

TEALS MINECRAFT PROJECT

Lecture 6: Basic Entities

A NEW MINECRAFT OBJECT: ENTITIES

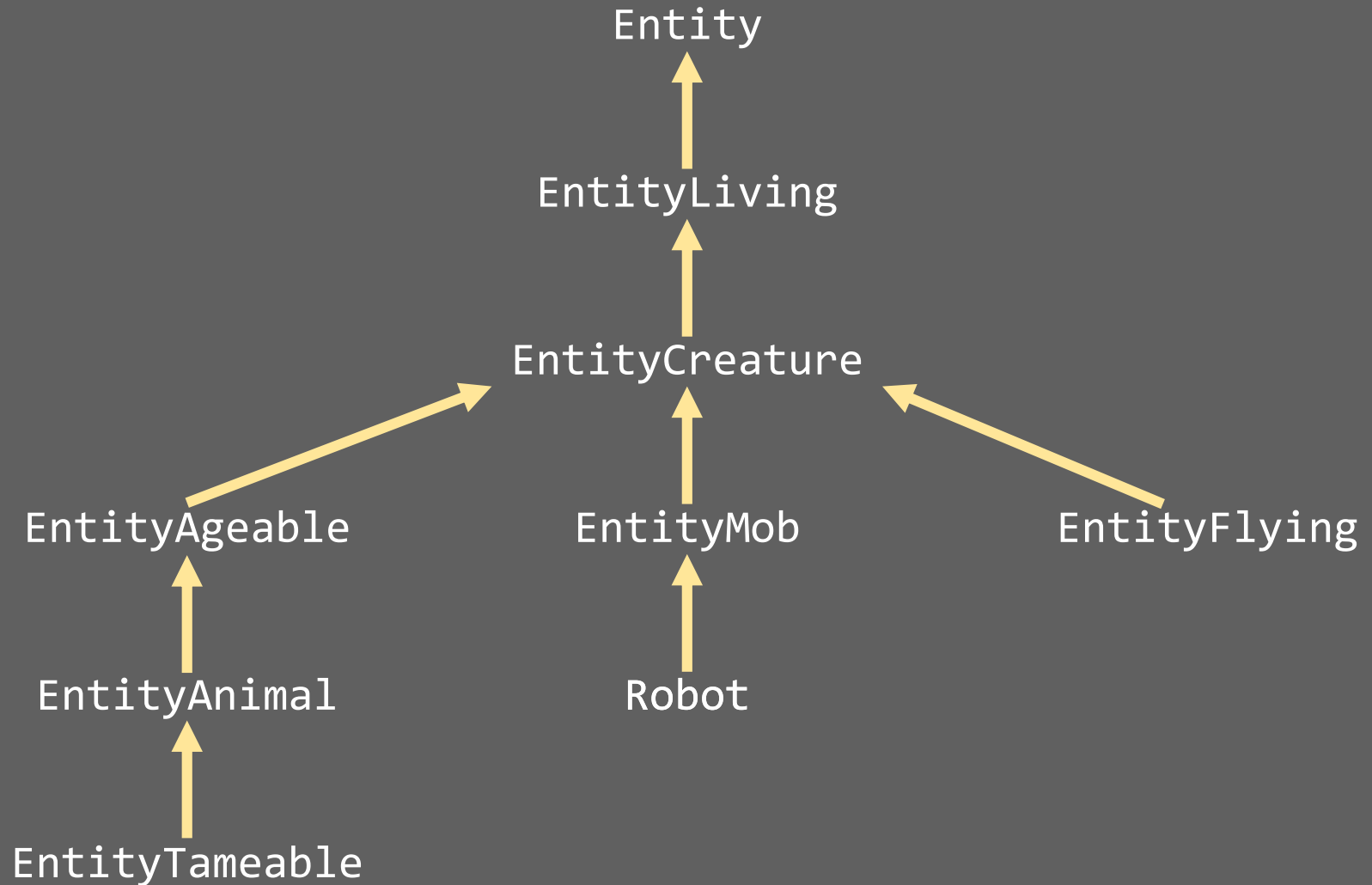
So far, you've learned about two different kinds of Minecraft objects: items and blocks. For the rest of this project, you'll work with *entities*.

Entities Have Unique State

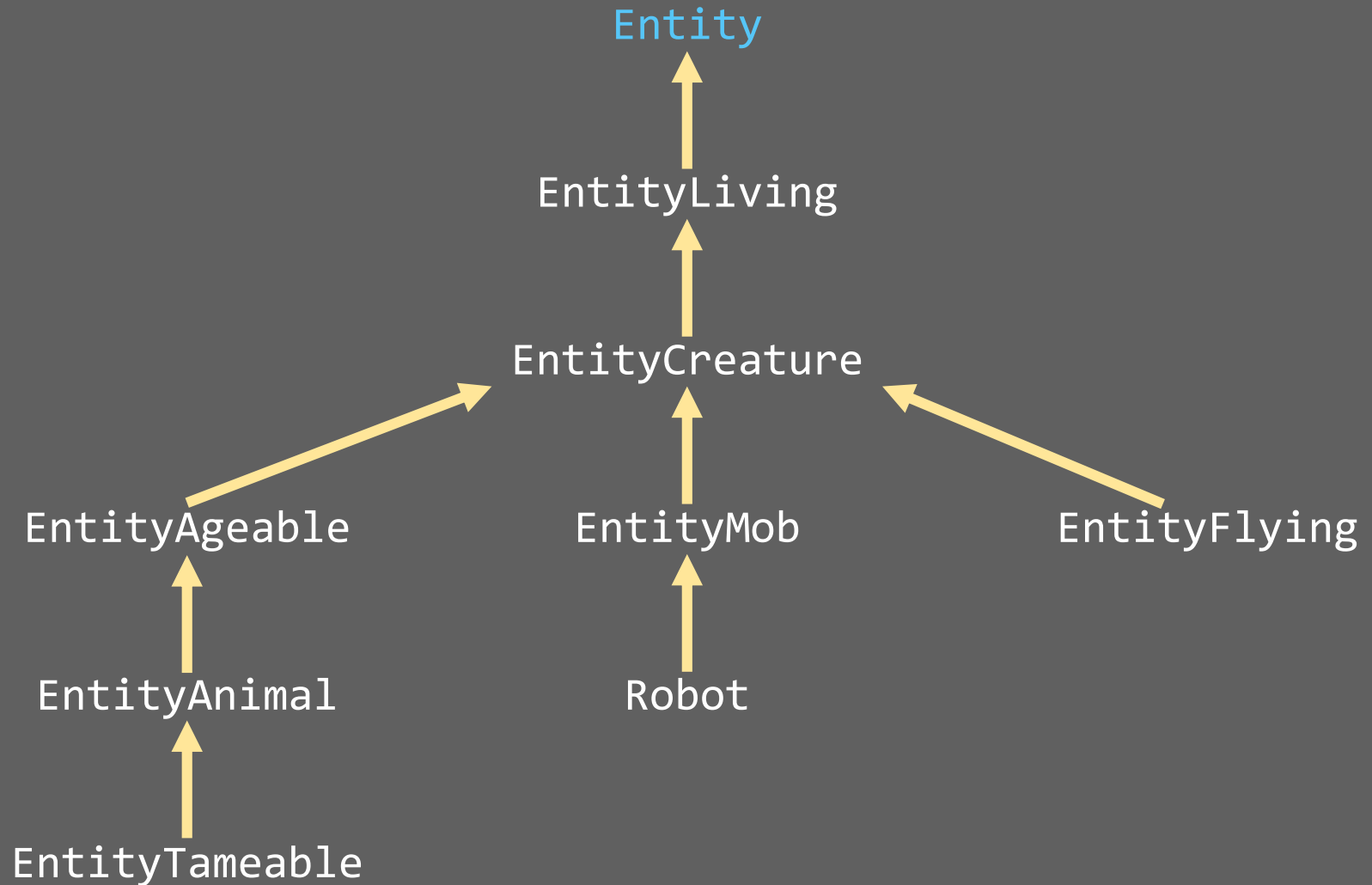
Each entity instance has its own state (health, items, damage, and so forth)

