

epiworld

0.0-1

Generated by Doxygen 1.9.1

1 Main Page	1
1.1 epiworld	1
1.2 Hello world	1
1.2.1 Tools	2
1.2.2 Contagion	2
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Class Documentation	7
4.1 cxxopts::values::abstract_value< T > Class Template Reference	7
4.2 AdjList Class Reference	9
4.2.1 Constructor & Destructor Documentation	9
4.2.1.1 AdjList()	9
4.3 cxxopts::values::parser_tool::ArguDesc Struct Reference	10
4.4 cxxopts::argument_incorrect_type Class Reference	10
4.5 DataBase< TSeq > Class Template Reference	11
4.5.1 Detailed Description	12
4.5.2 Member Function Documentation	12
4.5.2.1 get_today_total()	12
4.5.2.2 record_variant()	13
4.6 cxxopts::HelpGroupDetails Struct Reference	13
4.7 cxxopts::HelpOptionDetails Struct Reference	14
4.8 cxxopts::values::parser_tool::IntegerDesc Struct Reference	14
4.9 cxxopts::invalid_option_format_error Class Reference	15
4.10 cxxopts::ParseResult::Iterator Class Reference	16
4.11 cxxopts::KeyValue Class Reference	16
4.12 LFMCMC< TData > Class Template Reference	17
4.12.1 Detailed Description	17
4.12.2 Member Function Documentation	18
4.12.2.1 set_rand_engine()	18
4.13 Location< TSeq > Class Template Reference	18
4.14 cxxopts::missing_argument_exception Class Reference	19
4.15 Model< TSeq > Class Template Reference	20
4.15.1 Member Function Documentation	23
4.15.1.1 add_param()	23
4.15.1.2 add_status_susceptible()	24
4.15.1.3 init()	24
4.15.1.4 pop_from_adjlist()	25
4.15.1.5 reset()	25

4.15.1.6 reset_status_codes()	25
4.15.1.7 set_backup()	26
4.15.1.8 set_rand_engine()	26
4.15.1.9 set_rewire_fun()	26
4.15.1.10 set_user_data()	27
4.15.1.11 write_data()	27
4.15.1.12 write_edgelist()	27
4.16 cxxopts::Option Struct Reference	28
4.17 cxxopts::option_exists_error Class Reference	29
4.18 cxxopts::option_has_no_value_exception Class Reference	30
4.19 cxxopts::option_not_exists_exception Class Reference	31
4.20 cxxopts::option_not_has_argument_exception Class Reference	33
4.21 cxxopts::option_not_present_exception Class Reference	34
4.22 cxxopts::option_required_exception Class Reference	36
4.23 cxxopts::option_requires_argument_exception Class Reference	37
4.24 cxxopts::option_syntax_exception Class Reference	39
4.25 cxxopts::OptionAdder Class Reference	40
4.26 cxxopts::OptionDetails Class Reference	40
4.27 cxxopts::OptionException Class Reference	41
4.28 cxxopts::OptionParseException Class Reference	42
4.29 cxxopts::OptionParser Class Reference	43
4.30 cxxopts::Options Class Reference	43
4.31 cxxopts::OptionSpecException Class Reference	44
4.32 cxxopts::OptionValue Class Reference	45
4.33 cxxopts::ParseResult Class Reference	45
4.34 Person< TSeq > Class Template Reference	46
4.35 PersonTools< TSeq > Class Template Reference	47
4.35.1 Detailed Description	47
4.36 PersonViruses< TSeq > Class Template Reference	48
4.36.1 Detailed Description	48
4.37 Progress Class Reference	48
4.37.1 Detailed Description	49
4.38 Queue< TSeq > Class Template Reference	49
4.38.1 Detailed Description	49
4.39 RandGraph Class Reference	50
4.40 cxxopts::values::detail::SignedCheck< T, B > Struct Template Reference	50
4.41 cxxopts::values::detail::SignedCheck< T, false > Struct Template Reference	50
4.42 cxxopts::values::detail::SignedCheck< T, true > Struct Template Reference	50
4.43 cxxopts::values::standard_value< T > Class Template Reference	51
4.44 cxxopts::values::standard_value< bool > Class Reference	53
4.45 Tool< TSeq > Class Template Reference	54
4.45.1 Detailed Description	55

