

Fixed

- (Flowgraph) Dragging an edge to alter its spline results in the link being unselectable.
- (Flowgraph) Don't copy unremoveable nodes.
- (Flowgraph) Adding a new module port causes all nodes with assigned entities to blank out with the red "Choose Entity" error.
- (GameToken) Graph token pre-selection in game token dialog.
- (GameToken) Crash: Create tokens map also in game mode, for graph tokens.
- (Viewport) Invalid view to world position.
- (Shaders) Broken SilPOM.
- (Shader) ToneMapping.
- (Render) Fixed cubemap glossiness issue.
- (Particles) Fixed GPU particles rendering (commitdevicestates was missing for drawinstancedindirect). Generalized single flush validate.
- (Particles) Temporary disabled instanced vertex rendering
- (Particles) Missing triple buffering changes.
- (Particles) Rendering - D3D10 leftovers.
- (GeomCache) Missing SDKs.
- (GeomCache) Missing TString.
- (GeomCache) Fix GeomCache motion blur.
- (CryDesigner) Fixed a bug of connected/grow selection tool.
- (CryDesigner) Made switch of modes by pressing ESC become expected.
- (CryDesigner) Made designer objects saved in the middle of creation.
- (CryDesigner) Got the entire size to be displayed when extruding a face or creating primitives shapes (CE-2681).
- (CryDesigner) Fixed a crash bug caused when selecting a cloned object after cloning.
- (CryDesigner) Fixed a Material bugs of CryDesigner so that replaceme texture will be assigned when the material of the designer object isn't invalid.
- (CryDesigner) Fixed a crash bug caused when selecting a designer object. It caused by changing an edit tool to an another edit tool in a function in the current edit tool (CE-2591).
- (Trackview) Fix crashes with nodes without position/scale/rotation tracks.
- (Trackview) Prevent keys being created in recording mode when changing sequences.
- (Trackview) Fix not being able to move object in base position if it is active in TV without a position track.
- (Trackview) Remove TrackGizmo when entity gets deleted.
- (Trackview) Fix TrackView not displaying missing entities in red after delete.
- (Trackview) When switching to sequence cam viewport is not immediately updated.
- (Trackview) Fix sequences not being loaded properly when imported from a layer.
- (Trackview) Fix not being able to transform entities when adding them to TrackView without reselecting them (CE-2572).
- (Trackview) Fix sequence shown as expanded when it's not.
- (Trackview) Fix sequence not being expanded by default.
- (Memreplay) Invalid token in label xml.
- (Game) Crash: Don't call hud events during shutdown.
- (Game) Optimization turned off.
- (RC) ImageCompiler: fixed bug that created temporary files in the root folder.
- (RC) Fixed 32-bit ImageCompiler (it was failing to compute gamma table properly).
- (RC) Image extension helper flag clash.
- (RC) AlembicCompiler compilation.
- (3DEngine) Debug info rendering not working.
- (Flash) Intro movies use native resolution.
- (Flash) Use native resolution.
- (MMRM) Merging of FPE fixes. FPE fixes. Fixed FPEs in merged meshes.
- (MMRM) Fix for missing rotation when align to terrain is active.
- (MMRM) Various bugfixes - fixed bug when rendermesh is missing when too many instances are placed - fixed scale and orientation for unsimulated meshes - fixed wrong material id assignment.
- (MMRM) Fixed tangents broken after last optimization.
- (MMRM) Compile fix for debug builds.
- (MMRM) Fix for stack corruption on 32bit (and probably also other platforms).
- (MMRM) Fix for keeping deformable nodes around when assigning new statobj's.
- (MMRM) Proper fix for bounding volume approx.
- (MMRM) Disable SSE on PC.
- (MMRM) Fix for missing grass.
- (Editor) LayerStreaming: Mirror game code hide logic in editor (will hide/unhide children).
- (Game) Fix player receiving equipment-pack without "NoWeapon" (CE-1290).
- (Game) Menu was in wrong state after quickload (CE-2240).
- (Game) rotation for camera using helper was incorrect in FirstPerson camera for vehicles.
- (Game) Screenfader draws texture upside down (CE-1264).
- (Game) Player health showing wrong values in MP/SP (CE-1344).
- (Game) Linked entities can't be EntityID'd in FG (CE-981).
- (Game) Fixed issue with player health not regenerating in SP (CE-772).
- (Game) AI actors were not getting equipped in multiplayer games (CE-2345).
- (Game) Screen resolution not being remembered (CE-2274).
- (Game) Unified screen resolution between UISettings and ScreenResolution.
- (Game) Fire OnGraphicChanged when engine is started and read the profile attribute.
- (Game) "Use"-Action doesn't work on first time triggered in the level (CE-2364).
- (Game) Re-synced hit types across all files.
- (Game) Players were no longer exposed to the AI system after they respawned on a dedicated server (CE-2355).
- (Game) Removed overriding scopemask from PlayMannequinFragment node.
- (Game) Vehicles didn't re-queue looping idle actions correctly (CE-2470).
- (Sandbox) Restored "Align to Grid" functionality through menu bar so hotkeys can be assigned to it (CE-2116).