Even items are a type of entity (`EntityItem`)

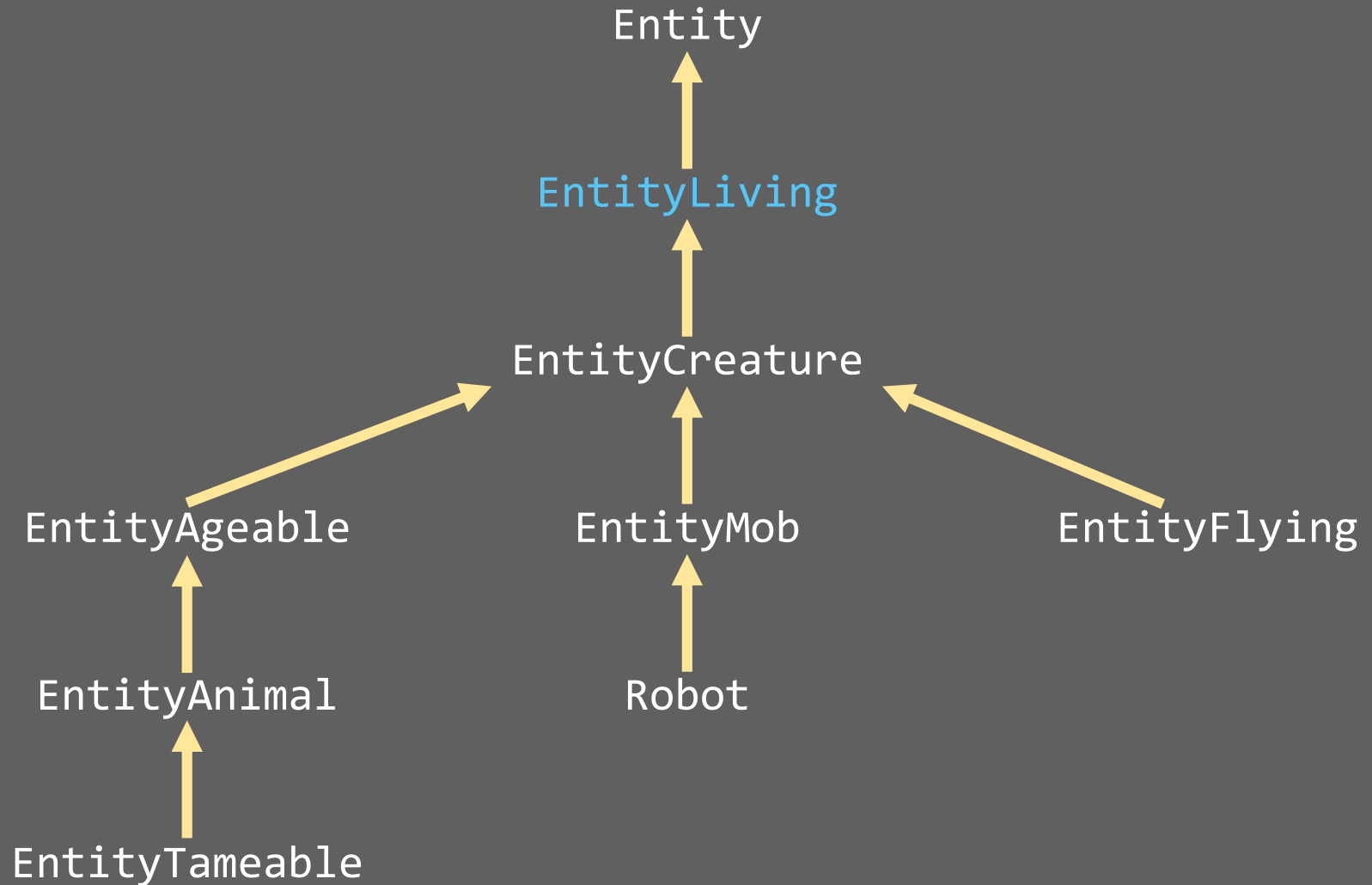
ENTITY BASICS – ENTITY HIERARCHY



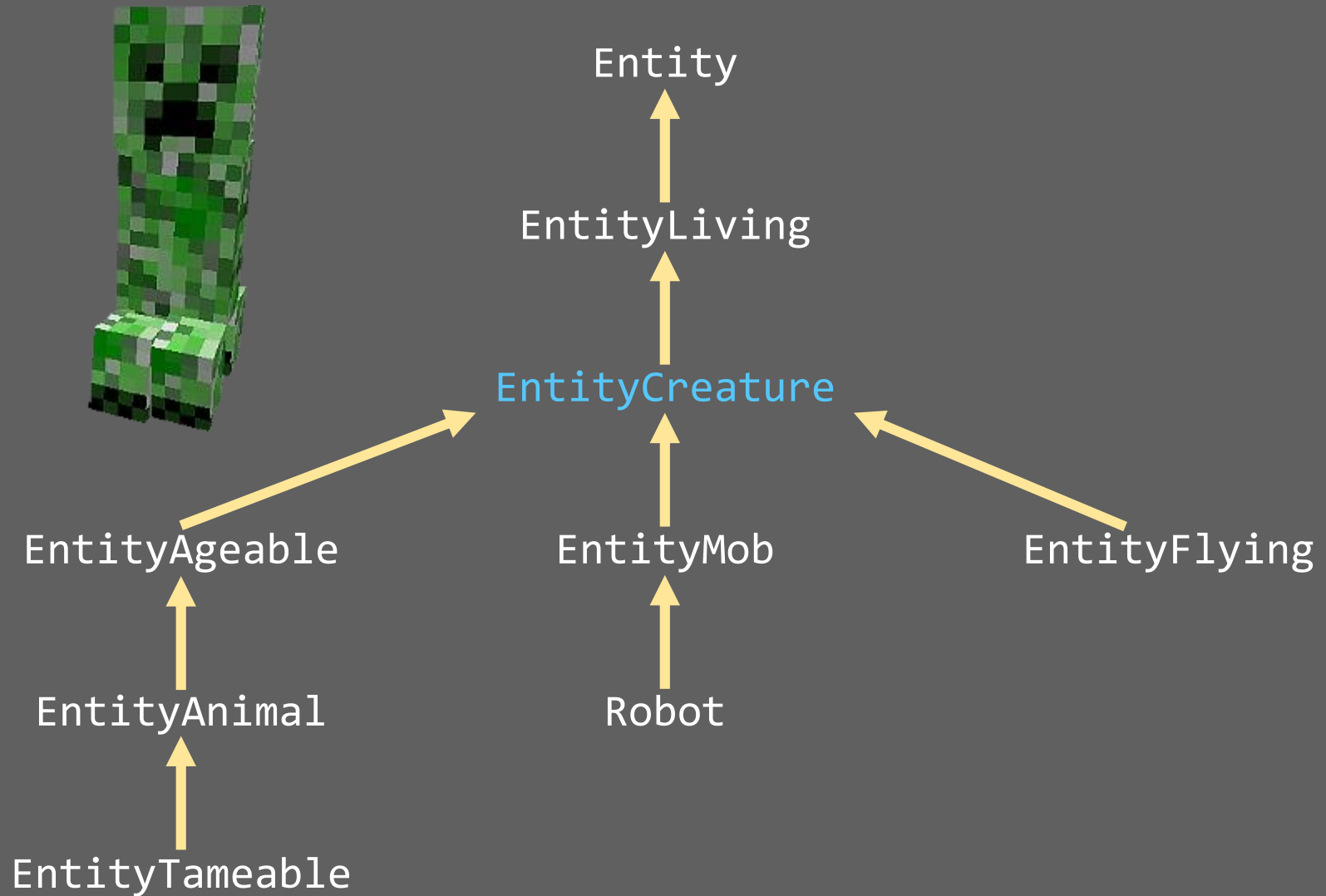
ENTITY BASICS – ENTITY HIERARCHY



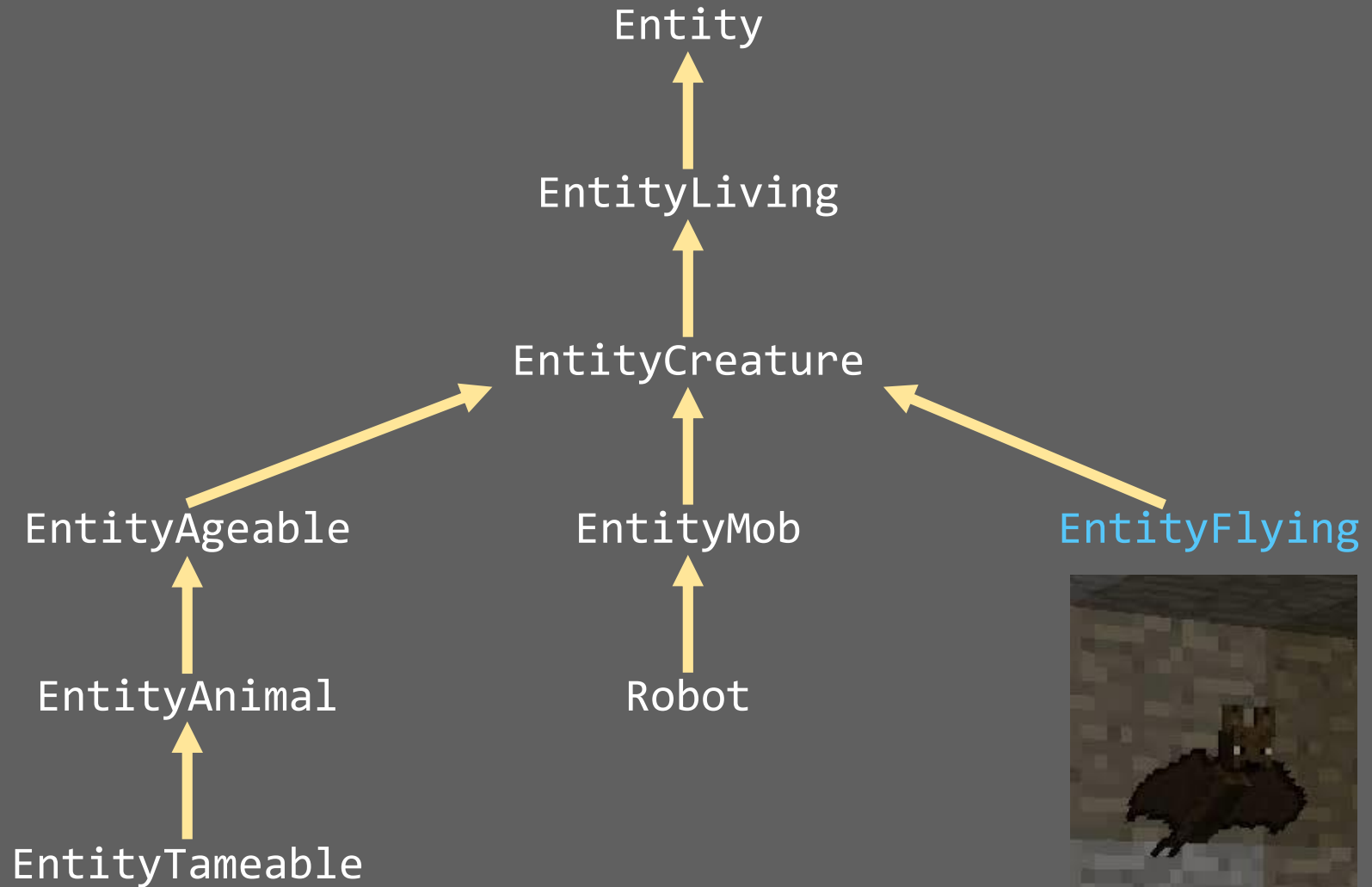
ENTITY BASICS – ENTITY HIERARCHY



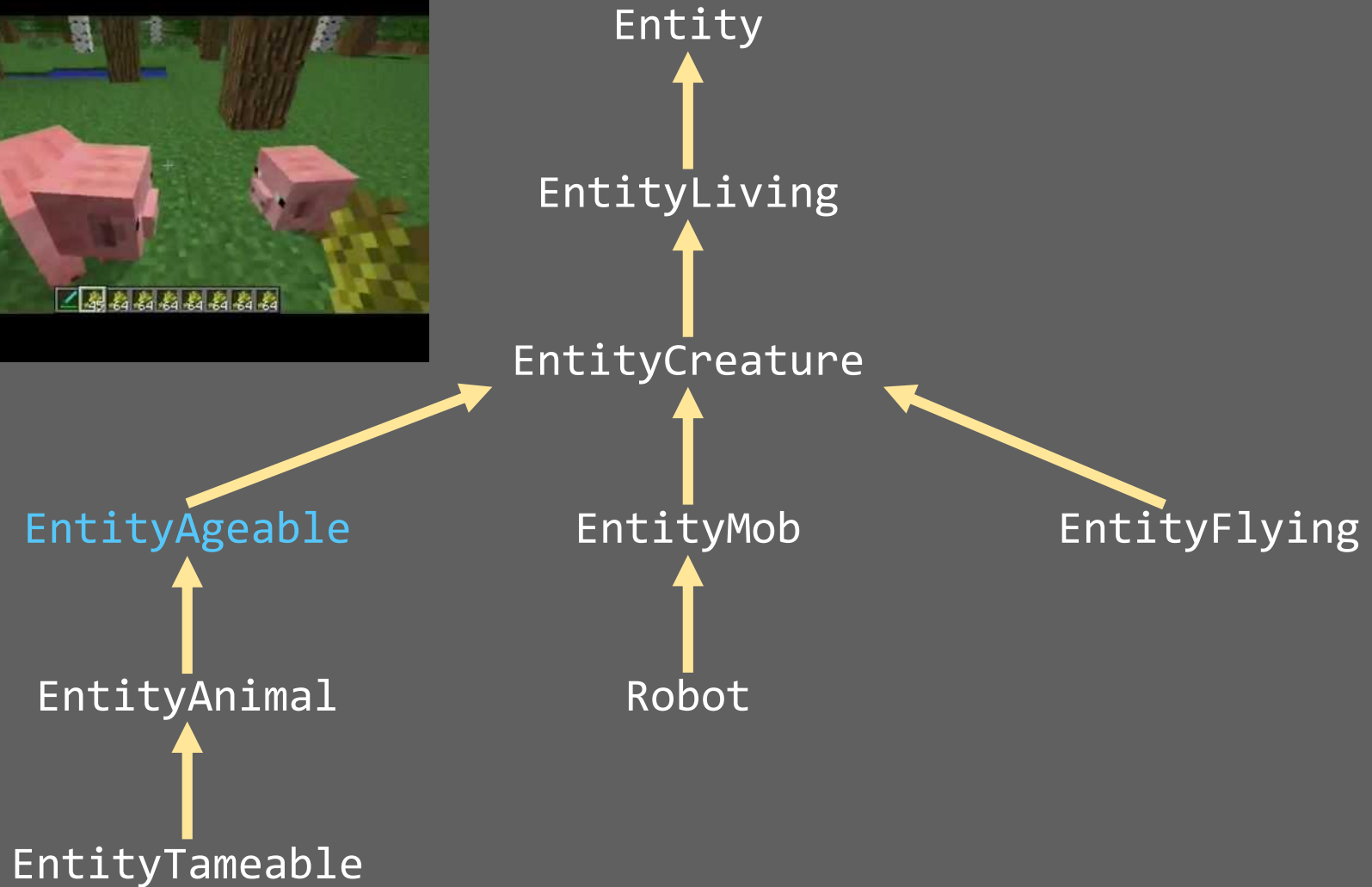
ENTITY BASICS – ENTITY HIERARCHY



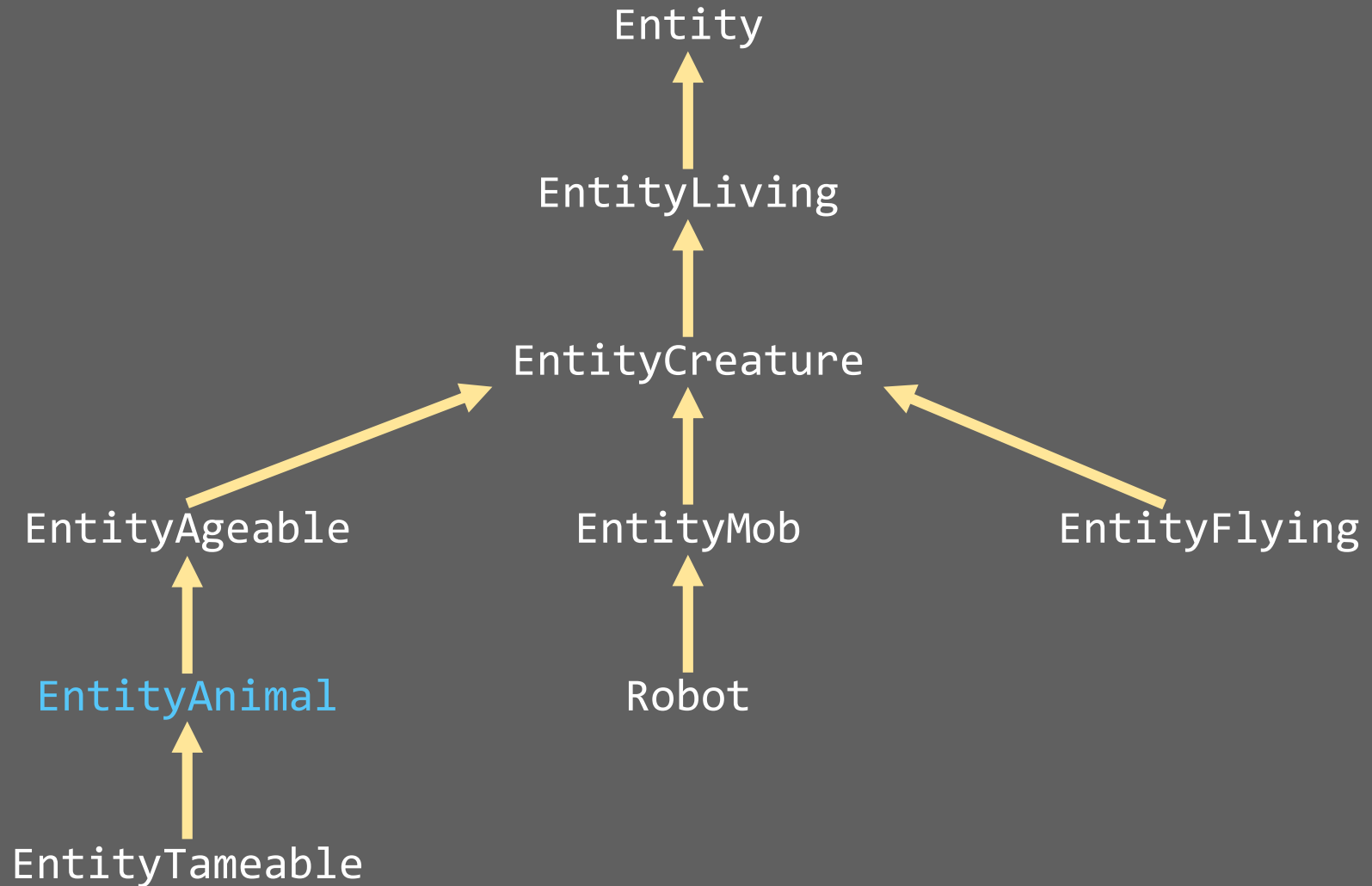
ENTITY BASICS – ENTITY HIERARCHY



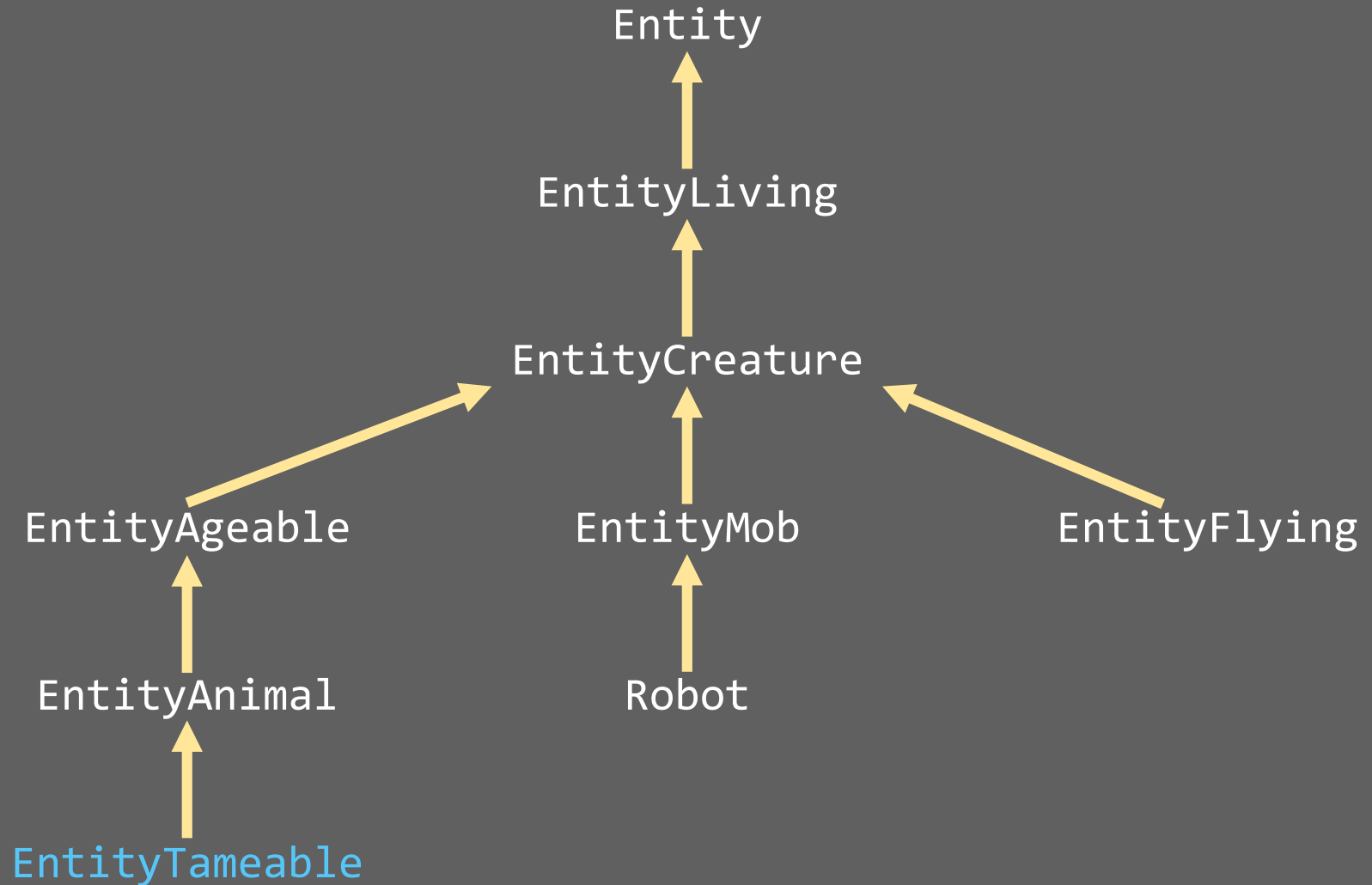
ENTITY BASICS – ENTITY HIERARCHY



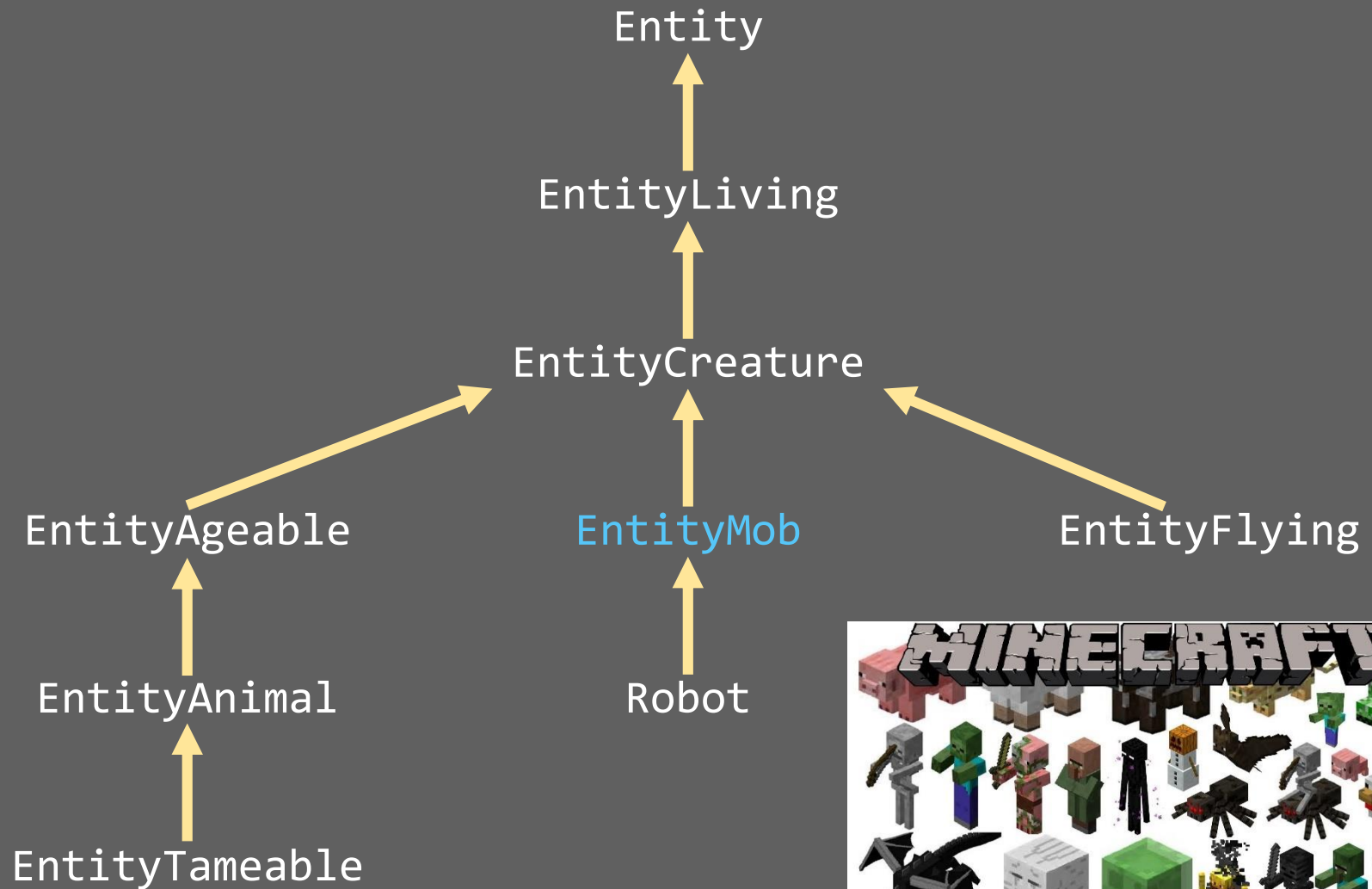
ENTITY BASICS – ENTITY HIERARCHY



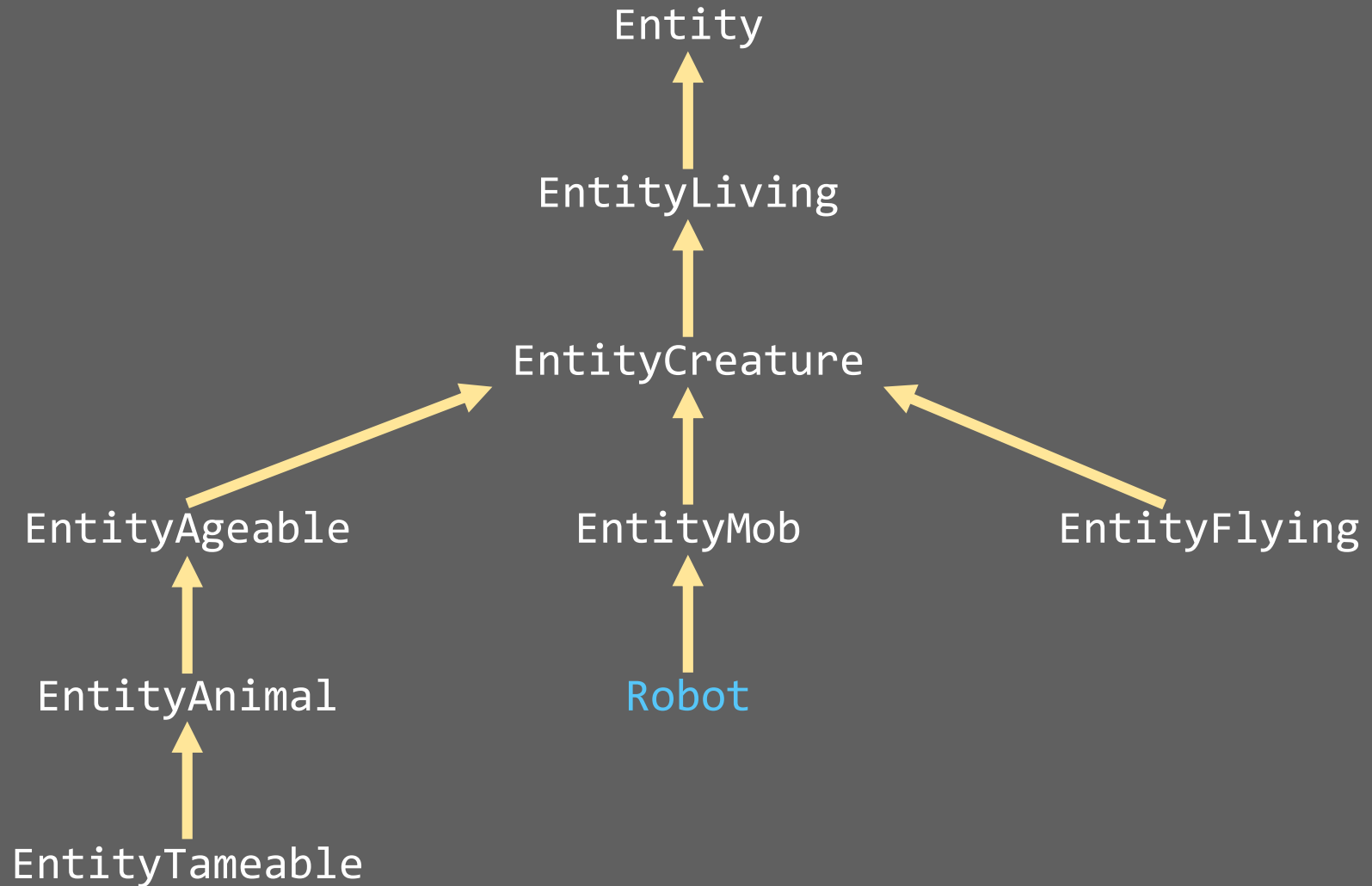
ENTITY BASICS – ENTITY HIERARCHY



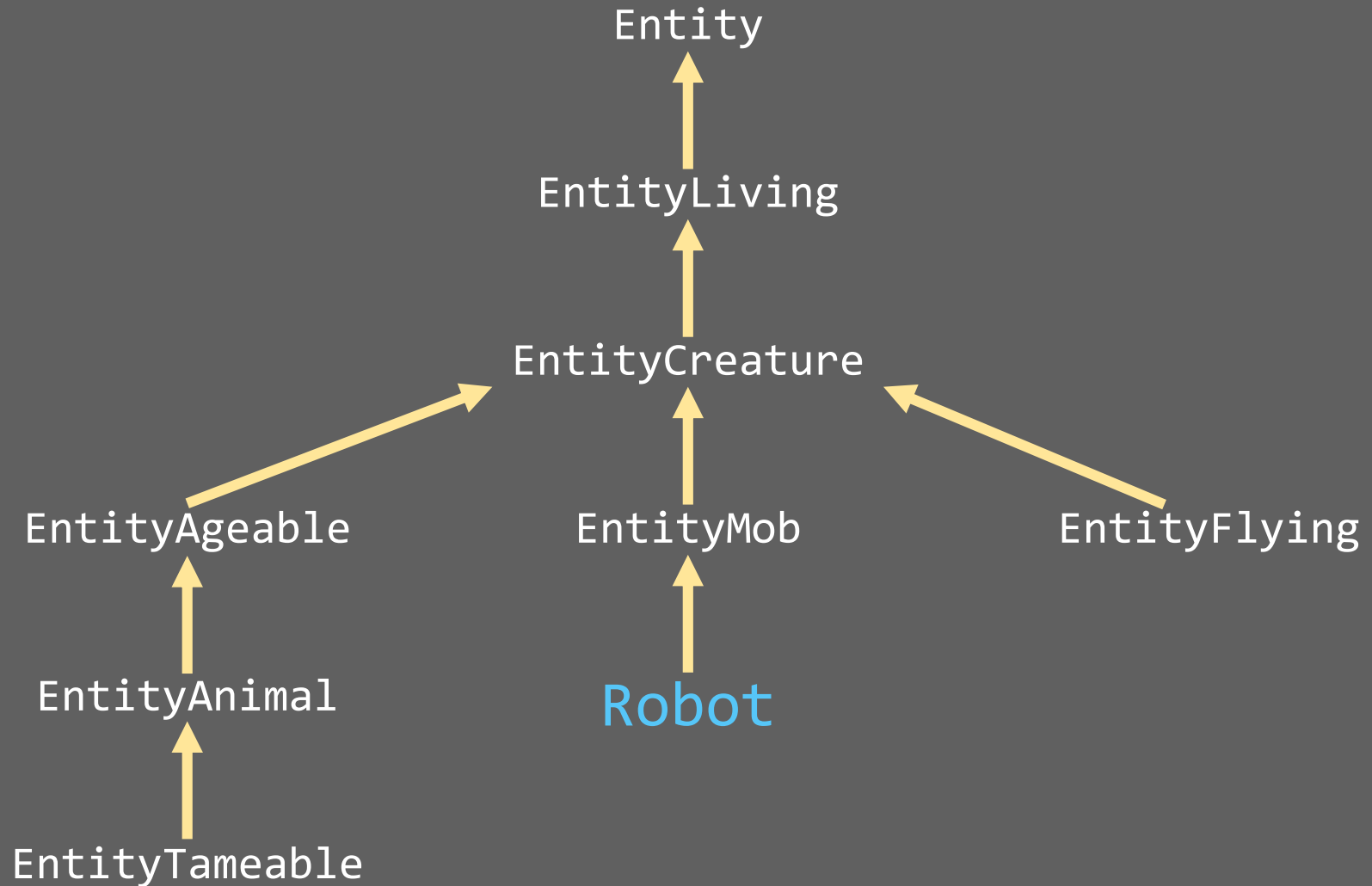
ENTITY BASICS – ENTITY HIERARCHY



ENTITY BASICS – ENTITY HIERARCHY



ENTITY BASICS – ENTITY HIERARCHY



EntityAITasks

EntityAITasks

- Add behaviors to EntityLiving entities

EntityAITasks

- Add behaviors to EntityLiving entities
- Derive from EntityAIBase

EntityAITasks

- Add behaviors to EntityLiving entities