4.45.2 Member Function Documentation	56
4.45.2.1 get_susceptibility_reduction()	56
4.46 cxxopts::values::type_is_container< T > Struct Template Reference	56
4.47 cxxopts::values::type_is_container< std::vector< T > > Struct Template Reference	56
4.48 UserData< TSeq > Class Template Reference	57
4.49 cxxopts::Value Class Reference	58
4.50 vecHasher< T > Struct Template Reference	59
4.51 Virus< TSeq > Class Template Reference	59
4.51.1 Detailed Description	60
4.51.2 Member Function Documentation	60
4.51.2.1 get_prob_infecting()	60
Index	63

Chapter 1

Main Page

1.1 epiworld

This C++ template-header-only library provides a general framework for epidemiologic simulation. The main features of the library are:

1. Four key classes: `Model`, `Person`, `Tool`, and `Virus`.
2. The model features a social networks of `Persons`.
3. `Persons` can have multiple `Tools` as a defense system.
4. `Tools` can reduce contagion rate, transmissibility, death rates, and improve recovery rates.
5. `Viruses` can mutate (generating new variants).
6. `Models` can feature multiple states, e.g., `HEALTHY`, `SUSCEPTIBLE`, etc.
7. `Models` can have an arbitrary number of parameters.
8. **REALLY FAST** About 6.5 Million person/day simulations per second.

1.2 Hello world

Here is a simple SIRS model implemented with

```
#include "../include/epiworld/epiworld.hpp"
using namespace epiworld;
int main()
{
    // Creating a model
    Model<> model;
    // Adding the tool and virus
    Virus<> virus("covid 19");
    virus.set_post_immunity(1.0);
    model.add_virus_n(virus, 5);

    Tool<> tool("vaccine");
    model.add_tool(tool, .5);
    // Generating a random pop
    model.pop_from_random(100000);
    // Initializing setting days and seed
    model.init(100, 123);
}
```

```
// Running the model
model.run();
model.print();

}
```

And you should get something like the following:

Running the model...

```
||||| done.

SIMULATION STUDY
Population size      : 100000
Days (duration)     : 100 (of 100)
Number of variants  : 1
Last run elapsed t   : 280.00ms
Rewiring             : off
Virus(es):
- covid 19 (baseline prevalence: 5 seeds)
Tool(s):
- vaccine (baseline prevalence: 50.00%)
Model parameters:
Distribution of the population at time 100:
- Total healthy (S)   : 99995 -> 97390
- Total recovered (S) : 0 -> 2554
- Total infected (I)  : 5 -> 56
- Total removed (R)   : 0 -> 0
(S): Susceptible, (I): Infected, (R): Recovered
```

Which took about 0.280 seconds.

1.2.1 Tools

1.2.2 Contagion

Susceptible individuals can acquire a virus from any of their infected connections. The probability that susceptible individual i gets the virus v from individual j depends on how three things:

1. The transmissibility of the virus, P_v in $[0,1]$,
2. The contagion reduction factor of i , Cr in $[0,1]$, and
3. The host's transmission reduction factor, Tr $[0,1]$.

The last two are computed from i and j 's tools. Ultimately, the probability of i getting virus v from j equals:

$$P(\text{Virus } v) = P_v * (1 - Cr) * (1 - Tr)$$

Nonetheless, the default behavior of the simulation model is to assume that individuals can acquire only one disease at a time, if any. This way, the actual probability is:

$$P(\text{Virus } v | \text{at most one virus}) = \text{Prcond}(i, v, j)$$

The latter is calculated using Bayes' rule

$$\begin{aligned} \text{Prcond}(i, v, j) &= P(\text{at most one virus} | \text{Virus } v) * P(\text{Variant } v) / P(\text{at most one virus}) \\ &= P(\text{Only Virus } v) / P(\text{Virus } v) * P(\text{Virus } v) / P(\text{at most one virus}) \\ &= P(\text{Only Virus } v) / P(\text{at most one virus}) \end{aligned}$$

