

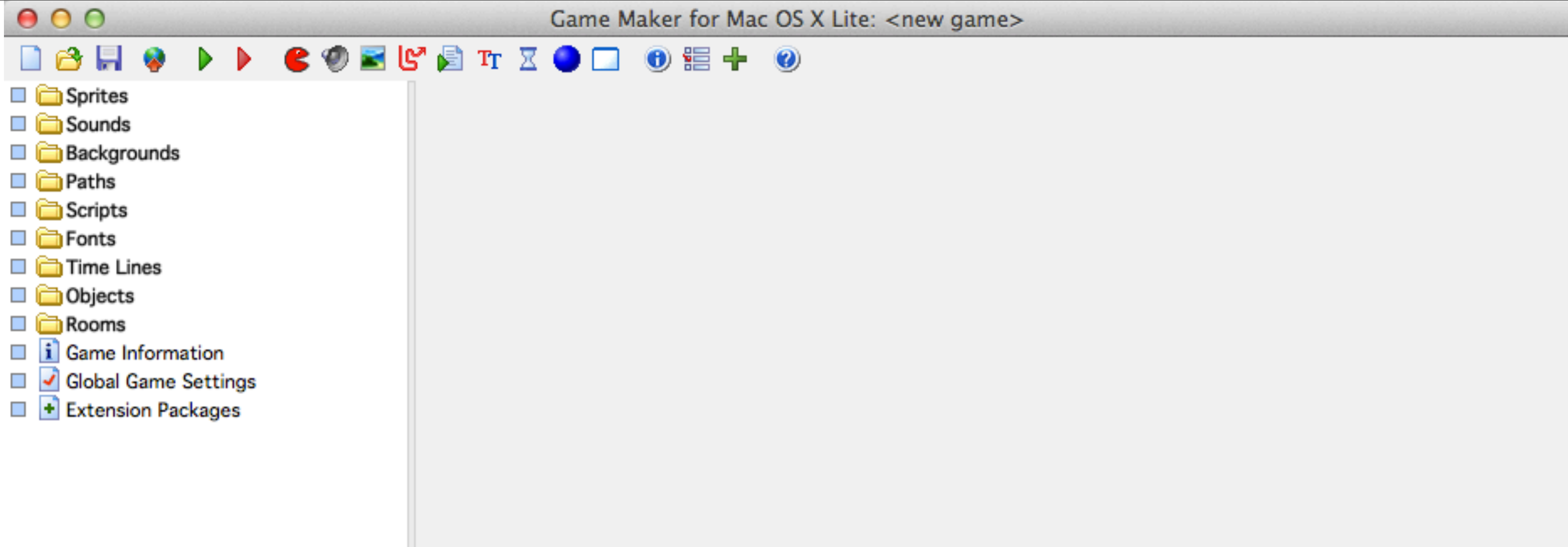


# GameMaker!



Tuesday, December 3, 13

Welcome to GameMaker! This is a picture of Spelunky, a popular roguelike originally made in GameMaker.



# What is GameMaker?



# What is GameMaker?

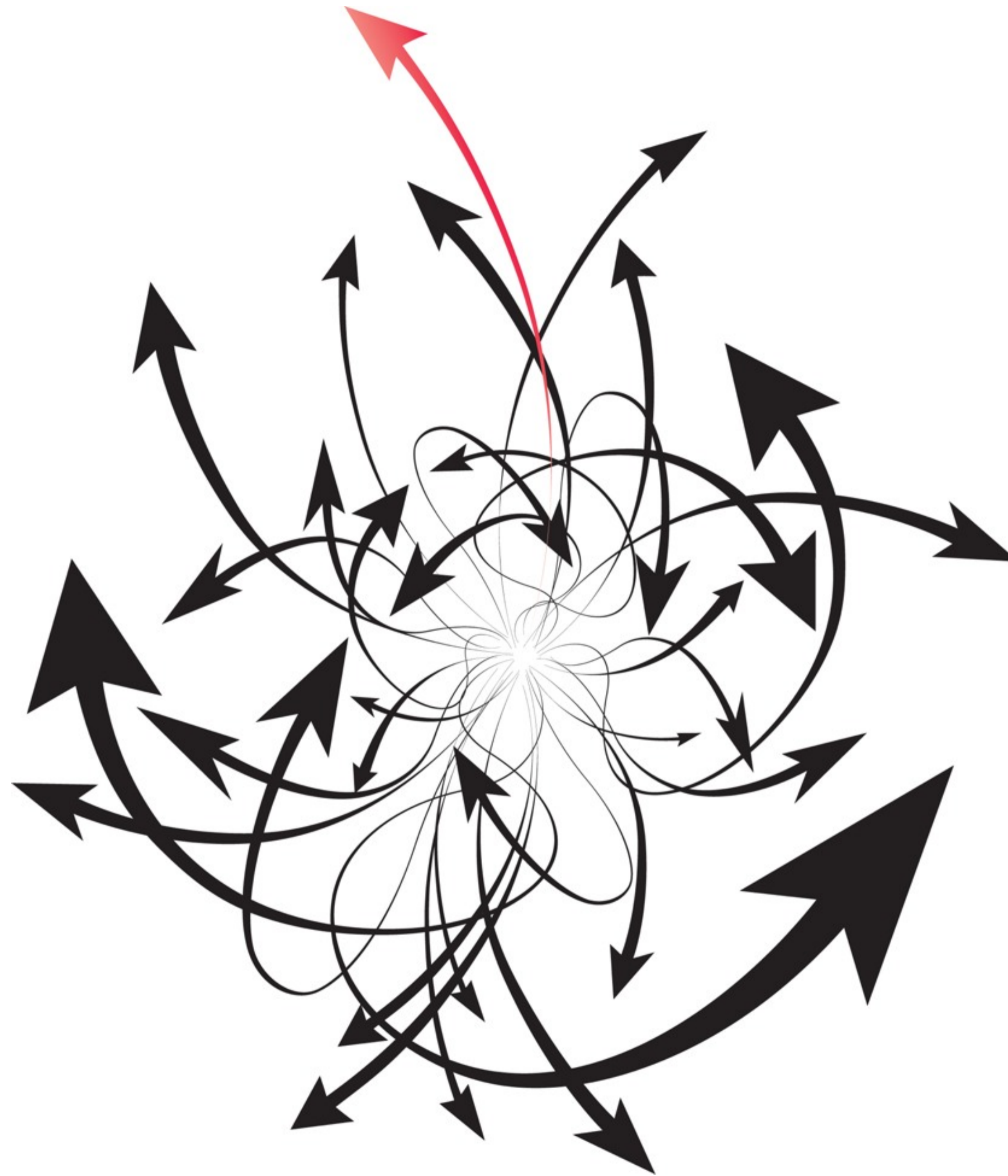
- GameMaker: a game-creation system specifically designed for novice game programmers
- Two main ways of developing with it:
  - Drag-and-drop actions, which are associated with events
  - Programming in GML, aka GameMaker Language (like C++/Java)
- You can absolutely develop with both in the same game (using the drag-and-drop for certain things and GML for others)



