

# Key Abstractions in Game Maker

Foundations of Interactive Game Design  
Prof. Jim Whitehead  
January 19, 2007



Creative Commons  
Attribution 2.5

[creativecommons.org/licenses/by/2.5/](http://creativecommons.org/licenses/by/2.5/)

UC SANTA CRUZ



# Upcoming Assignments

- Today: Second gamelog assignment
  - ▶ Must write your gamelogs about a game from classics list
  - ▶ Have until 5pm today to complete
- Monday: Team selection for game design projects
  - ▶ Maximum of two members
    - ❖ Are working on getting a second XBox 360, if that's an issue for you
  - ▶ Single person teams are OK, but far more work
  - ▶ Need to turn in
    - ❖ Name of team
    - ❖ Names of team members
    - ❖ Two pieces of contact information for each team member, such as an email address or a cell phone number.
    - ❖ Two times during the week when the team can meet for at least one hour.
    - ❖ A location that can be used for meetings
- <http://www.soe.ucsc.edu/classes/cmpps080k/Winter07/>

# What is Game Maker?

- Game Maker is a tool mostly for creating 2D games
  - ▶ (though there is limited 3D support in the latest version)
- It uses a property-sheet approach to create a game
  - ▶ Create a set of sprites, objects, rooms, sounds
  - ▶ Each one of these has a series of properties that you can set – these affect its behavior
- Game Maker does not require programming
  - ▶ It does have a scripting language (Game Maker Language, GML) you can use for more advanced features
  - ▶ It's possible to make very interesting games without knowing this language
- Though, of course, if you do learn the language, you'll be able to make more complex games
  - ▶ Creating games is somewhat less tedious using GML

# Game Maker OS & Cost

- Game Maker runs on Windows PCs
  - ▶ No Mac support, sorry
  - ▶ Basic version is a free download
  - ▶ [www.gamemaker.nl](http://www.gamemaker.nl)
  - ▶ See “Tools” section of CS 80K website
  - ▶ Most students find they want and need the full version
    - ❖ This costs \$20, register via Game Maker website
    - ❖ Can reduce the cost with a bulk order
    - ❖ Have a signup sheet at front of class for those who are interested
  - ▶ Full version of Game Maker is also available in ITS PC Labs
    - ❖ UCSC has a site license
- Wait, I have a Mac! Don't do this to me!
  - ▶ CS 80K “Tools” page has links to “Power Game Factory” and “Unity”
  - ▶ See also <http://www.idevgames.com/>

# Game Maker Community

- A web site hosting discussion boards for users of Game Maker
  - ▶ <http://forums.gamemaker.nl/>
- Has tutorials/examples board
- A novice user Q&A board
- Postings of games created by people who used Game Maker
- Pointers to third-party libraries to add neat features
  - ▶ Sophisticated particle systems
  - ▶ Lighting models

# Simple Ball Example

- In-class demonstration of using Game Maker to create the simple ball game
  - ▶ This example is described on pages 9-10 of the Game Maker manual
  - ▶ It is highly recommended that you also follow this example and create the simple game
  - ▶ Provides a relatively gentle introduction to how to use Game Maker

# Sprites

- Sprites are the visual representation of objects in the game
  - ▶ A sprite can be a single image that never changes
  - ▶ You can also animate a sprite by having a sequence of images display in a row
  - ▶ Show Pacman sprites in Game Maker
- To add a sprite:
  - ▶ Menu Add -> Add Sprite



Source: [www.molotov.nu](http://www.molotov.nu)

# Sprite Issues

- The “Transparent” choice when creating a sprite indicates whether the background color of the sprite should be transparent
  - ▶ Most of the time, this is the correct choice
  - ▶ Only want your item to show, not a square with the item inside it
- Exception might be when you have a wall
  - ▶ Even then, might take advantage of transparency to create interesting effects
- Background color is the leftmost, bottommost pixel of the image



Source:  
[www.molotov.nu](http://www.molotov.nu)

# Objects (Briefly)

- Objects add behavior to Sprites
  - ▶ A Sprite just tells you what an item looks like
  - ▶ A Sprite conveys no information on how it works in the game world
  - ▶ Does it move, bounce, explode, etc? A Sprite doesn't say anything about these issues – Objects do
- Every character, monster, wall, item, ball in the game has its visual representation made with a Sprite, and its behavior determined by an Object
- **Sprite**: visual appearance only
- **Object**: behavior

# Basic Model of an Object

- An **Object** reacts to **Events** by performing one or more **Actions**
  - ▶ For example, a *ball* reacts to a *collision* with a *wall* by *bouncing*
    - ❖ Ball and wall are objects
    - ❖ A collision is one kind of event
    - ❖ Bouncing is one kind of action
- **Event:** a notable occurrence in the game
  - ▶ An object hitting another object
  - ▶ A mouse click on an object
  - ▶ An object being created
  - ▶ A clock tick
- **Action:** an activity that can occur in the game
  - ▶ Bouncing off a wall
  - ▶ Starting to move in a particular direction

# Important Events

- **Create**

- ▶ When an object is created
- ▶ Useful for setting the initial motion of an object

- **Collision**

- ▶ When two objects collide, each object receives this event
- ▶ Useful for collisions with walls, enemies, bullets, etc.

- **Step**

- ▶ The game progresses in “ticks”
- ▶ Each tick is 1/30th of a second
- ▶ Receive a step event each tick
- ▶ Useful for updating game state, checking for whether the player is close to another object, and so on

# Rooms

- A room is where the action takes place in a game
- Can represent many things:
  - ▶ Opening screen
  - ▶ Fields of play
    - ❖ Levels, dungeons, rooms, villages, outdoors, etc.
  - ▶ Help screens
  - ▶ Cut scenes
- Typically each level is a separate room in Game Maker
- Class demos have been taking place in default room
  - ▶ Automatically created by Game Maker

# Gathering Keypresses

- Three events control key input
- **Keyboard Event**
  - ▶ Generates an event as soon as a key is pressed
  - ▶ Also continues to generate events while the key is kept pressed
  - ▶ Good for repeated firing of a weapon
  - ▶ Generally not good for movement, or jumping
- **Key Press Event**
  - ▶ Generates one and only one event for a key being pressed down
  - ▶ Good for changing direction of a player agent
  - ▶ Good for starting some action (movement) of the player agent
- **Key Release Event**
  - ▶ Generates one and only one event for an already down key being released
  - ▶ Good for ending something started with a Key Press event

# Demonstration of Key Press Differences

- *Demonstration with Game Maker, keypresses, and sounds*
  - ▶ Keyboard event will cause sound to be played repeatedly, while key is held down
    - ❖ Challenging to get just a single sound instance to play, as you typically get many keyboard events for pressing a key
  - ▶ Key Press event
    - ❖ Will cause “down” sound to play just once
  - ▶ Key Release event
    - ❖ Will cause “up” sound to play just once