Where

$$\begin{aligned} P(\text{Only Virus } V) &= P(\text{Virus } V) * \text{Prod}(m \neq V) (1 - P(\text{Virus } m)) \\ P(\text{at most one virus}) &= P(\text{None}) + \text{Sum}(k \text{ in viruses}) P(\text{Virus } k) * \text{Prod}(m \neq k) (1 - P(\text{Virus } m)) \\ P(\text{None}) &= \text{Prod}(k \text{ in Viruses}) (1 - P(\text{Virus } k)) \end{aligned}$$

This way, viruses with higher transmissibility will be more likely to be acquired when competing with other variants.

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

AdjList	9
cxxopts::values::parser_tool::ArguDesc	10
DataBase< TSeq >	11
DataBase< bool >	11
std::enable_shared_from_this	
cxxopts::Value	58
cxxopts::values::abstract_value< bool >	7
cxxopts::values::standard_value< bool >	53
cxxopts::values::abstract_value< T >	7
cxxopts::values::standard_value< T >	51
std::exception	
cxxopts::OptionException	41
cxxopts::OptionParseException	42
cxxopts::argument_incorrect_type	10
cxxopts::missing_argument_exception	19
cxxopts::option_not_exists_exception	31
cxxopts::option_not_has_argument_exception	33
cxxopts::option_not_present_exception	34
cxxopts::option_required_exception	36
cxxopts::option_requires_argument_exception	37
cxxopts::option_syntax_exception	39
cxxopts::OptionSpecException	44
cxxopts::invalid_option_format_error	15
cxxopts::option_exists_error	29
cxxopts::option_has_no_value_exception	30
cxxopts::HelpGroupDetails	13
cxxopts::HelpOptionDetails	14
cxxopts::values::parser_tool::IntegerDesc	14
cxxopts::ParseResult::Iterator	16
cxxopts::KeyValue	16
LFMCMC< TData >	17
Location< TSeq >	18
Model< TSeq >	20
Model< bool >	20
Model< TSeq >	20

cxxopts::Option	28
cxxopts::OptionAdder	40
cxxopts::OptionDetails	40
cxxopts::OptionParser	43
cxxopts::Options	43
cxxopts::OptionValue	45
cxxopts::ParseResult	45
Person< TSeq >	46
Person< bool >	46
PersonTools< TSeq >	47
PersonTools< bool >	47
PersonViruses< TSeq >	48
PersonViruses< bool >	48
Progress	48
Queue< TSeq >	49
Queue< bool >	49
RandGraph	50
cxxopts::values::detail::SignedCheck< T, B >	50
cxxopts::values::detail::SignedCheck< T, false >	50
cxxopts::values::detail::SignedCheck< T, true >	50
Tool< TSeq >	54
cxxopts::values::type_is_container< T >	56
cxxopts::values::type_is_container< std::vector< T > >	56
UserData< TSeq >	57
vecHasher< T >	59
Virus< TSeq >	59

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

cxxopts::values::abstract_value< T >	7
AdjList	9
cxxopts::values::parser_tool::ArguDesc	10
cxxopts::argument_incorrect_type	10
DataBase< TSeq >	
Statistical data about the process	11
cxxopts::HelpGroupDetails	13
cxxopts::HelpOptionDetails	14
cxxopts::values::parser_tool::IntegerDesc	14
cxxopts::invalid_option_format_error	15
cxxopts::ParseResult::Iterator	16
cxxopts::KeyValue	16
LFMCMC< TData >	
Likelihood-Free Markov Chain Monte Carlo	17
Location< TSeq >	18
cxxopts::missing_argument_exception	19
Model< TSeq >	20
cxxopts::Option	28
cxxopts::option_exists_error	29
cxxopts::option_has_no_value_exception	30
cxxopts::option_not_exists_exception	31
cxxopts::option_not_has_argument_exception	33
cxxopts::option_not_present_exception	34
cxxopts::option_required_exception	36
cxxopts::option_requires_argument_exception	37
cxxopts::option_syntax_exception	39
cxxopts::OptionAdder	40
cxxopts::OptionDetails	40
cxxopts::OptionException	41
cxxopts::OptionParseException	42
cxxopts::OptionParser	43
cxxopts::Options	43
cxxopts::OptionSpecException	44
cxxopts::OptionValue	45
cxxopts::ParseResult	45

