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.BaseBhvComponent: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 peformance 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_physcannon
    - 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
- ents.Entity:GetUri, :GetMemberValue, :SetMemberValue, :DebugPrintComponentProperties
- ents.EntityComponent:Log, :GetUri, :GetMemberUri, :GetDynamicMemberIndices, :GetStaticMemberCount, :GetMemberIndices
- ents.ModelComponent:ReloadRenderBufferList, :IsDepthPrepassEnabled, :SetDepthPrepassEnabled, :IsAutoLodEnabled, :SetAutoLodEnabled
- ents.BaseEnvLightSpotComponent:GetBlendFractionProperty, :GetOuterConeAngleProperty, :GetConeStartOffsetProperty
- ents.CameraComponent:GetFocalDistance, :SetFocalDistance, :GetFocalDistanceProperty
- ents.PanimaComponent:DebugPrint, :GetCurrentTime, :SetCurrentTime, :GetCurrentTimeFraction, :SetCurrentTimeFraction, :ReloadAnimation
- ents.LightComponent:SetLightIntensityType, :SetLightIntensity
- ents.ReflectionComponent:GetLocationIdentifier, :GenerateFromEquirectangularImage
- ents.LightComponent:IsBaked, :SetBaked
- ents.add_component_creation_listener
- ents.find_installed_custom_components, ents.get_registered_entity_types
- ents.RenderComponent:IsInPvs
- ents.TransformComponent:Teleport
- ents.WorldComponent:RebuildRenderQueues
- ents.ModelComponent:UpdateRenderMeshes
- file.is_empty
- debug.render_mesh
- 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
- geometry.calc_rect_circle_touching_position
- gui.create_render_target, gui.create_color_image, gui.register_default_skin, gui.set_focus_enabled, gui.is_focus_enabled
- game.DrawSceneInfo:AddSubPass
- gui.Base:CheckPosInBounds
- gui.DropDownMenu:FindOptionIndex
- gui.Element:GetRelativePos, :SetLocalAlpha, :GetLocalAlpha, :IsRemovalScheduled, :ResetRotation, :GetRotationMatrix, :SetRotation, :SetStencilEnabled, :IsStencilEnabled, :SetLocalRenderTransform, :ClearLocalRenderTransform, :DrawToTexture, :GetLocalRenderTransform, :AnchorWithMargin, :ClearFocus, :SetParentAndUpdateWindow
- gui.Text:GetTextHeight, :UpdateSubLines, :SetMaxLineCount, :GetMaxLineCount
- gui.Shape:ClearBuffer, :SetShape, :GetBuffer, :SetBuffer, :GetVertexCount
- gui.WIContextMenu:IsPopulated
- gui.ProgressBar:SetLabelVisible
- 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
    - locale.localize
    - math.snap_to_gridf, math.get_power_of_2_values, math.calc_bezier_point
    - math.Mat4.create_reflection
    - math.Quaternion:MirrorAxis
    - math.Transform:ToPlane, math.Transform:Reflect
    - math.Vector2:GetAngle, math.Vector:Reflect
    - Model:GetExtensionData, :GetSubMesh, :FindSubMeshId
    - Model.Mesh.Sub:GetTriangle, :GetIndexType, :SetIndexType, :AddIndex, :SetIndex, :GetIndex, :HasUvSet, :GetExtensionData, :ReverseIndices, IndexType and MaxIndex enums
    - Renamed Model.Mesh.Sub:GetTriangles to :GetIndices
    - panima.Animation.load, :RemoveChannel
    - panima.Channel:Resize, :GetSize, :SetTime, :SetValue, :ClearValueExpression, :GetValueExpression, :RemoveValue, :FindInterpolationIndices, :FindIndex, :RemoveValueRange, :AddValueRange, :SortValues, :GetInterpolatedValue, :FindIndexRangeInTimeRange
    - prosper.blur_texture
    - prosper.RenderTarget:GetColorAttachmentTexture, :GetDepthStencilAttachmentTexture
    - prosper.Window:GetMonitorBounds, :Maximize, :IsMaximized
    - shader.BaseTexturedLit3D:IsDepthPrepassEnabled, :SetDepthPrepassEnabled
    - table.is_empty, table.table_to_map
    - util.File:GetSize, util.import_assets, util.generate_hair_file, util.generate_hair_data
    - 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
    - input.is_ctrl_key_down, is_alt_key_down, is_shift_key_down
    - udm.Array:ToTable, udm.PropertyWrapper:ToTable, :ToAscii
    - udm.Property:AddValueRange, :RemoveValueRange, :SetValueType, :RemoveValue, :Copy
    - Material:SetLoaded
- Added classes/enums:
    - util.ThreadPool
    - util.ZipFile
    - gui.DrawToTextureInfo
    - asset.AssetLoadFlags
    - 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 cengine 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:GetReferenceId**
- 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