**YOU**

**GML**

**DRAG 'N DROP**

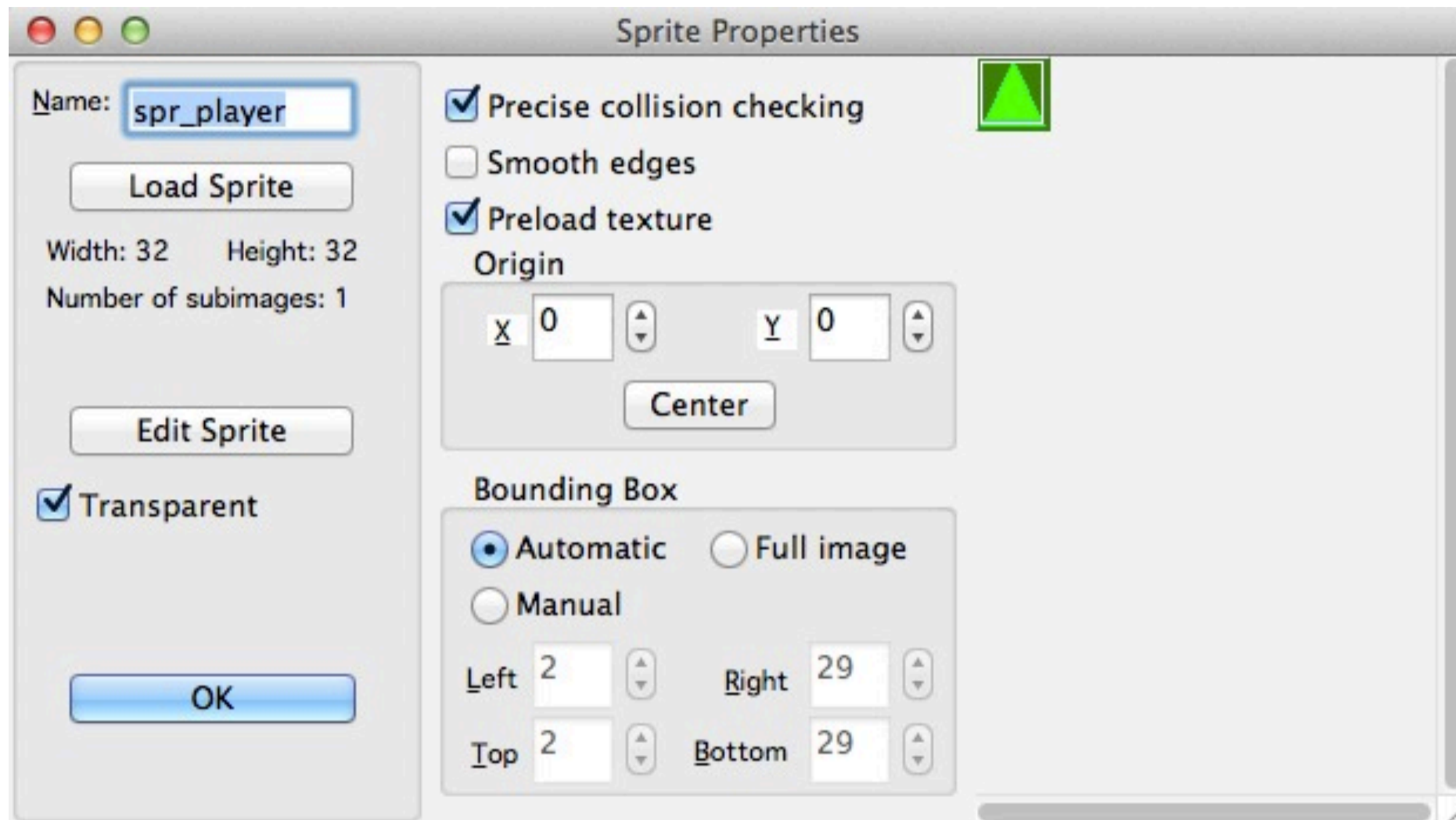


What's the GM workflow?

# General workflow

- Make the SPRITE: an image/series of images to represent a thing
- Make the OBJECT: a set of rules for how the thing works/behaves
  - Define important EVENTS (e.g. pressing a certain key)
  - Define ACTIONS for when events happen (e.g. spawn a thing)
- Place specific objects in the ROOM: a discrete space (a level, particular area, etc.)

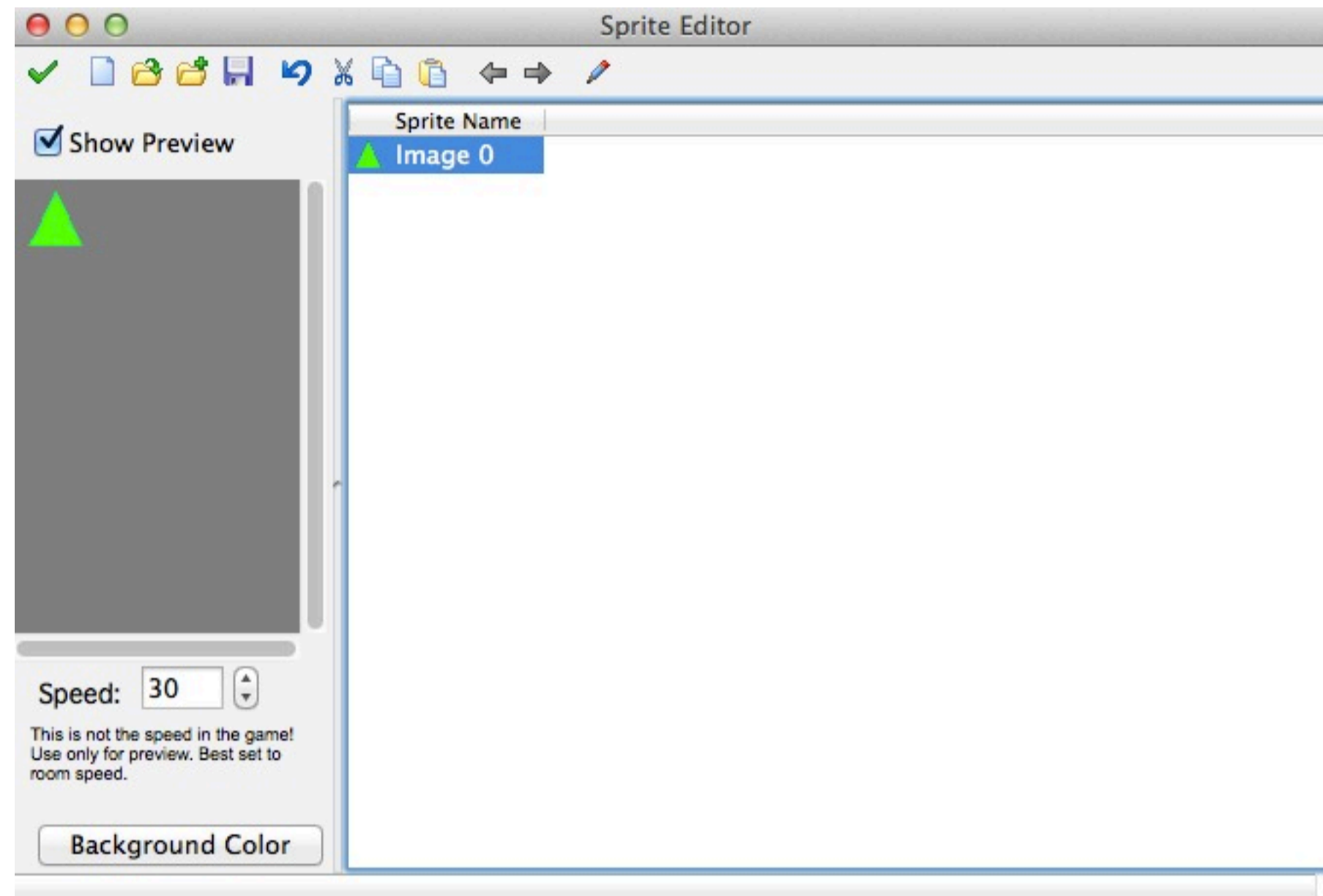
# Sprites



- Transparent: transparent background or not (if so, assumes lowest-left pixel color should = transparent)
- Origin: center point
- Bounding box: what points would count as a collision with the object

