_from\_filename, .matches, .relative\_path\_to\_absolute\_path, .absolute\_path\_to\_relative\_path, .get\_udm\_format\_extension, .get\_asset\_root\_directory, .exists, .find\_file, .is\_loaded**
- Added **file.find\_absolute\_path**
- Added new overload for **file.remove\_file\_extension** with parameter to specify which extensions to remove
- Added **udm.AssetData:SetData**
- Added **gui.find\_focused\_window, gui.get\_primary\_window, gui.find\_window\_under\_cursor, gui.get\_base\_elements, gui.add\_base\_element**
- Added **gui.Element:GetRootElement, :GetRootWindow**
- Added **shader.BasePbr** base class
- Added **prosper.create\_window, prosper.WindowCreateInfo** and **prosper.Window** classes
- Fixed **ents.Entity:SetEnabled** Lua binding not behaving correctly in certain cases
- Scripts loaded with **lua\_exec** and **lua\_exec\_cl** now auto reload when changed (Unless "nocache" argument is set)

- **util.register\_class** now defines a default constructor and tostring method
- **util.register\_class** now returns class object (or nil) instead of boolean

## Version 0.9.14 [21-04-16]

### Filmmaker

- Added animation and flex retargeting system
- Transform widget can now be toggled between world/local/view space
- Added basic posing and animating capabilities
- Improved hair rendering quality with LuxCoreRender
- Added hair configuration options to material editor
- Added live raytracing preview

### Engine

- Added "util\_export\_asset" option to export assets to Source Engine model/material/texture format with optional fake pbr conversion
- Transitioned asset file formats to UDM system
- Added console commands: "udm\_convert", "save", "crash", "debug\_render\_memory\_budget" and "debug\_render\_memory\_stats"
- Exporting model with glTF exporter now includes additional UV sets (such as lightmap UVs)
- Fixed rare random deadlock issue
- Added entity UUIDs to uniquely identify entities
- Added additional crash dump debug information

### Lua

- Added **asset.get\_supported\_extensions**, **asset.export\_texture\_as\_vtf**, **asset.TYPE\_MAP**
- Added **ents.GetUuid**, **ents.AttachableComponent:GetBone**
- Added **Quaternion:Distance**
- Added class **ents.IteratorFilterUuid**
- Added **udm** library
- Added **game.get\_material**
- Added **util.BSPTree.Node:GetIndex**
- Added **game.Material:Reset**
- Added **game.Model.Animation.Load**, **game.Model.FlexAnimation.Load**
- Added **game.Model.Mesh.Sub:Copy**
- Added **game.Model.Animation:GetBoneId**
- Added **game.Model.Eyeball:GetUpperLidFlexIndices**, **:GetUpperLidFlexAngles**, **:GetLowerLidFlexIndices**, **:GetLowerLidFlexAngles**, **:GetUpperLidFlexIndex** and

### **:GetLowerLidFlexIndex**

- Added **game.Model.FCOPY\_BIT\_FLEX\_ANIMATIONS** flag
- Added **ents.ModelComponent:GetBodyGroups**, **:SetBodyGroups**
- Added **ents.AnimatedComponent:GetBaseAnimationFlags**, **:SetBaseAnimationFlags**, **:GetLayeredAnimationFlags**, **:SetLayeredAnimationFlags**, **ents.AnimatedComponent:SetAnimatedRootPoseTransformEnabled**, **:IsAnimatedRootPoseTransformEnabled**, **:AddRootPoseBone**, **:SetRootPoseBoneId**, **:GetRootPoseBoneId** and **ents.AnimatedComponent.FPLAYANIM\_LOOP**
- Added **ents.LightComponent:SetMorphTargetsInShadowsEnabled**, **:AreMorphTargetsInShadowsEnabled**, **.BakeSettings.globalLightIntensityFactor** and **.import\_lightmap\_atlas**
- Added **ents.ReflectionProbeComponent:GetIBLMaterialFilePath**, **:CaptureIBLReflectionsFromScene** and **:RequiresRebuild**
- Added **prosper.ImageCreateInfo:FLAG\_ALLOCATE\_DISCRETE\_MEMORY\_BIT**, **FLAG\_DONT\_ALLOCATE\_MEMORY\_BIT**, **FLAG\_SRGB\_BIT**, **FLAG\_NORMAL\_MAP\_BIT**
- Added **prosper.Image:IsSrgb**, **:IsNormalMap**, **:SetSrgb**, **:SetNormalMap** and **:Convert**
- Added **prosper.Mesh:ClearBuffers**
- Added **ents.LightMapComponent:SetLightmapAtlas**
- Added **ents.RenderComponent:ClearBuffers**
- Removed **game.Model.Eyeball:GetUpperFlexDesc**, **:GetLowerFlexDesc**, **:GetUpperTarget**, **:GetLowerTarget**, **.lowerLidFlexDesc** and **.upperLidFlexDesc**

