Bullet i dźwięki

- Wprowadzenie
  - Bullet
  - Rola
  - Wykorzystanie
  - Wygląd obiektów
- Punkty kontaktowe
  - Co to
  - Podstawowe zadania
  - Generacja
  - optymalizacje

- Fizyczne dźwięki
  - Dwa dźwięki
  - Dźwięk uderzenia
  - Dźwięk przesuwania

Autor: Paweł Mogiła

Bullet

- opensource
- darmowy
- silnik fizyczny
- gry (realtime) / filmy

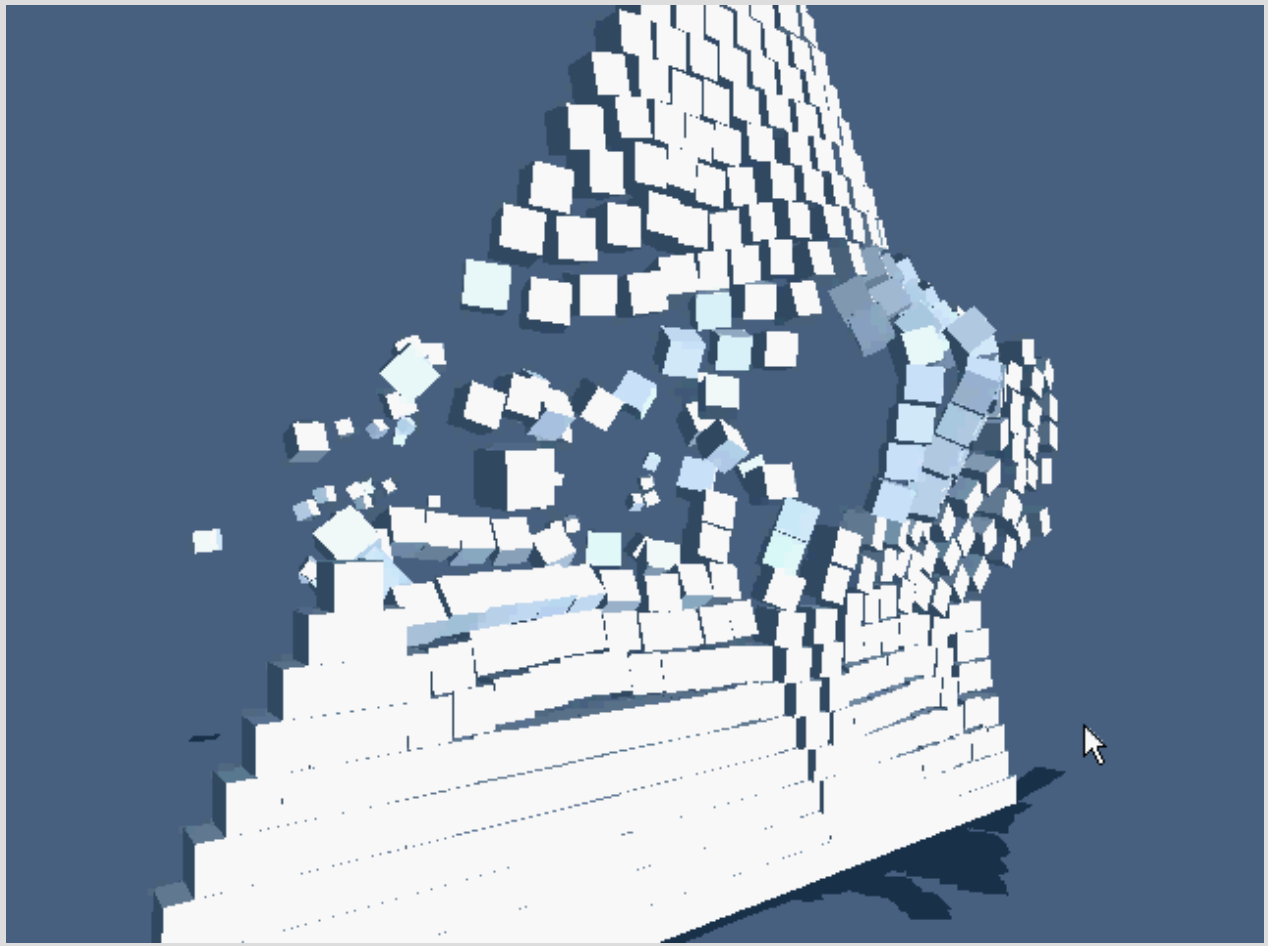
np. GTA4 ,toy story 3 ,2012