- (Sandbox) SplineObject: Fix Extended gizmo, fix cursor.
- (Sandbox) Fixed crash when closing error report pane after pressing Esc (CE-2373).
- (Sandbox) Spline Objects: Delete control points by Delete key (CE-2501).
- (Sandbox) Fix editor crash due to wrong coordinates (CE-2518).
- (Sandbox) Fix editor crash when "NoRecentDocsHistory" policy is set to 1.
- (Sandbox) Removed "Brush" menu because all sub-menus under "Brush" are obsolete (CE-592).
- (Sandbox) Crash while creating new map after resetting TOD values (CColorGradientCtrl::OnPaint) (CE-2432).
- (Scaleform) Don't handle change focus event during shutdown, shared resources might be gone already.
- (Action) Added missing ICryMannequinTagDefs interface to project filter.
- (Flowgraph) Crash when editing graph tokens or flowgraph modules.
- (EntitySystem) Area wasn't clearing its cached area data when starting the game from inside the editor (CE-1291).
- (CryDesigner) Fixed a bug so that CastShadows and RenderFlags of Designer object can be saved and loaded well (CE-2507).
- (CryDesigner) Fixed a crash caused when selecting a designer object. It caused by changing an edit tool to an another edit tool in a function in the current edit tool.
- (CryDesigner) Made all designer objects shown in Exclusive edit mode.
- (CryDesigner) Enabled to export designer objects to .obj files.
- (CryDesigner) Restored Extrusion Tool.
- (CryDesigner) Fixed incorrect undo on Group/Prefab. Inverted the CObjectManager::DeleteObject(CBaseObject *obj) method into the original one.
- (CryDesigner) Fixed a bug so that CastShadows and RenderFlags of Designer object can be saved and loaded well (CE-2507).
- (CryDesigner) Fixed a crash error happening when Object Creat tool cancelled after undoing in Designer Creation Mode (CE-2383).
- (CryDesigner) Improved the Grid Snapping in creating primitive shapes and extrusion functions (CE-202 + CE-2472).
- (CryDesigner) Got the entire size to be displayed when extruding a face or creating primitives shapes (CE-2681).
- (CryDesigner) Fixed a bug about not loading designer object when the level have prefabs which have designer objects (CE-2687).
- (CryDesigner) Fixed a bug about designer objects disappearing when making a prefab object out of designer objects.
- (CryDesigner) Fixed a crash bug caused when selecting a cloned object after cloning.
- (CryDesigner) Fixed a Material bugs of CryDesigner so that replaceme texture will be assigned when the material of the designer object isn't invalid.
- (CryDesigner) Improved a way of the edit tool switch.
- (CryDesigner) Made switch of modes by pressing ESC become expected.
- (CryDesigner) Made designer objects saved in the middle of creation.
- (CryDesigner) Fixed a crash bug caused when selecting a designer object. It caused by changing an edit tool to an another edit tool in a function in the current edit tool.
- (CryDesigner) Enabled to export Designer objects to .obj file. Enabled to draw a debug outline of a designer object as it is selected. Made a pivot of the selected designer object with transformation be center when PivotToCenter button pressed (CE-2403).
- (CryDesigner) Fixed a bug of Connected/Grow selection tool.
- (CryDesigner) Fixed a crash bug happening when loading a level (CE-2226).
- (CryMannequin) Fixed several mannequin issues.
- (Particles) Texture Tiling: Fixed erroneous handling of FirstFrame with AnimFrames and VariantCount.
- (Particles) Focus Axis & Azimuth: Randomness is now properly per-emitter.
- (Particles) Facing=CameraX particles didn't rotate correctly. Stretch and TailLength are now exclusive options. Facing=Water particles now set static bounds using water environment.
- (Particles) Connected: Fixed ConnectToOrigin behavior, again properly holds particles at negative age. Fixed TextureMapping=PerParticle particle assignment for multiple emission streams.
- (Particles) Fixed emission count calculation, now based on current rather than max lifetime params.
- (GameObject) RefCount underflow if game object got created with ChangeExtension + eCE_Activate.
- (Profiling) Fixed debugdraw 17, removed deprecated debugdraw 18.
- (Assert) Removed 100% repro assert in HUDEventDispatcher.
- (ScriptSystem) script debugger was omitting a lot of variables since metatables got enabled to preserve memory (CE-2439).
- (Vehicle) MH60_blackhawk: Cleared HelicopterTree parameter from script to prevent helicopter from activating by itself. Users wishing to use AI with helicopter should select HelicopterTree as ModularBehaviorTree.
- (Shaders) Fixed GeometryBeam shader not rendering in launcher (CE-1994).
- (Render) Fixed r_NoDrawNear not working (CE-2312).
- (Render) Added workaround to fix ocean fog flickering on horizon (CE-1864).
- (Render) Fixed Vegetation color issue.
- (System) initialize render CVars before loading configs.
- (AISystem) Fixed warning of navigation volume not being registered in the system.
- (Physics) Fpe in sphere buoyancy.
- (Flowgraph) Crash when trying to enable/disable debugging on a group/folder instead of a flowgraph
- (Flowgraph) Gap between port text and port arrow leads to moving the node instead of creating an edge. Re-added play button to recover from breakpoints without pressing F5. Output port breakpoint bigger than input port breakpoint.
- (Flowgraph) Crash when no flownode exist.
- (Assets) Fixed incorrect texture path on Claymore.
- (Scripts) Fixed issue with HitDeathReactions on startup. Hit Types need to be separated (CE-1710).
- (Scripts) Fixed errors showing up from BodyDamage scripts (CE-1710).
- (Scripts) Switch entity button orientation does not match switch base orientation at some angles (CE-2358).
- (UIActions) Fixed positioning of image in gamepadlayout menu.
- Make sure articulated entity's RemoveGeometry can remove full joints.
- Box-cylinder un-projection issue.
- Fixed tangent frame interpolation (CE-1892).
- Fixed e_DebugDraw 10 to correctly debug draw simple lines/triangles for skinned attachments.
- Fix assert in debug due to some object not having "Pos".
- Saved GUID flag in navigation file to fix data compatibility issue.
- Correctly initialize CStatObj::m_idmatBreakable.
- Potential fpe in characters.
- Fix for crash in GlobalAnimationHeaderAIM::ProcessDirectionalPoses when passing in look poses.
- Fixed an incorrect comparison with shader texture resources (comparing name to itself).
- Fix for water areas on dynamic entities.
- Fix for a race condition shutting down the netlog.