- Derive from EntityAIBase
- Run with a specified *priority*

EntityAITasks

- Add behaviors to EntityLiving entities
- Derive from EntityAIBase
- Run with a specified *priority*
- Are named EntityAIXXX, where XXX is the description of the task

EntityAITasks

- Add behaviors to EntityLiving entities
- Derive from EntityAIBase
- Run with a specified *priority*
- Are named EntityAIXXX, where XXX is the description of the task

Example Tasks

- EntityAIWander — move from current location to a random spot occasionally
- EntityAI Panic — run around frantically for a period of time
- EntityAIFleeSun — try to move to a darker spot
- EntityAIWatchClosest — look at closest nearby entity
- EntityAIAvoidEntity — avoid another entity
- EntityAIEatGrass — find grass. Eat grass.

THE GAME LOOP

All tasks are executed in the "game loop".

The Minecraft game loop runs 20 times every second. Thus, one Minecraft "tick" is 50ms (50 milliseconds) long.

In one tick, every animated thing in Minecraft is updated.

THE GAME LOOP

If all tasks finish *before* the tick is over (less than 50ms), the game sits idle until the tick is done.

If all tasks take *longer* than 50ms to complete, then the world slows down as it struggles to get all the work done. Hopefully this rarely happens.

The game can address this situation by doing one of two things:

1. Reduce the number of entities and blocks loaded at any time
2. Increase the performance of the task code

This is why you often see the world fade in and out in the distance — the game is managing how many things it has to keep track of at the same time.

THE GAME LOOP

You can adjust some of the Minecraft world workload by starting up the game, and going into options, then video settings. How can changing these settings reduce the work done in a single tick?



EntityAITask FUNCTIONS

EntityAITask classes have four primary methods to implement:

```
// Returns true if the task should be executed.  
public boolean shouldExecute();
```

```
// Called when the task begins execution.  
public void startExecuting();
```

```