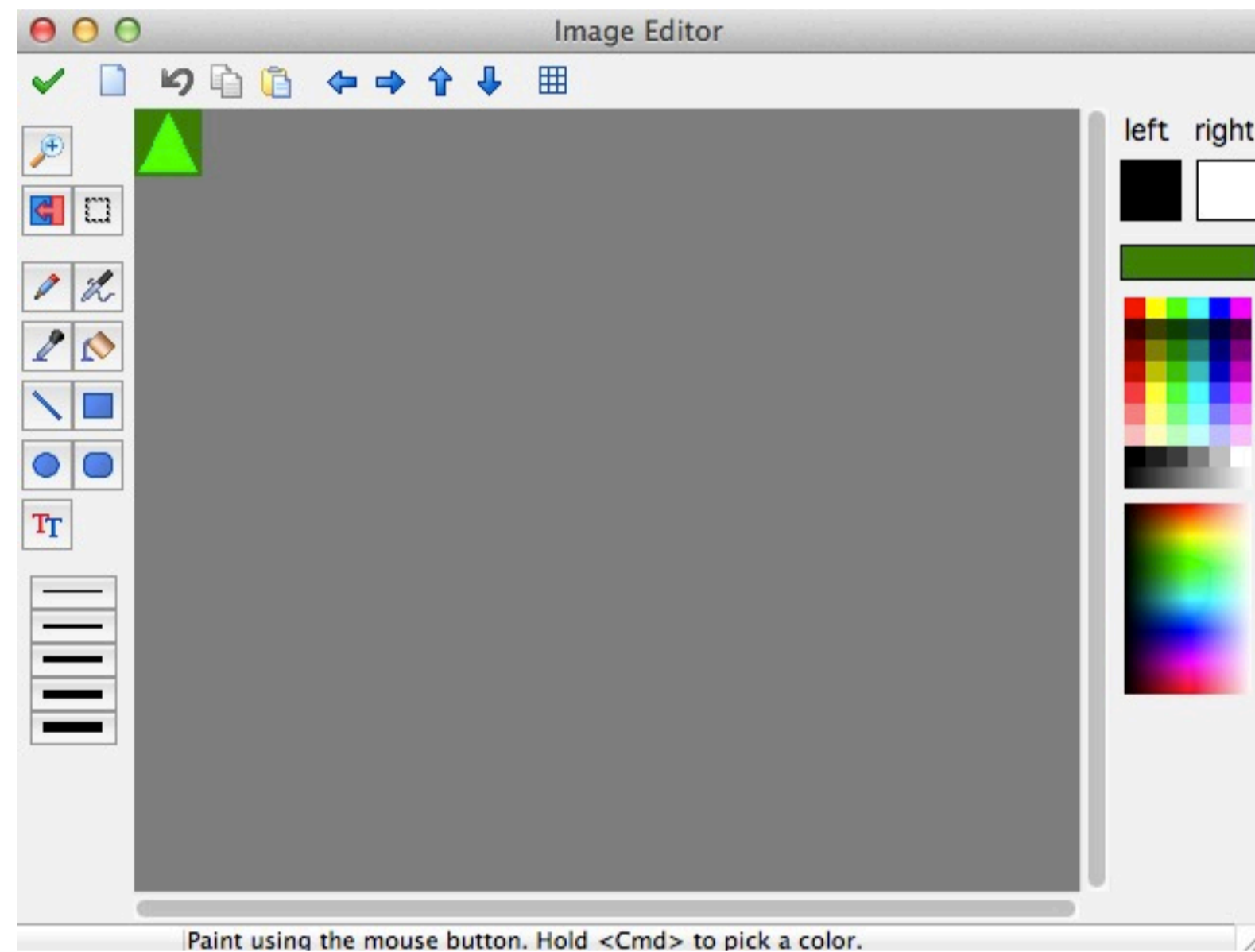


# Sprites



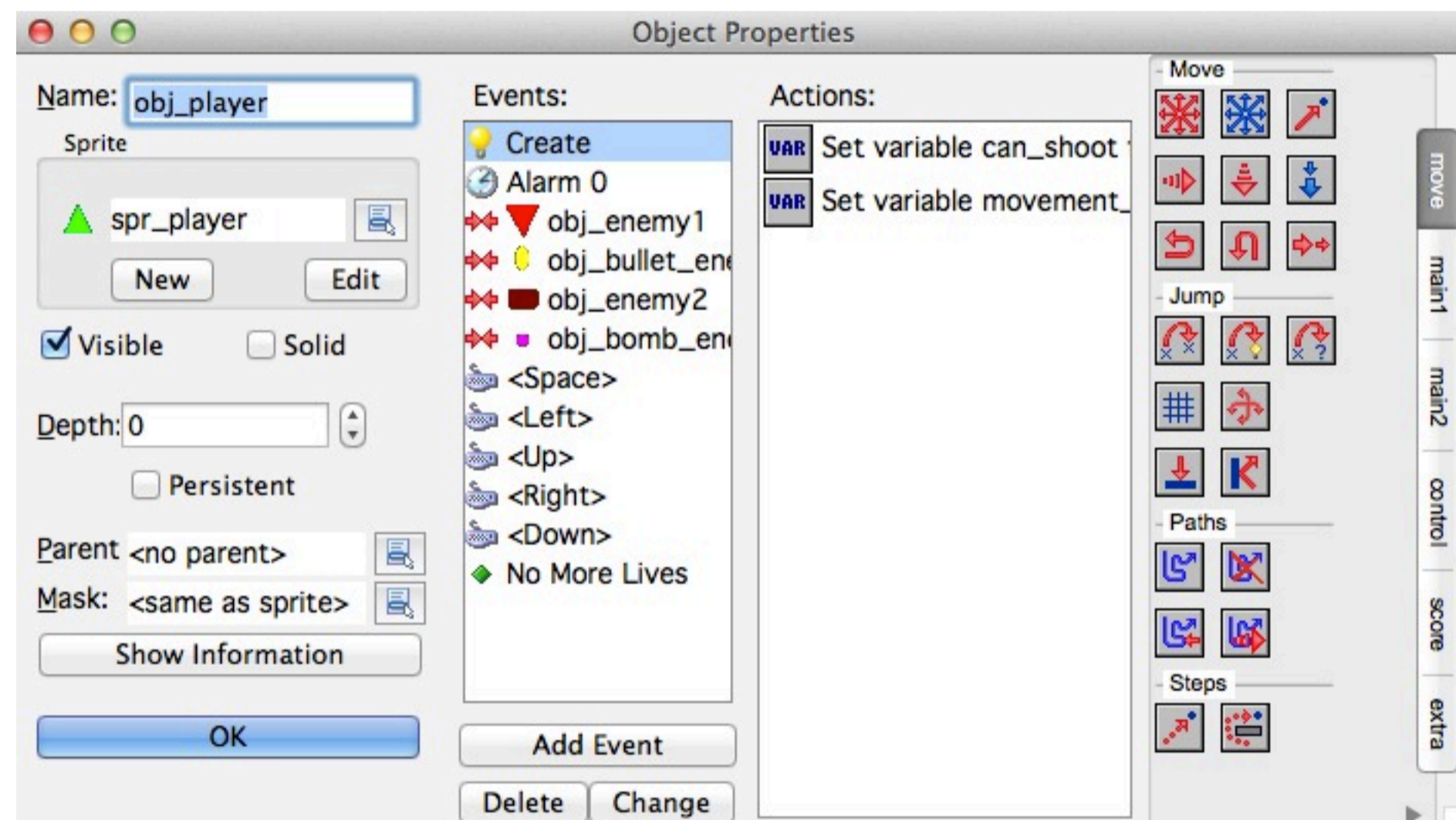
- Can create just one sprite or a series of images to form an animation
- Can click the file to make a new sprite from another image, or double-click an image to edit that sprite with the GM editor