## Version 0.9.13 [2021-03-14]

### Filmmaker

- Fully integrated LuxCoreRender
- Added support for transparent backgrounds/skies when rendering with LuxCoreRender
- Added new render tool commands (LuxCoreRender only):
  - **preview**: Creates a preview image of what's been rendered so far and opens it in the default image program
  - **pause**: Pauses the current render
  - **resume**: Resumes the current render
  - **stop**: Stops the render and uses what's been rendered so far to generate the final image
  - **suspend**: Creates a .rsm-file, which can be used to continue the render at a later date
  - **export**: Exports the entire scene in a LuxCoreRender format (including models and textures), which can be used to render it with the [standalone LuxCoreRender](#)

## [program](#)

- Fixed issue where importing textures via drag-and-drop would not generate mipmaps

## Engine

- Updated [build instructions](#)
- Morph targets now affect shadows
- Increased bone limit from 512 to 1024
- Added console commands "strip\_weapons" and "cl\_render\_frustum\_culling\_enabled"
- Reduced number of entities with unnecessary tick/logic events to improve performance
- Characters now blink if no eyeballs are defined in the model, but blink flex controllers exist
- Added [UDM](#) data format for unifying asset file formats
- Fixed issue where viewmodel entities would use incorrect field of view
- Fixed viewmodel not appearing in some cases if viewmodel was changed after weapon was deployed
- Changed entity component net event system so net events no longer have to be registered on the server *before* they are registered on the client
- Merged engine with client library and engine with shared

## Lua

- Removed gamemode system and replaced it with gamemode entity components
  - Allows changing gamemodes on the fly, as well as running multiple different gamemodes at the same time
- Added "player" and "gamemode" entity component types (and sub-directories)
- All entity components can now have individual tick/logic events instead of relying on the logic component
- Added **udm** library for loading/saving/working with udm assets
- Added **ents.ModelComponent:GetBodyGroups** and **:SetBodyGroups**
- Added **math.Quaternion:Distance**
- Added **ents.PhysicsComponent:EnableCollisions**, **:DisableCollisions**, **:SetCollisionsEnabled**, **:SetSimulationEnabled** and **:IsSimulationEnabled**
- Added **ents.CharacterComponent:SetMoveController** and **:GetMoveController**
- Added **util.DataStream:WriteVertex**, **:ReadVertex**
- Added **Vector:GetYaw** and **:GetPitch**
- Added **util.generate\_uuid\_v4**
- Added **ents.register\_component\_net\_event**
- Added **ents.AttachableComponent.FATTACHMENT\_MODE\_FORCE\_TRANSLATION\_IN\_PLACE**, **FATTACHMENT\_MODE\_FORCE\_ROTATION\_IN\_PLACE** and **FATTACHMENT\_MODE\_FORCE\_IN\_PLACE**

# Version 0.9.12 [2021-02-08]

## Filmmaker

- Added support for rendering with [LuxCoreRender](#)
- Added fur rendering options with LuxCoreRender