- Fixed the issue that vegetation with bending could mess up when set UseTerrainColor option on (CE-1160).
- Fix for RenderProxy stomping custom light materials.
- Fixed sw-skinning in Character Editor.
- Added CDeviceManager.DrawInstanced, restoring particle vertex instancing.
- Fixed tangent frame interpolation (CE-1892).
- Fixed typo in hdr-extension string.
- Fixed and added new Python interfaces.
- Fix FBXExporter compilation.
- Adapted some functions to the new vertex-format.
- Preventing having random data in AABB tree nodes saved to files (it didn't crash/break, but complicated file comparisons).
- MaxCryExport crashes when material contains empty sub-materials (CE-2574).
- Fixed a problem with iterator erase on std::map in CryMovie.
- Fixes for having more than 256 bones.
- Make sure matrices properly set when instancing used and no SSE.
- Make sure required RT flag gets enabled / make sure matrices properly set when instancing used and no SSE.
- Fixes for recounting of remapped bone indices. Fix for stack overflow in the editor.
- Prevent crash when MSAA is enabled: tiled deferred shading has to be disabled until it has proper MSAA support.
- Fix for deformables not updating positions or UV's during mesh updates from physics.
- Allow transparent objects to pick nearest probe even when shader uses ForceZPass (required for eyes).
- Fixed propagation of REQUIRES_NEAREST_CUBEMAP flag from skin attachments.
- Potential fix for rare crash on loading (buffers still pending in invalidation list that got released previously).
- Don't use diffuse alpha for alpha blending if glow mask is enabled.
- Small fix about redundantly overwriting UV positions.
- Fix out of bounds access in ComputeVertices sync variables. Hide particle emitter when it is assigned to a hidden render proxy. Adding missing initialization for output write mask in CCullRenderer::ReprojectHWDepthBufferAfterMerge. The REQUIRES_NEAREST_CUBEMAP flag is not set when an effect is hidden and then activated via a layer. Set it in the constructor since all effects require that flag anyway. Fix crash when moving portal. Fix missing RendElement type strings.
- Lock correct particle video memory buffer in RT_EndFrame. Change was modified during integration to remove the hardcoded number. Fixed crash at shutdown (depends on job refactoring).
- Fix for chr streaming, remap bones indices in complete callback.
- Fix broken FrameProfiler macro, remove duplicate references.
- Fixed e_ViewDistRatioLights not being affected by cvar/sys spec changes in Release.
- Fixed CS resource leaks.
- Fixed broken float2int code on several platforms.
- Stack corruption due to alloca running out of space (256k for job system workers).
- Don't skip individual faces for shadow casting point lights if there are no shadow casters.
- Integrate default geom cache engine asset.
- Fixed DXT1 NormalCompressionError format to be 32bit float.
- Fixed Dedicated Server startup hang.
- Area grid was not resizing its memory when number of areas was changing. Bug fix and clean up of area grid. Got rid of CAreaBitField.
- CRASH: CEntitySystem::Reset.
- Fix for massive overhead with recompiling the lua script for every placed instance now recompiles once and updates each instance with the new one.
- Fixed hair to work with fog volumes (could get random colors before since fog contribution was not bound for opaque hair).
- Fixed DOF being resolution dependent.
- Adjustable DOF depth dilation to avoid halos on focused objects.
- Fixed proxy size for probe parallax correction.
- Fixed error in water volume shader.
- Wind: fixed vertex bending being in applied in wrong space.
- Fixed flickering in PC launcher by reverting occlusion depth downsampling back to original code for now.
- Enable motion blur pass on hair.
- Fixed probe sorting for tiled shading.
- Fixed FPE when GlossFromNormals is accidentally used on an invalid normal map with zero length normals.
- R_ParticlesInstanceVertices default value set to 1. This fixes compilation problems in Release mode.
- Fix sunshaft integration issues.
- Fix r_motionBlurShutterSpeed.
- Enabled rendermesh updatedate reclamation.
- Fix for 32bit alignment crashes on PC.
- Fix for constant buffer deallocation crash on shutdown.
- Speculative fix for MSAA resolve when using PBR.
- Support for writing SSS amount to gbuffer.
- Preserve 1.0 during linearization pass.
- Use right cubemap format.
- Small cleanup for instancing (%_RT_INSTANCING_CONST got removed in SDK).
- Fix occlusion using wrong reprojection matrix.
- Fixed compilation due changed to VMath.
- Fix wrong viewport size causing short flicker in menu.
- Fixed particle Flickering (not all places used the new triple buffering correctly).
- Eye gets diffuse ambient from cubemap.
- Fix for glinting eye highlights.
- Bringing over DOF fixes.
- Fixed strong DOF bleeding and energy loss in out of focus areas.
- Fixed runtime skel-extensions in combination with ReamapTablePairs.
- Fixed Crash with AsyncDips. Fixed Compilation with Memreplay. Fixed copy and paste error. Other smaller improvements and optimizations. Added Frameprofiler support for Device/Context Wrapper. Optimized Cache DeviceWrapper Object.
- Added missing TextureCube token.
- Avoid releasing static shadow map twice.
- Fixed deferred decals.
- Disabled behavior that decals become deferred at a certain orientation.

