# Disease Model (dm) v1.0

_____

**SPECIFICATION**

Nolan McMahon
Last Updated: September 18, 2022

This document is free and open source and is licensed under the GFDL v1.3.

_____

## Overview

dm is a simulation of the spread of disease through a population.

## License

The software dm is licensed under the GNU General Public License version 3 (GPLv3) or, at your option, any later version. All documentation which accompanies dm, including this document, is licensed under the GNU Free Documentation License version 1.3 or, at your option, any later version.

dm makes use of, but in no way modifies, an external software project called TOML-Fortran or toml-f. According to its authors, toml-f may be licensed under either the Apache License, Version 2.0 or the MIT License, at my option. dm respects the decisions made by the authors of toml-f and grants the same freedoms to you, namely, releasing the integrated version of toml-f under either the Apache License, Version 2.0 or the MIT License, at your option.

dm makes use of, but in no way modifies, an external software project called test-drive. According to its authors, test-drive may be licensed under either the Apache License, Version 2.0 or the MIT License, at my option. dm respects the decisions made by the authors of test-drive and grants the same freedoms to you, namely, releasing the integrated version of test-drive under either the Apache License, Version 2.0 or the MIT License, at your option.

dm makes use of, but in no way modifies, an external software project called gnuplot. gnuplot is licensed under its own free source license, which is compatible with inclusion in dm.

The complete source code of dm, along with the full text of all of the aforementioned licenses, is made available at: https://git.nolanmcmahon.net/Disease_Model.

## Non-Goals

These are features that I do **not** plan to include (at least in version 1):

- A GUI. The only way to interact with the model will be through the command line.

# Command Line Interface

This program will be called from the command line by calling `dm`. Calling `dm` without any command line arguments will print the version as well as the GPLv3 license blurb. The command line argument will take exactly one of the following arguments:

- `--version`

  Prints current version of the program and exits, regardless of other flags and parameters passed.

- `-f [config]`

  The path to the custom config file to run the simulation with.

- `--covid-19`

  Run the COVID-19 prebuild simulation.

- `--spanish-flu`

  Run the 1918 Spanish Flu prebuilt simulation.

- `--black-death`

  Run the Black Death prebuilt simulation.

- `--sars`

  Run the SARS prebuilt simulation.

- `--mers`

  Run the MERS prebuilt simulation.

- `--ebola`

  Run the Ebola prebuilt simulation.

- `--swine-flu`

  Run the Swine Flu prebuilt simulation.

- `--normal`

  Run the normal times prebuilt simulation.

# Simulation Flow

On a high level, this is how `dm` simulates the spread of disease.

## People

`dm` assumes that all the people on the planet spend their entire lives living in one of seven possible states:

1. Alone
2. Not Alone
3. Travelling
4. Congregating
5. Quarantining
6. Hospitalized
7. In ICU

Alone and not alone are the default states that a person exists in.

The states of travelling and congregating are assumed to interrupt the default states of being alone or not alone. They are posed as "opportunities" which come around only once in a while. People are then able to exercise their own agency and decide, probabilistically, if they want to take advantage of the opportunity to congregate or travel.

Quarantining, hospitalization, and ICU admission are the last three states that a person can enter. A person will never willingly choose to be in any of these states. Instead, these states are foisted upon people because of their circumstances. Quarantining is a consequence of either being traced as a close contact of another person who has a disease, or as a result of travelling. Hospitalization is imposed when a person's infection is severe enough to require it. Likewise with ICU admission.

Each person is always in exactly one of these states for as long as they are alive in the simulation. Each of these states can carry a distinct risk of contracting each disease variant. These risks are relative to one another, and subject to the overall probability of a disease variant spreading.

## Regions

Regions are a construct in `dm` that handle a collective response to disease. They can be thought of being analogous to nation states. Regions are responsible for contact tracing individuals, deciding when to lock down society, whether or not to enforce quarantines, and taking care of public health through hospitals.

Each person in `dm` always exists within exactly one region, starting at birth. As long as they exist in a region, they are subject to the rules of the region, unless they choose to disregard them (see `comply_with_lock_down_prob` and `comply_with_quarantine_prob` variables below). So long as the receiving region permits it, all people are free to engage in travel. At the end of the travel period, the person in question will be assigned a new region. If the region that they started their travels in is not their home region (the one in which they were born) their next travel excursion will be to their home region. Once a person arrives in a new region, they are no longer subject to the rules of the region from which they came but only to the region in which they are currently. (This includes travel quarantines.)

If a region declares a lock down, the proportion of the people in the society who are complient with lock downs will no longer enter the congregation state. This only lasts as long as the lock down fatigue in the region has not been reached, after which point, all people ignore the lock down.

## Diseases, Variants, and Infections

`dm` handles illness in a higherarchy. People contract *infections*. Each infection is of a particular *variant*. Each variant belongs to a particular *disease.* A disease can contain many different variants, each of which can produce infections in many people.

When a person gets infected, they enter the incubation stage of the disease. At this point, they are not infectious, and have no symptoms. After the incubation period is over, they can either become symptomatic or asymptomatic. If they become symptomatic, then there is the option to do contact tracing which informs a subset of people in the region based on the state of the person during their incubation period. As the disease progresses, the person may need to enter the hospital, if there is a bed available. If a person needs a hospital bed, but one is not available, they immediately die. Once in the hospital, the person may worsen to the point where they need to be moved to the ICU. If this is the case but there are no available ICU beds, the person dies immediately. Eventually, the person either recovers from the disease and gains natural immunity to the variant, or they die.

Vaccines may or may not become available for different variants of the disease. Assuming that a vaccine becomes available, the people in who are not against vaccines will go and get immunized. This adds to their protection against infection.

Infected people who are not in the incubation phase are contagious. When a new infection is to be created, the target person is chosen randomly from the set of all other people in the same region as the person who is creating the new infection with a probability that is proportional to:

1. The state of the person at the moment of infection. $s_i$

  2. The vaccination protection level at the moment of infection. $v_i$

  3. The natural immunity protection level at the moment of infection. $q_i$

(Note that the protection provided by vaccines and natural recoveries compounds. For example, if a vaccine provides 80% coverage, then the booster shot provides 80% coverage of the remaining 20% susecptibility.) The probability $p$ that person $i$ in a region with $n$ other people will be the recipient of the new infection is:

$$p_i = \frac{s_i v_i q_i}{\sum_{j=1}^{n} s_j v_j q_j}$$

## Simulation of Passing Time

This simulation is based on a unit of time called a frame. The simulation starts at frame zero and counts up until either the simulation reaches some stopping condition (see Simulation Termination Conditions below), or the frame number reaches it's maximum value.

Each frame represents one minute of "real" time. That gives 1440 frames per day and 525,600 frames per year.

Many of the variables that can be specified in the `dm` configuration file make reference to an amount of time. This quantity is measured in frames.

## Statistics and Results

`dm` collects and returns a large swathe of data and statistics. This data is collected on a frame by frame level, where applicable, and returned as a CSV file. Plots are also made using gnuplot, if desired. This is the list of statistics that `dm` collects. Each of the following are collected on a frame by frame basis and on a cumulative basis, where appropriate. They are collected on a global scale and on a region by region scale.

- Number of People Currently Alive
- Number of People Born
- Average Life Span
- Number of People who have Died
- Number of People who have Died of Natural Causes
- Number of People who have Died of All Diseases
- Number of People who have Died by Disease
- Number of People who have Died by Disease Variant
- Number of People who have Recovered
- Number of People who Have Recovered By Disease
- Number of People who Have Recovered By Disease Variant
- Number of New Infections
- Number of New Infections by Disease
- Number of New Infections by Disease Variant

- Number of Resolved Infections
- Number of Resolved Infections by Disease
- Number of Resolved Infections by Disease Variant
- Number of People in Incubation Period
- Number of People in Incubation Period by Disease
- Number of People in Incubation Period by Disease Variant
- Number of Frames Spent in Incubation Period
- Number of Frames Spent in Incubation Period by Disease
- Number of Frames Spent in Incubation Period by Disease Variant
- Number of People in Symptomatic Period
- Number of People in Symptomatic Period by Disease
- Number of People in Symptomatic Period by Disease Variant
- Number of Frames Spent in Symptomatic Period