Silniki Fizyki - inne

- Havok
- PhysX

- ODE
- Box2d

Bullet – do czego



Bullet - Rola

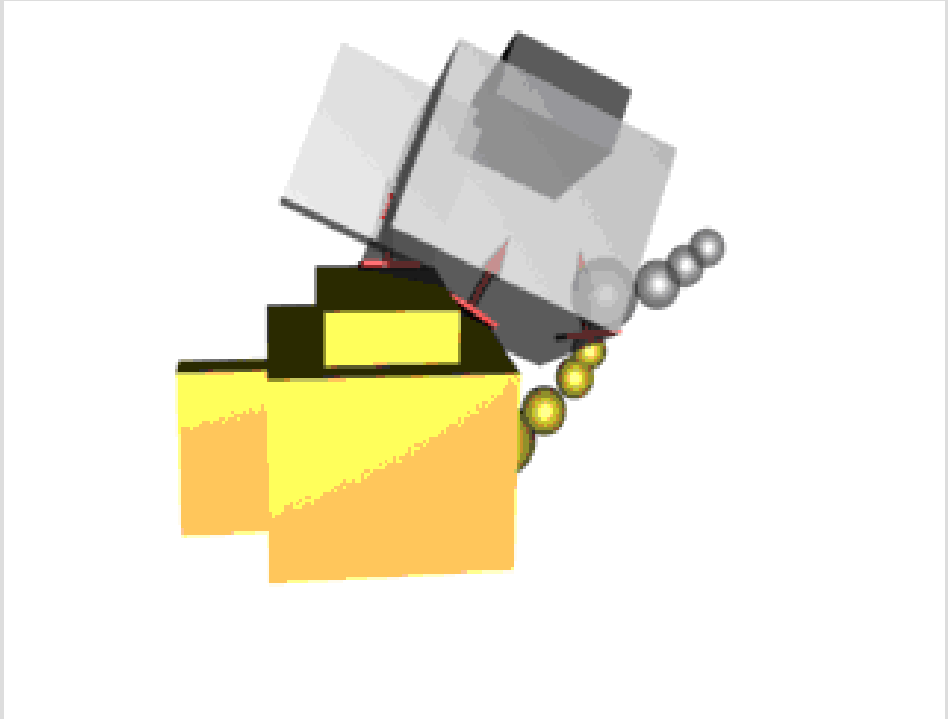
+ Symulacja ruchu
- masa
- odbijalność
- tarcie statyczne
- tarcie dynamiczne
- itp.

+ Wykrywanie kolizji
- sfera
- prostopadłościan
- siatka trójkątów (statyczna)
- wypukłe wielokąty
- wklęsłe wielokąty z siatki trójkątów (niezalecane)
- Kombinacje powyższych

