

# S6 Text – Installation instructions for EMULSION

Supporting Information for Picault *et al.*, “EMULSION: transparent and flexible multiscale stochastic models in human, animal, or plant epidemiology”

## Contents

|  |          |
|--|----------|
| <b>A Requirements</b>                                  | <b>1</b> |
| <b>B Install EMULSION in the Python environment</b>    | <b>1</b> |
| B.1 Linux and MacOS . . . . .                          | 1        |
| B.2 Windows . . . . .                                  | 2        |
| <b>C Install Graphviz (for state machine diagrams)</b> | <b>2</b> |
| C.1 Linux . . . . .                                    | 2        |
| C.2 MacOS . . . . .                                    | 2        |
| C.3 Windows . . . . .                                  | 2        |
| <b>D Test EMULSION installation</b>                    | <b>3</b> |
| <b>E Main command-line options for EMULSION</b>        | <b>6</b> |

This document provides instructions for installing EMULSION, testing the installation, and describes the main commands and options available for the `emulsion` command. The latest version of EMULSION and of the documentation is available on: <https://sourcesup.renater.fr/www/emulsion-public>

## A Requirements

**System** EMULSION has been designed under **MacOS** and **Linux**, and also works with **Windows 10**, with minor limitations such as the absence of shell completion. Installation procedures are specific to each system.

Also, in what follows, we assume that MacOS and Linux users are working with a **bash** shell, which often is the default configuration. To check this point, typing: `echo $SHELL` should print: `/bin/bash`. Otherwise, the commands below may require some adjustments (please refer to the documentation of your shell).

**Language** EMULSION is written in **Python 3** (version 3.6 or higher). In what follows, we assume that `python3` and `pip3` refer to the Python3 installation. If not, they should be replaced in the commands below by the proper names (e.g. with Windows: respectively `python.exe` and `pip.exe`).

The installation of Python 3 is fully described on Python website (<https://www.python.org/downloads/>). When installing on Windows, the box ”Add Python 3.x to PATH” must be checked to make python commands available from the terminal.

## B Install EMULSION in the Python environment

### B.1 Linux and MacOS

Open a terminal and type the following commands:

```
sudo pip3 install emulsion
init_emulsion
source $HOME/.bashrc
```

Notes:

1. Depending on how Python was installed on your system, `sudo` may be optional.
2. The second command (`init_emulsion`) initializes command-line completion (available with a bash shell, under Linux or MacOS), which allows to use TAB key to get suggestions on what is expected (options, files, parameters...) in the command. It also creates (or modifies) the environment variable named `PYTHONPATH` to make EMULSION able to find code add-ons located in the same directory as your models.

## B.2 Windows

Open a terminal (either "Command Prompt" also known as `cmd.exe`, or "Windows Power Shell"), then type:

```
pip.exe install emulsion
```

To make EMULSION able to find code add-ons in the directory where your model files are located, you must declare (or modify) an environment variable named `PYTHONPATH` to add the current directory (".").

To do so, you can use the graphical interface (from System Preferences), or the terminal. The syntax depends on the kind of terminal you are using:

- with the "Command Prompt" (`cmd.exe`), type:

```
setx PYTHONPATH ".;%PYTHONPATH%"
```

- with "Windows Power Shell", type:

```
setx PYTHONPATH ".;$Env:PYTHONPATH"
```

## C Install Graphviz (for state machine diagrams)

Graphviz (<http://www.graphviz.org>) is highly recommended (otherwise, simulations will run but the diagrams representing model structure will not be produced).

### C.1 Linux

```
sudo apt install graphviz
```

### C.2 MacOS

Graphviz can be installed on MacOS using for instance **homebrew** (<https://brew.sh/>):

```
brew install graphviz
```

### C.3 Windows

To install on Windows, download the latest stable version from Graphviz website ([https://graphviz.gitlab.io/\\_pages/Download/Download\\_windows.html](https://graphviz.gitlab.io/_pages/Download/Download_windows.html)) and update the `PATH` environment variable according to the location of executable files are (especially `dot.exe`).

