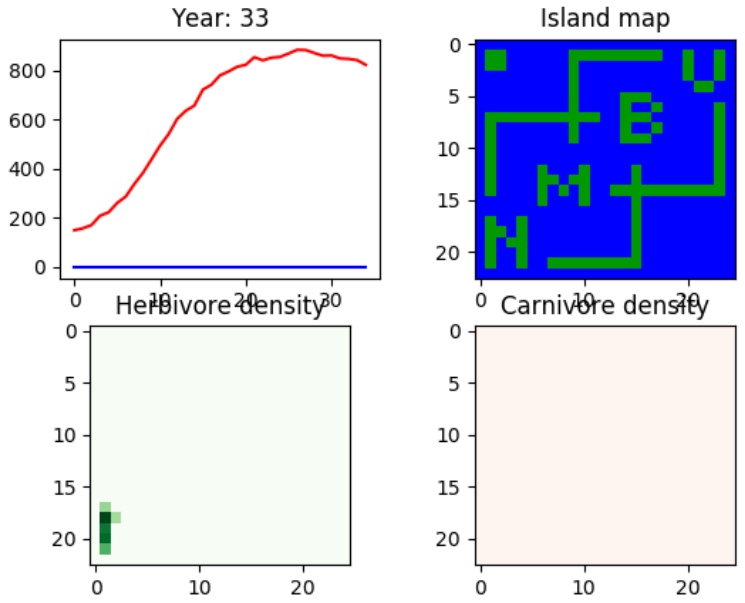
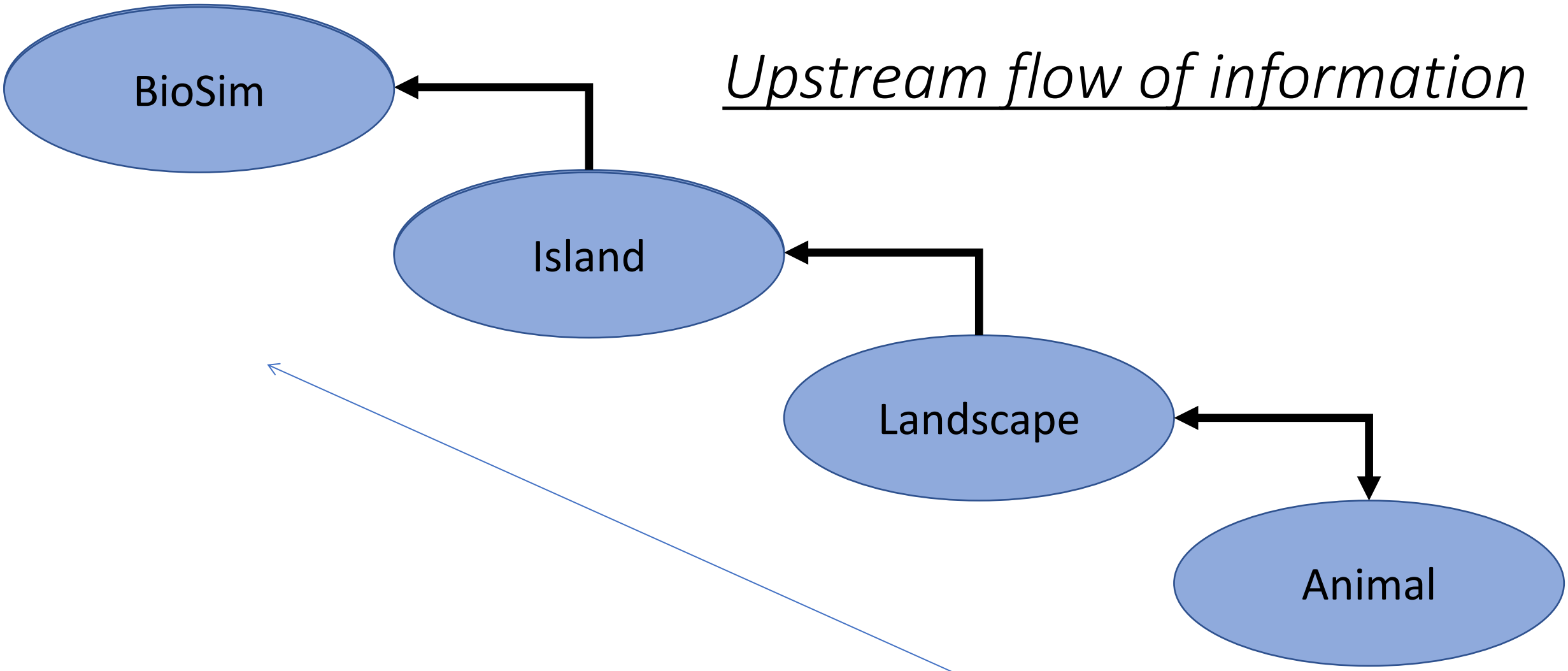
Population dynamics simulation