# Sprites



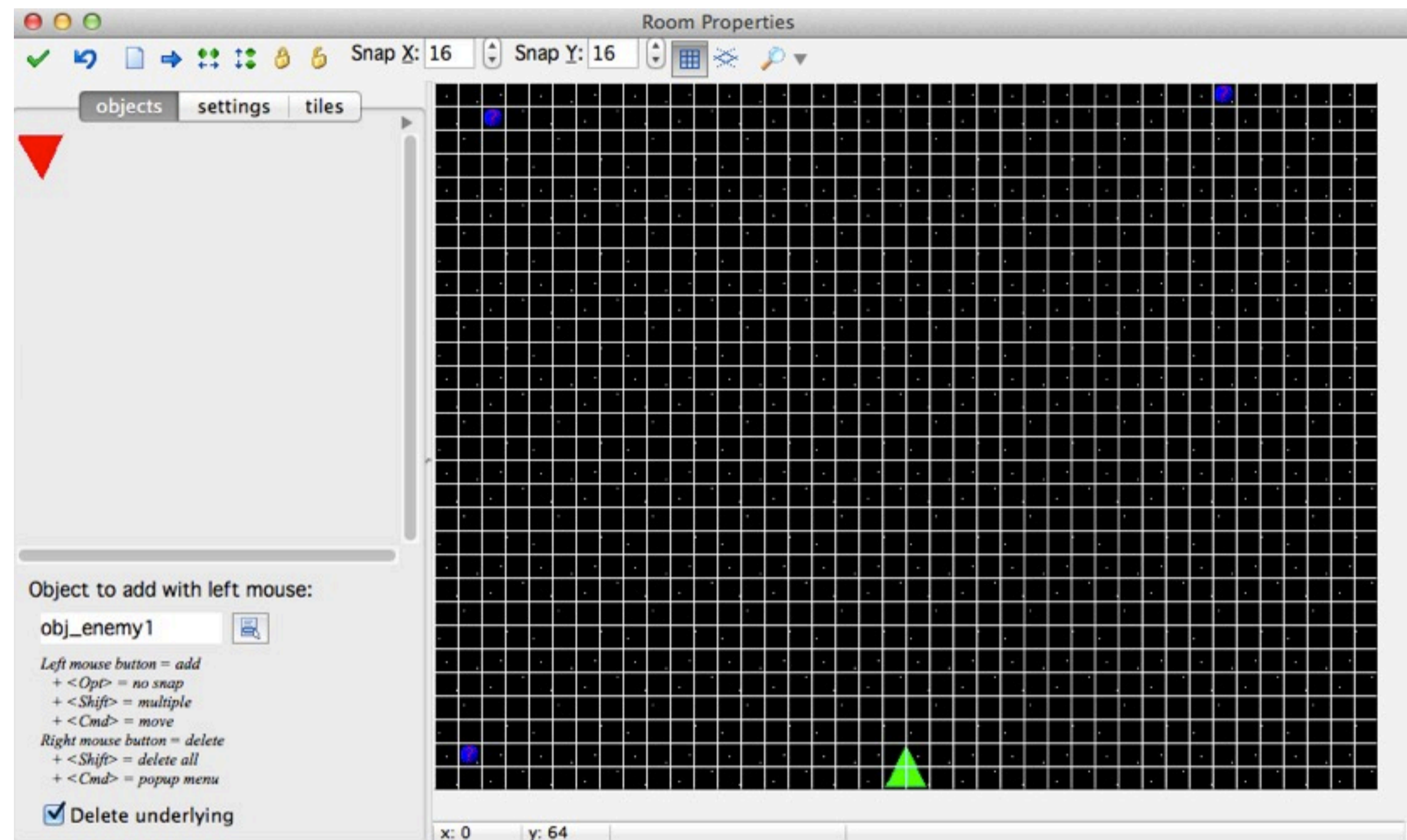
- Can edit sprite images in GM

# Objects



- Visible: can see object
- Solid: considered solid (you'll run into actions where it'll ask you whether it applies for all objects or just solids)
- Depth: deals with what overlaps; z-index
- Parent: inherit traits from another object
- Events: triggers that this object responds to
- Actions: what it does in response
- Persistent: lasts between rooms

# Rooms

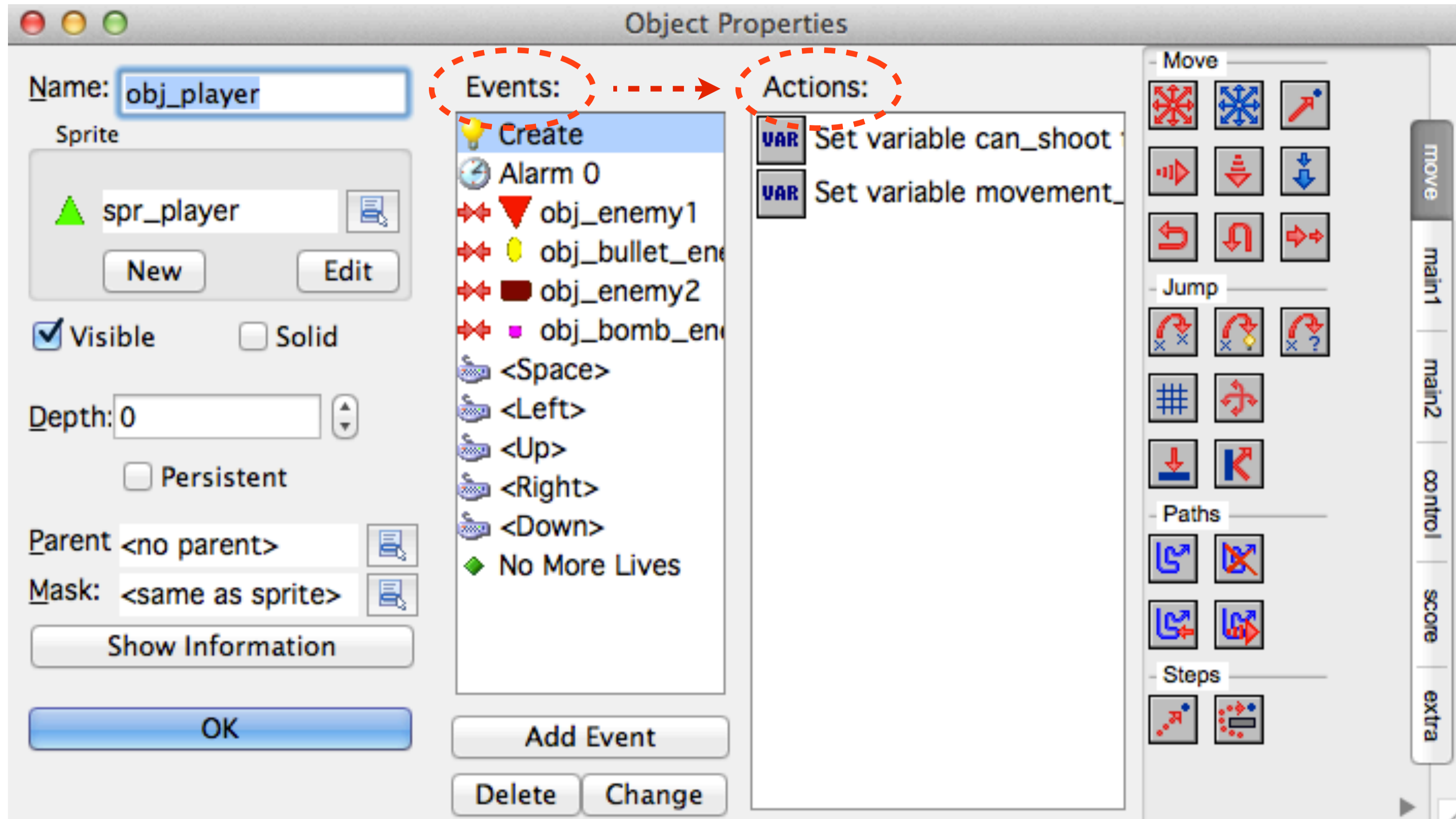


- Placing objects
- Setting backgrounds/background scrolls
- Assorted settings



How do I tell objects how to behave?