- Fix shadow gen for omni lights: Only clear relevant section of shadow pool when rendering individual faces.
- Fixed SM5 flag being set before OnCreateDevice logic (fixes water tessellation).
- Fix nearest shadow.
- Fix for broken grass.
- Some code cleanups regarding size_t usage.
- Removed some unused structures.
- Fix rain amount never increasing.
- Fix reflection distance fade. Support for rendering rain puddles directly to the GBuffer so that they get consistent shading. Added flownode to update rain entity properties
- Fixed samplers getting overwritten when a DX11 texture object with a slot higher than 15 is used.
- Enforced <type> semantic for all DX11 texture objects and added parser error.
- Specify swapchain flag for Full RGB content.
- Fixed RT stack issue when SwitchToNativeResolutionBuffer is called two times in a row (happens when level is loaded from UI).
- Gracefully handle stream preps that are missing some parts.
- Speculative fix for auto test crash, aborting a stream task on a texture that's no longer streaming.
- Various tiled shading fixes.
- Reduced artifacts caused by gbuffer filtering.
- Fixed compile error in Vegetation shader BLENDLAYER.
- Fixed merge bug in eye overlay.
- Fixed crash when using BC4 (pitch and blocksize were wrong).
- Use point sampling when downscaling depth for coverage buffer.
- Don't skip individual faces for shadow casting point lights if there are no shadow casters.
- Disabled async dips by default.
- Fix for accessing GetStats for Telemetry unlocked.
- Fixed access violation on SetSkinningDataVegetation.
- CRASH: CDeviceBufferManager::UpdateBuffer.
- CRASH: SRecursiveSpinLock::Lock.
- CRASH: CConstantBuffer::BeginWrite.

Added

- (CryDesigner) Added validation code for the designer object.
- (RC) Added new RC preset for LUTs.
- (RC) Added OpacityMap preset.
- CryTIF plugin for xNormals v3.18.2 (CE-864).
- CryMesh plugin for xNormals v3.18.2 (CE-864).
- (AISystem) new sample Behavior Tree SDK_Grunt_06.
- (Sandbox) PRT libs: Fix warning with link. Add libs for Visual Studio 2012.
- (Assets) Added generator_b asset.
- (Assets) Added wall_dirt decal assets.
- (Assets) Added various decal assets.
- (Assets) Added silhouette function to Generator in Forest Cave (CE-666).

Deleted

- (Configs) Removed last gen config support.
- Remove procedural creation code.
- Remove unused files.
- Delete unused lightmap compiler code.
- Deleted wrong LG renderer files.
- (Game) Removed obsolete and unreferenced UIObjective nodes, MissionStatelister replaces the functionality (CE-2263).
- (Renderer) Removed e_debugdraw 8 and 9 (CE-34).
- (Scripts) Removed obsolete Checkpoint scripts (CE-2589).
- (AISystem) removed deprecated SDK Behavior Trees (CE-2567).
- (3DEngine) Removed deprecated debugdraw 11 + 12 + old OCCLUSION_PROXY related code.
- Removed unused render flag ERF_USE_TERRAIN_COLOR.

Optimization

- (Build) Converted all remaining projects to VS11 toolset.
- (Build) Removed obsolete VS10 .vcproj files.
- (MMRM) 50% speed increase on x86/x64 platforms.
- (MMRM) Another round of optimizations (removed register dependency on cmplt_ps).
- (MMRM) Optimized deformables.
- (MMRM) Disabled tessellation support, SSE invariants defined as static const. Saved 5-10% of the execution time by removing CFloatToInt constructs on x86.
- (3DEngine) Avoid touching every RNTmpData to find out if it is has timed out - Saves ~1ms on MT. 3DEngine - Fix racecondition when creating RNTmpData.
- (Particles) Fixed crash involving dead RenderObject access. Don't run PrepareRenderObjects as job, move RenderObject flag computations into ComputeVertices jobs (CE-2280).
- (Particles) Added PosInt8 and TSmall support for uint8s that contain values 1..256; simplifies TextureTiling params. Moved or refactored CParticle collision members to preserve struct size.
- (Common) DynArray empty base class refactored out, to make class size again equal pointer size. Removed useless DynArrayRef intermediate class, replaced with #define for legacy code. Improved asserts for signed size_types.
- (CryDesigner) Optimized loading CryDesigner Object data. It can speed up loading a level consisting of only designer objects by around 80% at least.

