

# INTRODUCTION TO GAME MAKER: COLLISION EVENTS

Now that we have learned how to move objects in a game, it's time to learn how to check when an object collides against another object. For example, you may want an object to explode when it collides with a bullet or you may want an object to bounce off a wall when it comes in contact with it. Whatever the case may be, in order to make this happen, we will need to learn how the **Collision** event works in GameMaker.

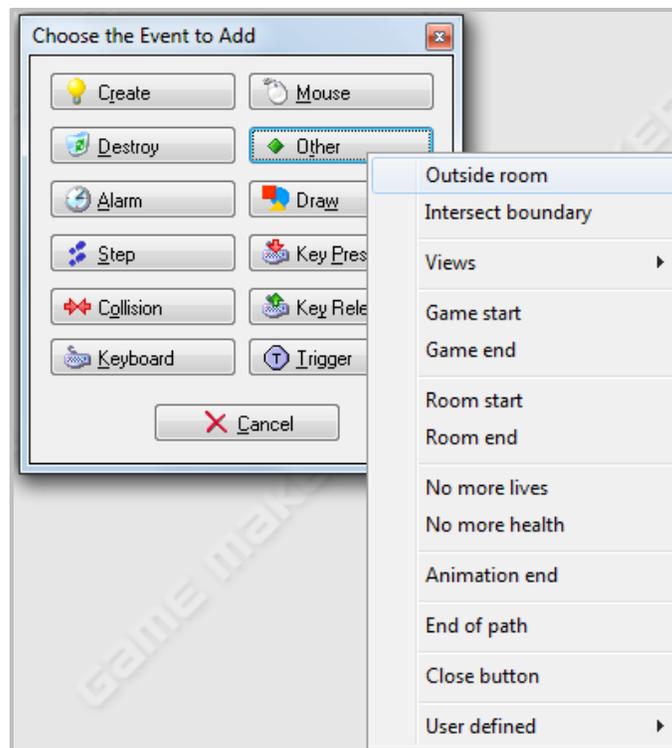


## OUTSIDE ROOM EVENT

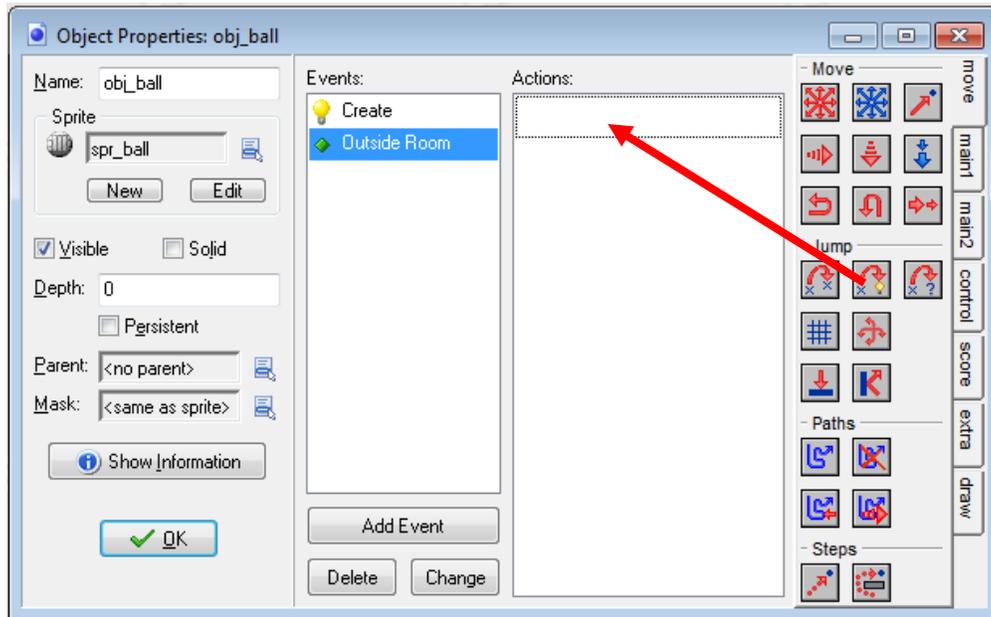
The first event we will look at is an event called **Outside Room**. The **Outside Room** event happens when an object lies completely outside a room. So, let's say we have a ball object moving across the screen. At some point the ball is going to keep moving and make its way outside the room. What we will do using the Outside Room event is make the ball come back into the room.