Tuesday, December 3, 13

GameMaker runs on an event-action system:

- Can tell the object what events it should pay attention to
- When those events happen, GameMaker will automatically run the actions that you associate with those events

# Events



- CREATE: the moment the object is created (instantiated) in-game
- DESTROY: the moment the object is destroyed in-game
- ALARM: the moment a timer that you set goes off
- STEP: each moment in game-time, by default every  $\sim .03s$  ( $1/\text{room\_speed}$ )
- COLLISION: when the object collides with a certain thing
- KEYBOARD: the duration from key press to key release

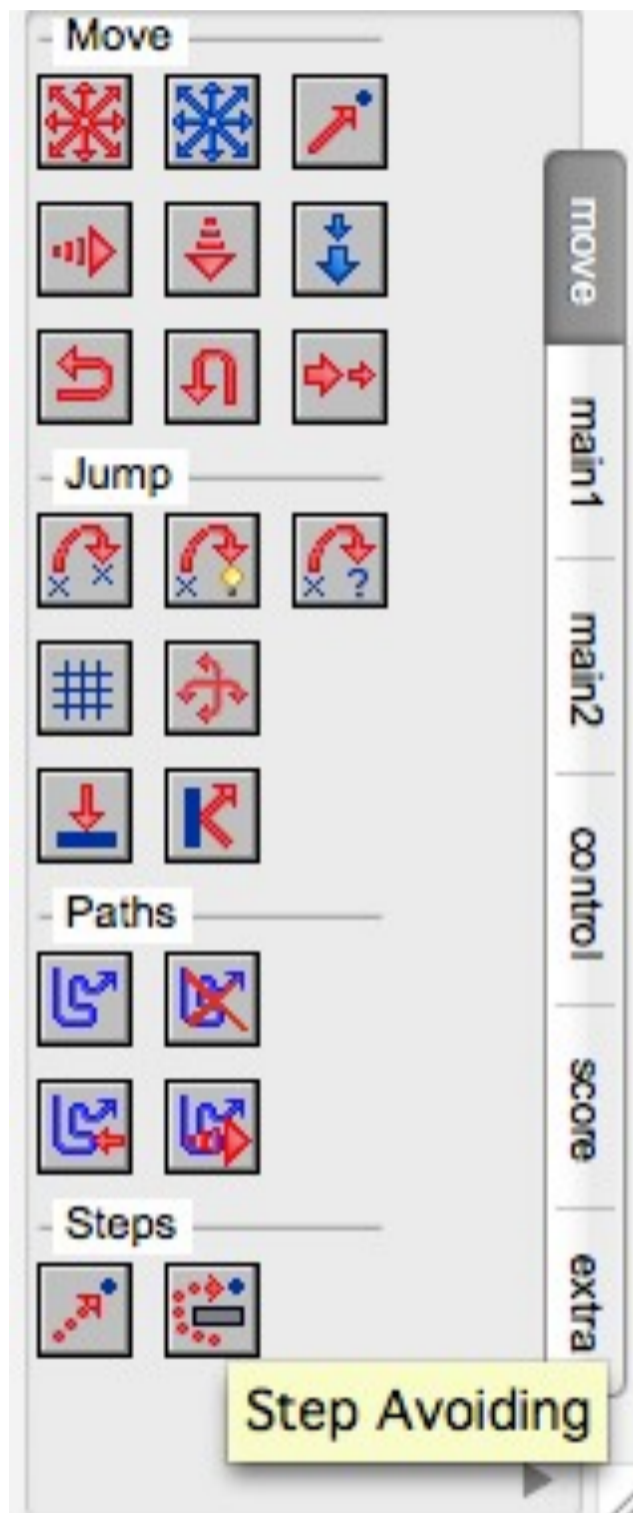
# Events



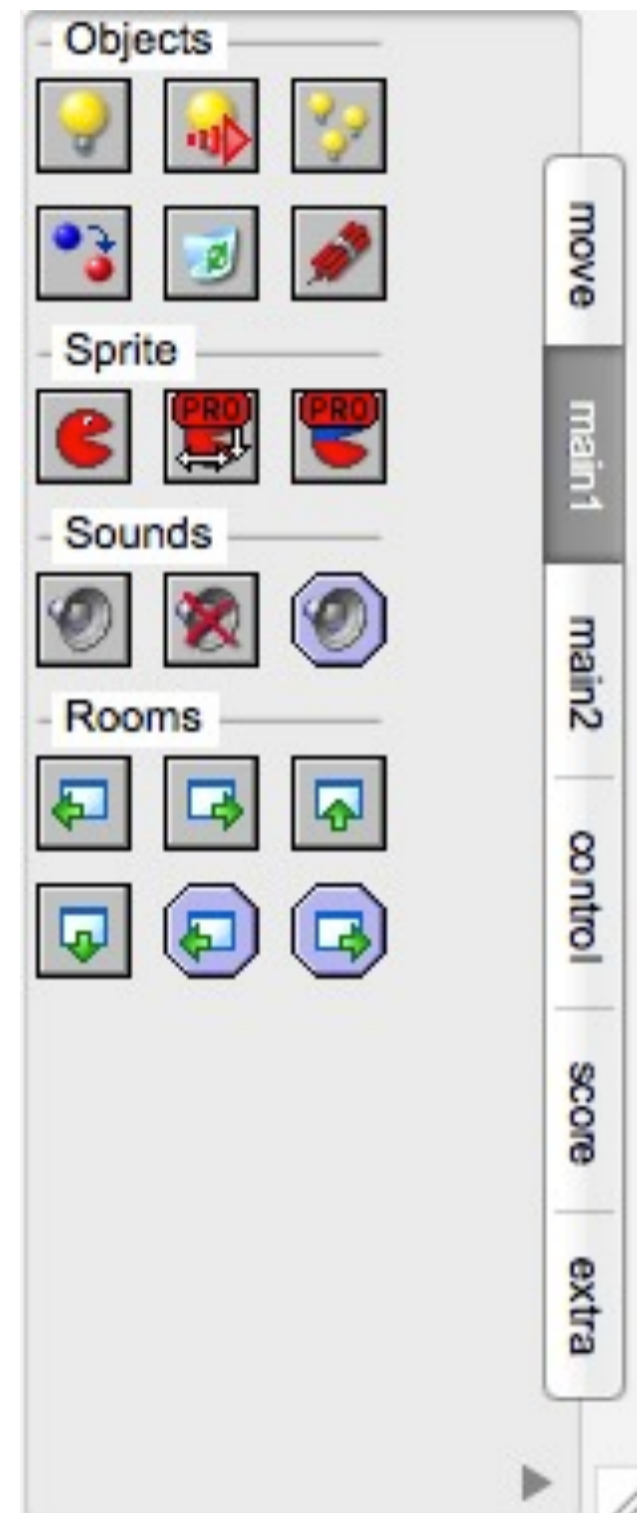
- **MOUSE:** when the mouse is pressed, released, etc. (Assumes on that object; global = anywhere.)
- **OTHER:** assorted other conditions
- **DRAW:** whenever GM is about to draw something. (Like step but for draw commands.)
- **KEYPRESS:** the single moment a key is pressed
- **KEYRELEASE:** the single moment a key is released