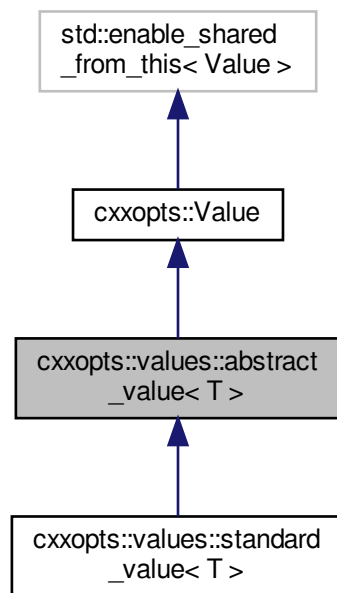
Person< TSeq >	46
PersonTools< TSeq >	
List of tools available for the individual to	47
PersonViruses< TSeq >	
Set of viruses in host	48
Progress	
A simple progress bar	48
Queue< TSeq >	
Controls which agents are verified at each step	49
RandGraph	50
cxxopts::values::detail::SignedCheck< T, B >	50
cxxopts::values::detail::SignedCheck< T, false >	50
cxxopts::values::detail::SignedCheck< T, true >	50
cxxopts::values::standard_value< T >	51
cxxopts::values::standard_value< bool >	53
Tool< TSeq >	
Tools for defending the host against the virus	54
cxxopts::values::type_is_container< T >	56
cxxopts::values::type_is_container< std::vector< T > >	56
UserData< TSeq >	57
cxxopts::Value	58
vecHasher< T >	59
Virus< TSeq >	
Virus	59

Chapter 4

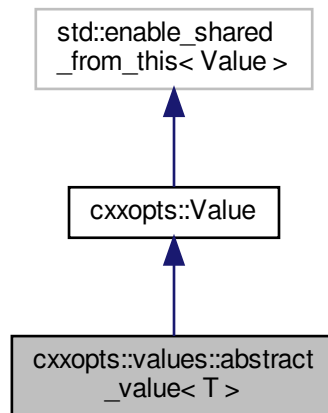
Class Documentation

4.1 cxxopts::values::abstract_value< T > Class Template Reference

Inheritance diagram for cxxopts::values::abstract_value< T >:



Collaboration diagram for `cxxopts::values::abstract_value< T >`:



Public Member Functions

- **abstract_value** (T *t)
- **abstract_value** & **operator=** (const **abstract_value** &)=default
- **abstract_value** (const **abstract_value** &rhs)
- void **parse** (const std::string &text) const override
- bool **is_container** () const override
- void **parse** () const override
- bool **has_default** () const override
- bool **has_implicit** () const override
- std::shared_ptr< **Value** > **default_value** (const std::string &value) override
- std::shared_ptr< **Value** > **implicit_value** (const std::string &value) override
- std::shared_ptr< **Value** > **no_implicit_value** () override
- std::string **get_default_value** () const override
- std::string **get_implicit_value** () const override
- bool **is_boolean** () const override
- const T & **get** () const

Protected Attributes

- std::shared_ptr< T > **m_result** {}
- T * **m_store** {}
- bool **m_default** = false
- bool **m_implicit** = false
- std::string **m_default_value** {}
- std::string **m_implicit_value** {}