## Engine

- Massive general performance improvements
- Added automatic rendering instancing system
- Added flex animation system
- Added new bloom/glow implementation
- Added lightmap baking with Cycles
- Added console commands **debug\_render\_octree\_dynamic\_find**, **debug\_render\_stats**, **render\_vsync\_enabled**, **render\_draw\_static**, **render\_draw\_dynamic**, **render\_translucent**, **cl\_render\_shadow\_lod\_bias**, **render\_queue\_worker\_thread\_count**, **debug\_textures**, **sky\_override**, **render\_queue\_entities\_per\_worker\_job**, **render\_queue\_worker\_jobs\_per\_batch**, **render\_instancing\_threshold** and **render\_instancing\_enabled**
- Fixed issue where camera would move upwards continuously in fullscreen windowed resolution mode
- Fixed first LOD being skipped when importing Source Engine models
- Fixed model LODs not working
- Fixed various crash bugs
- Fixed bone scaling not working correctly in some cases

## Lua

- Added **ents.RenderComponent:GetLocalRenderBounds**, **ents.RenderComponent:GetLocalRenderSphereBounds**, **ents.RenderComponent:GetAbsoluteRenderBounds**, **ents.RenderComponent:GetAbsoluteRenderSphereBounds**, **ents.RenderComponent:SetLocalRenderBounds**, **ents.RenderComponent:ShouldCastShadows**, **ents.RenderComponent:ShouldDraw**, **ents.RenderComponent:ShouldDrawShadow**, **ents.RenderComponent.EVENT\_ON\_RENDER\_MODE\_CHANGED** and **ents.RenderComponent.EVENT\_UPDATE\_INSTANTIABILITY**
- Added **util.DataBlock:GetVector2**
- Added **game.Material.detail\_blend\_mode\_to\_enum** and **game.Material.DetailBlendMode** enums
- Added **ents.SceneComponent:GetRenderQueue**, **ents.SceneComponent:BuildRenderQueue** and **ents.SceneComponent.CreateInfo**
- Added **math.Vector:ToScreenUv**

- Added **game.set\_default\_game\_render\_enabled**, **game.is\_default\_game\_render\_enabled**, **game.render\_scenes**, **game.set\_debug\_render\_filter** and **game.update\_render\_buffers**
- Added **game.RENDER\_FLAG\_TRANSLUCENT\_BIT**
- Added **util.ImageBuffer:SwapChannels**
- Added **prosper.get\_api\_identifier**, **prosper.get\_api\_abbreviation**, **prosper.wait\_for\_current\_swapchain\_command\_buffer\_completion**, **prosper.create\_primary\_command\_buffer** and **prosper.create\_secondary\_command\_buffer**
- Added **prosper.CommandBuffer:Flush**
- Added **game.DrawSceneInfo.FLAG\_FLIP\_VERTICALLY\_BIT**, **game.DrawSceneInfo.FLAG\_DISABLE\_RENDER\_BIT** and **game.DrawSceneInfo.flags**
- Added classes **game.RenderQueue**, **game.DepthStageRenderProcessor**, **game.LightingStageRenderProcessor**, **shader.GameWorld**, **game.Model.FlexAnimation** and **game.Model.FlexAnimation.Frame**
- Added **ents.ModelComponent:SetMaxDrawDistance**, **ents.ModelComponent:GetMaxDrawDistance** and **ents.ModelComponent.EVENT\_ON\_RENDER\_MESHES\_UPDATED**
- Added **ents.AnimatedComponent:AreSkeletonUpdateCallbacksEnabled**, **ents.AnimatedComponent:SetSkeletonUpdateCallbacksEnabled**, **ents.AnimatedComponent.EVENT\_ON\_SKELETON\_UPDATED**, **ents.AnimatedComponent.EVENT\_ON\_BONE\_MATRICES\_UPDATED** and **ents.AnimatedComponent.EVENT\_ON\_BONE\_BUFFER\_INITIALIZED**
- Added **util.BSPTree:FindLeafNodesInAABB**
- Added **ents.LightmapComponent.bake\_lightmaps**, **ents.LightmapComponent:SetExposure**, **ents.LightmapComponent:GetExposure**, **ents.LightmapComponent:GetExposureProperty** and **ents.LightmapComponent.BakeSettings**
- Added **ents.FlexComponent:PlayFlexAnimation**, **ents.FlexComponent:StopFlexAnimation**, **ents.FlexComponent:GetFlexAnimationCycle**, **ents.FlexComponent:SetFlexAnimationCycle** and **ents.FlexComponent:SetFlexAnimationPlaybackRate**
- Added **game.Model:GetFlexAnimations**, **game.Model:GetFlexAnimationNames**, **game.Model:GetFlexAnimationCount**, **game.Model:AddFlexAnimation**, **game.Model:LookupFlexAnimation**, **game.Model:GetFlexAnimation**, **game.Model:GetFlexAnimationName**, **game.Model:ClearFlexAnimations**, **game.Model:RemoveFlexAnimation**, **game.Model.AddFlexControllerId**, **game.Model.SetFlexControllerValue**, **game.Model.LookupLocalFlexControllerIndex**, **game.Model:GetLOD** and **game.Model.Mesh:RemoveSubMesh**
- Added **game.RasterizationRenderer:ReloadPresentationRenderTarget**
- Added **file.to\_relative\_path**
- Added **geometry.calc\_point\_on\_triangle** and **geometry.calc\_triangle\_area**
- **util.save\_image** now has an option for saving cubemap textures