# Actions



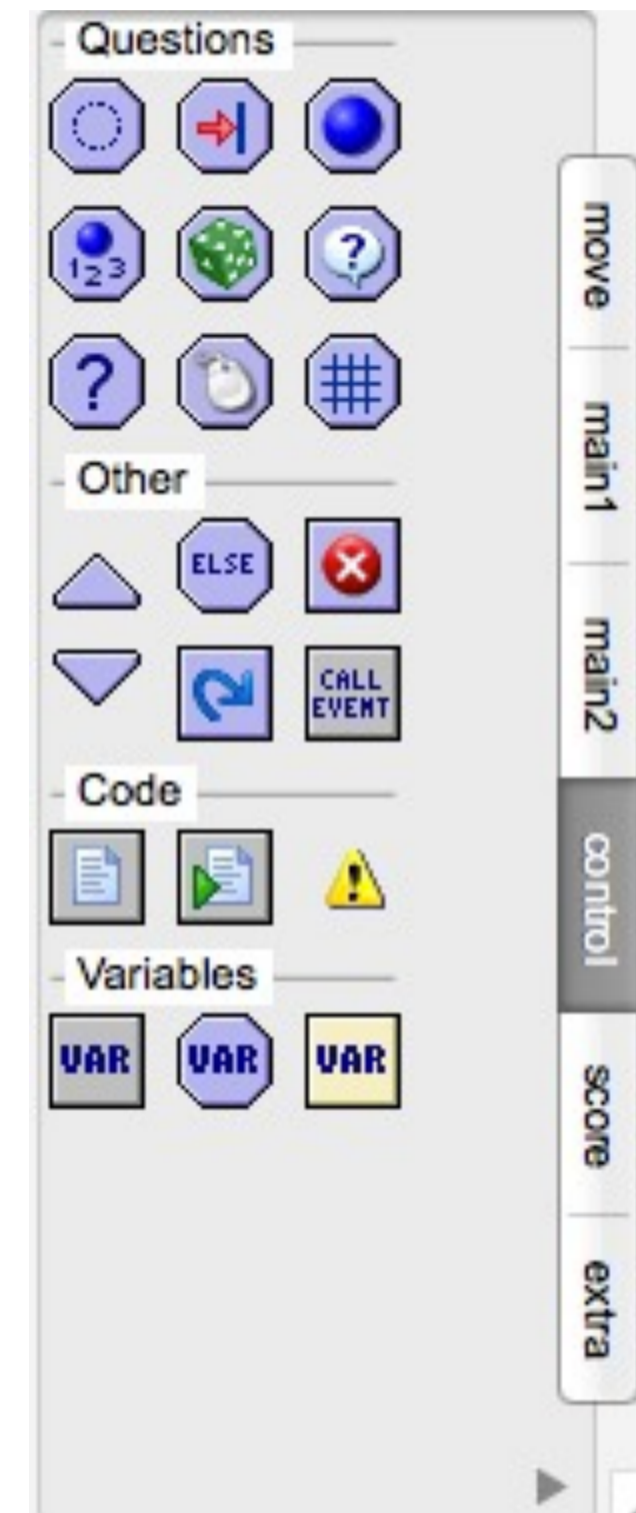
- Movement



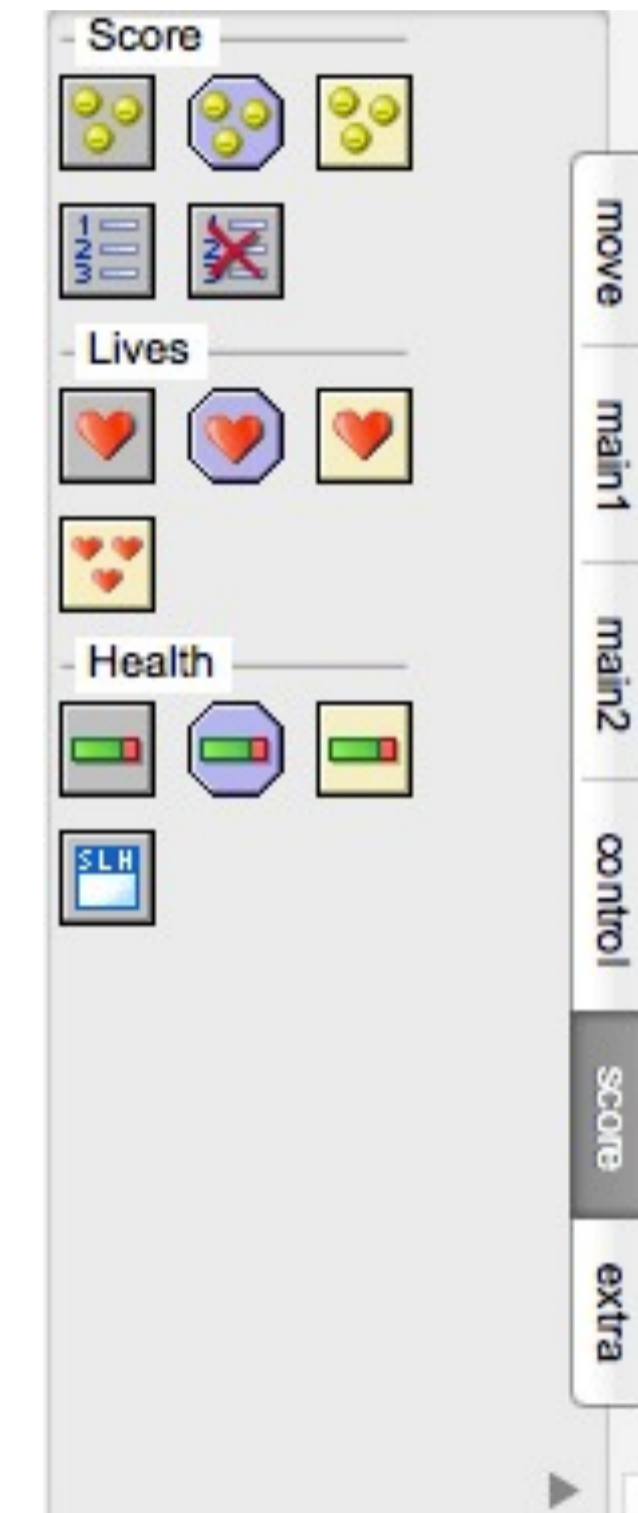
- Spawn things  
- Rooms  
- Sounds



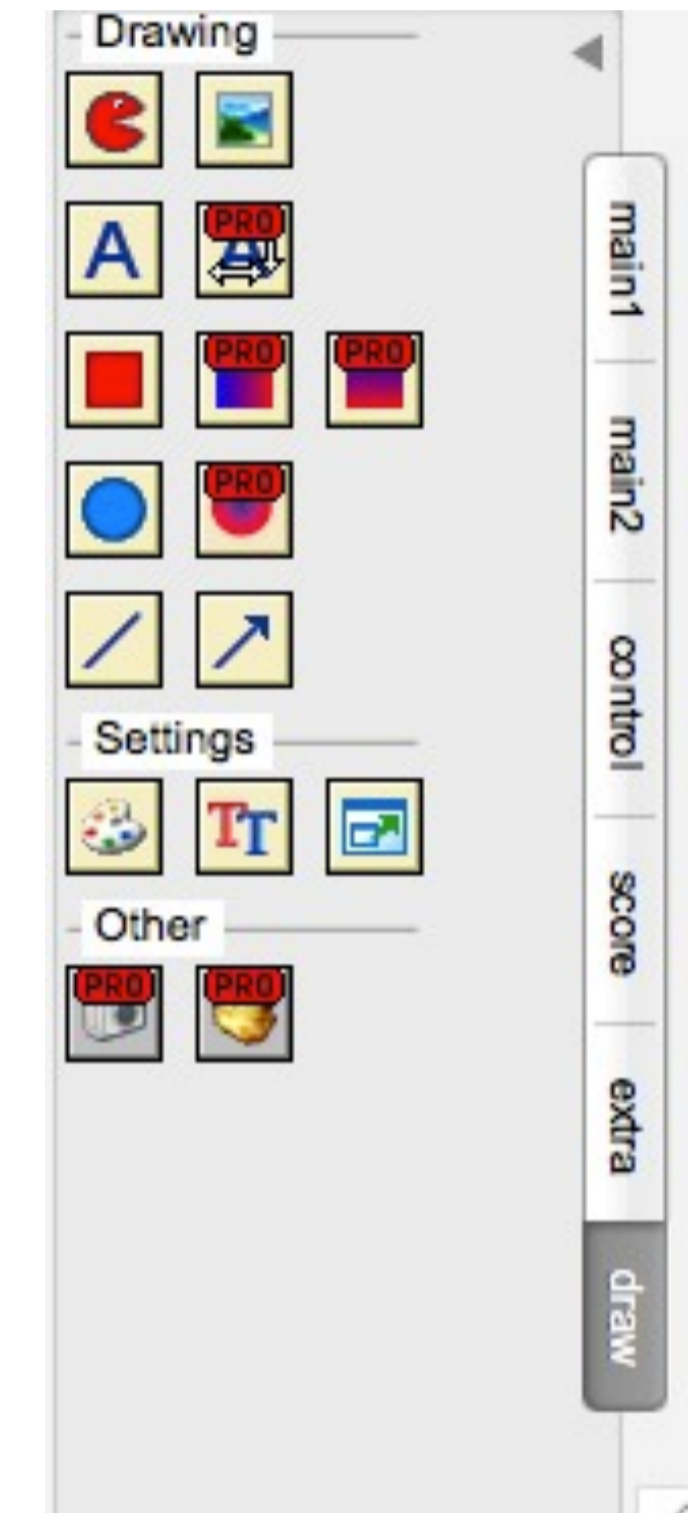
- Timers  
- Start/stop/  
restart



- Tests  
- Variables  
- Code/GML



- Score  
- Health  
- Lives



- Drawing

# How do I tell objects how to behave?

- Decide what event is important to you, and what should result
- Select the event, and then drag in the corresponding action(s)
- Can have many events and many actions per event