- Number of Frames Spent in Symptomatic Period by Disease
- Number of Frames Spent in Symptomatic Period by Disease Variant
- Number of People in Asymptomatic Period
- Number of People in Asymptomatic Period by Disease
- Number of People in Asymptomatic Period by Disease Variant
- Number of Frames Spent in Asymptomatic Period
- Number of Frames Spent in Asymptomatic Period by Disease
- Number of Frames Spent in Asymptomatic Period by Disease Variant
- Number of Frames Spent Alone
- Number of Frames Spent Not Alone
- Number of Frames Spent Travelling
- Number of Frames Spent Quarantining
- Number of Frames Spent Hospitalized
- Number of Frames Spent in ICU
- Number of Hospital Beds Used
- Number of Hospital Beds Free
- Number of ICU Beds Used
- Number of ICU Beds Free
- Number of New Hospital Admissions
- Number of New ICU Admissions
- Number of Hospital Beds Used By Disease
- Number of Hospital Beds Used By Disease Variant
- Number of ICU Beds Used By Disease
- Number of ICU Beds Used By Disease Variant
- Number of Frames Spent Hospitalized By Disease
- Number of Frames Spent Hospitalized By Disease Variant
- Number of Frames Spent in ICU by Disease
- Number of Frames Spent in ICU by Disease Variant
- Number of People Quarantining
- Number of Sick People Quarantining

- Number of Healthy People Quarantining
- Number of People Quarantining Because of Travel
- Number of People Quarantining Because of Contact Tracing
- Number of Frames Spent Quarantining Because of Travel
- Number of Frames Spent Quarantining Because of Contact Tracing
- Number of Sick People Quarantining by Disease
- Number of Sick People Quarantining by Disease Variant
- Number of People Vaccinated
- Number of People Vaccinated By Disease
- Number of People Vaccinated By Disease Variant
- Number of Frames of Lock Down
- Number of People Locked Down
- Number of Anti-Vaxers
- $R_0$ By Disease
- $R_0$ By Disease Variant
- Average Natural Immunity Level By Disease
- Average Natural Immunity Level By Disease Variant
- Average Vaccine Immunity Level By Disease
- Average Vaccine Immunity Level By Disease Variant
- Average Total Immunity Level By Disease
- Average Total Immunity Level By Disease Variant
- Number of Active Diseases
- Number of Active Diseases By Variant
- Number of Extinct Diseases
- Number of Extinct Diseases By Variant

# Simulation Process

The simulation process is the name given to the procedure that produces all of the necessary simulation objects and populates them with data. Note that the backend process of creating the simulation data is completely independent of the frontend process to visually represent this data in a dynamic, visual way to the end user. Because of this, and array language benefits that FOR-TRAN provides, there is no sense in performing simulation calculations on a frame by frame basis. Instead, wherever possible, objects will be created in their entirety at the time of their construction. There are 8 object classes from which objects need to be constructed to complete a single simulation of the spreading of a disease:

1. One Utilities Object
2. One Globe Object
3. At Least One Region Object
4. At Least Ten Person Objects for Each Region Object
5. At Least One Disease Object
6. At Least One Variant Object for Each Disease Object
7. At Least One Infection Object
8. One Results Object

The descriptions of these classes are listed below, in the "Objects Specifications" section. There are two components to create the simulation: initialization and propagation.

## Initialization

1. First, a Utilities object needs to be created. This object will exist for the duration of the simulation initialization and propagation object since it contains useful procedures and properties in the construction of other objects.
2. Second, a Globe object is created. This is the highest level object in the simulation. It contains references to all other objects that are in the simulation. There is only ever one of these objects.
3. Third, at least one Region object is created. The true number of created Region objects is controlled by the end user through the value of the `n_regions` parameter. Region objects are only created in the initialization step, they are then never again created or destroyed.
4. Fourth, Person objects are created for each region.
5. Fifth, at least one Disease object is created. It's parameters are controlled by the end user through a set of variables provided.
6. Sixth, at least one Variant object is created for each Disease object that was created. Their properties are conrolled by the end user through the provided variables.
7. Seventh, at least one Infection object is created for each Variant object that was created. These objects are associated with Person objects according to the parameters provided by the user or randomly.
8. Eighth, a Results object needs to be created to store all of the statistics from the simulation.

## Propagation

1. Check simulation termination conditions.
2. Create all new Diseases for the given frame based on user input, if applicable. If applicable, add user specified number of infections to infection schedule on the current frame.

Also create Variant objects according to user input for the new disease. Update Results object.

3. Create all new Variant objects as applicable as a consequence of the number of new infections on this frame in the infection schedule. Update Results Object.

4. Create all relevant Infection objects that start on the given frame number. New Infections should be totally populated upon creation at this stage. Infection objects should have their ids associated with their respective Person objects. New upcoming consequential infections should be added to the infection schedule. Update Results Object.

5. Modify all existing Person objects with deceased/recovered statuses etc where applicable based on the results of the previous step. Update all travel, and quarantining related data for affected Persons as well. Update the birth schedule to accomodate unexpected deaths. Update Results Object.

6. Update all Region objects with consequences as a result of travel changes, quarantining, infection levels, etc. Update Results Object.

7. Update Globe object as a consequence of changes to Region objects, as applicable. Update Results Object.

8. Create all new Person objects as described by the birth schedule. These new Person objects should have all of their fields populated (to the greatest extent possible) at this point after construction. Update the birth schedule. Update Results Object.

9. Move to next frame with a birth or an infection according to the birth and infection schedules.

## Simulation Termination Conditions

The conditions for termination of the simulation are as follows:

1. The sum of the infection schedule from the current frame forward is zero and there are no future diseases specified by the end user. (Nobody will ever get sick again.)

2. There are no Person objects which are not in the deceased state. (Everybody died.)

3. The user specified end frame is reached.

4. The limit of the value of the frame number is reached.

## Schematic

The simulation considers a set of individuals and a set of diseases that can spread through them. Each person has a set of properties that impact how they respond to each disease. Every person is assigned to a region. Each region also has a set of properties that impact how the collective of individuals within that region respond to the diseases as a whole. At the beginning of the simulation, exactly one person is sick with each initialized disease variant. As the simulation progresses, other people can contract the diseases and their variants. The simulation ends when there are no more infected people.

The simulation is thus broken down into a number of different components which interact with one another.

• Globe - Interacts with all other components of the simulation.

• Regions - Analogous to nation states. Controls collective responses to disease.

• People - Controls individual responses to disease.

• Diseases - Controls a category of variants of an illness.

• Variants - Controls individual properties of a disease.

• Infections - Measures the impact of a variant on a person.

There are a number of different toggles that can be used to influence the outcome of the simulation. Most variables are related to one of the above objects. A sample configuration file, which can

be used to modify the behaviour of the simulation, is included in a file called `spec/sample.toml` and as Appendix A to this specification. The list of all variables that `dm` understands is given below.

## Default Values

By default, all variables are given a type, and there are restrictions on the values that variables are permitted to contain. Unless otherwise specified, these are the restrictions on the values that variables can have:

- Logical - [true, false].
- Character - Any sequence of ASCII characters.
- Integer - Any non-negative value which is $\leq 2^{64} - 1$.
- Real - Any non-negative value which is $\leq 1$.

One other note: all variables below that are time related request values in a unit called a "frame". For more detail about what a "frame" is, see the section "Simulation of Passing Time" above.

### General

Controls the flow of the simuation program as a whole.

- `verbose` - [Logical]

Whether or not produce verbose output while generating the simulation results.

### Output

Controls how and what the simulation outputs for results.

- `output_file` - [Character]

Path to the output csv file to be created, which contains all of the simulation details.

- `use_gnuplot` - [Logical]

Whether or not to use gnuplot to generate plots of the simulation output.

### Termination

Controls when the simulation terminates.

- `maximum_simulation_length` - [Integer]

The maximum amount of time that the simulation will be allowed to run for, measured in minutes.

### Globe

- `number_of_regions` - [Integer]

The number of regions, or nations, to simulate.

### Globe.Persons

There are two different ways to set values for Person objects: globally, meaning for all people in all regions, or regionally, meaning only for the people in that region. This section holds Global Person values, which can also be thought of as default values. If a region does not have a `Region.Person` entry, or if the entry is missing a variable assignment, or if one of it's variable assignments is set to `nan`, then the associated `Globe.Person` entries are used. Note that wherever possible, `Region.Person` entries are preferred to `Globe.Person` entries.

- `anti_vax` - [Real]

The probability that a person will be against vaccines for all diseases. A value of 1 means that the person is certainly anti-vaccine, a value of 0 means that the person is certainly pro-vaccine.

- `alone_prob` - [Real]

The raw probability that a Person will be alone, when not travelling, congregating, quarantining, or recovering in the hospital. The probability that the Person is not alone is then set to $1 - \text{alone\_prob}$. Set to `nan` to use `alone_prob_mean` and `alone_standard_deviation` to set the probability of being alone probabilistically using a normal distribution. `alone_prob` is always used if not set to `nan`.

- `alone_prob_mean` - [Real]

The average probability of a Person being alone when not travelling, congregating, quarantining, or recovering in the hospital. The mean is used along with the `alone_prob_standard_deviation` to calculate the actual probability of being alone (and as a consequence, the probability of not being alone) using a normal distribution.

