Dan Hervé
CSCD 349
Dan Tappan

Task 2

A_Entity

A_EntityAttacking   A_EntityTree

EntityGhost   A_EntityBird   EntityDog   Entity_Clam   EntityFish

EntityBirdAir   EntityBirdGround   EntityBirdWater   EntityTreeOak   EntityTreePine

1.

Attack Graph

EntityBirdWater

EntityBirdAir

EntityDog

EntityGhost

EntityFish
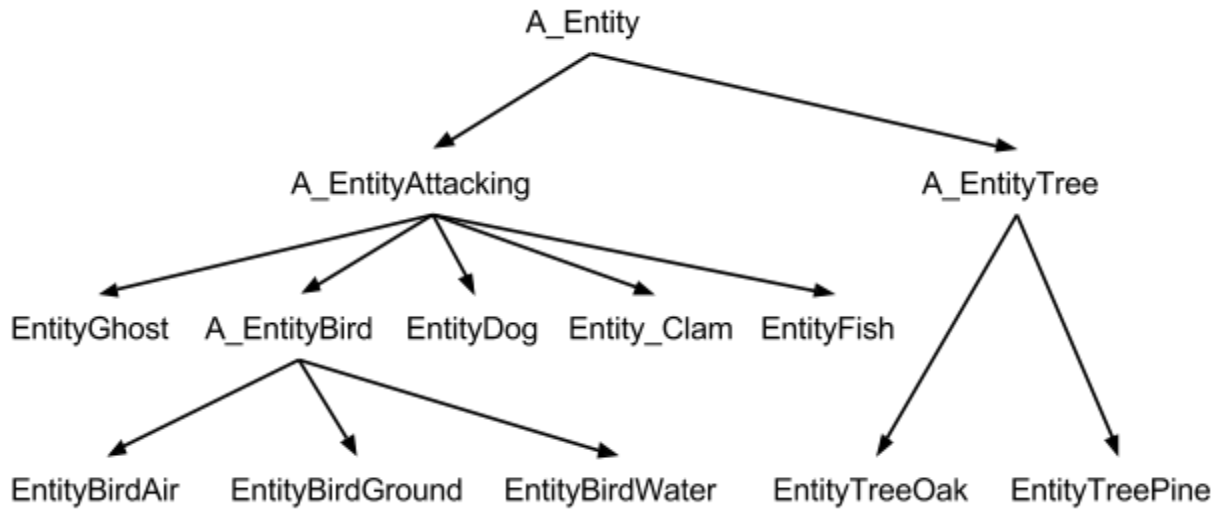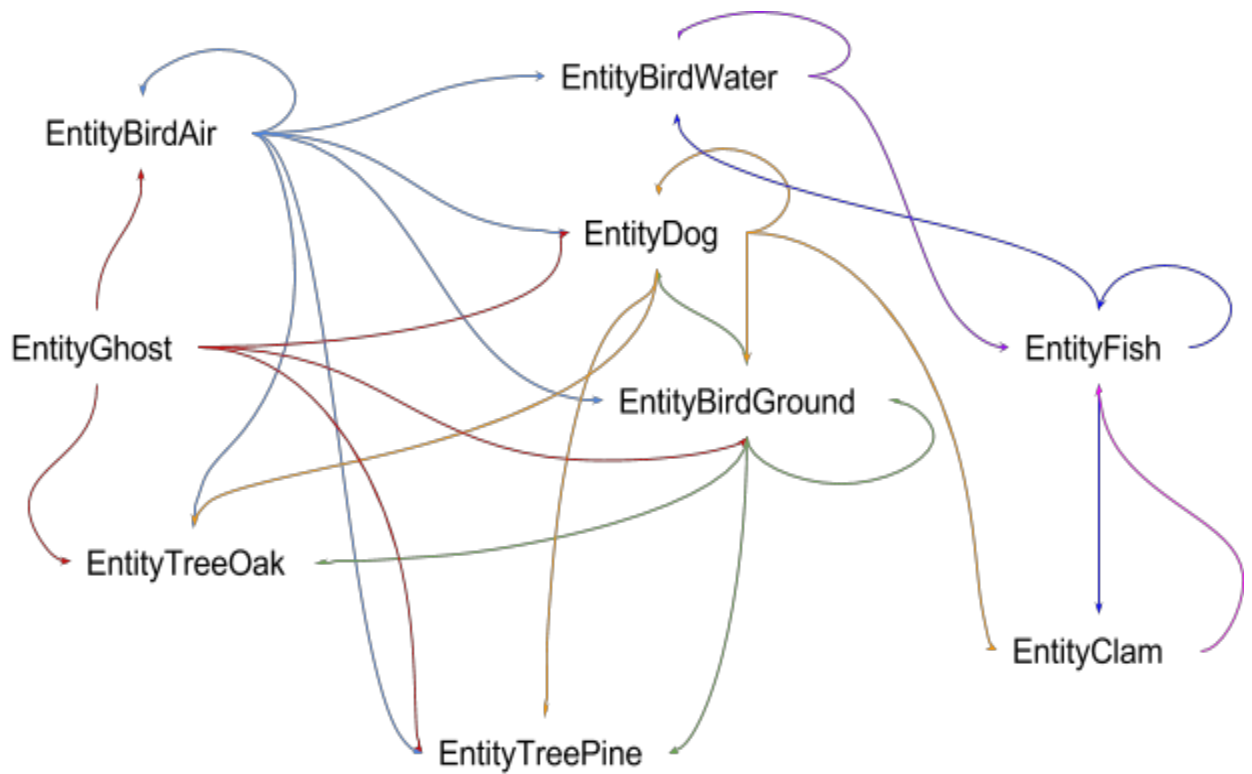
EntityBirdGround

EntityTreeOak

EntityClam

EntityTreePine

2.

Can Attack

3.

	EntityBird Air	EntityBird Ground	EntityBirdW ater	Entity Dog	Entity Fish	Entity Clam	Entity Ghost	EntityTree Oak	EntityTree Pine
EntityBirdAir	X	X	X	X				X	X
EntityBirdGround		X		X				X	X
EntityBirdWater			X		X				
EntityDog		X		X		X		X	X
EntityFish			X		X	X			
EntityClam					X				
EntityGhost	X	X		X				X	X
EntityTreeOak									
EntityTreePine									

4. There a number of assumptions this pattern seems to follow:

   4.1. Air entities can attack anything that is not ethereal nor that lives primarily under the water.

   4.2. Likewise, ghosts can attack anything air entities can attack excepting water birds since ghosts cannot travel under nor over water

   4.3. Ground entities that are not trees can attack other ground entities.

   4.4. Trees cannot attack anything

   4.5. Water entities can, in general, attack other water entities.

   4.6. Clams are a bit anomalous, since they are water entities, but can be attacked by dogs and fish and only attack fish.

5. See additional material

6. EntityBirdAir overrides the acceptAttackBy and releaseAttackBy methods from its super

   class(A_Entity) while EntityBirdWater does not. Also, they implement different interfaces.

7. By implementing different interfaces for each attacking relationship, the code checks for

   errors at compile time by making sure that the constraints created by the interfaces hold

   true.