// Returns true if an already-executing task should continue.  
public boolean continueExecuting();
```

```
// Called once per tick (20 times per second)  
public void updateTask();
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```


EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

EntityAITask CODE

```
package tealsmc.anything;
import net.minecraft.entity.EntityLiving;
import net.minecraft.entity.ai.EntityAIBase;

public class EntityAIExampleTask extends EntityAIBase {

    private EntityLiving entity;    // Target Entity

    public EntityAIExampleTask (EntityLiving entity) {
        this.entity = entity;
    }

    @Override
    public boolean shouldExecute() {
        // This abstract method must be implemented.
        return true;
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving; {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```


USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving; {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

USING EntityAITask

```
package tealsmc.anything;

import net.minecraft.entity.EntityLiving;
import net.minecraft.world.World;

public class ExampleEntity extends EntityLiving; {

    public ExampleEntity (World world) {
        super(world);

        tasks.addTask (1, new EntityAIExampleTask(this));
    }
}
```

OTHER ENTITY FIELDS

```
// Entity World Position  
double posX, posY, posZ;
```

```
// The world in which the entity is located  
World worldObj;
```

```
// Whether the entity is dead  
boolean isDead;
```

```
// Whether the entity is in water  
boolean inWater;
```

OTHER ENTITY METHODS

```
// Drops items on the ground at the entities location.  
EntityItem dropItem (Item item, int amount);
```

```
// Sets the entities location to the specified x, y and z.  
void moveEntity (double x, double y, double z);
```

```
// Resize the entity if it's resizable.  
void setSize (float width, float height);
```

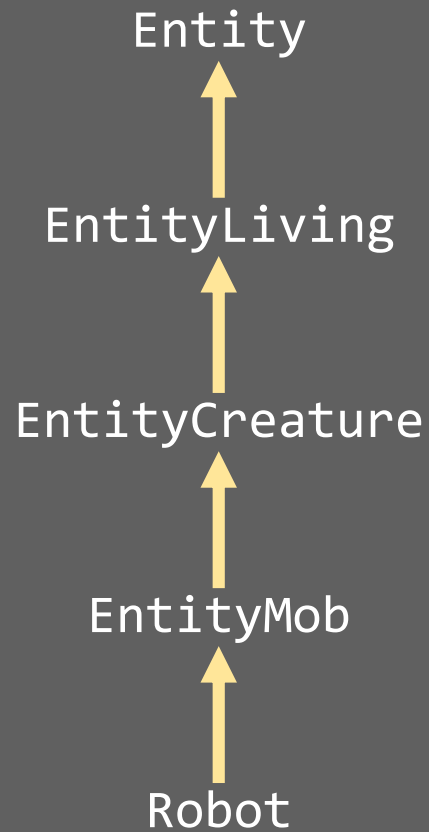
```
// Called when the entity collides with a player.  
void onCollideWithPlayer (EntityPlayer player);
```