Modelling the Ecosystem of Rossumøya

Filip Rotnes & Sigve Sørensen

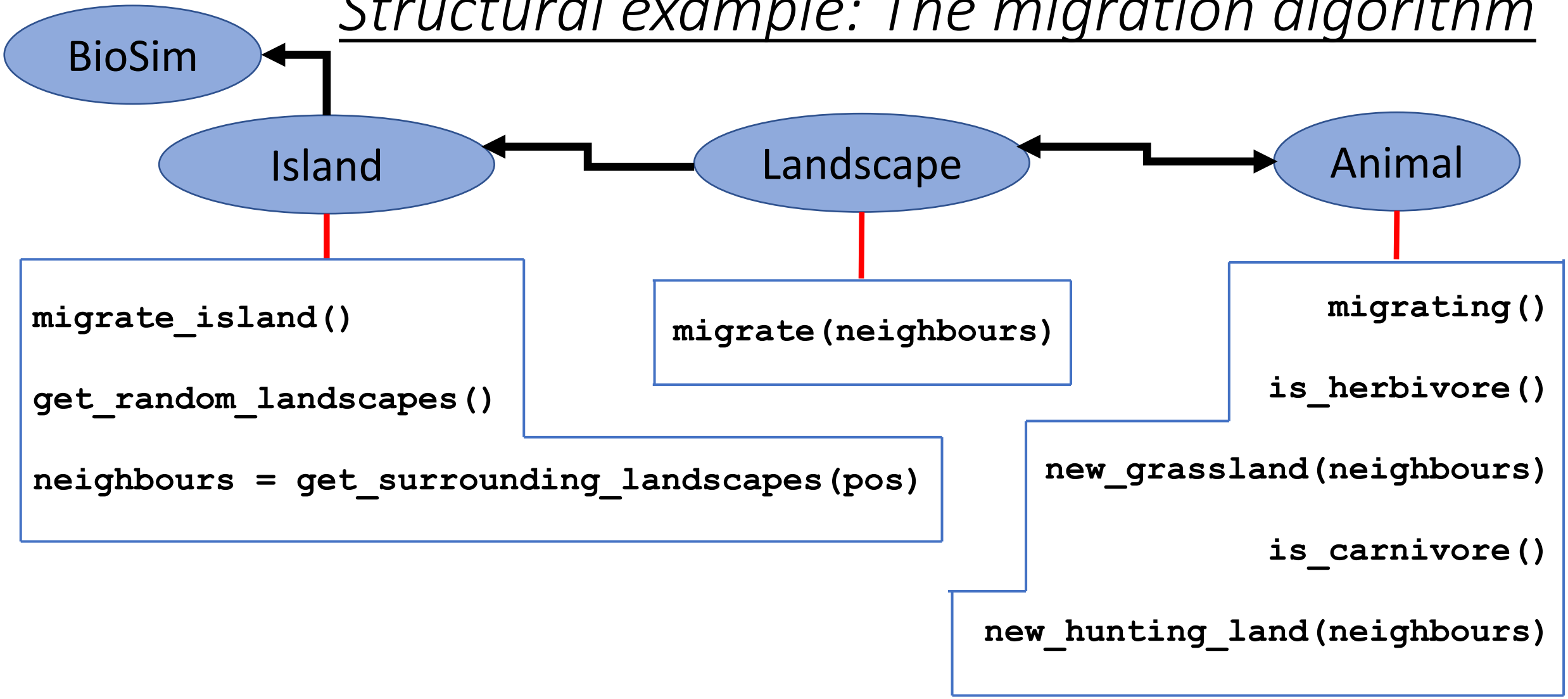


