Ragdoll

Description

Ragdoll is physically simulated character body skeleton usually enabled after player death. It is consist of several physical meshes that represents all main body parts (head, forearms, upper arms, legs, etc.). This physical meshes connected using joints, that configured to restrict relative movement of corresponding bodies. Entire configuration of physical mesh hierarchy is performed in 3D studio max, and exported using PhysX plugin. During character physical mesh creation, game engine loads this files and create all necessary meshes in physics scene. Client code can switch from animated character to physically simulated character using appropriate functions.

Associated classes and structures

ActorBone	Structure that describes single physical mesh that correspond single bone of character skeleton.
r3dPhysSkeleton	Physical skeleton used for ragdoll simulation

class r3dPhysSkeleton

Instance of this class is created for every character presented on game level, and can perform switching between physical body simulation and animated mesh. Also it perform synchronizing graphical and physical positions and orientations.

Sources: r3dPhysSkeleton.h, r3dPhysSkeleton.cpp

Important methods

void syncAnimation()

If skeleton in ragdoll mode (physics simulation enabled), this function will propagate transformation matrices from physics engine to render engine. With this we can properly render all body motion simulated by ragdoll.

If skeleton in animation-controlled mode, this function will propagate bones motion to physics meshes to setup proper initial position for the ragdoll mode switching time. In animation-controlled mode physical simulation for ragdoll is disabled.

void SwitchToRagdoll()

Toggle ragdoll and animation-controlled modes. Initial mode is animation-controlled.

void SwitchToRagdollWithRayImpulse(const r3dRay &impulse)

Same as SwitchToRagdoll function, but when new mode is ragdoll physics simulation (not animation-controlled mode) we apply ray impulse to the physics body.

Parameters:
  impulse – force origin and direction that define ray impulse. Magnitude of direction vector represents force amount.

void SwitchToRagdollWithDirectionalForce(const r3dPoint3D &force)

Same as SwitchToRagdoll function, but when new mode is ragdoll physics simulation (not animation-controlled mode) we apply directional area force field to the physics body.

Parameters:
  force – force direction. Magnitude of this vector represents force amount.

bool IsRagdollMode() const

Return true if the current mode is ragdoll physics simulation, and false if skeleton is in animation-controlled state.

r3dBoundBox getWorldBBox() const;

Calculate and return bounding box of current mesh state.

struct ActorBone

This is simple structure, that used for linking animation bone with physics bone mesh. Using this information we can pass pose matrices from animations to physics and vice versa.

Important members

PxRigidDynamic* actor
Physics actor pointer, that represents physics mesh for bone.

int boneID
Bone identifier for game engine animation system.