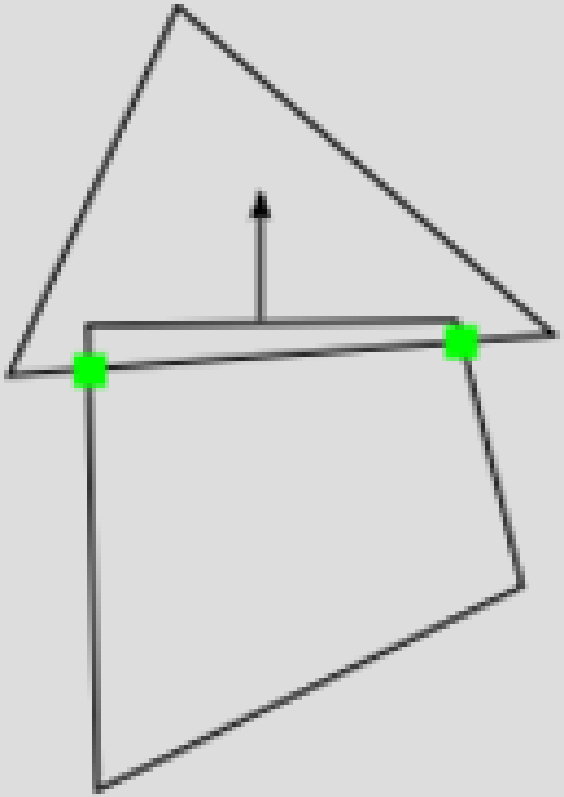
Wyglad obiektów

Grafika

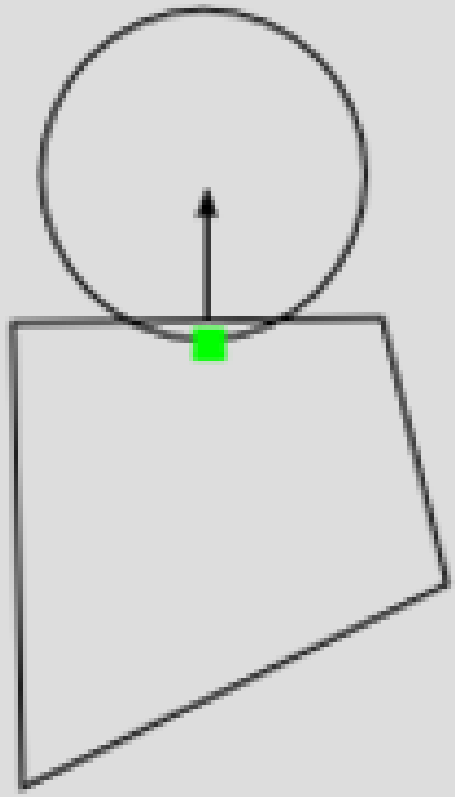
Fizyka



Punkty kontaktowe

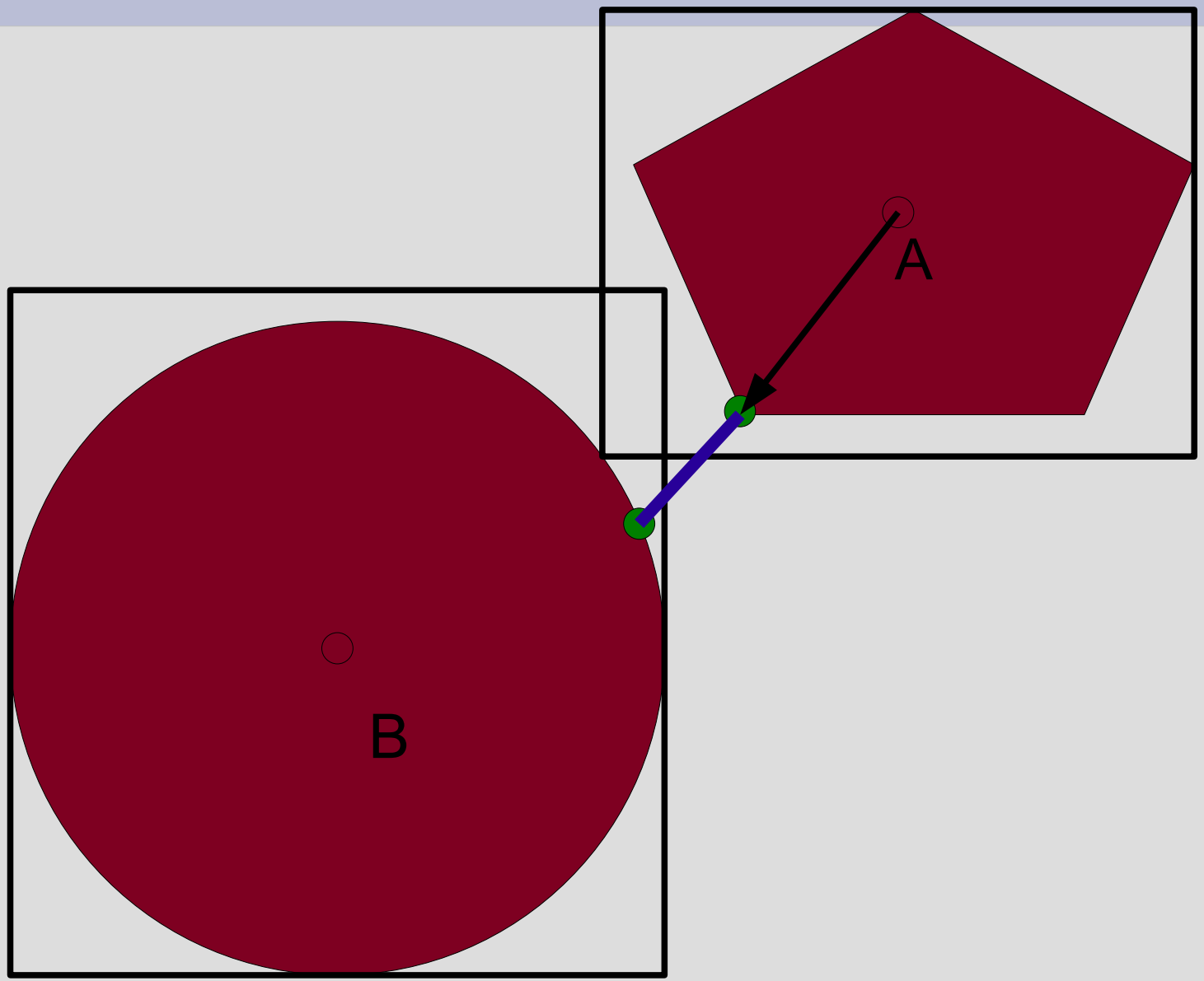


two points, one normal



two points, one normal

Punkty kontaktowe



Punkty kontaktowe- specyfikacja bulleta

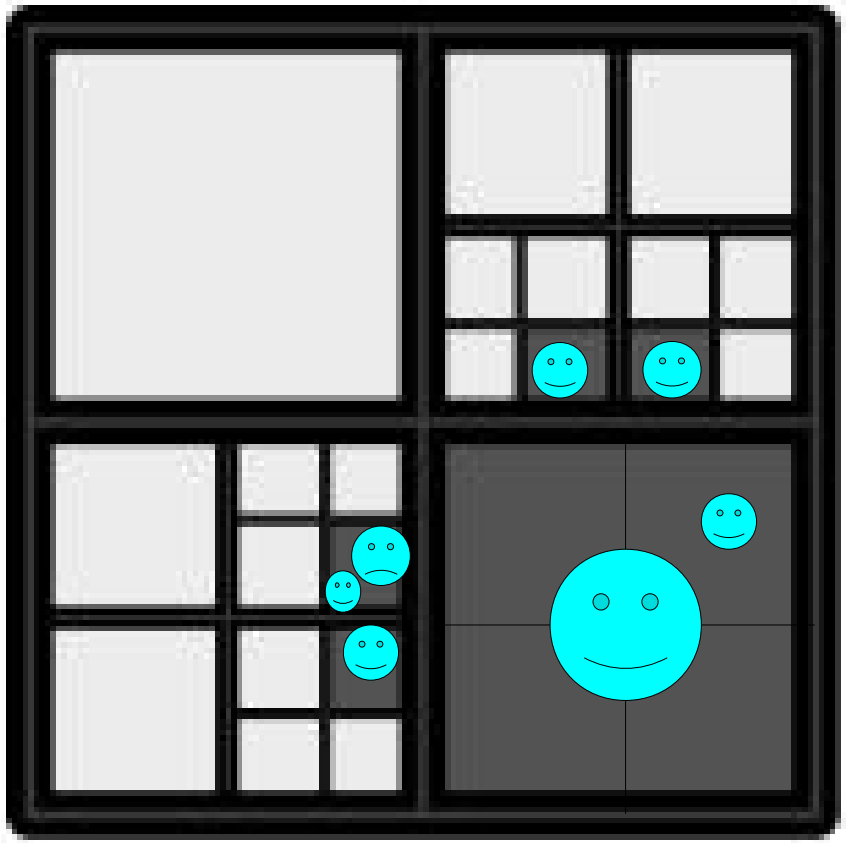
```
btScalar                getDistance () const
int                     getLifeTime () const
const btVector3 &       getPositionWorldOnA ()
const btVector3 &       getPositionWorldOnB ()

btScalar    getAppliedImpulse () const

btVector3   m_localPointA
btVector3   m_localPointB

btRigidBody  *    getBody0 ()
btRigidBody  *    getBody1 ()
```