Here are the steps that will make that happen:

1. Go back to the previous lesson where we had a ball object moving and open the ball object.
2. Click **Add Event** and select **Other > Outside room**.



3. Now add the **Jump to Start** action to the event.



4. With this action you can let an object wrap around, that is, when it leaves one side of the room it can reappear where it first started. Keep in mind that the object must have a speed for wrapping to work because the direction of wrapping is based on the direction of the motion.

All you will need to indicate is whether you want the object to wrap only **horizontal**, only **vertical**, or **in both directions**.

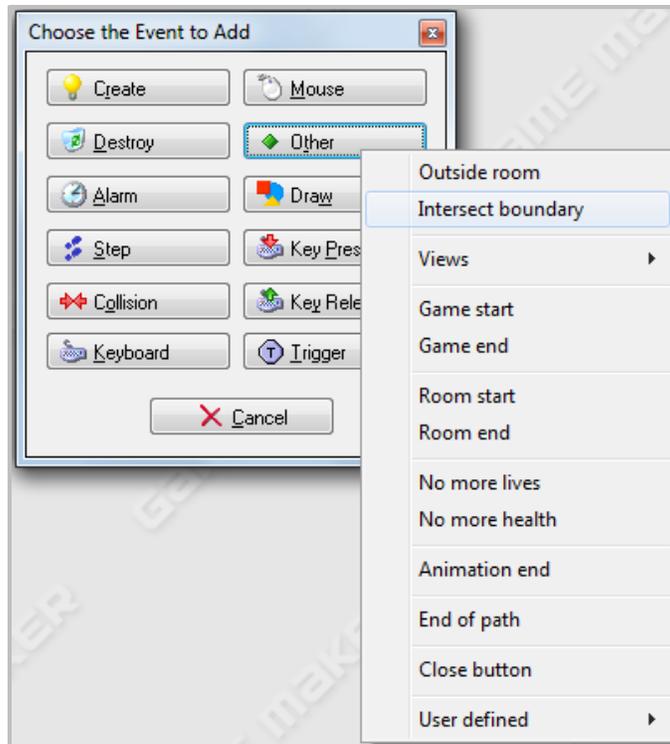
For our purposes, let's select in both directions so that if the ball goes off the screen either horizontally or vertically, it will return to the room in either case.