- (Assets) Speedboat: improved proxy (CE-2295).
- Changed cloth unprojection to soft.
- Switched to BC4 (3Dcp) for height/displacement maps.
- Skip DOF passes when DOF amount is 0.
- Native backbuffer improvements.
- Basic eye shader updates.
- Optimized and unified depth downsampling.
- Devbuffer: removed handle resolve approach for most operations (saves 1ms).
- Packing velocity output into g-buffer.
- Don't process water volume caustics when they are not enabled on any entity.
- Disable anisotropic filtering on alpha tested geometry (can be expensive for vegetation planes, especially since HiZ is not working there).
- Start Prepare Occlusion job earlier (moved from beginning 3DEngine to before SyncAnimation, brings ~2ms more time for its computation).
- Numerous shadow rendering optimization and refactoring.
- Water updates: Fixed precision issues on water volume reflections. Filter out NaNs in water volume reflections after switch to floating point format. Extended range of EnvCubeScale parameter for water volumes. Quick ocean polishing pass (reduce gloss in distance, more physically based Fresnel computation, improved SSS).
- Using exact sRGB conversion for colors in editor instead of gamma 2.2 approximation.
- When turning a boolean game token value into a string output either "true" or "false" rather than "0" or "1". Fixes warnings while reading boolean game tokens from XML files specified as "true" and "false" (previously complained that they should be written as "1" or "0" respectively).
- Loading and streaming linear tiled textures (uses move engine to optimally tile).
- Static textures now created in pool (built on defrag allocator).
- Streaming occurs in place, with ping-pong through ring buffer/move engine for final tiling.
- Rain rendering bottom part.
- Numerous worker/job system optimizations and refactoring.
- Raised priority of streaming update job, to prevent stalls when syncing with it.
- Precomputed dev texture sizes for each mip chain to avoid expensive recomputation at runtime.
- Added specialized StreamGetFilesToRead for splitted files.
- Deferred D3D texture creation for streamed textures to worker thread.
- Added table of precomputed texture layouts.
- Changed lens optics format to mini float (10F).
- Split texture prep streaming update away from standard stream state update - during precache phase, prep update runs on RLT rather than RT. Reduced lock requirements for prep update to stream state acquisition/release to minimize RT/RLT contention.
- Removed linear depth from gbuffer and introduced a linearization pass.
- Mini async dips.
- Prevent sw-skinned attachments from spending 12 ms copying data around.
- Reduced contention with devbuffer.
- Made FC_TARGET flag application for m_nCommitFlags implicit.
- Devbuffer: transient buffers now use first-fit for performance.
- Removed remapping table copying if char global joints enabled.
- Reduced cachemisses in rendermesh stream accesses.
- Added samplers to device states.
- Reduced footprint of CREMeshImpl::mfPrepare by using incrementing mask.
- Devbuffer: moved constants to direct vmem access (can be disabled via BUFFER_ENABLE_DIRECT_ACCESS).
- Fixes and optimizations to DevBuffer.
- Reduced anisotropy to 4 except in VeryHighSpec.
- Execute DOF only when it is really enabled.
- Switched to CPU friendly batching.
- Splitted JobStates for SkinningData/VertexAnimation to prevent sync point. Start ShadowMapRenderJobs only after RenderBufferedMeshes, as shadow frustums for dynamic lights are generated in RenderBufferedMeshes.
- Tweaked Job Priorities to ensure no sync time even if two workers are taken by other threads.
- Re-ordered submitting of shadow render jobs, to allow them to run mostly earlier than the general pass ones, non job ones are still executed in their old place.
- Changed RT Sync points to sync only to shadows first, and after the shadow pass for the other jobs (allows jobs 2-4ms more time to run).
- Splitted CoverageBuffer Reprojection Part into multiple SubParts to allow more parallelism (Omitted from Integration). CParticleEmitter::Render is now invoked as a job if possible (not for the speed of the function itself, but this allows removing sync points from the MT). Merged Cull/Compute Vertices into a single job as Particle Merging is now gone (can likely be even more optimized).
- Start WaterVolume::Render direct from jobs, without going over the MT.
- Triple Buffer Particle Vertex/Index buffer for direct video memory access to prevent possible stalls waiting for the GPU.
- Avoid querying cvar from pConsole for every shadow casting light.
- Disabled ambient occlusion passes when tiled deferred shading is enabled, as they won't have any effect (SSDO is fully applied in tiled shading).
- Use sRGB backbuffer view for native upscaling. Moved native resolution upscaling to PostAA and changed smoothstep function.
- Optimized Area::getminmax by removing it, introducing special box holder structure containing all area's bbs in a contiguous array to reduce cache misses. Optimized UpdateLightSources by replacing qsort with a std::sort (comparison predicate gets inlined, no aliasing issues).

Refactored

- (Game) Removed last gen code.
- (MMRM) Removed CFloatToInt and CIntToFloat constructs (x86 pays for the conversion via double).
- (MMRM) Removed __fres usage.
- (Trackview) Remove Animation.h from precompiled header as it includes TrackView objects.
- (Trackview) Remove some CString usage from TrackView code.
- (Trackview) Remove most of TrackView specific code from EntityObject.
- (CryDesigner) Removed duplicated flags of the designer such as eOpType_Copy and eDesignerMode_ExcludeFrontFace so that programming designers will be more intuitive.
- (CryDesigner) Changed a way of saving and loading DesignerObjects to the binary format.
- (CryDesigner) Made it unnecessary to deal with backfaces in manipulating geometries in terms of a programmer.
- (Flowgraph) Moved "Input" nodes back into Input folder, instead of Debug folder. Nodes still marked as "Debug" category. Substitutions will handle existing flowgraphs with these nodes (CE-1556).