Generacja punktów

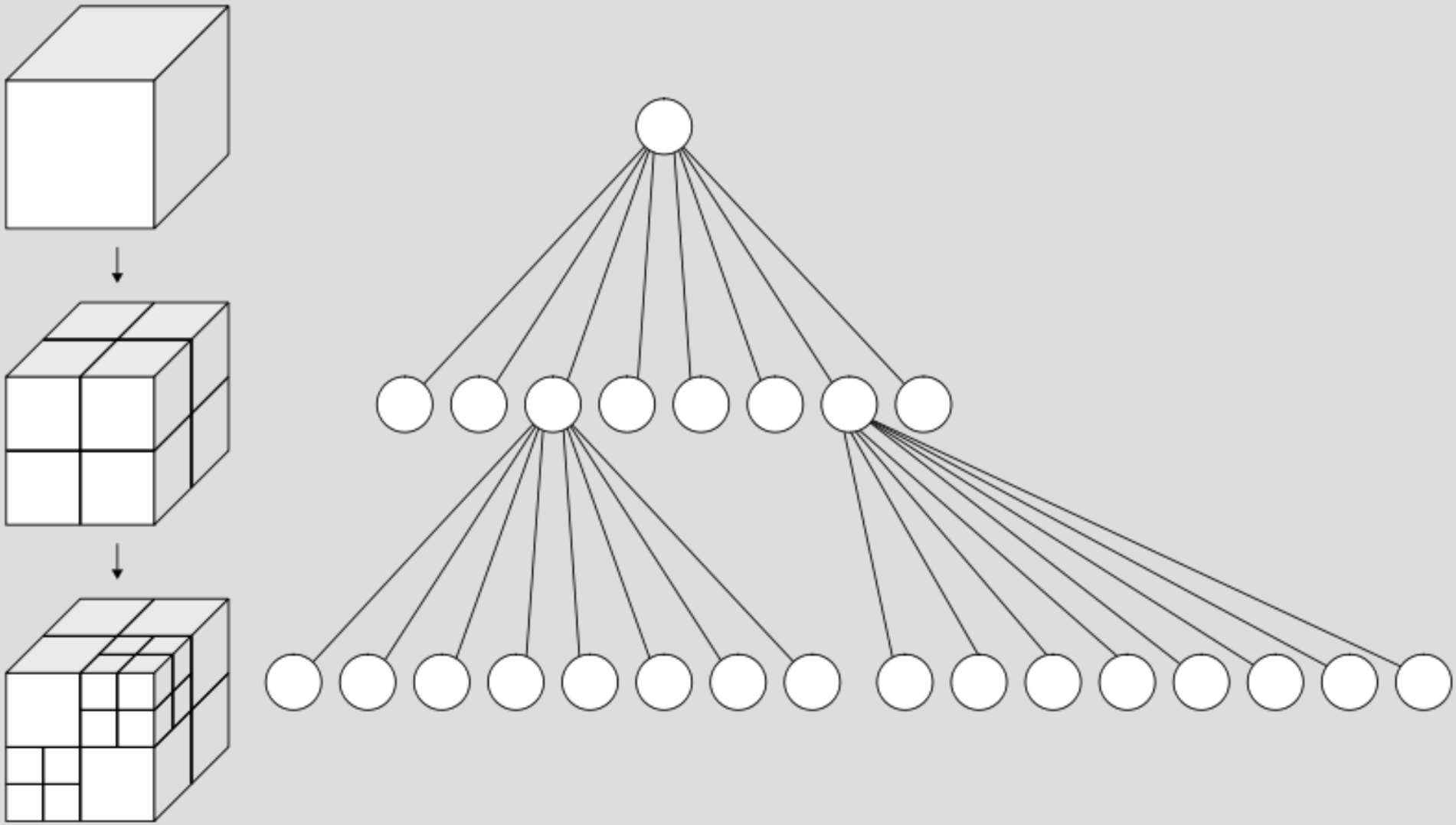
- brute force - złożoność kwadratowa

+ optymalizacje
- otaczający prosty kształt
  (bounding box, bounding sphere)
- umieszczanie obiektów w drzewa
(quadtree ,octree)
- „usypianie” obiektów