## INTERSECT BOUNDARY EVENT

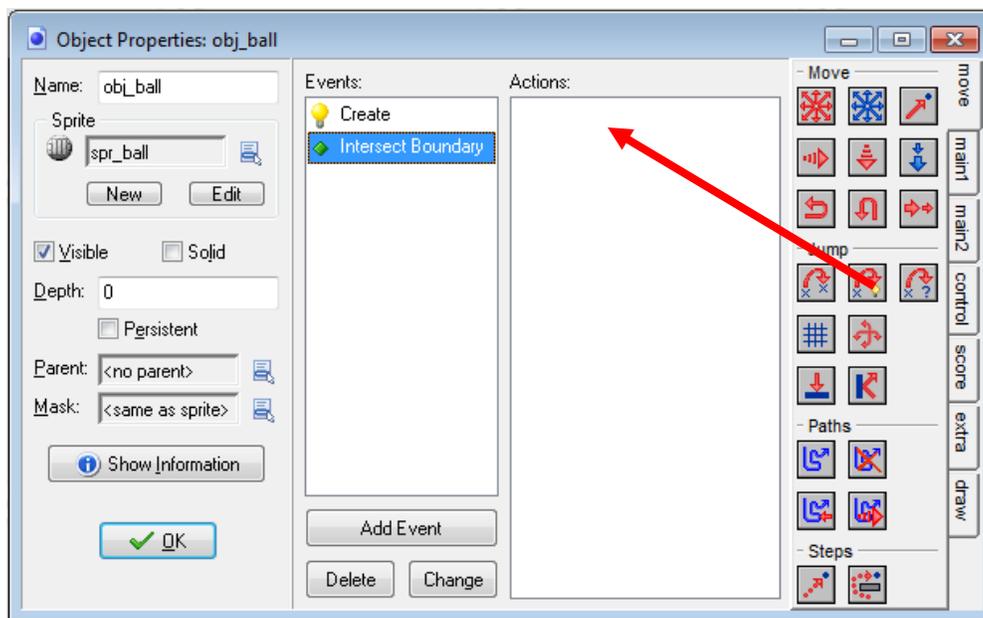
The **Intersect Boundary** event works somewhat similar to the **Outside Room** event. This event happens when an object intersects the boundary of the room, that is, it lies **at least partially** outside the room.

To use the Intersect Boundary event, you will need to follow these steps:

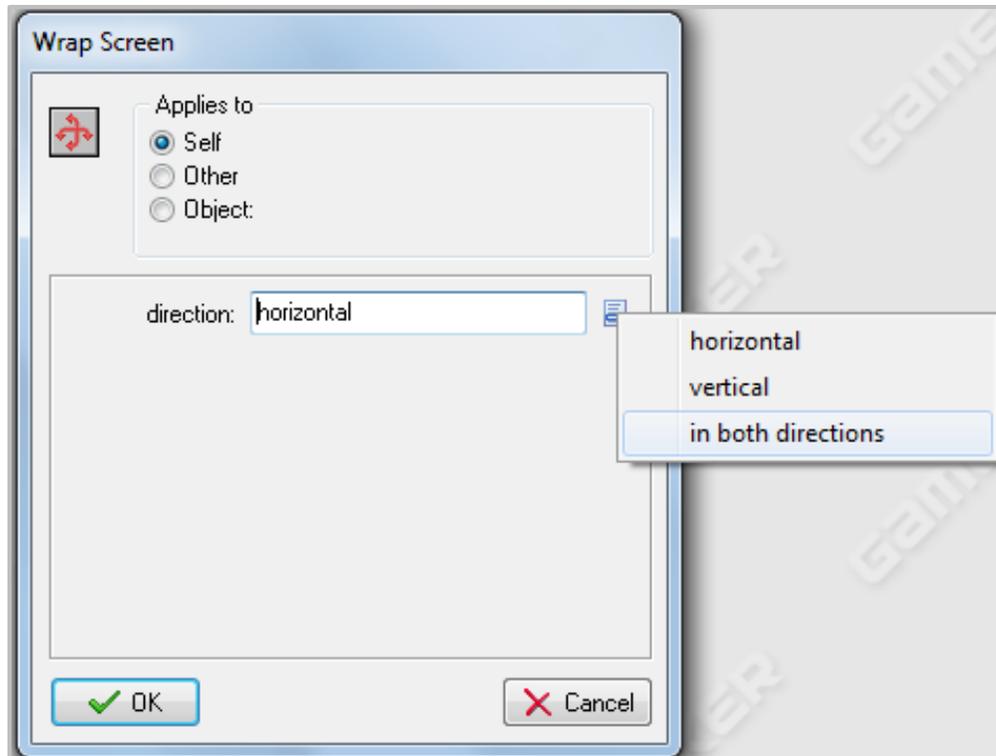
1. Click **Add Event** and select **Other > Intersect boundary**.