Norges miljø- og biovitenskapelige universitet

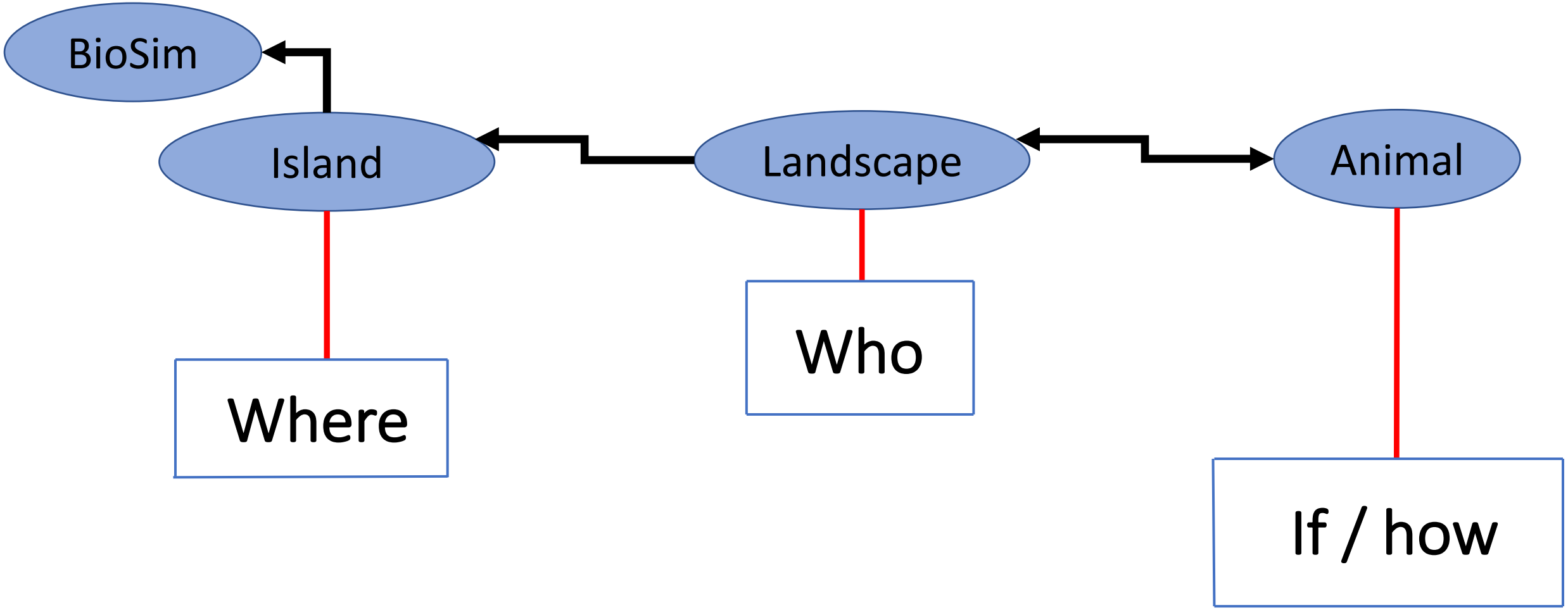