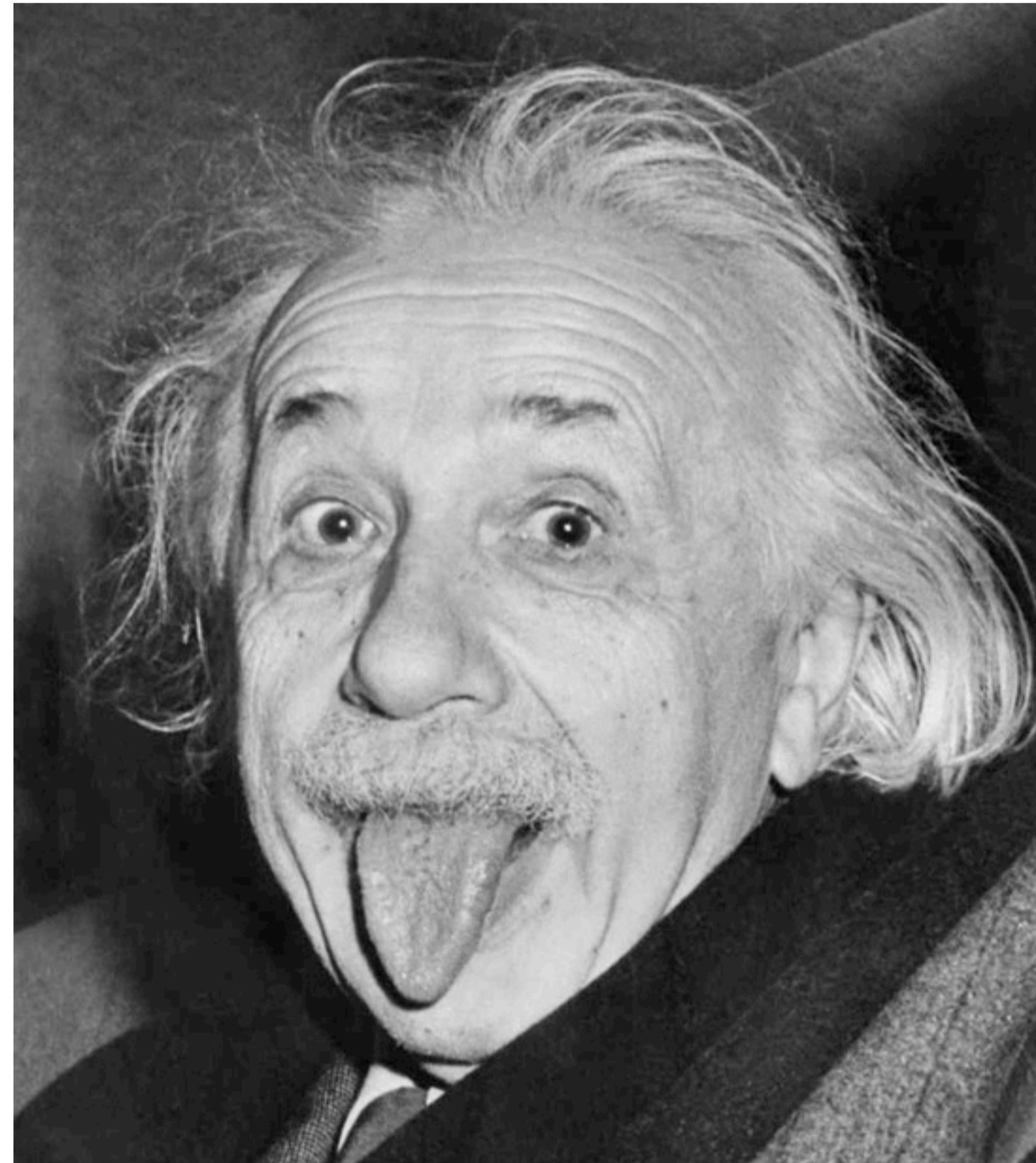
Question:

If I wanted to set the initial speed of an object right when it's created, what event would I use?

Bonus question:

If I wanted to store that in a new variable, can you find which action I'd use to do so?

# What's this 'relative' checkbox?



# What's this 'relative' checkbox?

- Generally means “in relation to the current value”
- Health example:
  - If you set health at beginning of the game, it ALWAYS equals, say, 3
  - If your ship crashes into an enemy at some point, what does it equal?  
How do you write that rule?
  - Can't say that health = 3 or 2 or any other literal
  - CAN say that we subtract 1 from the current value--new value is RELATIVE to the old one

## Question:

Let's say I want to program my object such that:

[Press l-arrow -> jump to position 3 pixels left of where I am now]

Do you think this is the kind of command that requires 'relative' to be checked? Why or why not?

(Don't worry about which action you'd use yet.)

[For those with programming experience:  
you can think of relative as +=, -=, \*=, etc.]

# Variables

- Variables are useful for storing data that **may change** throughout the course of your app (e.g. your player's health)

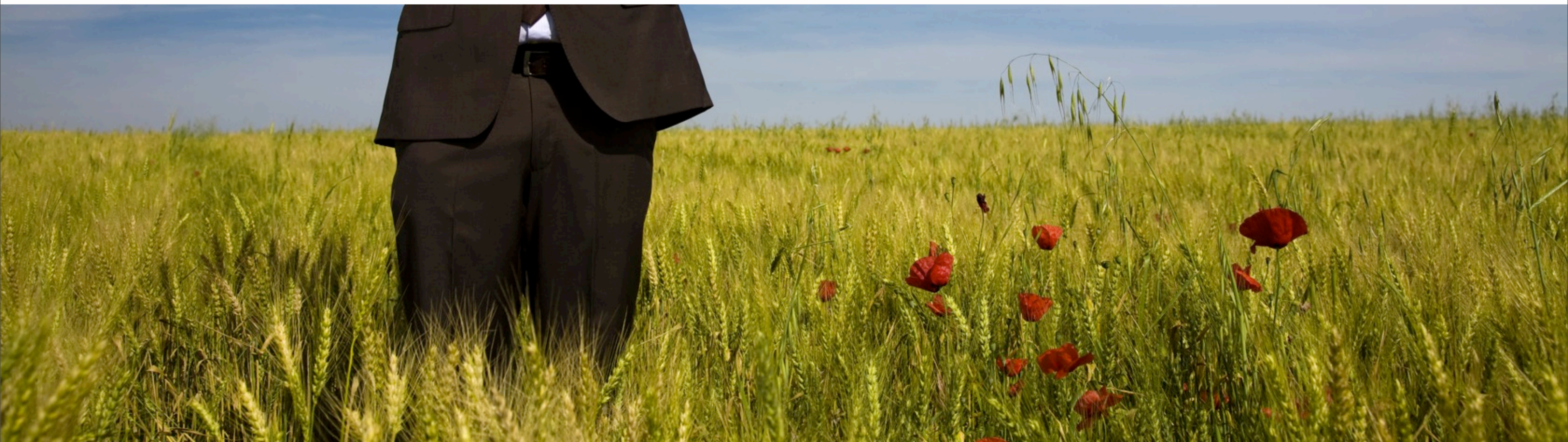


# Some Variable Types

- **Float:** a decimal number (“I’m 5.4 feet tall.”)
- **Integer:** a whole number (“I’m 25 years old.”)
- **Boolean:** a true/false condition (“I’m not from California.”)
- **String:** text (“My name is Jane.”)
- **Char:** a single letter (“You all get an A in programming!”)