2. Add the **Jump to Start** action to the event.



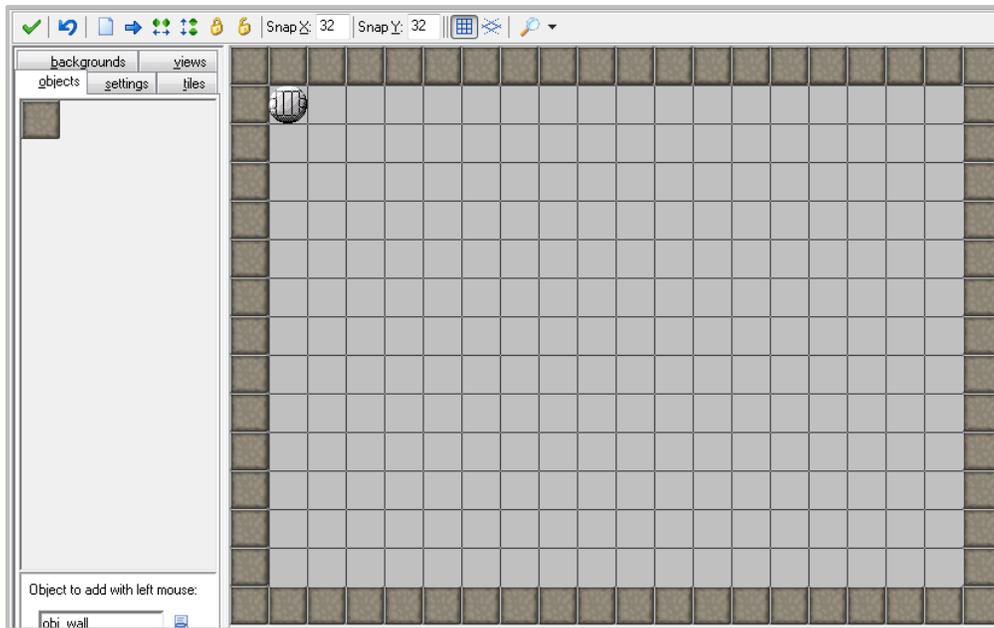
3. Select **in both directions**.



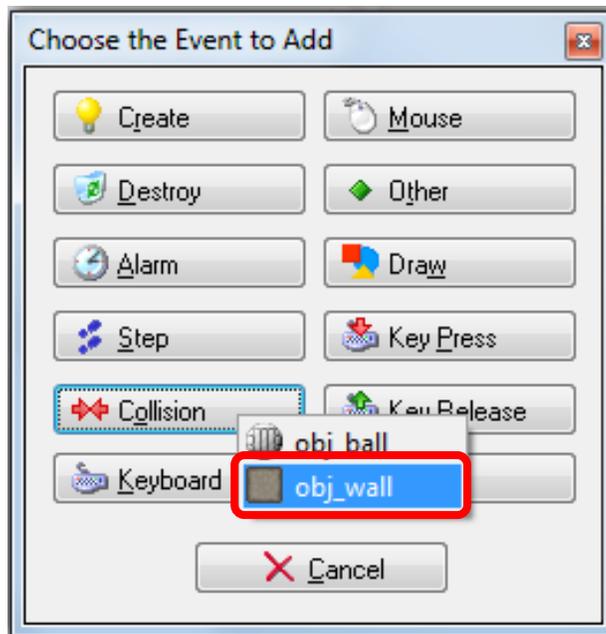
Notice what happens now when you run the program? The ball will return to its starting position once it is partially outside the room's boundary. When we used the **Outside Room** event, the ball only returned to its starting position once it was completely outside the room. It's a slight difference in terms of how an object behaves, but it is a difference that you may want to take into account when creating a game.