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.2 AdjList Class Reference

Public Member Functions

- [AdjList](#) (const std::vector< unsigned int > &source, const std::vector< unsigned int > &target, bool directed, int min_id=-1, int max_id=-1)
Construct a new Adj List object.
- void **read_edgelist** (std::string fn, int skip=0, bool directed=true, int min_id=-1, int max_id=-1)
- std::map< unsigned int, unsigned int > **operator()** (unsigned int i) const
- void **print** (unsigned int limit=20u) const
- unsigned int **get_id_max** () const
- unsigned int **get_id_min** () const
- size_t **vcount** () const
- size_t **ecount** () const
- std::map< unsigned int, std::map< unsigned int, unsigned int > > & **get_dat** ()
- bool **is_directed** () const

4.2.1 Constructor & Destructor Documentation

4.2.1.1 AdjList()

```
AdjList::AdjList (
    const std::vector< unsigned int > & source,
    const std::vector< unsigned int > & target,
    bool directed,
    int min_id = -1,
    int max_id = -1 ) [inline]
```

Construct a new Adj List object.

It will create an adjacency list object with `maxid - minid + 1` nodes. If `min_id` and `max_id` are not specified (both < 0), then the program will try to figure them out automatically by looking at the range of the observed ids.

Parameters

<i>source</i>	Unsigned int vector with the source
<i>target</i>	Unsigned int vector with the target
<i>directed</i>	Bool true if the network is directed
<i>min_id</i>	int min id.
<i>max_id</i>	int max id.

The documentation for this class was generated from the following files:

- include/epiworld/adjlist-bones.hpp
- include/epiworld/adjlist-meat.hpp

4.3 cxxopts::values::parser_tool::ArguDesc Struct Reference

Public Attributes

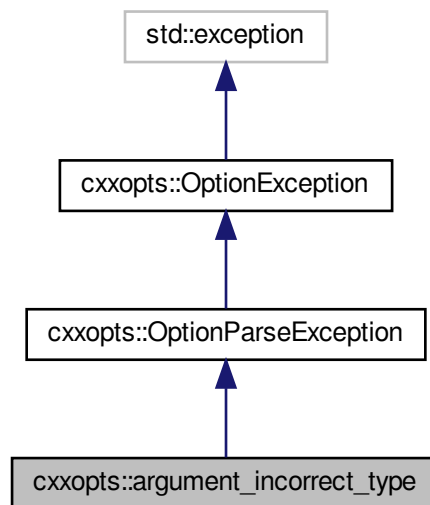
- std::string **arg_name** = ""
- bool **grouping** = false
- bool **set_value** = false
- std::string **value** = ""

The documentation for this struct was generated from the following file:

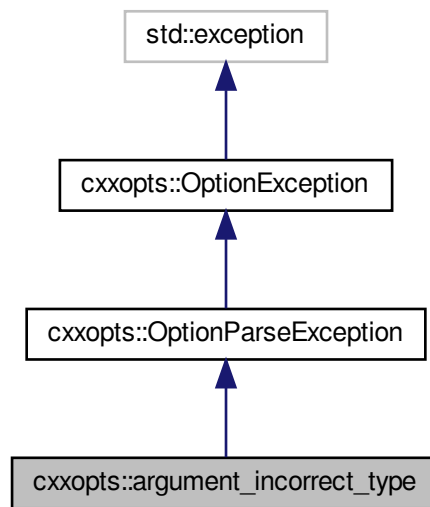
- include/cxxopts/cxxopts.hpp

4.4 cxxopts::argument_incorrect_type Class Reference

Inheritance diagram for cxxopts::argument_incorrect_type:



Collaboration diagram for cxxopts::argument_incorrect_type:



Public Member Functions

- **argument_incorrect_type** (const std::string &arg)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.5 DataBase< TSeq > Class Template Reference

Statistical data about the process.

```
#include <database-bones.hpp>
```

Public Member Functions

- **DataBase** (int freq=1)
- void **record_variant** (Virus< TSeq > *v)
Registering a new variant.
- void **set_seq_hasher** (std::function< std::vector< int >(TSeq)> fun)
- void **set_model** (Model< TSeq > &m)
- Model< TSeq > * **get_model** ()
- void **record** ()
- const std::vector< TSeq > & **get_sequence** () const
- const std::vector< int > & **get_nexposed** () const