Upstream flow of information

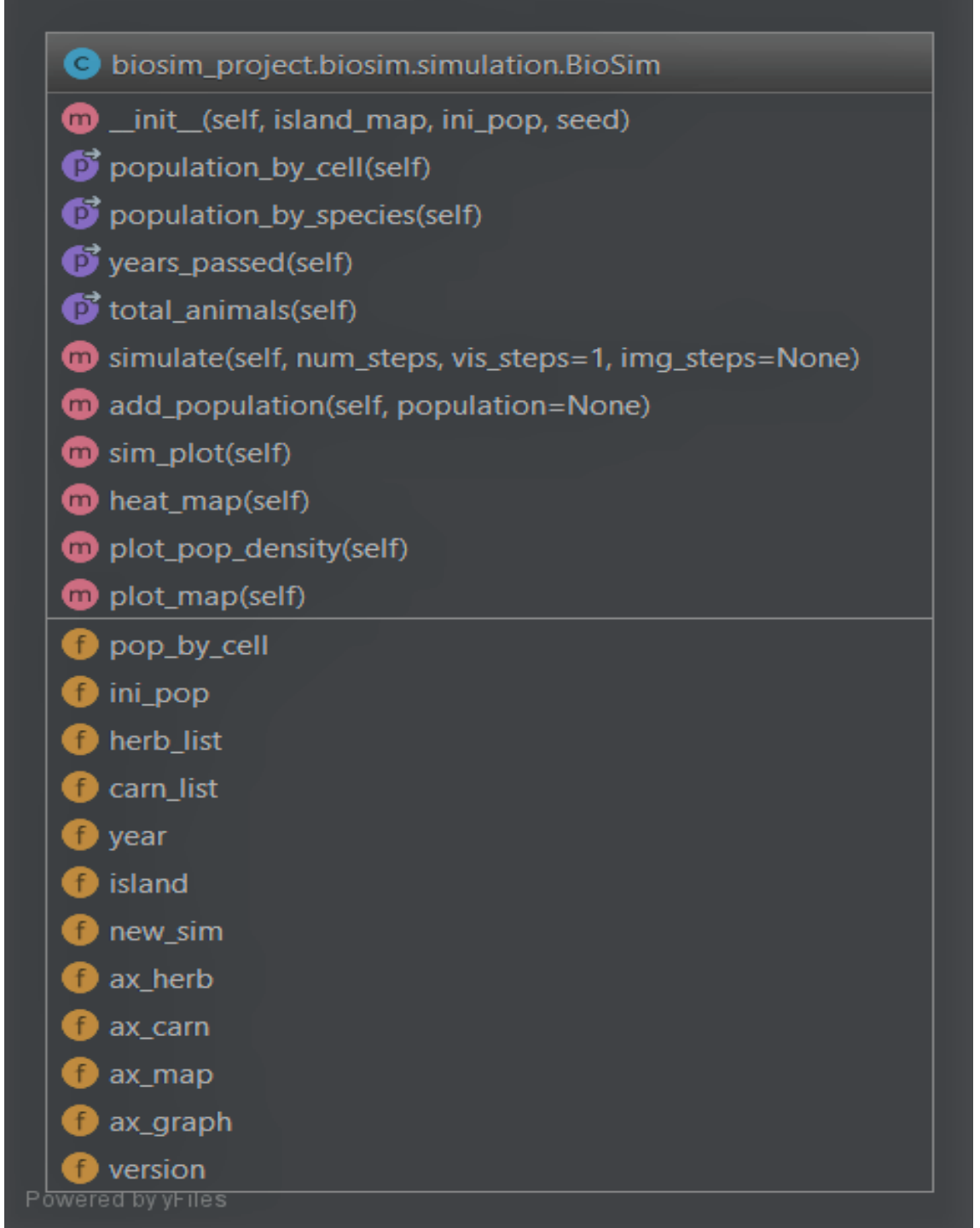
Structural example: The migration algorithm