- `alone_prob_standard_deviation` - [Real]

The standard deviation of the probability of a person being alone when not travelling, congregating, quarantining, or recovering in the hospital. The standard deviation is used along with the `alone_prob_mean` to calculate the actual probability of being alone (and as a consequence, the probability of not being alone) using a normal distribution.

- `travel_willingness` - [Real]

The likelihood that if a travel opportunity comes along, the person will choose to travel. A value of 1 means the person is certain to travel at every opportunity, a value of 0 means the person will never travel.

- `travel_time` - [Integer]

The raw amount of time, in frames, that a Person spends travelling each time they choose to do so. Set to `nan` to use `travel_time_mean` and `travel_time_standard_deviation`. `travel_time` is always used if it is set.

- `travel_time_mean` - [Integer]

The average amount of time that a Person spends travelling each time that they choose to travel. This mean is used, along with the `travel_time_standard_deviation` to calculate the actual travel duration of each trip with a normal distribution.

- `travel_time_standard_deviation` - [Integer]

The standard deviation of the time that a Person spends travelling each time that they choose to travel. This standard deviation is used, along with the `travel_time_mean` to calculate the actual travel time duration of each trip with a normal distribution.

- `travel_opportunity_separation` - [Integer]

The number of frames between opportunities to travel. Smaller values mean that the person has more opportunities to travel if they choose to (see `travel_willingness`), larger values mean fewer opportunities to travel if they choose to. Set to `nan` to use `travel_opportunity_separation_mean` and `travel_opportunity_separation_standard_deviation`. `travel_opportunity_separation` is always used if set.

- `travel_opportunity_separation_mean` - [Integer]

The average amount of time that a Person waits, in frames, between potential trips. This mean is used along with the `travel_opportunity_separation_standard_deviation` to calculate the actual length of time between travel opportunities with a normal distribution.

- `travel_opportunity_separation_standard_deviation` - [Integer]

The standard deviation of the amount of time that a Person waits, in frames, between potential trips. This means is used along with the `travel_opportunity_separation_mean` to calculate the actual length of time between travel opportunities with a normal distribution.

- `comply_with_lock_down_prob` - [Real]

The probability that a Person will comply with lock down orders, so long as lock down fatigue has not yet set in (see `time_before_lock_down_fatigue`). This can also be seen as the percentage of people in a society that comply with lock down, both formulations are equivalent.

- `comply_with_quarantine_prob` - [Real]

The probability that a Person will comply with quarantine orders received, for example after travelling or being contact traced.

- `congregation_willingness` - [Real]

The likelihood that if a travel opportunity comes along, the person will choose to travel. A value of 1 means the person is certain to congregate at every opportunity, a value of 0 means the person will never congregate.

- `congregation_visit_time` - [Integer]

The raw amount of time, in frames, that a Person spends at a congregation location, if they choose to do so. Set to `nan` to use `congregation_visit_time_mean` and `congregation_visit_time_standard_deviation`. `congregation_visit_time` is always used if it is set.

- `congregation_visit_time_mean` - [Integer]

The average amount of time that a Person spends, in frames, in the congregation location. This mean is used, along with the `congregation_visit_time_standard_deviation` to calculate the actual length of each congregation location visit with a normal distribution.

- `congregation_visit_time_standard_deviation` - [Integer]

The standard deviation of the amount of time that a Person spends, in frames, in the congregation location. This standard deviation is used, along with the `congregation_visit_time_mean` to calculate the actual length of each congregation location visit with a normal distribution.

- `congregation_visits_separation` - [Integer]

The amount of time between potential congregation locations visits. Set to `nan` to use `congregation_visits_separation_mean` and `congregation_visits_separation_standard_deviation`. `congregation_visits_separation_time` is always used if it is set.

- `congregation_visits_separation_mean` - [Integer]

The average amount of time that a Person waits, in frames, between potential visits to a congregation location. This mean is used, along with the `congregation_visits_separation_standard_deviation` to calculate the actual length of time between potential congregation location visits with a normal distribution.

- `congregation_visit_separation_standard_deviation` - [Integer]

The standard deviation of the amount of time that a Person waits, in frames, between potential visits to a congregation location. This standard deviation is used, along with the `congregation_visits_separation_mean` to calculate the actual length of time between potential congregation location visits with a normal distribution.

## Regions

There are two different ways to set values for Region objects: globally, meaning for all regions, or individually on a region by region basis. This section holds global default values for all the parameters that control how regions behave. If there are an insufficient number of `Region` entries are created given the value of `number_of_regions`, or if any `Region` entry is missing a variable assignment, or if one if its variable assignments is set to `nan`, then the default `Regions` value will be used. Note that wherever possible, `Region` values are preferred to `Regions` values.

- `name` - [Character]

The prefix to use when naming regions by default. The names of the regions are then "name-1", "name-2", "name-3", ... If the key "World" is passed, then the names of the countries of the world are used instead.

- `starting_number_of_people` - [Integer]

The number of people to start the region off with.

- `birth_rate` - [Real] $\geq 0, \leq 5$

The average number of children people born in the region will have in their lifetime.

- `life_span` - [Real] $\geq 25, \leq 100$

Average life span of people born in the region.

- `hospital_beds` - [Integer]

The number of non-ICU hospital beds available in the region.

- `icu_beds` - [Integer]

The number of ICU beds available in the region.

- `infection_tolerance_for_travel` - [Real]

The percentage of people that can be infected with any one disease variant before travel is forbidden to and from the region. 0 means shutdown travel if one person is infected. 1 means never shutdown travel.

- `infection_tolerance_for_lock_down` - [Real]

The percentage of people that can be infected with any one disease variant before lock down is imposed in the region. 0 means lock down if one person is infected. 1 means never lock down.

- `time_before_lock_down_fatigue` - [Integer]

The amount of time after the first lock down begins before lock downs are completely ignored in the region.

- `infection_tolerance_for_quarantine` - [Real]

The percentage of people that can be infected with any one disease variant in the region before quarantining is imposed. 0 means every sick person must always quarantine, 1 means quarantine is never imposed.

- `length_of_quarantine` - [Integer]

The amount of time that infected people must quarantine after either displaying symptoms or being informed that they are a close contact of a sick person.

- `region_contact_tracing` - [Logical]

Does the region do contact tracing?

- `infection_tolerance_for_contact_tracing` - [Real]

The percentage of people in a region that can be infected before contact tracing is stopped. 0 means never do contact tracing, 1 means always done.

- `contact_tracing_delay` - [Integer]

The amount of time after a sick person develops symptoms and when close contacts of the person are informed.

**Diseases**

Controls variables that are common to all variants of a given disease.

- `name` - [Character]

The name of the disease.

- `first_seen_on` - [Integer]

The frame number of the simulation when patient zero is infected with the first variant of this disease.

- `new_variant_prob` - [Real]

The probability that any new infection of this disease, of any variant, will be a new variant.

- `vaccine_for_disease` - [Logical]

Whether or not there will be a vaccine made available for any of the variants of this disease.

- `unlimited_new_variants` - [Logical]

Whether or not this disease is allowed to create an infinite number of new variants or must only create the variants supplied in the configuration file.

**Variants**

- `name` - [Character]

The name of the variant. This name is combined with the name of the disease with a "-".

- `first_seen_on` - [Integer]

The frame number of the simulation when patient zero of this variant is infected. Only applies if the disease itself is still active. If no new infections are triggered on this frame, this variant will appear on the next applicable frame. Leave as `nan` to allow the variant to first appear randomly.

- `alone_infection_modifier` - [Integer]

The relative likelihood that a person, when alone, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `not_alone_infection_modifier` - [Integer]

The relative likelihood that a person, when not alone (but also not congregating with a large number of other people), will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `travelling_infection_modifier` - [Integer]

The relative likelihood that a person, when travelling, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `congregation_infection_modifier` - [Integer]

The relative likelihood that a person, when congregating with a large number of other people, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `quarantine_infection_modifier` - [Integer]

The relative likelihood that a person, when quarantining, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `hospitalized_infection_modifier` - [Integer]

The relative likelihood that a person, when hospitalized but not in the ICU, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned to the Person object.

- `icu_infection_modifier` - [Integer]

The relative likelihood that a person, when in the ICU, will contract a new infection of this Variant. This value is relative to all of the other `infection_modifier` variables assigned

to the Person object.

- `transmission_prob` - [Real]

The probability that a new infection will be created on each frame. 0 is a disease that cannot spread, 1 is a disease that always spreads.

- `death_prob` - [Real]

The probability of succumbing to the disease.

- `asymptomatic_prob` - [Real]

The probability that a infection of this variant will leave the person asymptomatic. 0 means the person will always have symptoms, 1 means that the person will never have symptoms.

- `incubation_time` - [Integer]

The amount of time between the onset of the infection and the point where symptoms show up, if the person is not asymptomatic. During this stage, the infection is not contagious. `incubation_time` is an absolute value, if a distribution of incubation times are preferred, set it to `nan` and instead use `incubation_time_mean` and `incubation_time_standard_deviation`. `incubation_time` is always preferred if it is set.