## COLLIDING AGAINST OBJECTS

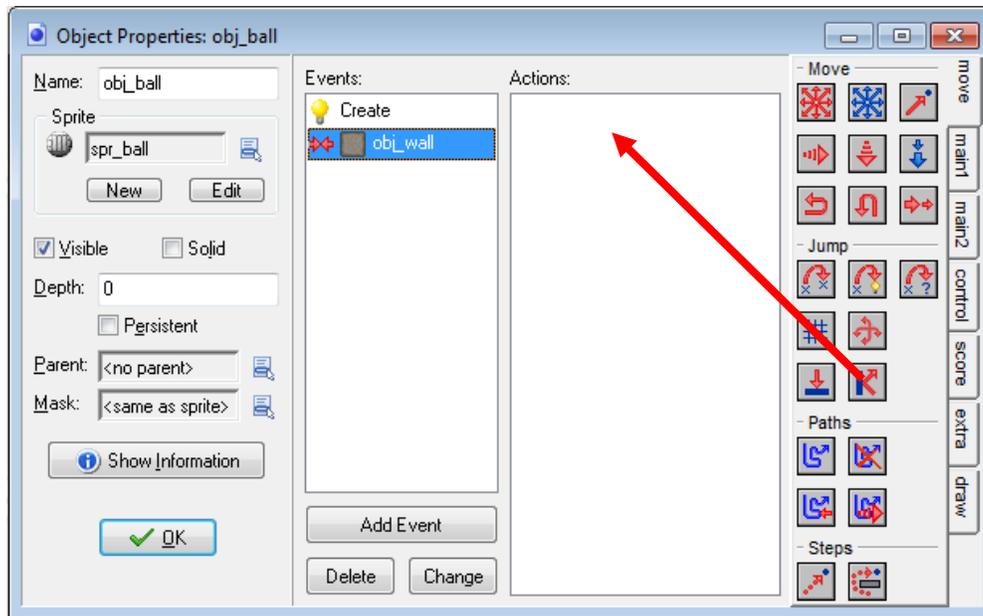
Let's now see how we can make an object do things when it collides against other objects. Let's start by putting walls around the perimeter of our room.



1. Open up the ball object and click **Add Event > Collision > obj\_wall**.



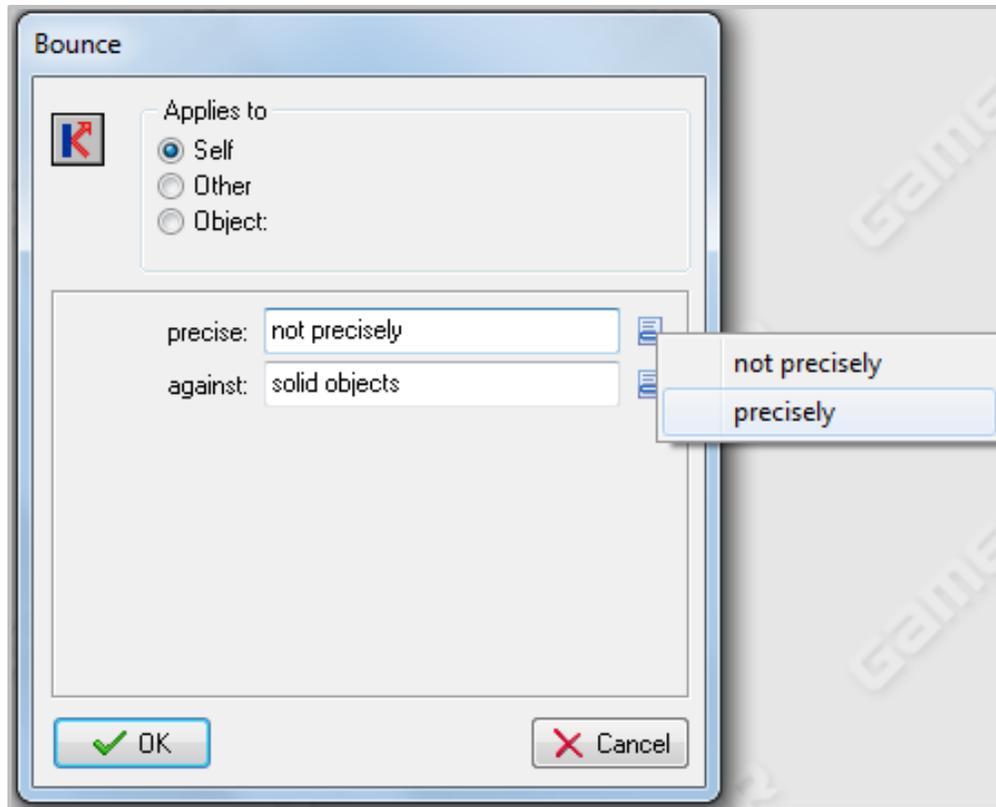
2. Add the **Bounce** action to the event.