- Removed **ents.SceneComponent:PrepareRender**, **ents.TransformComponent:GetPosProperty**, **ents.TransformComponent:GetRotationProperty**, **ents.TransformComponent:GetScaleProperty**, **game.draw\_scene**, **ents.RenderComponent:GetRenderBounds**, **ents.RenderComponent:SetRenderBounds**, **ents.RenderComponent:GetRenderSphereBounds**, **ents.RenderComponent:GetRenderModeProperty**, **ents.RenderComponent:SetDepthBias**, **ents.RenderComponent:GetDepthBias**, **game.DrawSceneInfo.flipVertically** and **ents.RenderComponent.RENDERMODE\_AUTO**
- Changed parameters of **ents.SceneComponent:Render** and **ents.RenderComponent:UpdateRenderBuffers**

## Version 0.9.11 [2020-11-11]

### Filmmaker

- Cycles render settings are now saved with the project
- Changed frame render order when rendering animations with external render tool for easier quality control
- Added support for detail maps when rendering with Cycles
- Fixed emission textures having no effect when rendering with Cycles
- Fixed IOR value being set improperly when opening material in material editor
- Fixed material editor not updating properly when switching to a different material
- Fixed HDR output format not working properly when rendering with Cycles
- Changed default gamma correction value when rendering with Cycles to 2.2 (was 2.4 before)

### Engine

- Added support for **\$detail**, **\$detailscale**, **\$detailblendfactor**, **\$detailblendmode** and **\$detailtint** VMT parameters
- Added CS: GO to default game mount list
- Fixed issue where entities with morph target animations would get visually corrupted if they were made invisible and then visible again
- Fixed issue where ingame cursor position would mismatch actual cursor position in fullscreen windowed resolution mode
- Fixed issue where camera would move upwards continuously in fullscreen windowed resolution mode

### Lua

- Replaced class **game.Scene** with **ents.SceneComponent**
- Added **ents.SceneComponent.OcclusionCullingMethod** enums
- Added **game.Material.DetailBlendMode** enums
- Added **game.Material.detail\_blend\_mode\_to\_enum**
- Added **ents.SceneComponent:SetOcclusionCullingMethod**
- Added **game.Model:GetBodyGroupMesh**
- Added **game.Material:SetShader**
- Added **ents.Entity:GetBodyGroup** and **:SetBodyGroup** overloads for body group IDs
- Added **ents.BaseEntityComponent:Save**, **ents.BaseEntityComponent:Load** and **ents.BaseEntityComponent:Copy**
- Added **ents.Entity:RemoveFromAllScenes** and **ents.Entity:AddChild**
- Added **debug.breakpoint** (Only available if `-luaext` was specified in the launch options)
- Added **math.Mat4:ApplyProjectionDepthBiasOffset**
- Added **ents.RenderComponent:SetReceiveShadows** and **ents.RenderComponent:IsReceivingShadows**
- Added **ents.EyeComponent:ClearViewTarget**
- Added **util.DataBlock:GetVector2**
- Renamed **game.create\_scene** to **ents.create\_scene**
- **util.remove** now has a second argument for removing objects safely (if available)
- **toboolean** and **toint** now accept numbers and booleans as arguments
- Changed parameter order of **ents.create\_prop**

