Bullet Sensors

Sensors

Bullet doesn't feature sensors as Box2D did. So we have to find another way to deal with colliders that inform about a collision but don't calculate a contact response.

We will use Bullet **flags**.

Sensors

```cpp
void PhysBody3D::SetAsSensor(bool is_sensor)
{
    if(this->is_sensor != is_sensor)
    {
        this->is_sensor = is_sensor;
        if(is_sensor == true)
            body->setCollisionFlags(body->getCollisionFlags() | btCollisionObject::CF_NO_CONTACT_RESPONSE);
        else
            body->setCollisionFlags(body->getCollisionFlags() &~ btCollisionObject::CF_NO_CONTACT_RESPONSE);
    }
}
```

Uh...

If those operators look weird, don’t worry.

Bit-wise operators aren’t used frequently, but it’s good to know they exist.

```
//Add “CF_NO_CONTACT_RESPONSE””” to Current Flags
CurrentFlags | btCollisionObject::CF_NO_CONTACT_RESPONSE);

//Remove “CF_NO_CONTACT_RESPONSE” from Current Flags
CurrentFlags &~ btCollisionObject::CF_NO_CONTACT_RESPONSE);
```

&	|	~	^	>>	<<
AND	OR	NOT	XOR	Right Shift	Left Shift

Useful work

How about implementing a function that can send events like:

CONTACT_BEGIN and CONTACT_END,

instead of calling "OnContact" every single frame?

AND NOW . . .

Let's make a game!