3. When choosing the **Bounce** action, you need to specify whether you want the check to be precise or not. If you select **precisely**, it will do a precise check; if you select **not precisely** it will check based on the bounding box of the object.

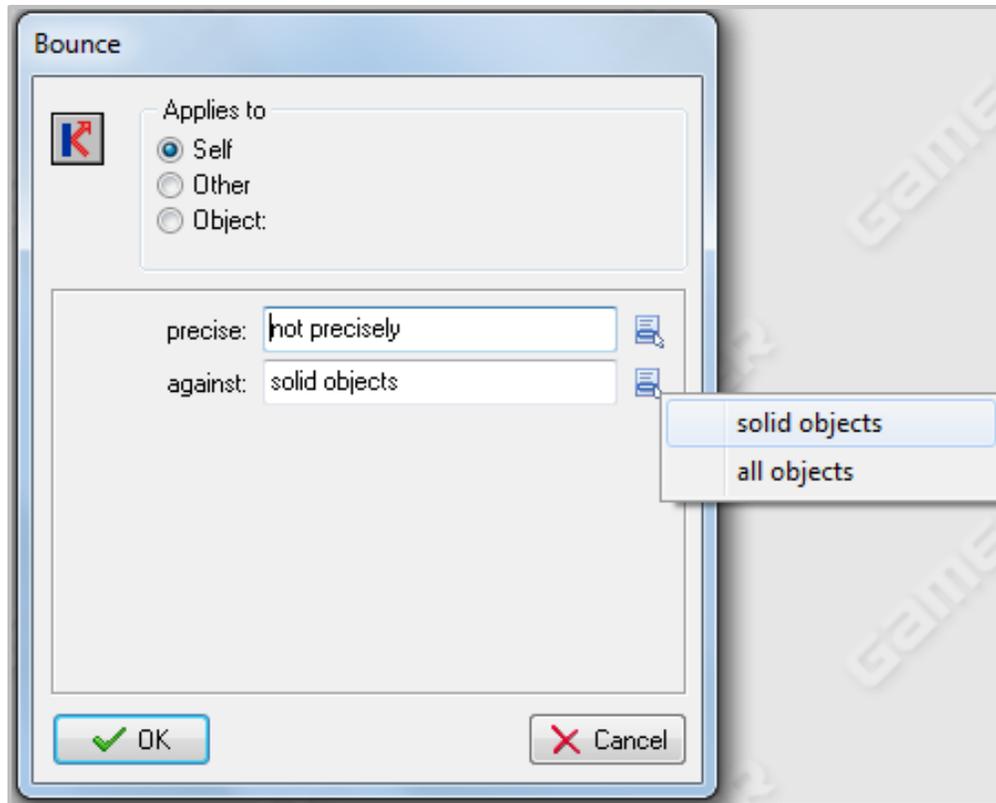
Keep in mind that precise checking is only done when the sprite for the object has the precise collision checking selected.

So let's select **precisely**.



The other option we have here is whether we want the object to bounce against **solid objects** only or **all objects** (that is, whether or not they're solid).

Let's select **solid objects** since we only want the ball to bounce off solid objects.



Now when you run the program, you will notice that the ball will bounce off the walls.