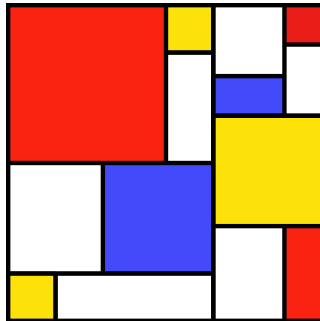


# SLiM

## Workshop Series



#16: Separate Sexes

# Separate Sexes

- Hermaphroditic models are the default
- Separate sexes can be modeled easily:
  - Enable with `initializeSex("A")`
  - Test sex with `individual.sex` ("M" or "F")
  - Sex-specific recombination/mutation rates
  - Other support for sex (sampling, output)

# Sexual WF Models

- Turn on separate sexes:
  - `initializeSex("A")`
- The sex ratio (M:M+F) is a parameter:
  - `sim.addSubpop("p1", 500, 0.6)`
  - `p1.setSexRatio(0.6)`
- The first parent is always female
- Selfing is not allowed

# A complete WF sexual model

```
initialize() {  
    initializeMutationRate(1e-7);  
    initializeMutationType("m1", 0.5, "f", 0.0);  
    initializeGenomicElementType("g1", m1, 1.0);  
    initializeGenomicElement(g1, 0, 99999);  
    initializeRecombinationRate(1e-8);  
    initializeSex("A");  
}  
1 {  
    sim.addSubpop("p1", 500);  
}  
2000 {  
    sim.simulationFinished();  
}
```

- Calls `initializeSex("A")`
- Nothing else is needed!

# Sexual nonWF Models

- Turn on with `initializeSex("A")`
- Sex-specific `reproduction()` callbacks:
  - `reproduction(<subpop>, "F")`
- Offspring sex can be controlled:
  - `p1.addCrossed(f, m, sex="M")`
  - `p1.addCrossed(f, m, sex=0.6)`
- The sex ratio is emergent

# A complete nonWF sexual model

```
initialize() {  
    initializeSLiMModelType("nonWF");  
    initializeSex("A");  
    initializeMutationType("m1", 0.5, "f", 0.0);  
    m1.convertToSubstitution = T;  
    initializeGenomicElementType("g1", m1, 1.0);  
    initializeGenomicElement(g1, 0, 99999);  
    initializeMutationRate(1e-7);  
    initializeRecombinationRate(1e-8);  
}  
reproduction(NULL, "F") {  
    mate = subpop.sampleIndividuals(1, sex="M");  
    subpop.addCrossed(individual, mate);  
}  
1 early() { sim.addSubpop("p1", 10); }  
early() { p1.fitnessScaling = 500 / p1.individualCount; }  
2000 late() { sim.outputFull(); }
```

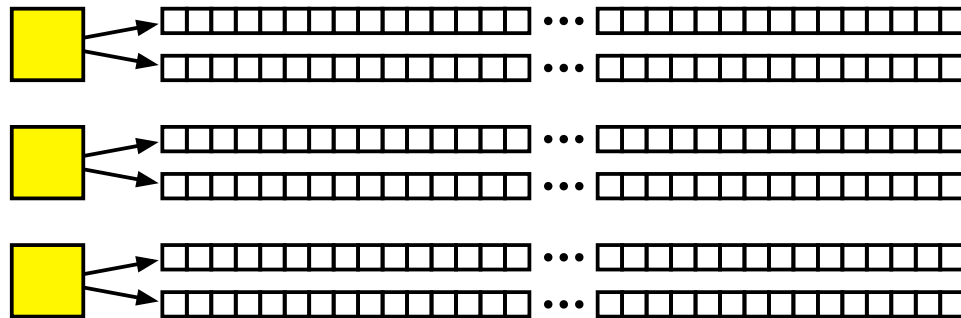
- Sex-specific reproduction() callback

# Modeling Sex Chromosomes

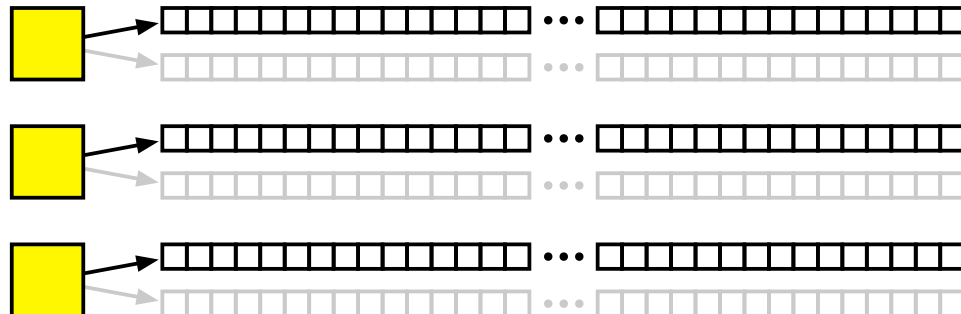
- Choose "X" or "Y" instead of "A":
  - `initializeSex("X")`
  - `initializeSex("Y")`
- Every individual still has two genomes
- Unmodeled genomes are “null”
  - test with `genome.isNullGenome`
  - do not attempt to manipulate these!