- Removed the brush argument setting to each tool.
- Rain improvements.
- Move CSimpleEntity etc. to their own files.
- Removed ICharacterModel and ICharacterModelSkeleton from SDK3.6.
- Move Python Libs to Editor/Python/Lib.
- Moved stridedptr into its own file.
- Removed separate render target for POM self-shadowing (if feature is desired, the occlusion term needs to be encoded in the gbuffer).
- Adjusted skin shader for PBR and removed obsolete options.
- Slightly simplified eye shader and added a simple light scattering approximation.
- Grain gets applied after PostAA.
- Cleanup: Removed SSII leftovers.
- Removed shader options that are obsolete with PBR (explicit env map, gloss_diffusealpha, aniso_specular, blendlayer_gloss, options to derive spec values and gloss from other maps).
- Removed specular level from material editor.
- Remove unnecessary includes/dependencies to rest of cryengine.
- Math cleanup - Replaced specialized clamp-functions by clamp_tpl().
- Cleanup & generalize python message/error printing by implementing stdout/stderr sink in BoostPythonHelpers and listening to that with observers.
- Refactored rainfall to render directly to HDR buffer and allow a single layer for lower costs.
- Motion Blur & PostAA refactoring and optimization.
- Fixed e_cameraFreeze. Added a cvar "r_reprojectonlystaticobjects" to split the zpass that only static objects are reprojected. Implemented direct VMEM access to read the downscaled Z-Buffer to remove a 1 frame latency (also saves ~0.6 ms on RT by moving some work into a job). Cleaned up FX_ProcessZPass to be more compact. Fix occlusion buffer reprojection jobs merge. Parallelize occlusion buffer merge. Prevent crash in CullThread when no frame is rendered. Occlusion buffer reprojection. Fix FPE in occlusion. Fix FPE, using _mm_load1_ps on static values confuses the compiler. Fix invalid occlusion buffer when switching camera.
- Refactor shader variables for clarity to have s_ for statics instead of m_.
- Various hair related engine updates.
- Packing velocity output into g-buffer (use r_MotionBlurGBufferVelocity to enable/disable) humanskin shader.
- Cubemap generation does not use MIRRORCULL any longer.
- Multi-layer eye shader prototype (needs cleaning + flag to enable old shader mode + proper iris shading).
- Eye overlay refactor.
- Switched to 3 channel gbuffer layout and fixed some issues with it.
- Switched probe format to BC6.
- Read gloss from attached normal map alpha.
- Introduce specific wrapper classed for Device, Context, PerformanceDevice and PerformanceContext - Those are simple inline wrapper which call the underlying device etc pointers - To support memreplay and logging, before and after each call, a DeviceHook is checked - Those can be un/registered during runtime to allow tracking device interactions.
- To make the wrapper more useable, all client code was refactored to use only GetDevice(), GetContext() etc functions - Direct usage of the device or the wrapper should not be done. Those GetDevice() etc calls are also instrumented with the required synchronization for ASYNC Dips.
- Use exclusively environment probe entities for cubemaps.
- Small editor cleanups regarding probes and ambient lights.
- Removed r_deferredShadingEnergyConservation CVar (always enabled now).
- Removed obsolete FEATURE_DEFERRED_SHADING, %_RT_DEFERRED_SHADING and r_FullDeferredShading.
- Removed FresnelBias and FresnelScale parameters from PBR based shaders.
- Radical cleanup of mathlib in SDK3.6.
- Removed obsolete FEATURE_PBR define.
- Fixing eye shader bugs.
- Eye overlay has its own render list for proper ordering.
- Devbuffer: made constant buffer bank sizes and thresholds for memory reclamation configurable. Optimized mfSetParams.
- Render mesh create skinned parts directly in vram.
- Cleanup: Removed old deferred list based shader activation code.
- Cleanup: Removed not used System event listener from Material Manager.

Tweak

- (Flowgraph) Highlight selected node edges.
- (Flowgraph) Improved Debug nodes visibility.
- (CryDesigner) Improved a way of the edit tool switch.
- (CryDesigner) (CE-202,2472) Improved the Grid Snapping in creating primitive shapes and extrusion functions.
- (CryDesigner) Restored Clone/Array tools.
- (CryDesigner) Optimized loading CryDesigner Object data. It can speed up loading a level consisting of only designer objects by around 80%.
- (Game) Changed Boid parameter from "Invincible" to "Invulnerable" for consistency.
- (Render) Removed obsolete level shader cache miss left overs.
- (RC) Preventing compiler warnings.
- (GameToken) Don't show Game Tokens root item if no game tokens exist.
- (Flowgraph) Added description text for Interpol nodes (CE-1836).
- (Flowgraph) Show debug only nodes in different color.
- (AISystem) Updated default MBT for Human_x (also tidied random tabbed spacing). Updated Characters EntityArchetypes to use 05 MBT.
- (Vehicles) Abrams: Tweaked damages. Uniformed hull -> body. Uniformed helper names. Removed unused elements. Lowered turret speeds. Grouped params for better readability. MH60_Blackhawk: Uniformed helper names. Grouped params for better readability. Speedboat: Uniformed chassis -> body. Uniformed helper names.
- (Game) Modified Gamestate node to work in UIActions (CE-2286).
- (Scripts) Updated helper information in Rifle.xml.
- (Scripts) Set secondary_damage to enable in SP and disable in MP to compensate for player lower health in MP.
- (Assert) Fixed an assert from running off the end of the bone array.
- (CryDesigner) Made adjusting the height in primitive creation tools and extrude tool more sensitive to a mouse movement.
- (CryDesigner) Changed the exclusive mode so that only one object is visible.
- (CryDesigner) Renamed "Enable Back Faces" to "Display Back Faces" and made backfaces not exported always.