- `incubation_time_mean` - [Integer]

The average amount of time between the onset of the infection and the point where symptoms show up, if the person is not asymptomatic. It is used along with `incubation_time_standard_deviation` to calculate the actual incubation time from a normal distribution.

- `incubation_time_standard_deviation` - [Integer]

The standard deviation of the time between the onset of the infection and the point where symptoms show up, if the person is not asymptomatic. It is used along with `incubation_time_mean` to calculate the actual incubation time from a normal distribution.

- `recovery_time` - [Integer]

The amount of time taken to recover after the recovery period is over. The person is infectious at this stage. `recovery_time` is an absolute value, if a distribution of recovery times are preferred, set it to `nan` and instead use `recovery_time_mean` and `recovery_time_standard_deviation`. `recovery_time` is always preferred if it is set.

- `recovery_time_mean` - [Integer]

The average amount of time for the person to recover or die from an infection of this variant. It is used along with `recovery_time_standard_deviation` to calculate the actual recovery time from a normal distribution.

- `recovery_time_standard_deviation` - [Integer]

The standard deviation of the amount of time for the person to recover or die from an infection of this variant. It is used along with `recovery_time_mean` to calculate the actual recovery time from a normal distribution.

- `vaccine_for_variant` - [Logical]

Whether or not there will be a vaccine for this variant.

- `vaccine_delay` - [Integer]

The amount of time between the onset of patient zero's infection and when the vaccine for the variant becomes available.

- `vaccine_round_delay` - [Integer]

The absolute amount of time between booster vaccines for this variant. `vaccine_round_delay` is an absolute value, if a distribution of vaccine round delays are preferred, set it to `nan` and instead use `vaccine_round_delay_mean` and `vaccine_round_delay_standard_deviation`. `vaccine_round_delay` is always

preferred if it is set.

- `vaccine_round_delay_mean` - [Integer]

The average amount of time between booster vaccines for this variant. `vaccine_round_delay_standard_deviation` to calculate the actual vaccine round delay from a normal distribution.

- `vaccine_round_delay_standard_deviation` - [Integer]

The standard deviation of the amount of time between booster vaccines for this variant. `vaccine_round_delay_mean` to calculate the actual vaccine round delay from a normal distribution.

- `vaccine_booster_effect` - [Real]

How effective is the vaccine at preventing infection? 1 is perfectly effective, 0 is perfectly ineffective.

- `vaccine_buildup_delay` - [Integer]

The amount of time before the vaccine has reached it's peak effectiveness.

- `vaccine_buildup_func` - [Character]

Must be one of "linear", "logarithmic", "exponential", "zero_step", or "one_step". Describes how the immunity changes after the vaccine has been administered but before it reaches full strength.

- `vaccine_full_protection_time` - [Integer]

How long does a vaccinated person remain at full protection? Use `inf` to indicate indefinite maximum protection.

- `vaccine_degradation_func` - [Character]

Must be one of "zero_step", "constant", "linear", "reciprocal", or "inverse_cosine". Indicates how the vaccine's effectiveness wanes over time.

- `vaccine_protection_degradation_time` - [Integer]

The amount of time that it takes for the vaccine's effectiveness to wear off. `vaccine_protection_degradation_time` is an absolute value, if a distribution of vaccine protection degradation times is preferred, set it to `nan` and instead use `vaccine_protection_degradation_time_mean` and `vaccine_protection_degradation_time_standard_deviation`. `vaccine_protection_degradation_time` is always preferred if set.

- `vaccine_protection_degradation_time_mean` - [Integer]

The average amount of time that it takes for the vaccine's effectiveness to wear off. Used with `vaccine_protection_degradation_time_standard_deviation` to calculate the actual vaccine protection degradation time from a normal distribution.

- `vaccine_protection_degradation_time_standard_deviation` - [Integer]

The standard deviation of the amount of time that it takes for the vaccine's effectiveness to wear off. Used with `vaccine_protection_degradation_time_mean` to calculate the actual vaccine protection degradation time from a normal distribution.

- `natural_effectiveness` - [Real]

How effective is natural immunity at preventing re-infection? 1 is perfectly effective, 0 is perfectly ineffective.

- `full_natural_protection_time` - [Integer]

How long does a person who has recovered from the disease have full protection? Use `inf` to indicate indefinite maximum protection.

- `natural_effectiveness_degradation_func` - [Character]

Must be one of "zero_step", "constant", "linear", "reciprocal", or "inverse_cosine".

- `natural_effectiveness_degradation_time` - [Integer]

The amount of time that it takes for the vaccine's effectiveness to wear off. `natural_effectiveness_degradation_time` is an absolute value, if a distribution of natural effectiveness degradation times is preferred, set it to `nan` and instead use `natural_effectiveness_degradation_time_mean` and `natural_effectiveness_degradation_time_standard_deviation`. `natural_effectiveness_degradation_time` is always preferred if set.

- `natural_effectiveness_degradation_time_mean` - [Integer]

The average amount of time that it takes for the vaccine's effectiveness to wear off. Used with `natural_effectiveness_degradation_time_standard_deviation` to calculate the actual natural effectiveness degradation time from a normal distribution.

- `natural_effectiveness_degradation_time_standard_deviation` - [Integer]

The standard deviation of the amount of time that it takes for the vaccine's effectiveness to wear off. Used with `natural_effectiveness_degradation_time_mean` to calculate the actual natural effectiveness degradation time from a normal distribution.

- `vaccine_booster_effect_against_variants` - [Real]

How effective is the vaccine at preventing infection from other variants of the disease? 1 is perfectly effective, 0 is perfectly ineffective.

- `vaccine_full_protection_time_against_variants`

How long does a vaccinated person remain at full protection against other variants of the disease? Use `inf` to indicate indefinite maximum protection.

- `vaccine_degradation_against_variants_func`

Must be one of "zero_step", "constant", "linear", "reciprocal", or "inverse_cosine". Indicates how the vaccine's effectiveness against variants wanes over time.

- `vaccine_protection_degradation_against_variants_time` - [Integer]

The amount of time that it takes for the vaccine's effectiveness to wear off. `vaccine_protection_degradation_against_variants_time` is an absolute value, if a distribution of vaccine protection degradation against variants times is preferred, set it to `nan` and instead use `vaccine_protection_degradation_against_variants_time_mean` and `vaccine_protection_degradation_against_variants_time_standard_deviation`. `vaccine_protection_degradation_against_variants_time` is always preferred if set.

- `vaccine_protection_degradation_against_variants_time_mean` - [Integer]

The average amount of time that it takes for the vaccine's effectiveness to wear off. Used with `vaccine_protection_degradation_against_variants_time_standard_deviation` to calculate the actual vaccine protection degradation against variants time from a normal distribution.

- `vaccine_protection_degradation_against_variants_time_standard_deviation` - [Integer]

The standard deviation of the amount of time that it takes for the vaccine's effectiveness to wear off. Used with `vaccine_protection_degradation_against_variants_time_mean` to calculate the actual vaccine protection degradation against variants time from a normal distribution.

- `natural_effectiveness_against_variants` - [Real]

How effective is natural immunity at preventing infection from other variants of the disease? 1 is perfectly effective, 0 is perfectly ineffective.

- `full_natural_protection_time_against_variants`

How long does a person who has recovered from the variant remain at full protection against other variants of the disease? Use `inf` to indicate indefinite maximum protection.

- `natural_effectiveness_degradation_against_variants_func`

Must be one of "zero_step", "constant", "linear", "reciprocal", or "inverse_cosine". Indicates how natural effectiveness against variants wanes over time.

- `natural_effectiveness_degradation_against_variants_time` - [Integer]

The amount of time that it takes for the vaccine's effectiveness to wear off. `natural_effectiveness_degradation_against_variants_time` is an absolute value, if a distribution of natural effectiveness degradation against variants times is preferred, set it to `nan` and instead use `natural_effectiveness_degradation_against_variants_time_mean` and `natural_effectiveness_degradation_against_variants_time_standard_deviation`. `natural_effectiveness_degradation_against_variants_time` is always preferred if set.

- `natural_effectiveness_degradation_against_variants_time_mean` - [Integer]

The average amount of time that it takes for the vaccine's effectiveness to wear off. Used with `natural_effectiveness_degradation_against_variants_time_standard_deviation` to calculate the actual natural effectiveness degradation against variants time from a normal distribution.