ROBOTS

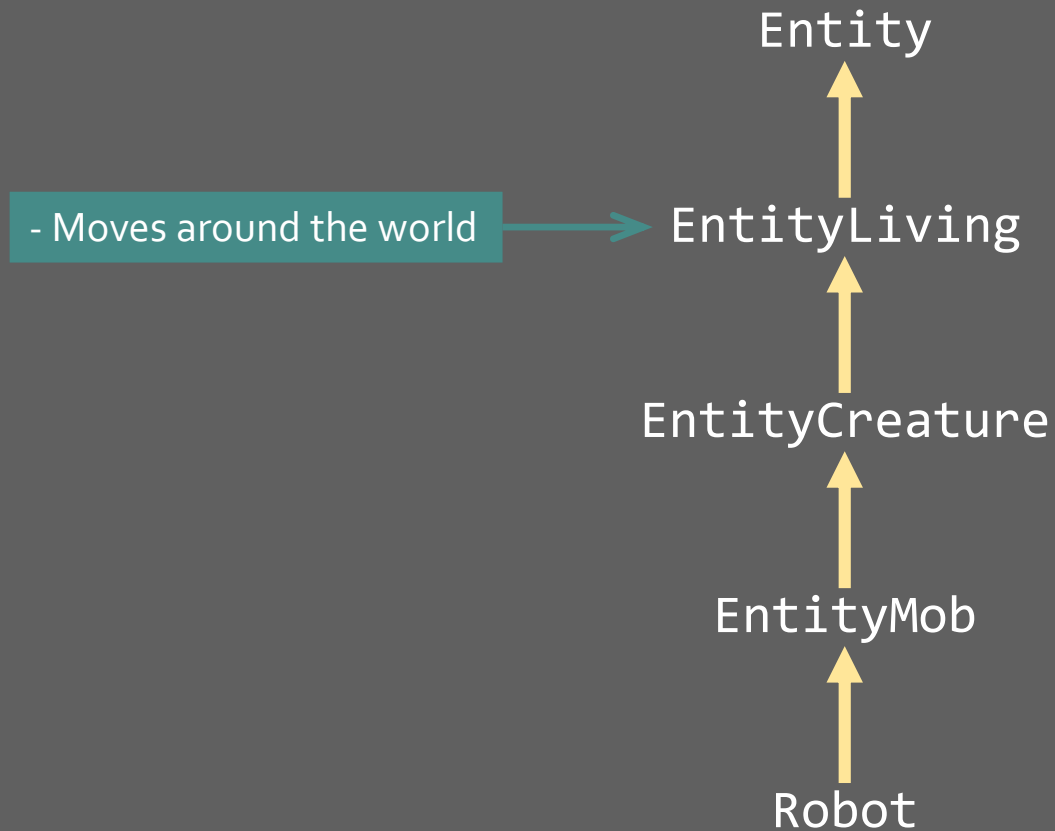
Robots are entities with special properties that you will use as a base for entities you create in the entity labs. Robots:

- look like rectangular prisms with solid colors and outlines,
- short-circuit and take 5 damage for every second they're in water,
- play sounds similar to the iron golem,
- attack in groups (if something harms or obstructs a robot, nearby robots will all attack the offending entity).