Quadtree



Octree



Fizyczne dźwięki

Fizyczne dźwięki

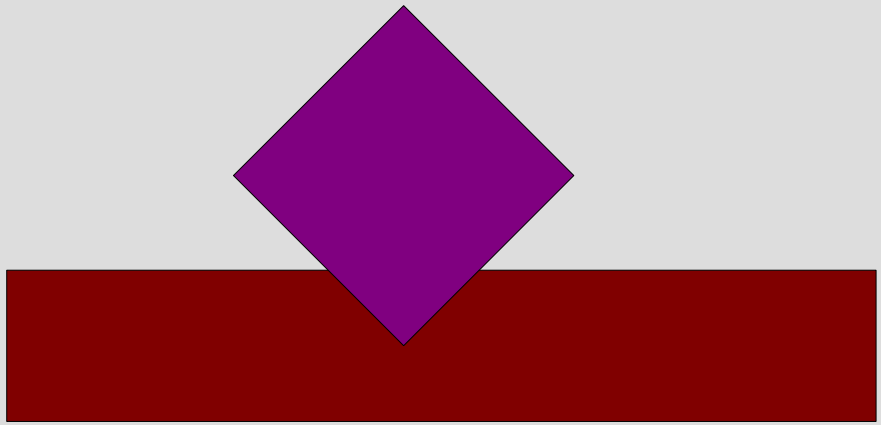
+Przeglądamy tablice punktów kontaktowych
+nakładamy warunki

Dwa rodzaje dźwięków:
- dźwięk uderzenia
- dźwięk przesuwania

Dźwięk uderzenia

Warunki:

- getDistance () < 0.f
- getAppliedImpulse () > MAGIC_VALUE
- getLifeTime () == 1

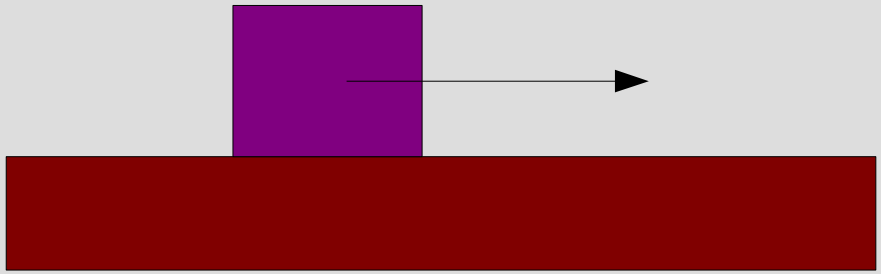


Dźwięk przesuwania

Warunki:

getDistance()< SMALL_MAGIC_VALUE_1

(getBody0()->getLinearVelocity()-
getBody1()->getLinearVelocity()).length()
> SMALL_MAGIC_VALUE_2



Kod

```
for (int i=0;i<  dynamicsWorld->getDispatcher()->getNumManifolds();i++)
{
    btPersistentManifold* contactManifold =
      dynamicsWorld->getDispatcher()->getManifoldByIndexInternal(i);
    for(int t=0;t<getNumContacts();++t)
    {
        auto point=contactManifold->getContactPoint(t);
        if(point.getDistance()<-0.5f && point.getLifeTime()==1 &&
              point.getAppliedImpulse()>100.f)
        {
            //play impact sound
        }else
        if(((contactManifold->getBody0()->getLinearVelocity()-contactManifold
      ->getBody1()->getLinearVelocity()).length()>4.f) &&
             point.getDistance()<1.5f )
        {
            //play sliding sound
        }
    }
}
```

Przykład

Ulepszenia

- natężenie dźwięku
- wysokość dźwięku
- zależność dźwięku od materiałów

Powiązane linki:

- http://bulletphysics.org
- http://szamq.wordpress.com/

Powiązane linki:

- http://bulletphysics.org
- http://szamq.wordpress.com/

- Pytania?