# Modeling the X

Females (XX):



Males (XY):

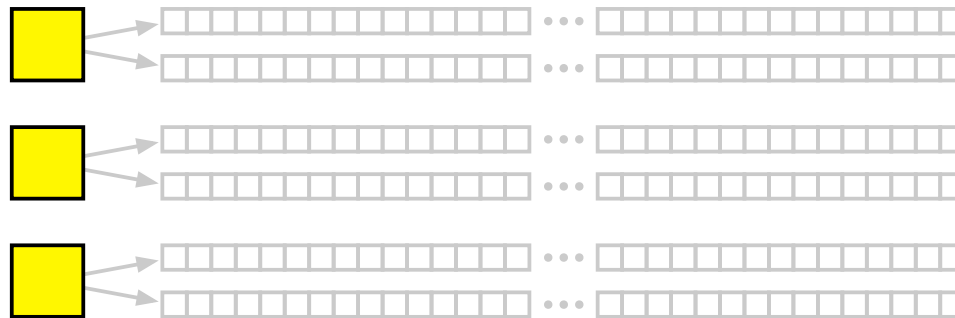


genome2  
is null

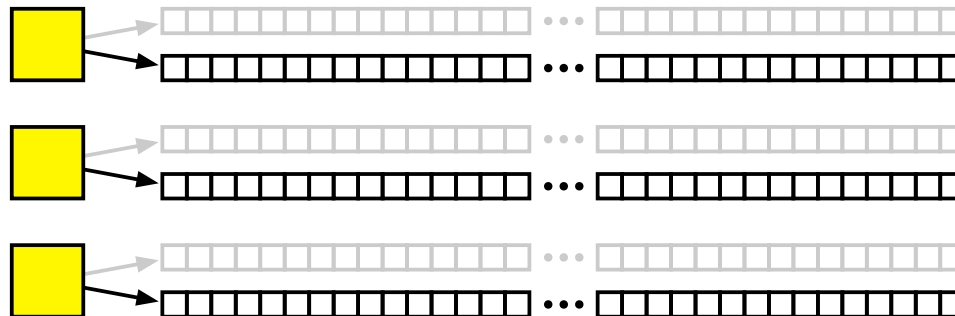


# Modeling the Y

Females (~~XX~~):



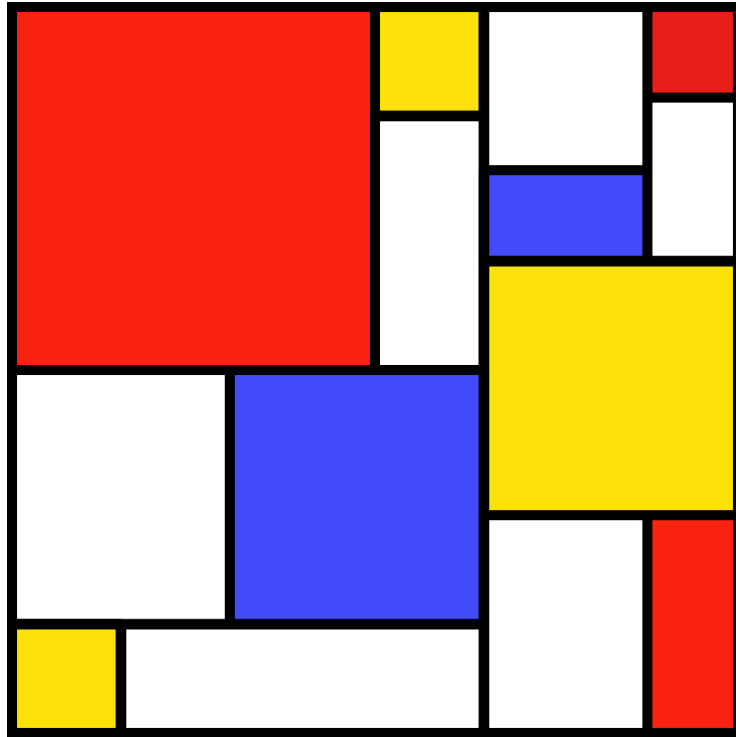
Males (~~XY~~):



genome1  
is null

# Modeling Sex Chromosomes

- More complex models can be built:
  - both X and Y
  - X, Y, and autosomes
  - mitochondrial + nuclear DNA
  - haplodiploidy and other mating systems
- Such models are constructed in script
  - do not use `initializeSex()`
  - control mating, etc., yourself (nonWF)
  - use `addRecombinant()` to generate offspring



SLiM Workshop Exercise #16