

PhysX Tutorial 02

Denison Linus

dlmtavares@inf.ufrgs.br

<http://www.inf.ufrgs.br/~dlmtavares>

4 - Character Controller

CApplication.h

CApplication(...) : ... CPhysXCharacterControl(0)

private:

...

CPhysXCharacterControl* PhysXCharacterControl;

void characterFPSCameraUpdate();

bool CursorKeys[5];

f32 JumpSpeed;

void CApplication::run()

{

while(Device->run() && Driver)

if(Device->isWindowActive())

{

characterFPSCameraUpdate();

PhysicsManager->update();

4 - Character Controller

```
void CApplication::characterFPSCameraUpdate()
{
    if(!PhysXCharacterControl)
        return;

    if(CursorKeys[0])
    {
        ICameraSceneNode* camera = Smgr->getActiveCamera();
        vector3df position = camera->getPosition();
        vector3df target = camera->getTarget();
        vector3df irrDir = (target - position).normalize();
        PhysXCharacterControl->move(irrDir);
    }
    ...
}

bool CApplication::OnEvent(SEvent event)
{
    if(!Device)
        return false;

    if (event.EventType == EET_KEY_INPUT_EVENT)
    {
        if(PhysXCharacterControl)
        {
            if(irr::KEY_KEY_W == event.KeyInput.Key)
            {
                CursorKeys[0] = event.KeyInput.PressedDown;
            }
            ...
        }
    }
}
```

4 - Character Controller

- Add FPS Camera with Character Controller:

```
ICameraSceneNode* camera = Smgr->addCameraSceneNodeFPS(0, -100, 200, 500);  
camera->setPosition(vector3df(500, 150, 120));  
camera->setTarget(vector3df(499, 150, 120));
```

```
PhysXCharacterControl = PhysicsManager->createCharacterControl  
(  
    CPhysXManager::generateUniqueName(),  
    camera,  
    12, 4  
);
```

```
PhysXCharacterControl->setCharacterSpeed(100);  
PhysXCharacterControl->setAutoStepping(10);  
PhysXCharacterControl->setEyeHeight(vector3df(0,10,0));
```

4 - Character Controller

- Add Rigid Mesh:

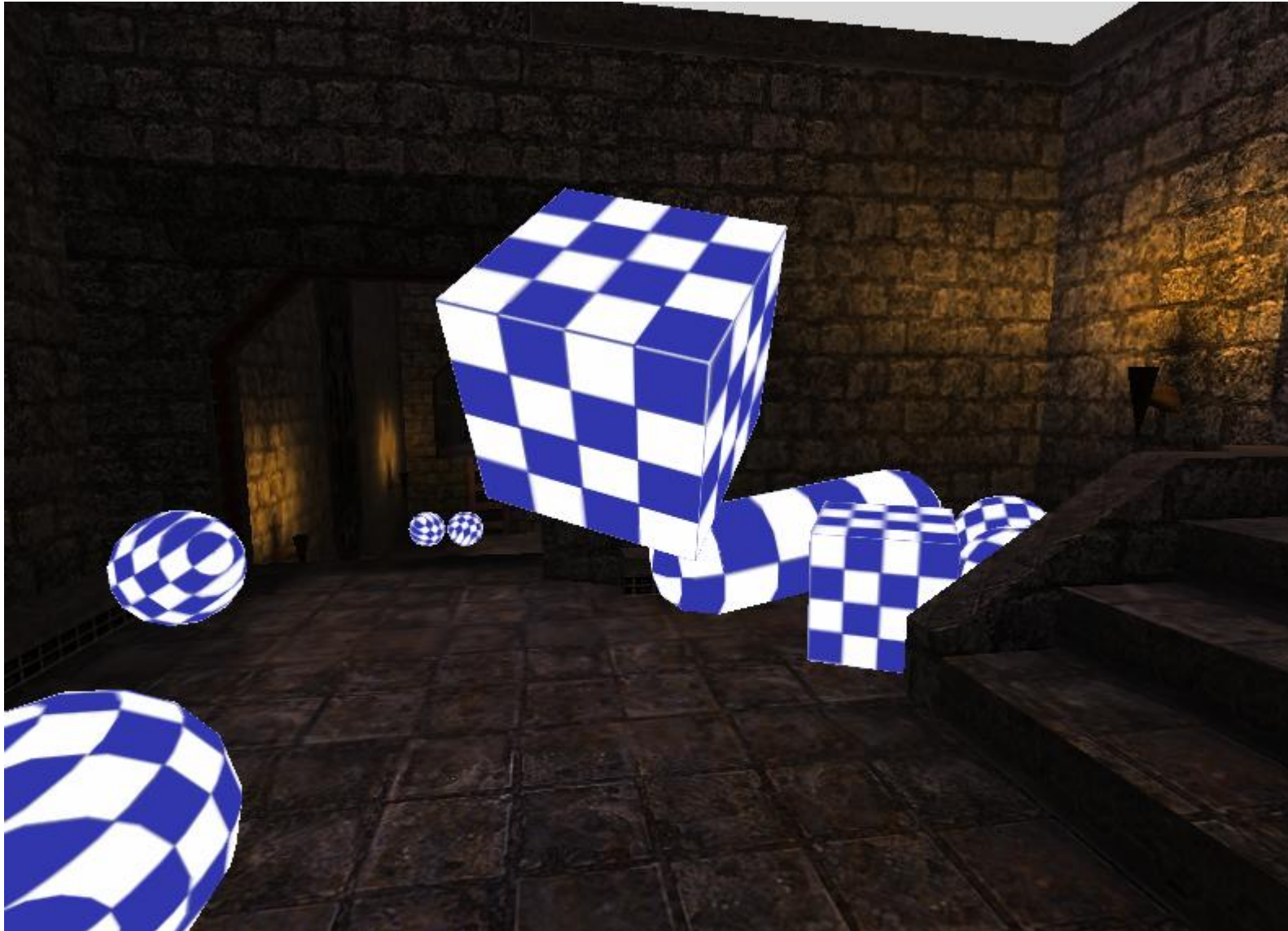
```
FileSystem->addZipFileArchive("Pak/map-20kdm2.pk3");  
IAnimatedMesh* sceneMesh = Smgr->getMesh("Pak/20kdm2.bsp");  
ISceneNode* sceneNode = Smgr->addOctTreeSceneNode(sceneMesh);  
sceneNode->setScale(vector3df(0.25f, 0.25f, 0.25f));  
sceneNode->setMaterialFlag(video::EMF_LIGHTING, false);  
sceneNode->setMaterialFlag(video::EMF_BACK_FACE_CULLING, false);
```

```
PhysicsManager->createRigidMesh  
(  
    CPhysXManager::generateUniqueName(),  
    sceneNode, sceneMesh->getMesh(0),  
    "Pak/20kdm2.bin",  
    ECCG_COLLIDABLE_NON_PUSHABLE  
);
```

4 - Character Controller

- **Static Mesh With Character Control**
 - **Edit loadSceneBSPWithCharacterControl()**
 - Add 1 FPS with Character Controller
 - Add 1 Rigid Mesh ("Pak/20kdm2.bsp")
 - Clear the CursorKeys vector
 - **Edit characterFPSCameraUpdate()**
 - Complete the CursorKeys actions
 - **Edit OnEvent(...)**
 - Complete the key press events
 - **Edit loadSceneTerrainWithCharacterControl()**
 - Add 1 FPS with Character Controller
 - Add 1 Create Rigid Terrain ("PAK/terrain-heightmap.bmp")
 - Clear the CursorKeys vector

4 - Character Controller



5 - Constraints (Joints, Breakable joints)

- **Extra Source Code:**
 - http://www.inf.ufrgs.br/~dlmtavares/graduate/INF01019/Tutorial_PhysX_Extra.zip
 - `void loadSceneRagDoll();`
 - `void loadSceneRagDollBreakable();`
 - `void loadSceneSimpleJoints();`

Features Missing in the wrapper

- **Raycasting (collision detection with rays);**
- **Particles System (particle emitter);**
- **Force Fields (area of influence);**
- **Continuous Collision Detection (for fast moving objects);**
- **Fluid Particles (simulation of liquids and gases);**
- **Soft Bodies (volumetric deformable objects);**
- **Cloth (flags, clothing);**
- **Vehicles (wheel Shapes, suspension, torque, brakes and steering);**
- **...**