Subject: gameobject lists-> which&when
Posted by robbyke on Fri, 07 Oct 2011 00:34:59 GMT
View Forum Message <> Reply to Message

with the way with the object lists i cant find out wich to use.
my problem :

created object.
attached a script
find out if soldier is near object with attached script

that last step wont work ive tried out some lists but im doubting if i can still find my newly created object like this or maybe i first have to see under wich class its now divided

Subject: Re: gameobject lists-> which&when
Posted by Gen_Blacky on Fri, 07 Oct 2011 01:52:52 GMT
View Forum Message <> Reply to Message

This should work depending on what exactly you want to do. Just modify it to suit your needs. Like check to see if script is attached to what ever and set how far the player should be or whatever.

```
GameObject *Get_Nearest_Solider(GameObject *obj) // Get the nearest solider next to a object
{
 float closestdist = FLT_MAX;
 Vector3 obj_pos = Commands->Get_Position(obj);
 GameObject *object = 0;
 SLNode<BaseGameObj> *x = GameObjManager::GameObjList.Head();
 while (x)
 {
  GameObject *o = (GameObject *)x->Data();
  if (o && As_SoldierGameObj(o))
  {
   Vector3 player_pos = Commands->Get_Position(o);
   float dist = Commands->Get_Distance(player_pos,obj_pos);
   if (dist < closestdist)
   {
    closestdist = dist;
    object = o;
   }
  }
  x = x->Next();
 }
 return object;
}
```

Subject: Re: gameobject lists-> which&when
Posted by jonwil on Fri, 07 Oct 2011 13:32:12 GMT
View Forum Message <> Reply to Message

This code will work better

```
#include "GameObjManager.h"
#include "SoldierGameObj.h"
GameObject *Get_Nearest_Solider(GameObject *obj) // Get the nearest solider next to a object
{
 float closestdist = FLT_MAX;
 Vector3 obj_pos = Commands->Get_Position(obj);
 SmartGameObj *object = 0;
 for (SLNode<SmartGameObj>* node = GameObjManager::SmartGameObjList.Head(); node;
node = node->Next())
 {
  SmartGameObj* object = node->Data();
  if (o && o->As_SoldierGameObj())
  {
   Vector3 player_pos = Commands->Get_Position(o);
   float dist = Commands->Get_Distance(player_pos,obj_pos);
   if (dist < closestdist)
   {
    closestdist = dist;
    object = o;
   }
  }
 }
 return object;
}
```

That code will find the closest soldier to the passed in object.
The SmartGameObjList is for soldiers and vehicles and should be used in this case as you will have less objects to check.

Subject: Re: gameobject lists-> which&when
Posted by robbyke on Fri, 07 Oct 2011 17:58:02 GMT
View Forum Message <> Reply to Message

i had to use the first one i had to check if there was an object nearby with a certain script attached
ive found it now. i checked if my obj was a veh or a soldier and i shouldnt have done that

Subject: Re: gameobject lists-> which&when
Posted by Gen_Blacky on Sat, 08 Oct 2011 01:08:27 GMT
View Forum Message <> Reply to Message

jonwil wrote on Fri, 07 October 2011 06:32

The SmartGameObjList is for soldiers and vehicles and should be used in this case as you will have less objects to check.

Ah okay, So if I was looking for a building gameobject I would use BuildingGameObjList.