- `natural_effectiveness_degradation_against_variants_time_standard_deviation` - [Integer]

The standard deviation of the amount of time that it takes for the vaccine's effectiveness to wear off. Used with `natural_effectiveness_degradation_against_variants_time_mean`

- 

- `hospitalization_prob` - [Real]

The probability that an infection of this variant will require hospitalization. `hospitalization_admission_delay` - [Integer]

The amount of time between the end of the incubation period and the person's admission to the hospital, if necessary. `hospitalization_admission_delay` is an absolute value, if a distribution of hospitalization admission delays are preferred, set it to `nan` and instead use `hospitalization_admission_delay_mean` and `hospitalization_admission_delay_standard_deviation`. `hospitalization_admission_delay` is always preferred if it is set.

- `hospitalization_admission_delay_mean` - [Integer]

The average amount of time between the end of the incubation period and the person's admission to the hospital, if necessary. `hospitalization_admission_delay_standard_deviation` to calculate the actual hospitalization admission delay from a normal distribution.

- `hospitalization_admission_delay_standard_deviation` - [Integer]

The standard deviation of the amount of time between the end of the incubation period and the person's admission to the hospital, if necessary. `hospitalization_admission_delay_mean` to calculate the actual hospitalization admission delay from a normal distribution.

- `icu_prob` - [Real]

The probability that an infection of this variant will require a visit to the ICU. Only triggered if the person is already hospitalized.

- `icu_admission_delay` - [Integer]

The amount of time between the person's admission to the hospital and the person's admission to the ICU, if necessary. `icu_admission_delay` is an absolute value, if a distribution of ICU admission delays are preferred, set it to `nan` and instead use `icu_admission_delay_mean` and `icu_admission_delay_standard_deviation`. `icu_admission_delay` is always preferred if it is set.

- `icu_admission_delay_mean` - [Integer]

The average amount of time between the person's admission to the hospital and the person's admission to the ICU, if necessary. `icu_admission_delay_standard_deviation` to calculate the actual ICU admission delay from a normal distribution.

- `icu_admission_delay_standard_deviation` - [Integer]

The standard deviation of the amount of time between the person's admission to the hospital and the person's admission to the ICU, if necessary. `icu_admission_delay_mean` to calculate the actual ICU admission delay from a normal distribution.

# Object Specifications

## Utilities

These are utility functions that don't belong inside of a single class since they may be shared between various parts of the program.

### Procedures

- [Real $\rightarrow$ Real] `f_constant_one`

A constant function that always returns one.

$$f(x) = 1$$

| **UT_1:** **[Math Check]** |
|---|
| Checks that: $f(0) = f\left(\dfrac{1}{2}\right) = f(1) = 1$. |

- [Real $\rightarrow$ Real] `f_constant_zero`

A constant function that returns zero.

$$f(x) = 0$$

| **UT_2:** **[Math Check]** |
|---|
| Checks that: $f(0) = f\left(\dfrac{1}{2}\right) = f(1) = 0$. |

- [Real $\rightarrow$ Real] `f_linear_dec`

A linear function that deceases at a constant rate from $x = 0$ to $x = 1$. Satifies:

$$\begin{cases} f(0) = 1 \\ f(1) = 0 \end{cases}$$

Defined as:

$$f(x) = -x + 1$$

| **UT_3:** **[Math Check]** |
|---|
| Checks that: $f(0) = 1$, $f\left(\dfrac{1}{2}\right) = \dfrac{1}{2}$, and $f(1) = 0$. |

- [Real → Real] `f_linear_inc`

A linear function that increases at a constant rate from $x = 0$ to $x = 1$. Satisfies:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Defined as:

$$f(x) = x$$

| **UT_4:** | **[Math Check]** |
|---|---|
| | Checks that: $f(0) = 0$, $f\left(\dfrac{1}{2}\right) = \dfrac{1}{2}$, and $f(1) = 1$. |

- [Real → Real] `f_zero_step_down`

A function which immediately steps down to zero. Satisfies:

$$\begin{cases} f(0) = 1 \\ f(1) = 0 \end{cases}$$

Defined as:

$$f(x) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0 \end{cases}$$

| **UT_5:** | **[Math Check]** |
|---|---|
| | Checks that: $f(0) = 1$, $f\left(\dfrac{1}{2}\right) = 0$, and $f(1) = 0$. |

- [Real → Real] `f_zero_step_up`

A function which immediately steps up to one. Satisfies:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Defined as:

$$f(x) = \begin{cases} 0, & x = 0 \\ 1, & x \neq 0 \end{cases}$$

| **UT_6:** | **[Math Check]** |
|---|---|
| | Checks that: $f(0) = 0$, $f\left(\dfrac{1}{2}\right) = 1$, and $f(1) = 1$. |

- [Real → Real] `f_one_step_down`

A function which steps down to zero at one. Satisfies:

$$\begin{cases} f(0) = 1 \\ f(1) = 0 \end{cases}$$

Defined as:

$$f(x) = \begin{cases} 0, & x = 1 \\ 1, & x \neq 1 \end{cases}$$

---

**UT_7:    [Math Check]**

Checks that: $f(0) = 1$, $f\left(\dfrac{1}{2}\right) = 0$, and $f(1) = 0$.

---

- [Real → Real] `f_one_step_up`

A function which steps up to one at one. Satisfies:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Defined as:

$$f(x) = \begin{cases} 1, & x = 1 \\ 0, & x \neq 1 \end{cases}$$

---

**UT_8:    [Math Check]**

Checks that: $f(0) = 0$, $f\left(\dfrac{1}{2}\right) = 0$, and $f(1) = 1$.

---

- [Real → Real] `f_reciprocal`

A reciprocal function which monotonically decreases. Satisfies:

$$\begin{cases} f(0) = 1 \\ f(1) = 0 \end{cases}$$

Defined as:

$$f(x) = \frac{2}{x + 1} - 1$$

---

**UT_9:    [Math Check]**

Checks that: $f(0) = 1$, $f\left(\dfrac{1}{2}\right) = \dfrac{1}{3}$, and $f(1) = 0$.

---

- [Real → Real] `f_inverse_cosine`

A trigonomic function which decreases monotonically. Satisfies:

$$\begin{cases} f(0) = 1 \\ f(1) = 0 \end{cases}$$

Defined as:

$$f(x) = \frac{2}{\pi} \cos^{-1}(x)$$

---

**UT_10:   [Math Check]**

Checks that: $f(0) = 1$, $f\left(\dfrac{1}{2}\right) = \dfrac{2}{3}$, and $f(1) = 0$.

---

- [Real → Real] `f_logarithmic`

A logarithmic function which satisfies:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Defined as:

$$f(x) = \log_2(x + 1)$$

---

**UT_11:   [Math Check]**

Checks that: $f(0) = 0$, $f\left(\dfrac{1}{2}\right) \approx 0.5849625$, and $f(1) = 1$.

---

- [Real → Real] `f_exponential`

An exonential function which satisfies:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Defined as:

$$f(x) = 2^x - 1$$

---

**UT_12:   [Math Check]**

Checks that: $f(0) = 0$, $f\left(\dfrac{1}{2}\right) \approx 0.41421356$, and $f(1) = 1$.

---

**Properties**

- [Int] `test_seed`
  *test_seed* = 42

The random seed used for all applicable tests.

- [Int] `test_small_region_people`
  *test_small_region_people* = 1,000

A small initial starting population for all relevant tests.

- [Int] `test_large_region_people`
  *test_large_region_people* = 1,000,000

A large initial starting population for all relevant tests.

- [Real] `test_zero_birth_rate`
  *test_zero_birth_rate* = 0

A birthrate of zero for applicable tests.

- [Real] `test_one_birth_rate`
  *test_one_birth_rate* = 1

A birthrate that perfectly replaces the population (on average).

- [Real] `test_two_birth_rate`
  *test_two_birth_rate* = 2

A birthrate that increases the size of the population.

- [Int] `test_lifespan_avg`
  *test_lifespan_avg* = 50

The average lifespan to use in appropriate tests.

- [Int] `test_single_n_regions`
  *test_single_n_regions* = 1

A one region globe to use in appropriate tests.

- [Int] `test_multiple_n_regions`
  *test_multiple_n_regions* = 10

A ten region globe to use in appropriate tests.

- [Real] `test_zero_transmission_prob`
  *test_zero_transmission_prob* = 0

A zero transmission probability for use in appropriate tests.

- [Real] `test_small_transmission_prob`
  *test_small_transmission_prob* = 3

A small transmission probability for use in appropriate tests.

- [Real] `test_large_transmission_prob`
  *test_large_transmission_prob* = 50

A large transmission probability for use in appropriate tests.

- [Int] `test_quick_recovery_time`
  *test_quick_recovery_time* = 1

A quick recovery time for appropriate tests.

- [Int] `test_long_recovery_time`
  *test_long_recovery_time* = 100

A long recovery time for appropriate tests.

- [Int] `test_zero_incubation_time`
  *test_zero_incubation_time* = 0

A zero incubation time for appropriate tests.