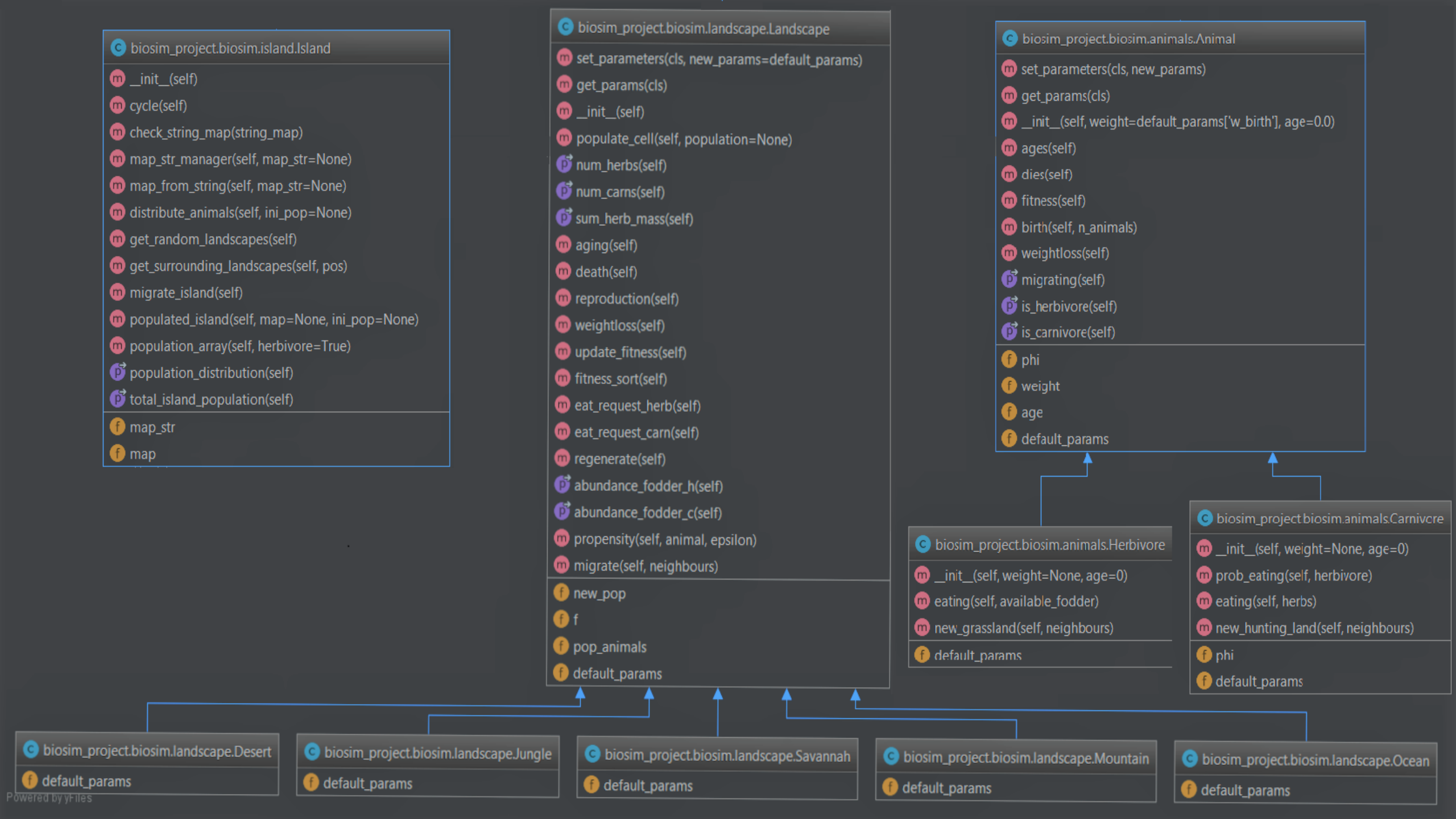


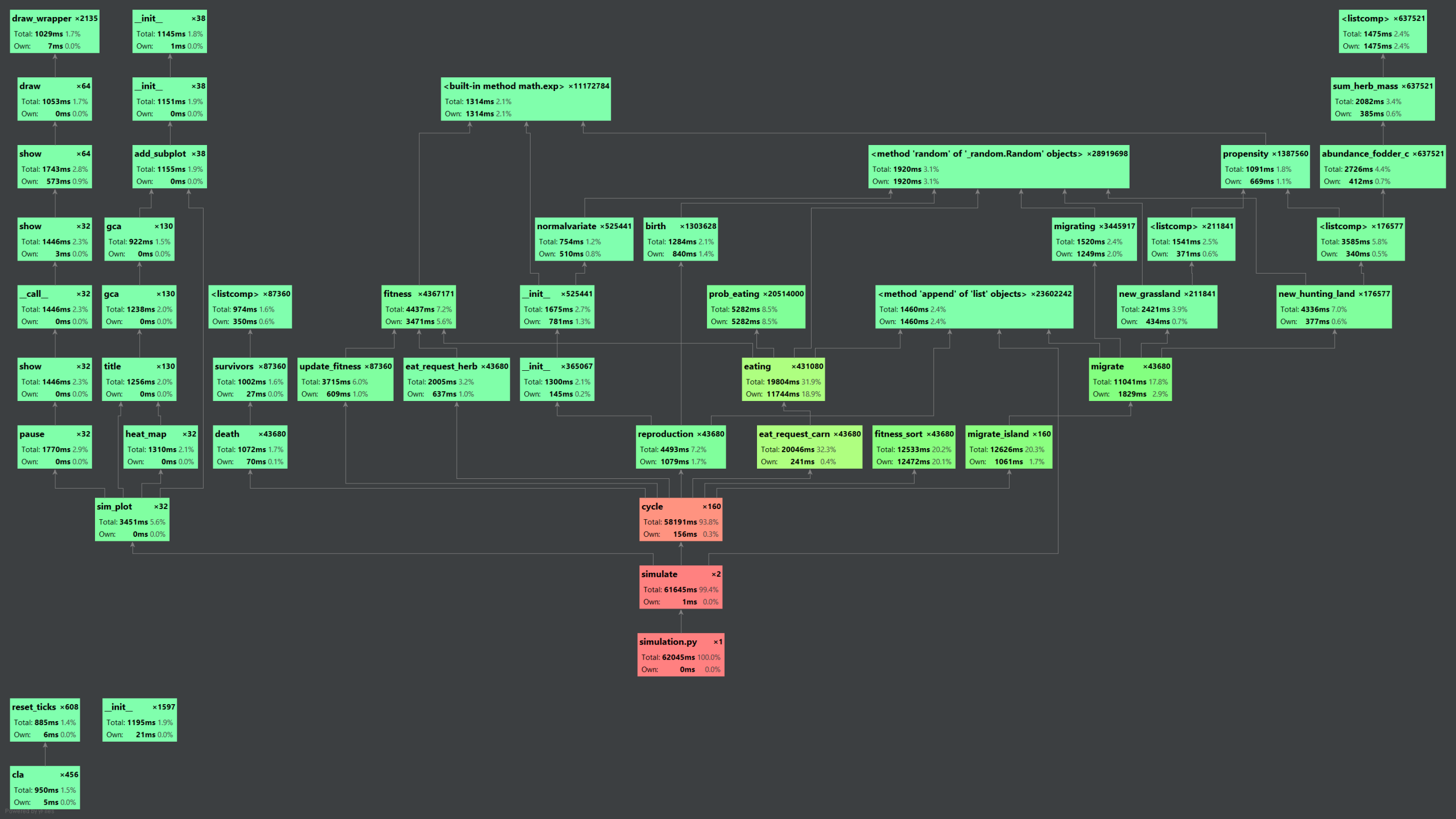
General structure of decision-making



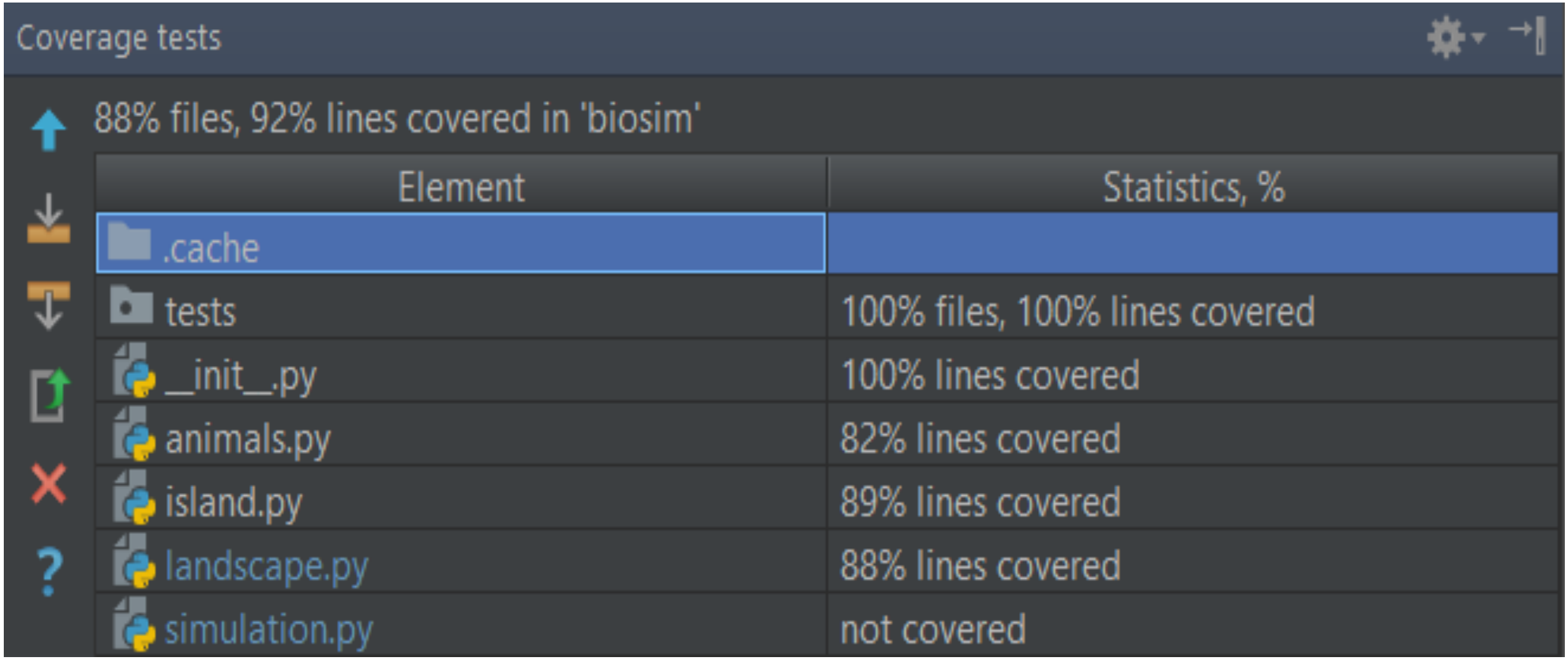
UML-diagrams




biosim_project.biosim.island.Island
__init__(self)
cycle(self)
check_string_map(string_map)
map_str_manager(self, map_str=None)
map_from_string(self, map_str=None)
distribute_animals(self, ini_pop=None)
get_random_landscapes(self)
get_surrounding_landscapes(self, pos)
migrate_island(self)
populated_island(self, map=None, ini_pop=None)
population_array(self, herbivore=True)
population_distribution(self)
total_island_population(self)
map_str
map
biosim_project.biosim.landscape.Landscape