For instance, if `dot.exe` is located in `C:\Program Files (x86)\Graphviz\bin`:

- using the "Command Prompt" (`cmd.exe`):

```
setx PATH "C:\Program Files (x86)\Graphviz\bin;%PATH%"
```

- using "Windows Power Shell":

```
setx PATH "C:\Program Files (x86)\Graphviz\bin;$Env:PATH"
```

## D Test EMULSION installation

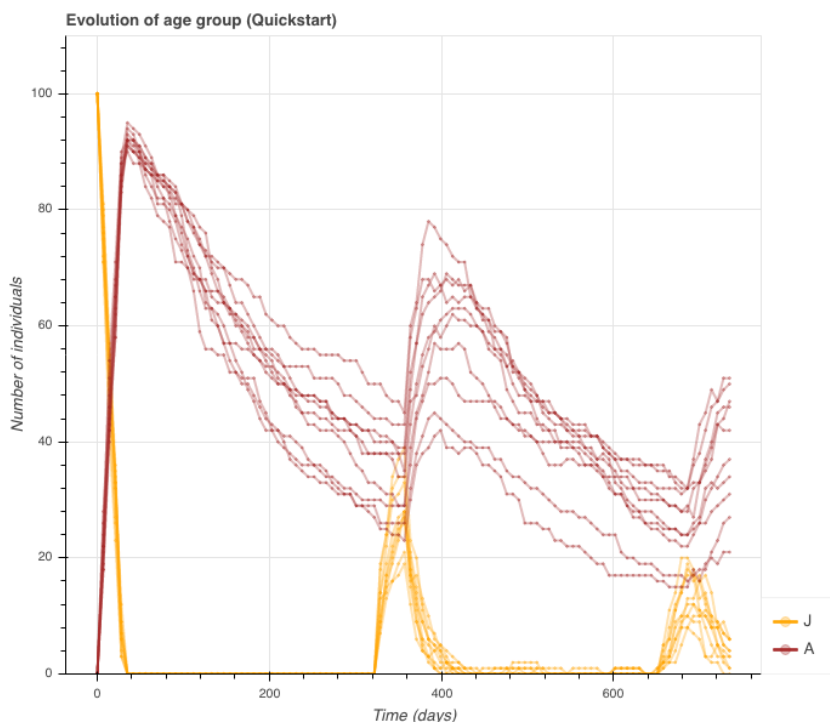
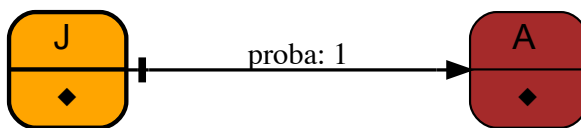
1. Download model examples: <https://sourcesup.renater.fr/www/emulsion-public/models.zip>
2. Extract the archive, open a terminal in the models directory, then type:

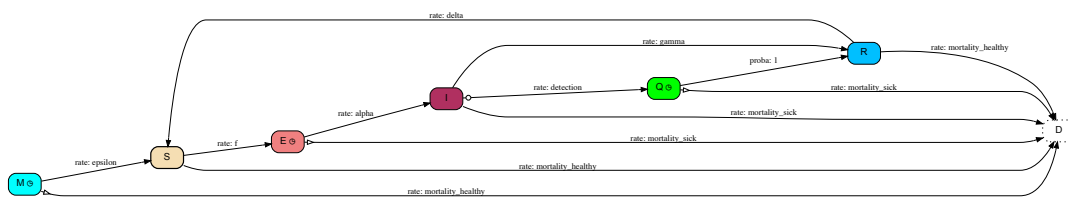
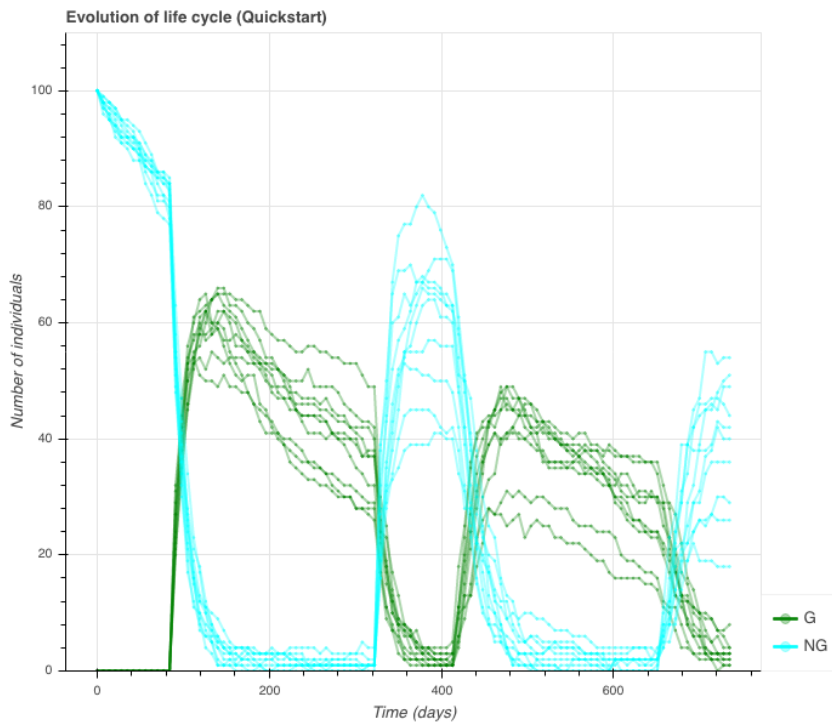
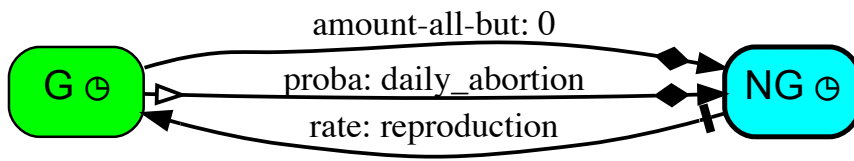
```
cd quickstart
emulsion run --plot quickstart.yaml --view-model --silent
```