- [Int] `test_quick_incubation_time`
  *test_quick_incubation_time* = 5

A short incubation time for appropriate tests.

- [Int] `test_long_incubation_time`
  *test_long_incubation_time* = 50

A long incubation time for appropriate tests.

- [Real] `test_zero_death_probability`
  *test_zero_death_probability* = 0

A zero probability of death for appropriate tests.

- [Real] `test_normal_death_probability`
  *test_normal_death_probability* = 50

A normal probability of death for appropriate tests.

- [Real] `test_certain_death_probability`
  *test_certain_death_probability* = 100

A certain probability of death for appropriate tests.

- [Real] `test_zero_asymptomatic_probability`
  *test_zero_asymptomatic_probability* = 0

A zero probability of being asymptomatic for appropriate tests.

- [Real] `test_normal_asymptomatic_probability`
  *test_normal_asymptomatic_probability* = 50

A normal probability of being asymptomatic for appropriate tests.

- [Real] `test_certain_asymptomatic_probability`
  *test_certain_asymptomatic_probability* = 100

A certain probability of being asymptomatic for appropriate tests.

- [Real] `test_zero_probability_new_variant`
  *test_zero_probability_new_variant* = 0

A zero probability that any given person will be patient zero for a new variant for appropriate tests.

- [Real] `test_normal_probability_new_variant`
  *test_normal_probability_new_variant* = 0.5

A normal probability that any given person will be patient zero for a new variant for appropriate tests.

- [Logical] `test_true_vaccine`
  *test_true_vaccine* = *True*

There will be a vaccine for appropriate tests.

- [Logical] `test_false_vaccine`
  *test_false_vaccine* = *False*

There will not be a vaccine for appropriate tests.

- [Int] `test_zero_vaccine_delay`
  *test_zero_vaccine_delay* = 0

A zero delay for the vaccine to be developed after the disease appears for appropriate tests.

- [Int] `test_normal_vaccine_delay`
  *test_normal_vaccine_delay* = 10, 000

A normal delay for the vaccine to be developed after the disease appears for appropriate tests.

- [Logical] `test_true_variant_vaccine`
  *test_true_variant_vaccine* = *True*

There will be a variant vaccine for appropriate tests.

- [Logical] `test_false_variant_vaccine`
  *test_false_variant_vaccine* = *False*

There will not be a variant vaccine for appropriate tests.

## Globe Class

The globe class contains all of the regions that are being simulated. It controls the passage of time. The globe class implements the following procedures and properties:

**Procedures**

**Properties**

- [Int] `current_frame`

The current frame of the simulation as it is being created.

- [Int] `n_regions`

The number of regions to simulate.

- [Int] `next_region_id`

The next id to associate to a region.

- [Int] `next_person_id`

The next id to associate to a person.

- [Int] `next_infection_id`

The next id to associate to an infection.

- [Int] `next_disease_id`

The next id to associate with a disease.

- [Int] `next_variant_id`

The next id to associate with a variant.

- [Array of Arrays of Ints] `birth_schedule`

The schedule for births of new Person objects into the Globe. It is of shape (n_frames x n_regions) and the value of each element represents the number of Person objects to be created on that frame and in that region.

- [Array of Arrays of Arrays of Arrays of Ints] `infection_schedule`

The schedule for new infections on the Globe. It is of shape (n_frames x n_diseases x n_variants x n_regions) and the value of each element represents the number of Infection objects to be created on that frame and in that region of that disease and of that variant.

## Region Class

The region class represents a uniform area within the simulation. It contains a number of people objects and keeps track of properties and methods that are shared amongst the people it contains. The region class implements the following procedures and properties:

**Procedures**

**Properties**

- [Int] `region_id`

The id number of the region.

- [Int] `region_people`

Represents the number of living people that are currently assigned to the region. Can be set by the user to reflect the initial value when the simulation starts, but changes as people die and reproduce. Does not reflect the number of people currently inside of the physical boundaries of the region (if travel between regions is permitted) only the number of people assigned to the region.

- [Real] `birth_rate`

Represents the average number of children that the members of a region will have in their lives. More specifically, `birth_rate` can be thought of as the expected value of the number of children per person random variable. Thus, the probability *p* that a person will "give birth" to another person on any given frame is given by:

$$p = \frac{\text{birth\_rate}}{\text{time\_to\_frames(average\_life\_span)}}$$

- [Int] `lifespan_avg`

Represents the average number of years that any given person in a region will live. The actual lifespan of each person is calculated upon the construction of the Person object using a normal distribution with the `lifespan_avg` as the mean and a standard_deviation of five years.

## Person Class

The person class represents an individual within the simulation. They are the smallest possible unit of simulation. The person class has the following procedures and properties:

**Procedures**

**Properties**

- [Int] `person_id`

The id number of the person.

- [Logical] `alive`

Denotes whether or not the person is alive. If alive, a person can move, get sick, spread disease, and potentially increase the number of people being simulated (if natural birth rate allows). Every person starts alive and becomes dead. Dead people cannot come back to life.

- [Int] `age`

The age (in frames) of the person.

- [Int] `death_age`

Denotes the age in minutes at which the Person will die assuming that they do not succumb to the disease or any of its variants. If a person dies naturally, they will die randomly at some point during the year of their death.

- [Logical] `anti_vaxer`
  *anti_vaxer* ∈ {True, False}

Whether or not the person is willing to be vaccinated or not. Moot if there is no vaccine available.

- [Logical] `infected`
  *infected* ∈ {True, False}

Whether or not the person has at least one active infection. If any of the infections that the person has contracted is not in the recovered state and the person is alive, then this value is true.

- [Logical] `asymptomatic`
  *aymptomatic* ∈ {True, False}

Whether or not the person is asymptomatic. Note that if even one active disease is symptomatic, this value must be false.

- [Logical] `symptomatic`
  *symptomatic* ∈ {True, False}

Whether or not the person is symptomatic. Note that if even one active disease is symptomatic, this value must be true.

- [Logical] `healthy`
  *healthy* ∈ {True, False}

Whether the person is healthy. The person is only healthy if none of the infections associated with him are active.

- [Logical] `quarantined`
  *quarantined* $\in$ {True, False}

Whether or not the person is quarantined.

- [Logical] `travelling`
  *travelling* $\in$ {True, False}

Whether or not the person is actively traveling between regions.

- [Logical] `in_hospital`
  *in_hospital* $\in$ {True, False}

Whether or not the person is in the hospital due to an active infection.

- [Logical] `in_icu`
  *in_icu* $\in$ {True, False}

Whether or not the person is in the ICU due to an active infection.

- [Array of Ints] `infection_ids`
  *infection_ids* $= \{id_1, id_2, \cdots id_n | id_i \in \mathbb{R}\}$

An array of infection ids associated with the person.

- [Int] `recovered_infection_count`

The number of associated infections from which the person has recovered.

- [Int] `current_infection_count`

The number of current, active infections assoicated with this person.

- [Int] `time_spent_healthy`

The number of frames this person has been healthy for. (Asymptomatic is not healthy.)

- [Int] `time_spent_sick`

The number of frames this person has been sick for.

- [Int] `time_spent_symptomatic`

The number of frames this person has been symptomatic for.

- [Int] `time_spent_asmyptomatic`

The number of frames this person has been asymptomatic for.

- [Int] `time_spent_quarantined`

The number of frames this person has been quarantined/locked down for.

- [Array of Logical] `healthy_time_series`

The historical record of healthiness for the person.

- [Array of Logical] `sick_time_series`

The historical record of sickness for the person.

- [Array of Logical] `symptomatic_time_series`

The historical record of symptomatic state for the person.

- [Array of Logical] `asymptomatic_time_series`

The historical record of asmymptomatic state for the person.

- [Array of Logical] `quarantined_time_series`

The historical record of quarantined state for the person.

- [Array of Int] `state_time_series`

The state of the person throughout their life, frame by frame. There are 6 states that a person can be in, each denoted by an integer and accompanied by an assoicated infection risk.

1. [0] Alone (×1 or `alone_infection_modifier`)
2. [1] Not Alone (×20 or `not_alone_infection_modifier`)
3. [2] Travelling (×50 or `travelling_infection_modifier`)
4. [3] Quarantined (×0 or `quarantined_infection_modifier`)
5. [4] Hospitalized (×15 or `hospitalized_infection_modifier`)
6. [5] In ICU (×10 or `icu_infection_modifier`)

The `state_time_series` is a record of the person's state throughout the simulation of their life.

## Disease Class

The disease class contains all of the information that is shared between all the variants of the disease. The variant class inherits from the disease class. This class implements the following procedures and properties:

**Procedures**

- [int, char] => [int] `time_to_frames(value, code)`

Converts a time in minutes, hours, days, weeks, months, or years to frames. `value` represents the absolute number of time units to convert and `code` represents the scale of the value ("m" for minutes, "h" for hours, "d" for days, "w" for weeks, "n" for months, and "y" for years). Uses the following formula:

$$f = \begin{cases} m \\ 60 \times h \\ 1440 \times d \\ 10080 \times w \\ 43830 \times n \\ 525960 \times y \end{cases}$$

depending on the `code` passed to the method.