set_parameters(cls, new_params=default_params)
get_params(cls)
__init__(self)
populate_cell(self, population=None)
num_herbs(self)
num_carns(self)
sum_herb_mass(self)
aging(self)
death(self)
reproduction(self)
weightloss(self)
update_fitness(self)
fitness_sort(self)
eat_request_herb(self)
eat_request_carn(self)
regenerate(self)
abundance_fodder_h(self)
abundance_fodder_c(self)
propensity(self, animal, epsilon)
migrate(self, neighbours)
new_pop
f
pop_animals
default_params
biosim_project.biosim.animals.Animal
set_parameters(cls, new_params)
get_params(cls)
__init__(self, weight=default_params['w_birth'], age=0.0)
ages(self)
dies(self)
fitness(self)
birth(self, n_animals)
weightloss(self)
migrating(self)
is_herbivore(self)
is_carnivore(self)
phi
weight
age
default_params
biosim_project.biosim.animals.Herbivore
__init__(self, weight=None, age=0)
eating(self, available_fodder)
new_grassland(self, neighbours)
default_params
biosim_project.biosim.animals.Carnivore
__init__(self, weight=None, age=0)
prob_eating(self, herbivore)
eating(self, herbs)
new_hunting_land(self, neighbours)
phi
default_params
biosim_project.biosim.landscape.Desert
default_params
biosim_project.biosim.landscape.Jungle
default_params
biosim_project.biosim.landscape.Savannah
default_params
biosim_project.biosim.landscape.Mountain
default_params
biosim_project.biosim.landscape.Ocean
default_params
Powered by yFiles