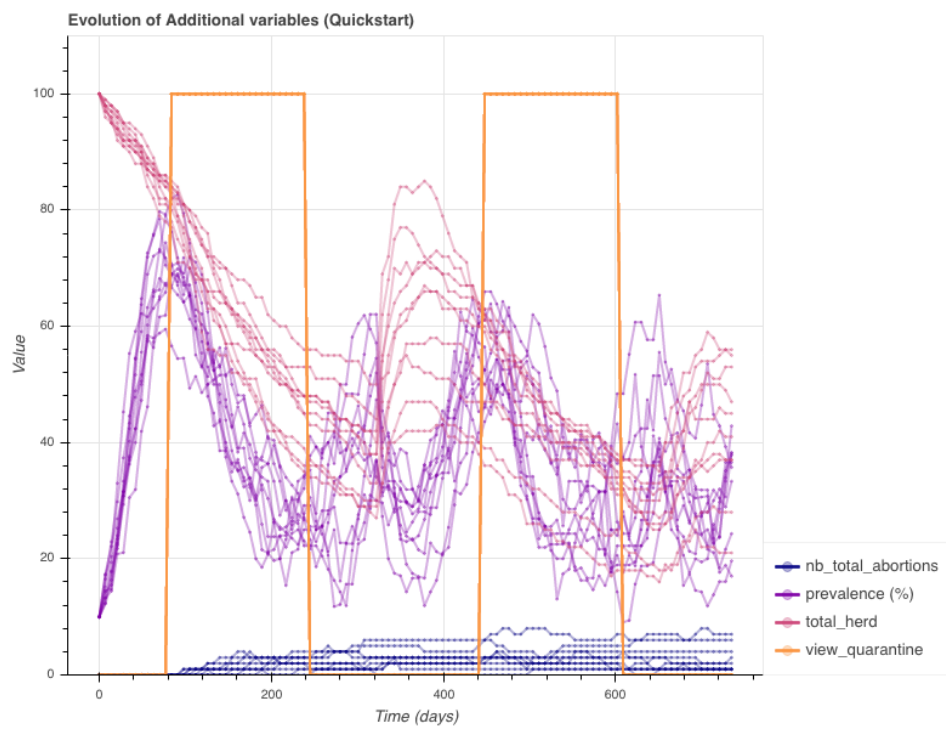
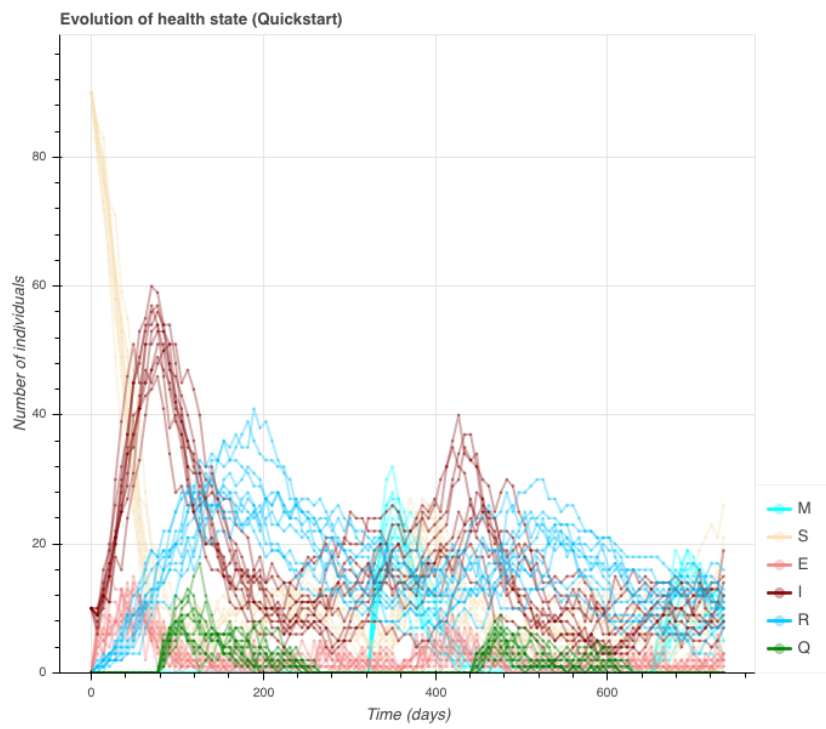
3. This should produce the following output in the terminal:

```
Simulation level:herd
Generated state machine diagram img/Quickstart_age_group_machine.svg
Generated state machine diagram img/Quickstart_life_cycle_machine.svg
Generated state machine diagram img/Quickstart_health_state_machine.svg
100%|*****| 10/10
Simulation finished in 12.37 s
Outputs stored in outputs/counts.csv
Outputs plot in file: img/Quickstart.html
```