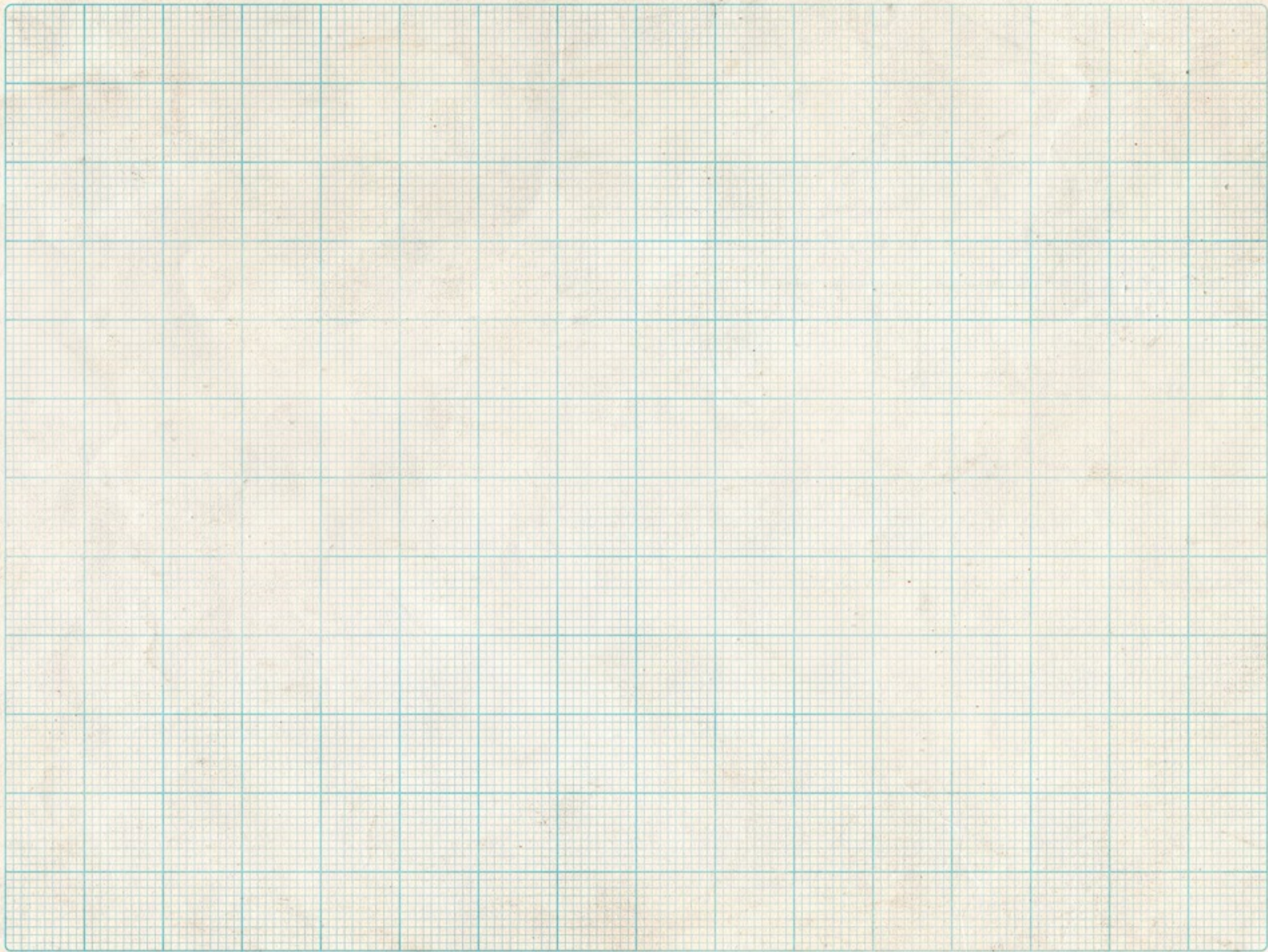
# How does positioning work?



Tuesday, December 3, 13

## HOW THE HECK DO YOU POSITION STUFF

- In order to draw something on screen, you have to tell the computer exactly where to put it



Tuesday, December 3, 13

### HOW THE HECK DO YOU POSITION STUFF

- You can think of the app window like a piece of graph paper, each pixel with its own coordinate location



$y=room\_height$

Tuesday, December 3, 13

HOW THE HECK DO YOU POSITION STUFF

- X: gets bigger as it goes to the right
- Y: gets bigger as it goes down
- Upper-leftmost corner is 0,0, bottom-rightmost corner is room\_width, room\_height



# Coordinate Plane

- **X**: horizontal axis, gets larger as you go right
- **Y**: vertical axis, gets larger as you go down

Question:

If I wanted a circle to move left, what value would I change--  
x or y?

Would I make it bigger or smaller?

What about if I wanted it to move up?

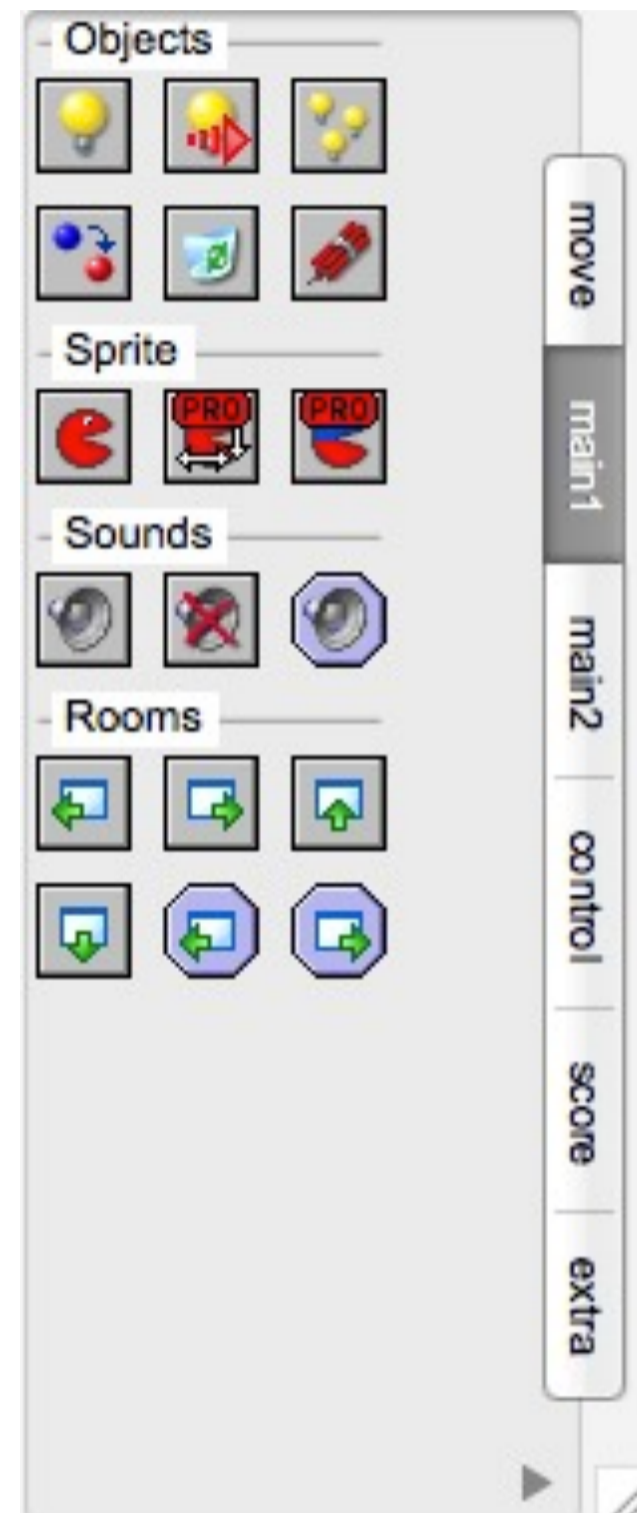


How do you make multiple rooms?





- GM will, by default, start with the first room on the list
- You can arrange your rooms in the order you want them...



- ...and then use the room commands to affect them (move to next, move to previous, restart, etc.)
- Can conceptualize start/end screens this way too

Question:

Let's say I wanted to make a start screen with a start button you click to go to the actual game.

How would I make this functionality?

Or, more specifically:

- Would the button need to be an object? Why or why not?
- Would we want the GLOBAL mouse event? Why or why not?