draw_wrapper ×2135
Total: 1029ms 1.7%
Own: 7ms 0.0%
__init__ ×38
Total: 1145ms 1.8%
Own: 1ms 0.0%
<listcomp> ×637521
Total: 1475ms 2.4%
Own: 1475ms 2.4%
draw ×64
Total: 1053ms 1.7%
Own: 0ms 0.0%
__init__ ×38
Total: 1151ms 1.9%
Own: 0ms 0.0%
<built-in method math.exp> ×11172784
Total: 1314ms 2.1%
Own: 1314ms 2.1%
sum_herb_mass ×637521
Total: 2082ms 3.4%
Own: 385ms 0.6%
show ×64
Total: 1743ms 2.8%
Own: 573ms 0.9%
add_subplot ×38
Total: 1155ms 1.9%
Own: 0ms 0.0%
<method 'random' of '_random.Random' objects> ×28919698
Total: 1920ms 3.1%
Own: 1920ms 3.1%
propensity ×1387560
Total: 1091ms 1.8%
Own: 669ms 1.1%
abundance_fodder_c ×637521
Total: 2726ms 4.4%
Own: 412ms 0.7%
show ×32
Total: 1446ms 2.3%
Own: 3ms 0.0%
gca ×130
Total: 922ms 1.5%
Own: 0ms 0.0%
normalvariate ×525441
Total: 754ms 1.2%
Own: 510ms 0.8%
birth ×1303628
Total: 1284ms 2.1%
Own: 840ms 1.4%
migrating ×3445917
Total: 1520ms 2.4%
Own: 1249ms 2.0%
<listcomp> ×211841
Total: 1541ms 2.5%
Own: 371ms 0.6%
<listcomp> ×176577
Total: 3585ms 5.8%
Own: 340ms 0.5%
__call__ ×32
Total: 1446ms 2.3%
Own: 0ms 0.0%
gca ×130
Total: 1238ms 2.0%
Own: 0ms 0.0%
<listcomp> ×87360
Total: 974ms 1.6%
Own: 350ms 0.6%
fitness ×4367171
Total: 4437ms 7.2%
Own: 3471ms 5.6%
__init__ ×525441
Total: 1675ms 2.7%
Own: 781ms 1.3%
prob_eating ×20514000
Total: 5282ms 8.5%
Own: 5282ms 8.5%
<method 'append' of 'list' objects> ×23602242
Total: 1460ms 2.4%
Own: 1460ms 2.4%
new_grassland ×211841
Total: 2421ms 3.9%
Own: 434ms 0.7%
new_hunting_land ×176577
Total: 4336ms 7.0%
Own: 377ms 0.6%
show ×32
Total: 1446ms 2.3%
Own: 0ms 0.0%
title ×130
Total: 1256ms 2.0%
Own: 0ms 0.0%
survivors ×87360
Total: 1002ms 1.6%
Own: 27ms 0.0%
update_fitness ×87360
Total: 3715ms 6.0%
Own: 609ms 1.0%
eat_request_herb ×43680
Total: 2005ms 3.2%
Own: 637ms 1.0%
__init__ ×365067
Total: 1300ms 2.1%
Own: 145ms 0.2%
eating ×431080
Total: 19804ms 31.9%
Own: 11744ms 18.9%
migrate ×43680
Total: 11041ms 17.8%
Own: 1829ms 2.9%
pause ×32
Total: 1770ms 2.9%
Own: 0ms 0.0%
heat_map ×32
Total: 1310ms 2.1%
Own: 0ms 0.0%
death ×43680
Total: 1072ms 1.7%
Own: 70ms 0.1%
reproduction ×43680
Total: 4493ms 7.2%
Own: 1079ms 1.7%
eat_request_carn ×43680
Total: 20046ms 32.3%
Own: 241ms 0.4%
fitness_sort ×43680
Total: 12533ms 20.2%
Own: 12472ms 20.1%
migrate_island ×160
Total: 12626ms 20.3%
Own: 1061ms 1.7%
sim_plot ×32
Total: 3451ms 5.6%
Own: 0ms 0.0%
cycle ×160
Total: 58191ms 93.8%
Own: 156ms 0.3%
simulate ×2
Total: 61645ms 99.4%
Own: 1ms 0.0%
simulation.py ×1
Total: 62045ms 100.0%
Own: 0ms 0.0%
reset_ticks ×608
Total: 885ms 1.4%
Own: 6ms 0.0%
__init__ ×1597
Total: 1195ms 1.9%
Own: 21ms 0.0%
cla ×456
Total: 950ms 1.5%
Own: 5ms 0.0%

Test coverage



The code:

- Tidy, categorical
- Expandable
- Focus on readability
- Task specific functions
- Benefits of the Numpy array

If we had better time:

- Package completeness
  - Examples -> Model potential
  - User friendliness -> UI
    - Terminal -> GUI
- Further statistical analysis
- Optimization
- Task specific functions
- Diversify output data