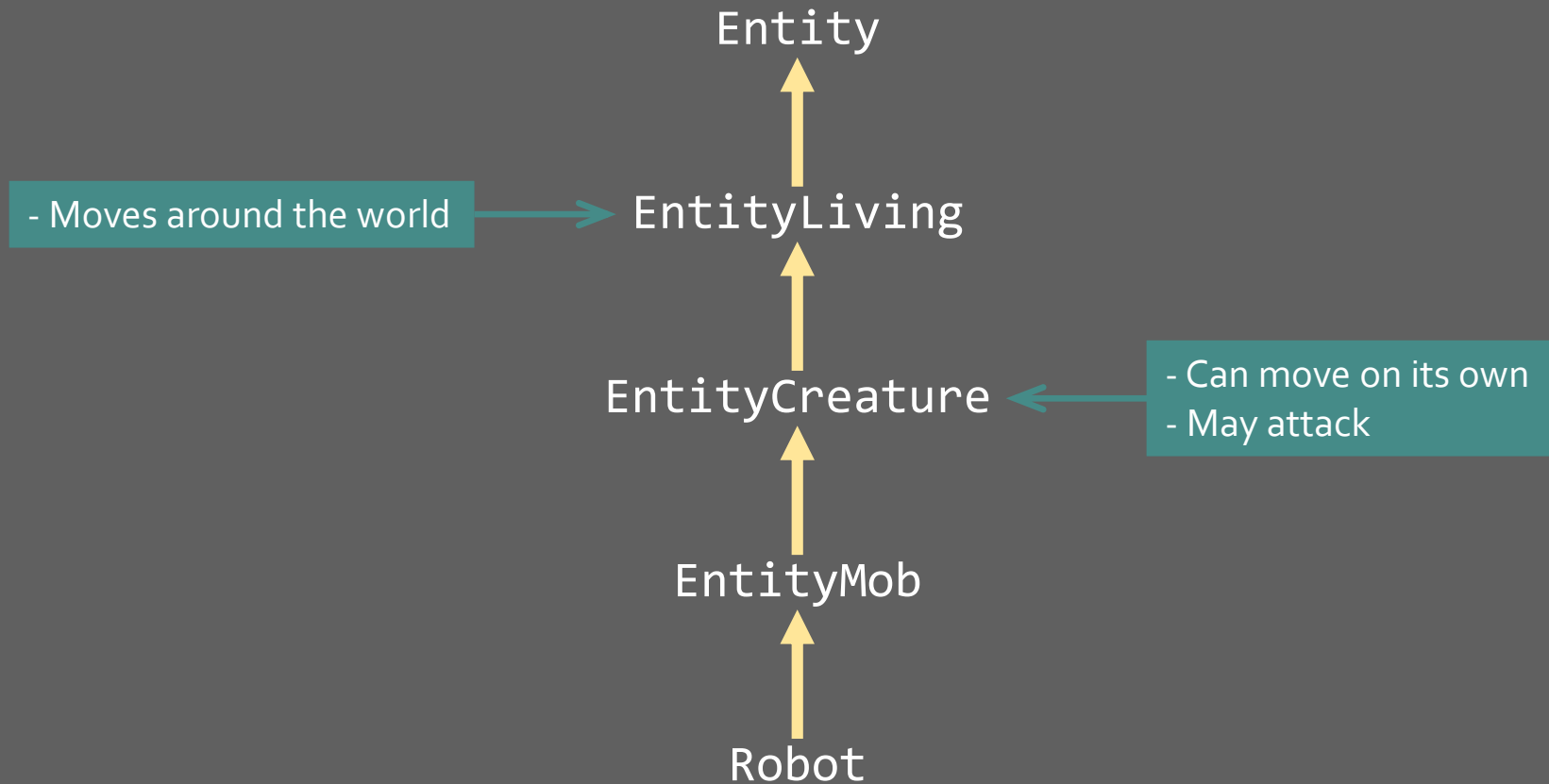
ROBOT HIERARCHY



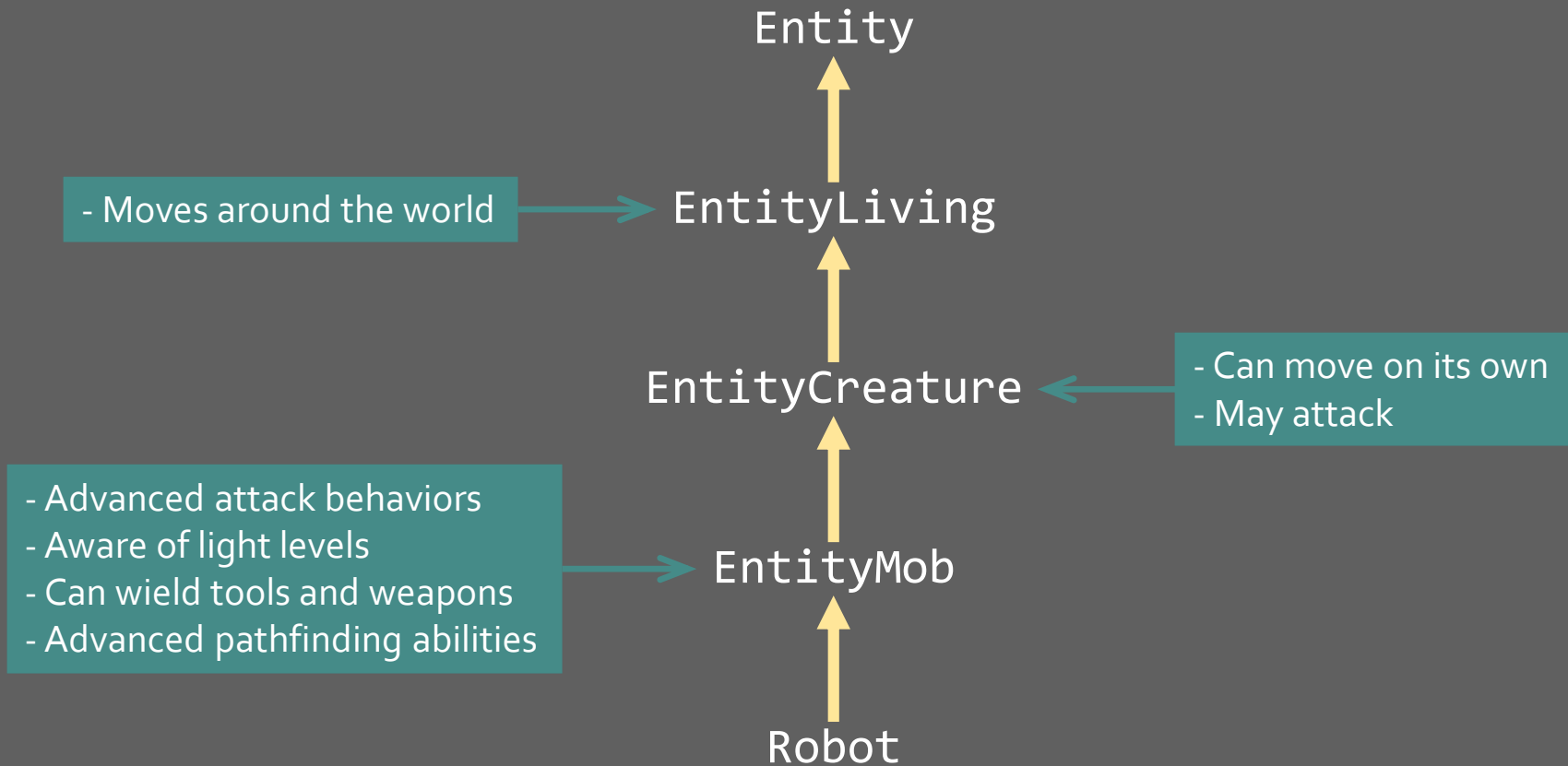
ROBOT HIERARCHY



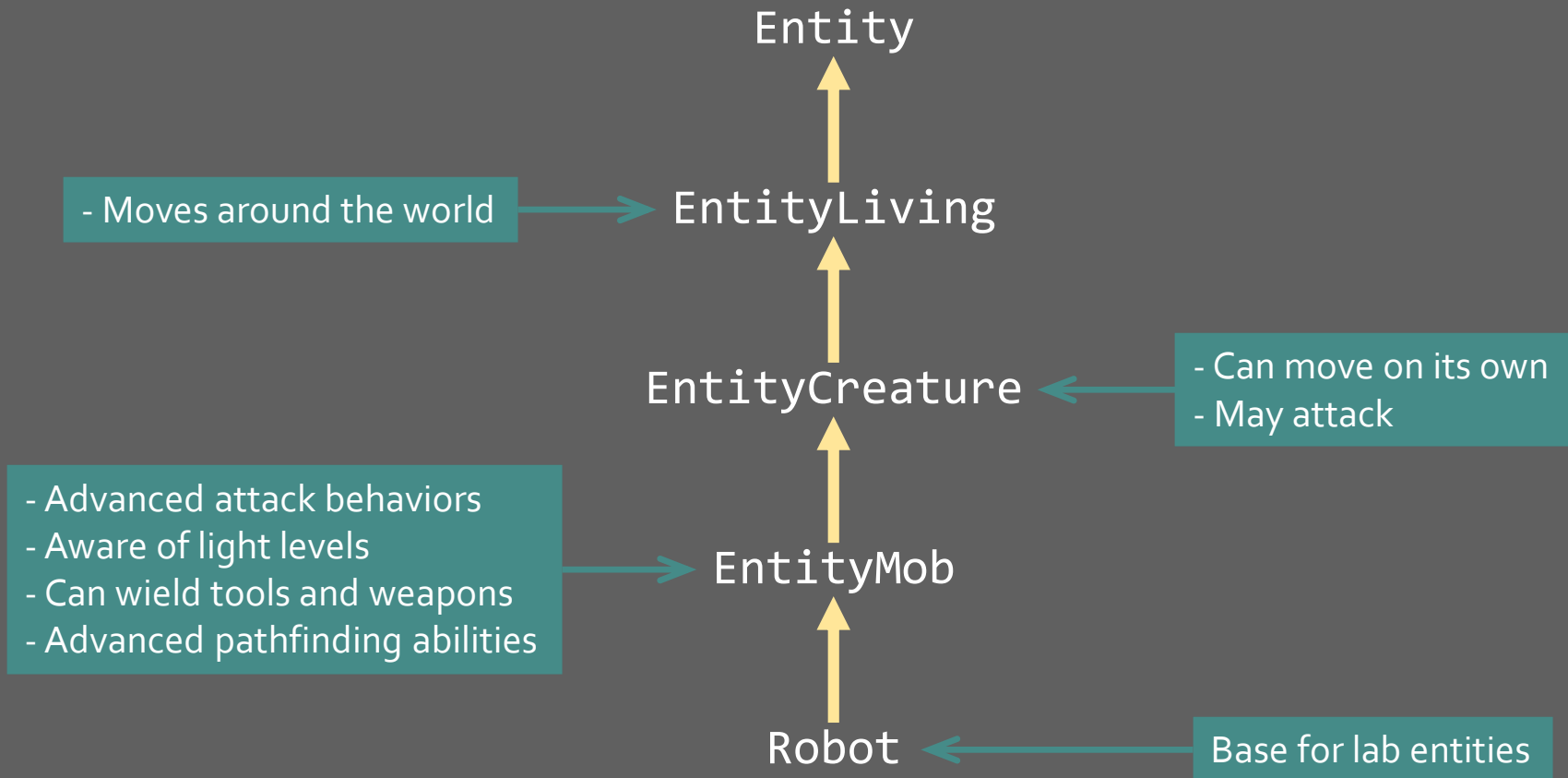
ROBOT HIERARCHY



ROBOT HIERARCHY



ROBOT HIERARCHY



SPAWNING ROBOTS

Once you've coded up a robot, you can summon them in game by typing

```
/robot <robot_name>
```

For lab 6, we'll be creating a leaf-converter robot of type "leafbot", so to summon a new leafbot, type the following command any time in the game:

```
/robot leafbot
```

LAB 6: LEAF-CONVERTER ROBOT

In Lab 6, you will create your first Minecraft entity: a leaf-converter robot. You will program this robot to (1) wander around the map on its own, and (2) randomly convert nearby blocks into leaves.