- `size_t size () const`
- `void up_exposed (Virus< TSeq > *v, epiworld_fast_uint new_status)`
- `void down_exposed (Virus< TSeq > *v, epiworld_fast_uint prev_status)`
- `void state_change (epiworld_fast_uint prev_status, epiworld_fast_uint new_status)`
- `void record_transition (epiworld_fast_uint from, epiworld_fast_uint to)`
- `int get_today_total (std::string what) const`
Get recorded information from the model.
- `int get_today_total (epiworld_fast_uint what) const`
- `void get_today_total (std::vector< std::string > *status=nullptr, std::vector< int > *counts=nullptr) const`
- `void get_today_variant (std::vector< std::string > &status, std::vector< int > &id, std::vector< int > &counts) const`
- `void get_hist_total (std::vector< int > *date, std::vector< std::string > *status, std::vector< int > *counts) const`
- `void get_hist_variant (std::vector< int > &date, std::vector< int > &id, std::vector< std::string > &status, std::vector< int > &counts) const`
- `void write_data (std::string fn_variant_info, std::string fn_variant_hist, std::string fn_total_hist, std::string fn←_transmission, std::string fn_transition) const`
@]
- `void record_transmission (int i, int j, int variant)`
- `size_t get_nvariants () const`
- `void reset ()`
- `void set_user_data (std::vector< std::string > names)`
- `void add_user_data (std::vector< epiworld_double > x)`
- `void add_user_data (unsigned int j, epiworld_double x)`
- `UserData< TSeq > & get_user_data ()`

Friends

- `class Model< TSeq >`

4.5.1 Detailed Description

```
template<typename TSeq>
class DataBase< TSeq >
```

Statistical data about the process.

Template Parameters

<i>TSeq</i>	
-------------	--

4.5.2 Member Function Documentation

4.5.2.1 get_today_total()

```
template<typename TSeq >
int DataBase< TSeq >::get_today_total (
```

```
std::string what ) const [inline]
```

Get recorded information from the model.

Parameters

<i>what</i>	std::string, The status, e.g., 0, 1, 2, ...
-------------	---

Returns

In `get_today_total`, the current counts of `what`.

In `get_today_variant`, the current counts of `what` for each variant.

In `get_hist_total`, the time series of `what`

In `get_hist_variant`, the time series of `what` for each variant.

In `get_hist_total_date` and `get_hist_variant_date` the corresponding dates @

4.5.2.2 record_variant()

```
template<typename TSeq >
void DataBase< TSeq >::record_variant (
    Virus< TSeq > * v ) [inline]
```

Registering a new variant.

Parameters

<i>v</i>	Pointer to the new variant. Since variants are originated in the host, the numbers simply move around. From the parent variant to the new variant. And the total number of infected does not change.
----------	--

The documentation for this class was generated from the following files:

- include/epiworld/database-bones.hpp
- include/epiworld/database-meat.hpp

4.6 cxxopts::HelpGroupDetails Struct Reference

Public Attributes

- std::string **name** {}
- std::string **description** {}
- std::vector< [HelpOptionDetails](#) > **options** {}

The documentation for this struct was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.7 cxxopts::HelpOptionDetails Struct Reference

Public Attributes

- std::string **s**
- std::string **l**
- String **desc**
- bool **has_default**
- std::string **default_value**
- bool **has_implicit**
- std::string **implicit_value**
- std::string **arg_help**
- bool **is_container**
- bool **is_boolean**

The documentation for this struct was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.8 cxxopts::values::parser_tool::IntegerDesc Struct Reference