## Version 0.9.10

### Filmmaker

- Added Cycles option for setting the number of frames to the end of a clip or session
- Render jobs now take up a lot less disk space for animations (static geometry is now stored shared instead of every frame)
- Slightly improved overall rendering speed
- Fixed rendering taking up your entire CPU resources when rendering with your GPU
- Fixed an issue where rendering several times consecutively could slow your PC down to a crawl
- Fixed VR renders being flipped horizontally
- Fixed a denoising issue which caused less than optimal denoising quality
- Fixed issue where the tiles in the final image would have mismatching sample counts and some tiles would be lower quality as a result
- Fixed render preview not being interactable when rendering VR images
- Fixed potential crash when doing a preview render with the Vulkan renderer being used

### Engine

- Minor occlusion culling performance optimizations

- Greatly improved reflection probe rendering speed
- Fixed issue where entities would get rendered with error materials if their skin would exceed the max skin count of their model
- Fixed bone scaling not working correctly
- Fixed massive performance hit for light sources that cast shadows in the past, but had shadow casting disabled
- Fixed BSP occlusion culling not working for imported Source Engine maps
- Fixed meshes not being rendered if entity skin exceeds maximum skin count
- Improved lightmap implementation
- Various minor bug fixes and improvements
- Improved loading times when starting a game

## Lua

- Added class **util.Version**
- Added **exec**
- Added **ents.create\_prop**
- Added **ents.AnimatedComponent:SetBindPose** and **ents.AnimatedComponent:GetBindPose**
- Added **ents.LightMapComponent:UpdateLightmapUvBuffers**, **ents.LightMapComponent:ReloadLightmapData** and **ents.LightMapComponent:SetLightmapAtlas**
- Added **ents.LightMapReceiverComponent:UpdateLightmapUvData**
- Added **gui.Element:InjectMouseClick** and **gui.Element:InjectKeyPress**
- Added **ents.Entity:IsDisabled** and **ents.Entity:IsTurnedOff**
- Added **util.Path>CreateFilePath** and **util.Path>CreatePath**
- Added **ents.MapComponent:GetMapIndex**
- Added **ents.AnimatedComponent:PlayAnimation** overloads for playing animations by id
- Added **util.DataBlock:ToString**
- Added **game.Model.Frame:Copy**, **game.Model.Skeleton.Bone:IsAncestorOf** and **game.Model.Skeleton.Bone:IsDescendantOf**
- Added **math.calc\_average\_rotation**
- Added **file.strip\_illegal\_filename\_characters**
- Added **game.Model.Mesh:GetReferenceld**
- Added **game.Model.Mesh.Sub:FlipTriangleWindingOrder**, **:SetVertexCount**, **:SetIndexCount**, **:AddUVSet**, **:GetUVSetNames**, **:SetVertexTangent** and **:HasUVSet**
- Added **string.hash**
- Added new overload for **game.Model.Mesh.Sub:GetUVs**, **:SetVertexUV**, **:GetVertexUV** with UV set parameter
- Changed argument format for **util.pack\_zip\_archive**
- Renamed **ents.AnimatedComponent:GetBindPose** to **ents.AnimatedComponent:GetBoneBindPose**
- **util.remove** now accepts a table as argument
- **file.write** now automatically creates the file path if it doesn't exist
- File write operations now accept paths that include "addons/<addonName>/" as prefix

---

Revision #90

Created 2020-11-11 18:38:01 UTC by Silverlan

Updated 2022-08-30 14:51:09 UTC by Silverlan