**Properties**

- [Character] `name`

The name of the disease. Defaults to `disease_1`, `disease_2`, etc.

- [Int] `disease_id`

The id number of the disease.

- [Real] `new_var_prob`
  $0 \le new\_var\_prob \le 1$

The probability that when any given person contracts any variant of this disease that they will instead be patient zero of a new variant of the disease.

- [Logical] `disease_vaccine`
  $disease\_vaccine \in \{True, False\}$

Will a vaccine ever be made available for this disease and all of its variants?

## Variant Class

The variant class contains all of the information that is specific to the one variant of the disease itself. It inherits from the disease class. This class implements the following procedures and properties:

**Procedures**

**Properties**

- [Character] `name`

The name of the variant, if provided. If not specified, use lowercase Greek letters.

- [Int] `variant_id`

The id number of the variant.

- [Real] `transmission_prob`
  $0 \leq transmission\_prob \leq 1$

Represents the probability that one person will get infected by this variant if they come into contact with an infected individual for one frame.

- [Int] `recovery_time_mu`
  $1 \leq recovery\_time\_mu \leq 525,960$

Represents the mean of the normal distribution of recovery time, measured in frames (in human readable form: 1 minute to 1 year). Recovery can end with either death or a state of health. Recovery time does not start until incubation time is up.

- [Real] `recovery_time_sd`
  $0 \leq recovery\_time\_sd < 144,000$

The standard deviation of recovery time for the variant. Ranges from 0 deviation to 100 days of deviation.

- [Int] `incubation_time_mu`
  $0 \leq incubation\_time\_mu \leq 288,000$

Represents the mean of the normal distribution of incubation time, measured in frames (in human readable form: instantly to 200 days). It is the amount of time between the moment of infection and the moment that symptoms appear (if symptomatic). During this incubation period, the infected individual can not spread the disease, recover from the disease, or die from the disease.

- [Real] `incubation_time_sd`
  $0 \leq incubation\_time\_sd \leq 144,000$

The standard deviation of incubation time for the variant. Ranges from 0 deviation to 100 days of deviation.

- [Real] `death_prob`
  $0 \leq death\_prob \leq 1$

Represents the mean of the normal distribution of the probability of death from the variant of the disease, measured in frames. Note that this normal distribution is capped at both ends, zero and one. The probability of surviving the disease or successfully recovering is given as $1 - death\_prob$.

- [Real] `asymptomatic_prob`
  $0 \leq asymptomatic\_prob \leq 1$

Represents the mean of the normal distribution of the probability that an infected individual will be asymptomatic after the incubation period is over, measured in frames. The probability that someone will be symptomatic is given by $1 - asymptomatic\_prob$.

- [Logical] `variant_vaccine`
  *variant_vaccine* ∈ {True, False}

Does this particular variant ever get a vaccine? Only matters if the value of the parent Disease object has `disease_vaccine` set to True.

- [Int] `vaccine_delay`
  0 ≤ *vaccine_delay* ≤ 144, 000, 000

The amount of time between the first incubation day for patient zero of the variant and the time when the vaccine first becomes available, measured in frames. Note that the range spans up to 100,000 days. Only applies if both the parent Disease object has `disease_vaccine` set to True and the variant object has `variant_vaccine` set to True.

- [Int] `vac_round_delay_mu`
  1 ≤ *vac_round_delay_mu* ≤ 14, 400, 000

Represents the mean of the normal distribution of the delay between vaccination rounds, measured in frames. It is capped on the lower end at one. There must be at least one frame between vaccination rounds.

- [Real] `vac_round_delay_sd`
  0 ≤ *vac_round_delay_sd* ≤ 5, 259, 600

The standard deviation for the delay between vaccination rounds. Ranges from precisely 0 deviation to 10 years of deviation.

- [Real] `vac_booster_effect`
  0 ≤ *vac_booster_effect* ≤ 100

Represents the probability that the booster will prevent an individual from contracting the disease the moment that the shot takes full effect. This protection level can decay over time.

- [Int] `vaccine_buildup_delay`
  0 ≤ *vaccine_buildup_delay* ≤ 525, 960

The amount of time, in frames, that it takes for the vaccine to reach full effectiveness: `vac_booster_effect`.

- [Function] `vaccine_buildup_func`
  $f: \{x | 0 \leq x \leq 1, x \in \mathbb{R}\} \rightarrow \{y | 0 \leq y \leq 1, y \in \mathbb{R}\}$

A function that describes how the immunity should be built up from the frame that the person receives the vaccine to the last frame of the `vaccine_buildup_delay`. The input *x* represents `vaccine_buildup_delay` rescaled to [0, 1] and the output *y* is a multiplier to `vac_booster_effect`. The function must conform to the following edge cases:

$$\begin{cases} f(0) = 0 \\ f(1) = 1 \end{cases}$$

Several prebuilt functions will be provided for this purpose, namely:

- `linear`
- `logarithmic`
- `exponential`
- `zero_step`
- `one_step`

- [Int] `vac_full_protection_time`
  *full_protection_time* ≥ 0

Represents the amount of time that the vaccine booster shot retains its effectiveness after reaching full effectiveness, in frames. If set to the special value of infinity, the effectiveness lasts forever.

- [Function] `vaccine_degredation_func`
  $f : \{x | x \geq 0, x \in \mathbb{R}\} \rightarrow \{y | 0 \leq y \leq 1, y \in \mathbb{R}\}$

A function that determines how the immunity granted by a vaccine varies over time. It starts at the moment that `vaccine_buildup_delay` ends. It continues indefinitely. It's effects stack with subsequent vaccines that are received for the same variant and with the natural immunity gained through recovery. The function must confom to the following edge cases:

$$\{f(0) = 1$$

Some prebuilt functions will be provided for this purpose, namely:

- `zero_step`
- `constant`
- `linear`
- `reciprocal`
- `inverse_cosine`

- [Real] `natural_effect`
  $0 \leq natural\_effect \leq 100$

Represents the probability that a person will be protected from contracting the same variant again by the natural immunity that they build up from getting the disease and then recovering. This level of protection is achieved as soon as the individual recovers from the disease and can decline over time.

- [Function] `natural_degredation_func`
  $f : \{x | x \geq 0, x \in \mathbb{R}\} \rightarrow \{y | 0 \leq y \leq 1, y \in \mathbb{R}\}$

A function that determines how the immunity granted by infection varies over time. It starts at the moment that a person recovers from the variant. It continues indefinitely. It's effects stack with vaccines that are received for the same variant. The function must confom to the following edge cases:

$$\{f(0) = 1$$

Some prebuilt functions will be provided for this purpose, namely:

- `zero_step`
- `constant`
- `linear`
- `reciprocal`
- `inverse_cosine`

- [Real] `hospital_prob`
  $0 \leq hospital\_prob \leq 100$

The probability that an infected individual will require hospitalization, but *not* ICU treatment.

- [Int] `hospital_delay_mu`
  $0 \leq hospital\_delay\_mu \leq 525,960$

Represents the mean of the normal distribution of hospital (but not ICU) admittance delay, measured in frames. (In human readable form: 0 minutes to 1 year). This represents the amount of time after the incubation time is up before the individual requires hospitalization.

- [Int] `hospital_delay_sd`
  $0 \leq hospital\_delay\_sd \leq 262,980$

Represents the standard deviation of the normal distribution of hospital admittance delay, measured in frames. (In human readable form: 0 minutes to 1/2 a year).

- [Real] `icu_prob`
  $0 \le icu\_prob \le 100$

The probability that an infected individual will require a stay in the ICU. Note that this does not relate whatsoever to the probability that an individual will require hospitalization.

- [Int] `icu_delay_mu`
  $0 \le icu\_delay\_mu \le 525,960$

Represents the mean of the normal distribution of ICU admittance delay, measured in frames. (In human readable form: 0 minutes to 1 year). This represents the amount of time:

- After incubation time has concluded.
- Before the individual has recovered or died.
- After the individual has been admitted to the hospital.

- [Int] `icu_delay_sd`
  $0 \le icu\_delay\_sd \le 262,980$

Represents the standard deviation of the normal distrubtuion of ICU admittance delay, measured in frames. (In human readable form: 0 minutes to 1/2 a year).

## Infection Class

The infection class describes the specific infection of an individual person. It inherits from the variant class and is associated with the person class. The infection class has the following procedures and properties:

**Procedures**

**Properties**

- [Int] `infection_id`

The unique id assigned to the infection.

- [Int] `disease_id`

The id of the disease associated with this infection.