- (CryDesigner) Renamed each item name of designer menu button enum list.
- (CryDesigner) Updated the Texture mapping tool so that it use not only the face selection tool and other related selection tools.
- (CryDesigner) Improved multiple edges' erase function.
- (CryDesigner) Added validation code for the designer object.
- (Assets) Removed "Optional" text in Forest objectives.
- (UIActions) Updated SP_Objectives to handle SecondaryMission text.
- Make new file dialog more intuitive.
- Changed it so bound keys with modifiers (ie, alt+f12) have an underscore between the modifier and the key. This way you can bind them from the console.
- Add alembic compilation to build.
- Increased minimum probe size.
- Allocate static shadow map at game start.
- Expose static shadow map resolution via CVar.
- Tessellation starts fading out only after some threshold.
- Disabled sharpening by default.
- Switched to SMAA 1TX by default.
- Adjusted job priorities to the RenderThread reordering.
- Added some additional profiling markers. Removed Particles Merging Code as it is and will be kept disabled.
- Teaked poly limit for rasterize job to prevent stalls.
- Disable Particle Geometry Render Jobs on 32Bit systems to prevent increasing the InfoBlock Size.
- Removed bias params from eye shader.
- Remove VF_CHEAT from e_ShadowsStaticMapUpdate, allow disabling e_ShadowsStaticObjectsLod.
- Disabled hardcoded alpha test value for hair.
- Disabled average luminance for auto exposure (parser does not work correctly here).
- Enable r_motionBlurGBufferVelocity.
- Updated r_ParticlesInstanceVertices, default now 0.
- Revived rain splashes for GBuffer technique.
- Removed invalid r_FullDeferredShading CVar.
- Made color chart code more tolerant to slight differences in the frame colors.
- Enabling writing chunk files in the new 0x746 format.
- When turning a Vec3 game token value into a string trim off extra zeros i.e. write "0.5.25,1.234567" instead of "0.000000,5.250000,1.234567". Fixes warnings while reading Vec3 game tokens from XML files specified as "0,0,0" etc (previously complained that it should be written as "0.000000,0.000000,0.000000").
- When turning a float game token value into a string trim off extra zeros i.e. write "7.125" instead of "7.125000". Fixes warnings while reading float game tokens from XML files specified as "0" etc (previously complained that it should be written as "0.000000").
- Also, don't display warnings in pop-up dialog boxes, just write them to the console and log.
- Remove redudant debug output.
- Add <float.h> include to Cry_Math.h. Reason: On Mac/Linux, in some cases where Cry_Math.h was included, the #define of FLT_EPSILON and related were not included yet.
- Do not include <intrin.h> on Mac or Linux, as it is a windows-only header..
- Updated CryAction Headers-project to include ICryMannequinTagDefs. Reason: GameCode export needs this file.
- Do not reference a function before it's declared. MSVC accepts this in templates, but GCC doesn't.
- FBXPlugin now uses same SDK as the rest of the code.
- Added missing file references to projects, so they will be exported.
- Removed persistent size from streaming computation, not required.
- (RC) Updated rc.ini: use average filtering mode where appropriate, updated normal gloss settings, new HDR sky preset.
- Output number of temp registers used when compiling shaders.
- Extended WrappedDX11Buffer to support more features like UAVs.
- Disabled chromatic aberration by default.
- Detail map gloss modifies roughness instead of specular color.
- (Renderer) Increased max tmus.
- Clamp min decal falloff to keep results consistent across the whole parameter range.
- Switched Skybox_highQ preset to BC7 and enabled mips.
- Increased maximum blend layer tiling.
- Added some runtime checks to prevent such issues.
- Disabled constant and height based ambient for PBR.
- Temporarily disabling SSSSS.
- Disabled parameter-not-found warning as it gives too many false positives with DX11 objects (requires a shader system refactoring to fix properly later).
- Disabled GI by default (environment probes should be used instead).
- Improve accuracy of automated static map bounding box generation.
- Re-added blending between cornea normal and sclera details.
- Added more control over cornea normal blending to eye shader.
- Cleaned up eye shader and removed deprecated shader path.
- Eye overlays depth bias not getting larger with depth.
- Using cubemap convolution params.
- Disabled light HDR dynamic power factor.
- Streamed texture array now contains all streamed textures, unloaded or not.
- 2D arrays are now streamable as far as the streamer is concerned, albeit without IO.
- Minor changes to ease debugging "CRASH: CDeviceBufferManager::UpdateBuffer".
- Capped worker threads to three to prevent inter-cluster communication (has to be checked after all optimizations is done, if adding an additional worker thread has more benefit that the cost of inter cluster communication). Increase number of worker threads to 4.

New Feature

- (Flowgraph) Modules support "Any" ports now.
- (Flowgraph) Assign/Unassign entities to multiple node selection.
- (GameToken) Show flowgraph tokens in token selection dialog.

