PhysX & APEX Programming and Optimization
Purpose of this Presentation

• Understand the story behind the Scene
• Tips on how to create similar effect in a Game engine
• Tips on Optimization
PhysX SDK Workflow

• PhysX Workflow
  – Asynchronous Simulation
  – PhysX SDK natively is multi-threaded

• PhysX-related Game State
  – Position
  – Velocity
  – Normal
  – ……

[Diagram]

Update PhysX State

Start Simulation

Do Other Stuff(Rendering AI etc)

Fetch Results

Update Game State

[gameworks.nvidia.com]
PhysX Fluid/Particle

- From 《P.L.A》 Benchmark
PhysX Particle Creation

• Creation of Particles
  – Directly calculate states(pos, vel) of particles
    • Create All particles upfront
    • Insert/Delete Particles each frame
  – Use NxFluidEmitter to emit particles automatically

• Main Properties
  – Collision radius and collision response
  – Velocity control of particles
  – SPH(fluids) only properties, e.g., viscosity and surfaceTension
Hardware Acceleration

• To increase the number of Particles
  – GPU PhysX particle (NX_FF_HARDWARE)
  – Adopt Instancing
    • UE3 material setting ➔
Share PhysX Particle SDK Sample

UserData Fluid Sample (hardware) www.fraps.com
F1 for help
GpuHeapUsageTotal = 3072, max = 6894 kB
GpuHeapUsageFluid = 1024, max = 4712 kB
GpuHeapUsageUtils = 0, max = 134 kB
GpuMeshCacheUsage = 0, max = 0 kB
Share PhysX Particle

• Max Particles
  – Maximum Number of Particles in a Particles system

• Particle Reserve
  – Old Particles in the FIFO will be deleted
  – Maximum Number of Particles to emit in a frame after the Maximum Particle number is reached
PhysX Cloth

- 《P.L.A》Cloth Sample
PhysX Cloth

• 《P.L.A》Cloth Sample，Explosion Interaction
Deformables

• Cloth & Softbody
• Particles + Constraints + Meshes
  – Cloth: triangle mesh
  – Softbody: Tetrahedral mesh
• Special features
  – Tearing
  – Self-collision
  – Pressure

[Diagram of cloth and softbody particles with constraints and pinned particles]
Deformables Creation Process

1. Export model from Max/Maya
2. Using NxCookingLib to cook cloth/softbody mesh for simulation
3. With cooked Mesh, fill in descriptors with other parameters
4. Last step, call NxScene’s API to create the cloth/softbody object
Cloth Optimization

• FreezeWhenNotRendered(pseudo-code)
  if (Cloth out of view frustrum)
  {
    Cloth->setFlag(NX_CLF_STATIC)
  }
• Cloth stops at the last position.
• Simulation stops to increase performance
Optimization for Tearable Cloth

- ValidBound Setting and Sample
ValidBound Opt Setting

• Cloth Flags
  • \texttt{NX\_CLF\_TEARABLE}
  • \texttt{NX\_CLF\_VALIDBOUNDS}

• \texttt{setValidBounds}(\texttt{const NxBounds3} & validBounds)
  • A World space cube
  • Cloth verts outside this cube will be deleted
  • Note: to remove the rendering triangles as well
APEX Destructible

- 《P.L.A》 Destructible Scene
APEX Destructible

- 《P.L.A》Destructible Scene
APEX vs PhysX Workflow

• PhysX
  – Simulate/fetchResults
  – Update Game State
    • Position, Rotation, Velocity, Normal, etc.

• APEX
  – Load Assets
  – Simulate/fetchResults
  – Update Game State
    • Vertex Buffer, Index Buffer, Bone Buffer, etc
Rigid Skinning Rendering

• Single VertexBuffer
Callback on Fracturing

• Callback function
  – Sample:

    ```
    void physx::apex::NxUserChunkReport::onDamageNotify ( const NxApexDamageEventReportData & damageEvent )
    {
      for each DamageEvent
      {
        Emit Fracturing Particles at the hit Location;
        Playback Fracturing Sound;
      }
    }
    ```

• NxApexDamageEventReportData
  – Destructible Actor
  – HitLocation, HitNormal
  – Fracturing Chunk index
  – ........
Jitter

• Problem:
  – Sometimes a pile of PhysX convexes can bounce around unnaturally for a long time.

• Possible Solution
  – Increase damping
    • Increase angular damping parameter to a value of 10-15
  – Lose energy
    • Increase friction and lower restitution
  – Equalize masses
    • If there is a large mass difference the body with the smaller mass can move quite a lot and keep the whole pile moving.
  – Put them to sleep
    • Increase the sleep energy threshold value will cause destructible chunks to fall asleep more easily.
Authoring Optimization

• Debris Timeout
Authoring Optimization

• Debris Setting with Code
  – setParamBool(*params,"destructibleParameters.flags.DEBRIS_TIMEOUT", true);
  – setParamI32(*params,"destructibleParameters.debrisDepth", 1);
  – setParamF32(*params,"destructibleParameters.debrisLifetimeMin", 0.0f);
  – setParamF32(*params,"destructibleParameters.debrisLifetimeMax", 0.0f);

• Similar for Debris Max Separation
Simulation Optimization

• Avoiding too many chunks on screen
  – setMaxDynamicChunkIslandCount
    • Maximum number of dynamic chunks island in the scene.
  – setMaxChunkCount
    • Maximum number of dynamic chunks in the scene. No less than maxDynamicChunkIslandCount
  – setLODResourceBudget
    • Overall LOD resource setting for an APEX scene(including destructible)
  – setMaxActorCreatesPerFrame
    • Cap the maximum number of Destructible actors in a frame
Hardware Acceleration

- Simulate Rigid body on GPU--GRB
Q & A?