Notes 20: UML for Game Design.

We could spend the entire quarter just on UML and OO analysis and design. This is intended only as a very brief intro. I have chosen just enough to give you a flavor of UML and OO analysis and design. For more info check out:

- UML Primer: http://www.vinci.org/uml/
- UML Distilled (a book):

  http://www.amazon.com/exec/obidos/tg/detail/-/0321193687/qid=1115059614/sr=8-1/ref=pd_csp_1/102-9221738-2149757?v=glance&s=books&n=507846

Class Diagrams

Class Name
Members
Methods/services

The three areas get filled in with just enough info to concisely describe the class. There are different levels of Class diagrams:

Examples:

BouncingBall_MovingMC
_x:Number // x-coordinate _y:Number // y-coordinate xv:Number // x velocity yv:Number // y velocity image:MovieClip // library name of MC width, height depth:Number // depth level

UpdateX()   // increments _x by xv
UpdateY()

_x, _y xv, yv rotation_speed   // + is clockwise, - is counter size   // 3== big, 2 == med, 1== small image   // library image name

Rotate()
IsHit()   // if size 3, break into 3 size 2, if size
    // 2, break into 3 size 1, if size 1, remove
UpdateLocation()   // move to new x,y coordinate

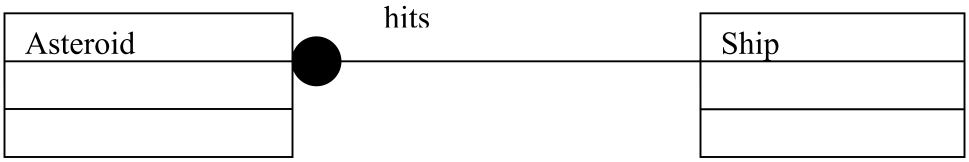
Ship
_x, _y currentSpeed shieldStrength   // 3, 2, 1. When zero ship destroyed image   // library name of MC

Thrust()   // check direction and adjust velocity
HitAsteroid() // decrement shield strength, destroy if 0

The above examples give enough info to have a good idea how to implement. OO Modelers really recommend three levels of modeling:

1) Conceptual
2) Specification (interface level)
3) Implementation (full data types, constructors, members, etc)

I like to do conceptual design with details that are not obvious to me thrown in. Classes also have associations with other classes.



How to Come Up With Initial Needed Classes:

1) Using “Things”
   - Identify individual or groups of things
   - Identify corresponding object

2) Pure Intuition

3) Using Nouns and Verbs
   Nouns become classes and members, verbs become methods (services)

   Example:

4) Model after previous approach (Design Patterns, past work)

   - Event Handlers:
     i. Mouse Listener
     ii. Keyboard Listener
     iii. OnEnterFrame

   - Keeping track of State:
     i. Scoring object
     ii. Timer object
     iii. Level/sprite counters

   - Implementation Details:
     i. Assets: which movie clips/graphics needed
     ii. Arrays?
     iii. Linked Lists?
     iv. Sounds?

Sequence Diagrams

Exercise 1:

Add in the mouse handler, keyboard handler , score manager, and sprite counters to the above Class Diagram?

Exercise 2:

In a group of 2 or 3, decide on a classic arcade game to design. Write a textual description of the game. Use this description to come up with an initial class diagram. Create an object diagram by using post-it notes to create instances of your objects. Move the objects around in to create scenarios that the objects need to interact. Write down sequence diagrams for these interactions. Does this modify your deisgn? If so, modify your design.