- (CryDesigner) Added a new advanced way of adjusting height when extruding or creating primitives or stairs etc, which the height will be adjusted at the position of the mouse cursor.
- (CryDesigner) Implemented the Pivot Tool.
- (CryDesigner) Implemented the Loop Selection Tool.
- (CryDesigner) Implemented the Border Selection Tool.
- (CryDesigner) Implemented the Ring Selection Tool.
- (CryDesigner) Implemented a separation of created primitive shapes on the ground.
- (CryDesigner) Implemented advanced ways of creating all types of primitive shape.
- (CryDesigner) Added the Rectangle Selection Function.
- (CryDesigner) Added the Grow selection, the All/None Selection and the Connection Selection.
- (CryDesigner) Added snapping to mirror plane in Extrude and Move tool.
- (CryDesigner) Added "Display Dimension Helper" check box.
- (CryDesigner) Added undo/redo for creating primitives.
- (CryDesigner) Restore the Convert tool from objects of brush and entity type.
- (CryDesigner) Restored Clone/Array tool.
- (Trackview) Add TrackView python unit test
- (RC) ImageCompiler: Added 'swizzle' option to allow simple pre-processing of the input image.
- (MMRM) Support for terrain alignment.
- (CryMannequin) New mannequin transition editor.
- (Render) Added CVar: r_WaterGodRaysDistortion to control amount of caustics distortion effect while underwater.
- (Particles) Allow Soft Particle option with all facing modes (CE-2464).
- (Particles) Texture Tiling: Connected particles can now use texture tiling: selected per-stream rather than per-particle. Added TextureTiling. bMirror flag.
- (Particles) New shape params: Aspect, PivotX, PivotY, TextureTiling.FlipChance for texture flipping.
- (Particles) New MaintainDensity and AdjustAlpha options; Connected.TextureFrequency can be negative, reversing texture V mapping. Tweaked mapping of Connected particles with TextureMapping=PerStream to map whole texture to partial streams (starting/stopping).
- (Flowgraph) Added HUD:SilhouetteOutline node for highlighting entities.
- (AISystem) navmesh accessibility is now also re-calculated when continuous update is enabled (CE-782).
- (AISystem) TPS queries for the sample Behavior Trees updated.
- (AISystem) name of the Behavior Tree is now also printed when debug-drawing a specific agent (CE-2620).
- (Physics) Support for clone-based non-uniform scaling.
- (UIActions) Added option to enable/disable filtering of levels/Multiplayer folder in MP host menu.
- Support of 8-weight skinning.
- Added "skip update" option for probes.
- Added box shaped environment probes.
- Added configurable parameters for BRDF model and adjusted glossFromNormals feature to use them.
- Geom cache 32 bit index support (requires vtx_idx to be set to uint32 at compile time).
- Cosine power filter for more accurate specular convolution.
- Add staging depthstencil textures for readback.
- Added support for cubemap arrays and more generic 2D array support.
- Added latest deferred SSSSS.
- Added refactored ambient lights.
- Environment BRDF.
- Manually updating hair shader with latest updates.
- Added translucency updates.
- (Rain) Added diffuse darkening. Prevent dynamic objects from being affected by rain puddles.
- Using DiffuseBRDF for lights.
- Controllable decal alpha falloff (useful for creating nicer fracture transitions).
- Added support for a dedicated opacity map for decals, so that the same diffuse map can be used with various blend masks.
- Added separate diffuse opacity parameter for decal geometry so that it is possible to use the diffuse color of the underlying geometry.
- Renderer/streaming support for composite 2d array textures.
- Bringing over glossy screenspace reflections.
- Bringing over latest SSDO updates and incorporating it better into shading model.
- Improved shading consistency of eyes (also sun shadows work now). Tiled shading support for eye.
- Enable nearest cubemap picking for hair even when it is not alpha-blended.
- New option to disable hardware gamma in Editor to avoid that color correction profiles get overwritten (enabled by default).
- Added option for gloss map filtering strength to RC.
- Improve goto selection behaviour & orbiting (behaves like Maya/Max now).
- Add some more python standard libraries.
- Added additional checks to support prefab filenames, as the prefab name string are in different format depending where they come from.
- Brushes generated procedurally when placing the procedural object in the editor were saved into the layer files when exported to the engine tiles. Added a flag to avoid that.
- Added functionalities to tell the Editor to use the bounding box for selection in the 3D view.
- Running out of physics pools for brushes, removed old brush pool (last gen only) - the PO and prefabs are generating lots of brushes.
- No need to check for Vista prerequisites anymore.
- Triple remapping of Bonelds.
- Shader parser support for RWBuffer.
- Integrated support for depth fix up.
- Shadow cascade debug overlay.
- Added support for blend mask tiling.
- Enable world streaming updates during FMV playback, to fix low res textures after FMV completes. Fixed broken pre-caching when resuming game.
- Shader support for using arbitrary AABB for probe parallax projection.
- Support for filtering of gloss in gbuffer based on normal variance to reduce specular aliasing.
- Added cvar to visualize gbuffer attributes (cvar got renamed, currently only works when postaa is disabled).
- New light attenuation model.
- Bringing over HDR changes. New bloom pass which uses a sum of Gaussians to approximate a more natural falloff. Bloom strength parameter is part of TOD settings. Different tonemapping curve which is less prone to oversaturating darks, adjusted scene key handling. Display average luminance and estimated scene key in HDRDebug.

- Better film grain simulation.
- Static shadow maps: Replace cascade 4 and 5 with 8k x 8k static shadow map that is only updated upon request via flowgraph. Only display static shadow map bounding box when `e_ShadowsStaticMap>2`. Don't allocate static shadow map from pool. Update static shadow map when new assets are streamed in. Reset incremental update state properly on dynamic shadow frustums. Fix flickering of static shadow map. Potential fix for log spam on shutdown.
- Integrated Tiled Shading.
- Clear separation of doubles and reals.
- Implement CopyResource for buffers, CopySubresourceRegion for Tex3D, depthstencil textures.
- Adding runtime prefabs, prefabs manager and some changes to the entity system to keep runtime prefabs 100% compatible with prefabs in SDK 3.6. Editor side changes to convert from/to runtime prefabs not included in this CL. Adding procedural entity, which is a proxy for runtime prefabs group over network. As an entity is easy to play with it with the Editor etc. Right now is not doing anything, the entire prefab manager is missing.
- Added Sandbox functionality to popmenu to right click and convert back and forth between prefab and procedural objects. Now it's very easy and smooth to iterate on randomized objects in the Editor.