- [Int] `variant_id`

The id of the variant of the disease associated with this infection.

- [Int] `incubation_time`

The actual incubation time of this infection, in frames.

- [Int] `recovery_time`

The actual recovery time of this infection, after which the patient is either healthy or dead, in frames.

- [Logical] `symptomatic`

Whether or not the infection is symptomatic.

- [Int] `hospital_delay`

The actual time between the end of the incubation time for the disease, and the admission of the associated person to the hospital as a consequence of this infection.

- [Int] `icu_delay`

The actual time between the admitance of the person to the hospital because of this infection, and the time when the person is moved to ICU because of this infection.

- [Int] `quarantine_time`

The amount of time that the person spent in quarantine because of this infection.

- [Logical] `killed_person`

Whether or not this infection killed it's associated person.

- [Int] `infection_onset`

The global frame number associated with the onset of the infection.

- [Int] `infection_duration`

The amount of time, in frames, from the initialization of this disease to the point when the associated person either dies or recovers.

- [Array of Int] `state_time_series`

The state of the infection throughout it's lifespan, frame by frame. There are 5 states that an infection can be in, each denoted by an integer and accompanied by an infectability modifier.

1. [0] Incubation ($\times 1$ or `incubation_infectivity_modifier`)
2. [1] Asymptomatic ($\times 10$ or `asymptomatic_infectivity_modifier`)
3. [2] Symptomatic ($\times 30$ or `symptomatic_infectivity_modifier`)
4. [3] Survived ($\times 0$ or `survived_infectivity_modifier`)
5. [4] Deceased ($\times 0$ or `deceased_infectivity_modifier`)

## Results

This object collects all of the interesting simulation data and produces plots and data files. When a person object is dead, it is deleted (freeing the memory allocated to it) and it's stats are transfered to the single instantiated Results object.

### Procedures

### Properties

- [Integer] `num_people_alive`

The current number of people alive on the entire globe.

- [Array of Integers] `num_people_alive_by_region`

The current number of people alive by region.

# Appendix A

This is a sample configuration file that dm can handle:

```
[General]
### High Level Program Parameters
verbose = boolean


[Output]
### Controls the Output of the Simulation


output_file = 'string'
use_gnuplot = boolean


[Termination]
### Puts a Limits on how Long the Simulation will Run
```

```
maximum_simulation_length = 123


[Globe]
### Impacts the State of the Earth, to Start


number_of_regions = 123


[Globe.Persons]
### Impacts Person Specific Parameters, but Applies to All Person Across
Regions as a Default if not Specified in Region.Person


anti_vax = 0.123


alone_prob = 0.123
alone_prob_mean = 0.123
alone_prob_standard_deviation = 0.123


travel_willingness = 0.123
travel_time = 123
travel_time_mean = 123
travel_time_standard_deviation = 123
travel_opportunity_separation = 123
travel_opportunity_separation_mean = 123
travel_opportunity_separation_standard_deviation = 123


comply_with_lock_down_prob = 0.123
comply_with_quarantine_prob = 0.123


congregation_willingness = 0.123
congregation_visit_time = 123
congregation_visit_time_mean = 123
congregation_visit_time_standard_deviation = 123


congregation_visits_separation = 123
congregation_visits_separation_mean = 123
congregation_visits_separation_standard_deviation = 123


[Regions]
### Impacts Region Specific Parameters, but Applies to All Regions Iden-
tically
```

```
name = 'string'


starting_number_of_people = 123
birth_rate = nan
life_span = nan


hospital_beds = nan
icu_beds = nan


infection_tolerance_for_travel = 0.123
infection_tolerance_for_lock_down = 0.123
time_before_lock_down_fatigue = 123
infection_tolerance_for_quarantine = 0.123
length_of_quarantine = 123


region_contact_tracing = boolean
infection_tolerance_for_contact_tracing = 0.123
contact_tracing_delay = 123


[[Region]]
### Impacts Region Specific Parameters, but Allows for Region by Region
Specificity


name = 'string'


starting_number_of_people = 123
birth_rate = nan
life_span = nan


hospital_beds = nan
icu_beds = nan


infection_tolerance_for_travel = 0.123
infection_tolerance_for_lock_down = 0.123
time_before_lock_down_fatigue = 123
infection_tolerance_for_quarantine = 0.123
length_of_quarantine = 123


region_contact_tracing = boolean
infection_tolerance_for_contact_tracing = 0.123
contact_tracing_delay = 123
```

```
[Region.Person]
### Impacts Person Level Details on a Region by Region Level

anti_vax = 0.123

alone_prob = 0.123
alone_prob_mean = 0.123
alone_prob_standard_deviation = 0.123

travel_willingness = 0.123
travel_time = 123
travel_time_mean = 123
travel_time_standard_deviation = 123
travel_opportunity_separation = 123
travel_opportunity_separation_mean = 123
travel_opportunity_separation_standard_deviation = 123

comply_with_lock_down_prob = 0.123
comply_with_quarantine_prob = 0.123

congregation_willingness = 0.123
congregation_visit_time = 123
congregation_visit_time_mean = 123
congregation_visit_time_standard_deviation = 123

congregation_visits_separation = 123
congregation_visits_separation_mean = 123
congregation_visits_separation_standard_deviation = 123

[[Disease]]
### Impacts the Disease Details

name = 'string'
first_seen_on = nan
new_variant_prob = 0.123
vaccine_for_disease = boolean
unlimited_new_variants = boolean

[[Disease.Variant]]
### Impacts Disease Variant Level Details
```

```
name = 'string'
first_seen_on = nan

alone_infection_modifier = 123
not_alone_infection_modifier = 123
travelling_infection_modifer = 123
congregation_infection_modifier = 123
quarantine_infection_modifier = 0
hospitalized_infection_modifier = 123
icu_infection_modifier = 123

transmission_prob = 0.123
death_prob = 0.123
asymptomatic_prob = 0.123

incubation_time = nan
incubation_mean = 123
incubation_standard_deviation = 123

recovery_time = nan
recovery_mean = 123
recovery_standard_deviation = 123

vaccine_for_variant = boolean
vaccine_delay = 123
vaccine_round_delay_mean = 123
vaccine_round_delay_standard_deviation = 123
vaccine_booster_effect = 0.123
vaccine_buildup_delay = 123
vaccine_buildup_func = function
vaccine_full_protection_time = 123
vaccine_degredation_func = function

vaccine_protection_degredation_time = 123
vaccine_protection_degredation_time_mean = 123
vaccine_protection_degredation_time_standard_deviation = 123

natural_effectiveness = 0.123
full_natural_protection_time = 123
natural_effectiveness_degredation_func = function
```

```
natural_effectiveness_degredation_time = 123
natural_effectiveness_degredation_time_mean = 123
natural_effectiveness_degredation_time_standard_deviation = 123


vaccine_booster_effect_against_variants = 0.123
vaccine_full_protection_time_against_variants = 123
vaccine_degredation_func_against_variants = function


vaccine_protection_degredation_against_variants_time = 123
vaccine_protection_degredation_against_variants_time_mean = 123
vaccine_protection_degredation_against_variants_time_standard_deviation
= 123


natural_effectiveness_against_variants = 0.123
full_natural_protection_against_variants = 123
natural_effectiveness_degredation_against_variants_func = function


natural_effectiveness_degredation_against_variants_time = 123
natural_effectiveness_degredation_against_variants_time_mean = 123
natural_effectiveness_degredation_against_variants_time_standard_devia-
tion = 123


hospitalization_prob = 0.123
hospitalization_admission_delay = 123
hospitalization_admission_delay_mean = 123
hospitalization_admission_delay_standard_deviation = 123
icu_prob = 0.123
icu_admission_delay_time = 123
icu_admission_delay_mean = 123
icu_admission_standard_deviation = 123
```

# GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright (C) 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <https://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

## 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

## 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does

not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

## 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

## 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

## 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A.  Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B.  List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C.  State on the Title page the name of the publisher of the Modified Version, as the publisher.

D.  Preserve all the copyright notices of the Document.

E.    Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F.    Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G.    Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H.    Include an unaltered copy of this License.

I.    Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J.    Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K.    For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L.    Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M.    Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N.    Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O.    Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties--for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

**5. COMBINING DOCUMENTS**

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

**6. COLLECTIONS OF DOCUMENTS**

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

**7. AGGREGATION WITH INDEPENDENT WORKS**

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

**8. TRANSLATION**

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant

Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

## 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

## 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See https://www.gnu.org/licenses/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

## 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus

published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

### ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (c)  YEAR  YOUR NAME.
Permission is granted to copy, distribute and/or modify this docu-
ment under the terms of the GNU Free Documentation License, Version
1.3 or any later version published by the Free Software Foundation;
with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
Texts.  A copy of the license is included in the section entitled
"GNU Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with...Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the
Front-Cover Texts being LIST, and with the Back-Cover Texts being
LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.