

Bangai HNC
Data Structures Design Specifications
Algorithms and Data Relation

Data Breakdown:

Entity – Super Class : Used by all players, monsters, items and projectiles

Sprite pointers (some composite entities will be supported)

Position, Frame

Player Number : -1 if not a player, otherwise the index of this player in the player list.

Think Function : called to give this entity new orders

Update Function : called to update the position and frame of this entity. Gonna be stock Update functions

Think Rate : how often we think

Update Rate : how often we update

Touch Function : We collided with something (as determined by Update), what now?

Health, Healthmax

move dir : Vector of movement

aim dir: Vector of aiming

Velocity : how fast we is movin

Max Velocity : How fast we can move

gravity : Are we effected by gravity or not

Other data needed to render all things in this game.

Bounding Box: For hit detection, players is smaller than it should be

Stand Level : At what point are we standing.

team number : used to prevent friendly fire and rockets from exploding before they leave the player

take damage : if it can take damage or not.

Player – Inherits Entity

Gun frame

Current Weapon : Pointer to weapon data

1st weapon : index of primary weapon

2nd weapon : index of secondary weapon

Inventory : Array indexed by the items in the game

Capacitance : How much ultimate charge we have for next Cell

Capacitors : How many fully charged cells we have for ultimate weapon

Cap Max : Total number of capacitors

Leg Unit : Index of Leg Unit equipped

Chest Unit : Index of Leg Unit equipped

Arm Units : Index of Arm Units equipped

Head Unit : Index of head unit equipped

Impact Armor: How much we absorb from physical damage

Energy Armor:How much we absorb from energy based damage

Score : how many things have you blownd up; bonuses for combos

Monster Data Type – instance of entity

Projectiles - instance of entity

- Bullets – Dumb Fire, small balls w/ small trail

- Rockets – Dumb Fire, wild fire and seeking w/ smoke trail

- Lasers – narrow & wide beams that reflect & cones that terminate. Particle trails

- Melee Blades – Sprite blades that appear near the player, particle trails.

Items – Data List referred to by index

- World Sprite : string

- Pickup Sprite : string

- bounding box : for touch purposes

weapons – Data list referred to by player via index

- player sprite : string path

- Fire Function : called when the player hits the attack button

- Ultimate Function : called when the player hits the ultimate attack button and has some capacitance.

- cool down : time between firings

- Ultimate cool down: The last star fighter , you are idle for a few seconds after an ultimate attack,

Tiles – used by maps for rendering and collision for reflection of lasers and bouncy missiles

- Block Angle : 0 – square, otherwise angled

- tile set : which tile is used to show this

- tile frame : which tile in the set is this one.

Map Objects – Entities that exist as part of the map.

- Think Function : if it needs one

- Touch : What happens when it touches something.

- Take Damage : some map objects can blow up

- Update : some map objects will need to be updated.

Particles – A superclass type that will be used for special effects. Particles will be used for explosions, engine thrust, projectile trails and lasers/beams.

- Position

- Sprite

- Color

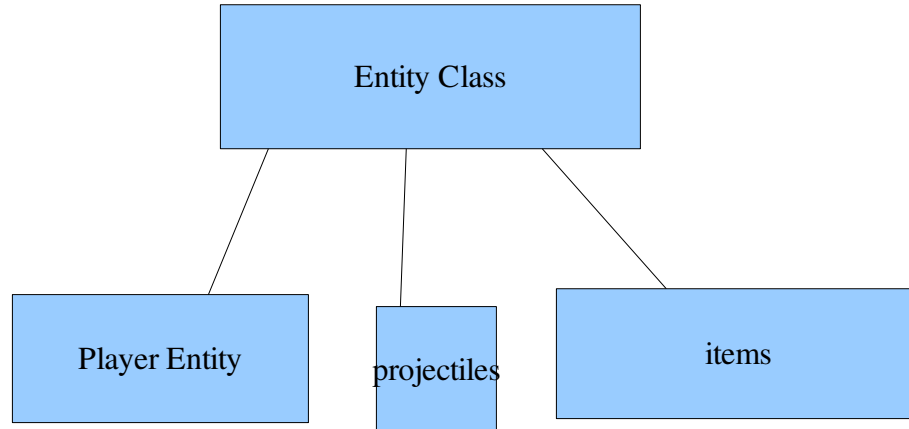
- color direction

- Velocity

- Acceleration

- Gravitate point

- Gravitate particle



Algorithms

The rocket launcher is going to take in a vector, spawn a rocket give it the vector and basic stats each frame it is going to update its' position and check for collision when it collides with something it will deal it damage and explode.

The explosion will damage anything in its' bounding box.