4. The default web navigator should open and display the following figures. They corresponds, for each state machine defined in the model, to the state machine diagram and the amounts of individuals in each state over time for each stochastic repetition, plus additional variables specified as `extra_vars` in the model file. All these outputs are stored in a file named `counts.csv` in the output directory (by default, `outputs/`).







## E Main command-line options for EMULSION

This is a summary of the most frequently used commands and options for EMULSION. To get all available features, type: `emulsion -h`

### Usage:

```
emulsion run [--plot] MODEL [options] [(-p KEY=VALUE)...]
emulsion diagrams MODEL
emulsion show MODEL [options] [(-p KEY=VALUE)...]
emulsion describe MODEL PARAM...
emulsion plot MODEL [options]
emulsion (-h | --help | -V | --version)
```

### Commands:

|                                      |  |
|--------------------------------------|--|
| <code>run MODEL</code>               | Run simulations based on the specified MODEL (MODEL is the path to the YAML file describing the model to run). |
| <code>diagrams MODEL</code>          | Generate diagrams for state machines defined in MODEL (without running simulations), open them and exit.       |
| <code>show MODEL</code>              | Print all MODEL parameter values and exit.   |
| <code>describe MODEL PARAM...</code> | Describe the role of specified PARAMETERS in the MODEL and exit.   |
| <code>plot MODEL</code>              | Plot outputs for MODEL (assumed already run) and exit.   |

### Options:

|                                    |  |
|------------------------------------|--|
| <code>-h --help</code>             | Display this help page and exit.   |
| <code>-V --version</code>          | Display version number and exit.   |
| <code>--plot</code>                | Plot outputs just after running the model.   |
| <code>-r RUNS --runs RUNS</code>   | Specify the number of repetitions of the same model [default: 10].   |
| <code>-t STEPS --time STEPS</code> | Specify the number of time steps to run in each repetition. If the model defines a <code>total_duration</code> , it is used as time limit, unless the <code>'-t'</code> option is explicitly specified. Otherwise, the default value is 100 steps. |
| <code>-p KEY=VAL</code>            | Change parameter named KEY to the specified VALUE.   |
| <code>--view-model</code>          | Produce diagrams to represent the state machines of the model (requires Graphviz). Figures are stored in <code>figure-dir</code> .   |
| <code>--silent</code>              | Show only the progression of repetitions instead of the progression of each simulation.  |
| <code>--output-dir OUTPUT</code>   | Specify a directory for simulation outputs [default: <code>outputs</code> ].   |
| <code>--input-dir INPUT</code>     | Specify a directory for simulation inputs [default: <code>data</code> ].   |
| <code>--figure-dir FIG</code>      | Specify a directory for graphic outputs (figures) [default: <code>img</code> ].  |
| <code>--log-params</code>          | When producing CSV outputs, insert the name and value of each parameter explicitly changed by option <code>-p</code> .   |