Public Attributes

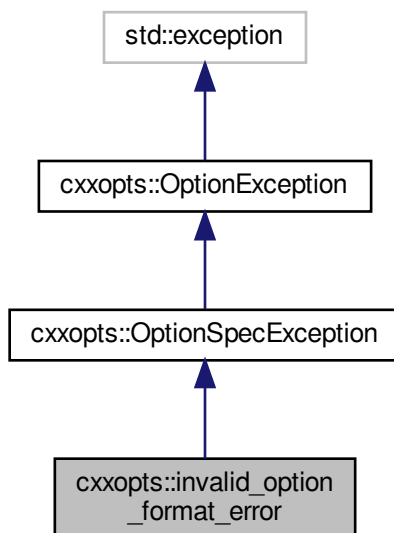
- std::string **negative** = ""
- std::string **base** = ""
- std::string **value** = ""

The documentation for this struct was generated from the following file:

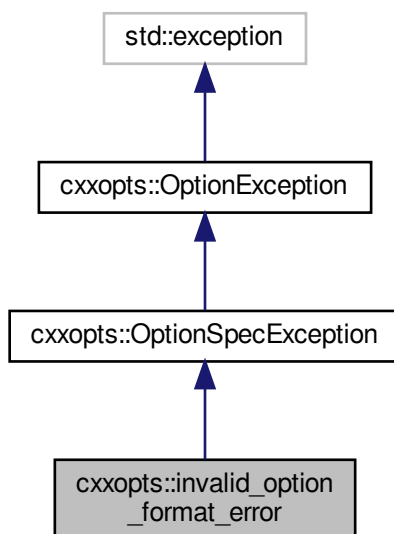
- include/cxxopts/cxxopts.hpp

4.9 cxxopts::invalid_option_format_error Class Reference

Inheritance diagram for cxxopts::invalid_option_format_error:



Collaboration diagram for cxxopts::invalid_option_format_error:



Public Member Functions

- **invalid_option_format_error** (const std::string &format)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.10 cxxopts::ParseResult::Iterator Class Reference

Public Types

- using **iterator_category** = std::forward_iterator_tag
- using **value_type** = [KeyValue](#)
- using **difference_type** = void
- using **pointer** = const [KeyValue](#) *
- using **reference** = const [KeyValue](#) &

Public Member Functions

- **Iterator** (const [Iterator](#) &)=default
- **Iterator** (const [ParseResult](#) *pr, bool end=false)
- [Iterator](#) & **operator++** ()
- [Iterator](#) **operator++** (int)
- bool **operator==** (const [Iterator](#) &other) const
- bool **operator!=** (const [Iterator](#) &other) const
- const [KeyValue](#) & **operator*** ()
- const [KeyValue](#) * **operator->** ()

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.11 cxxopts::KeyValue Class Reference

Public Member Functions

- **KeyValue** (std::string key_, std::string value_)
- CXXOPTS_NODISCARD const std::string & **key** () const
- CXXOPTS_NODISCARD const std::string & **value** () const
- template<typename T >
T **as** () const

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.12 LFMCMC< TData > Class Template Reference

Likelihood-Free Markov Chain Monte Carlo.

```
#include <lfmcmc.hpp>
```

Public Member Functions

- void **run** (VEC(epiworld_double) param_init, size_t n_samples_, epiworld_double epsilon_)
- **LFMCMC** (TData &observed_data_)
- void **set_observed_data** (TData &observed_data_)
- void **set_proposal_fun** (FUN< void(VEC(epiworld_double)&, LFMCMC< TData > *)> fun)
- void **set_simulation_fun** (FUN< TData(VEC(epiworld_double)&, LFMCMC< TData > *)> fun)
- void **set_summary_fun** (FUN< VEC(epiworld_double)(TData &, LFMCMC< TData > *)> fun)
- void **set_kernel_fun** (FUN< epiworld_double(VEC(epiworld_double)&, epiworld_double, LFMCMC< TData > *)> fun)
- void **set_rand_engine** (std::mt19937 &eng)
 - Random number generation.*
- std::mt19937 * **get_rand_engine** ()
- void **seed** (unsigned int s)
- void **set_rand_gamma** (epiworld_double alpha, epiworld_double beta)
- epiworld_double **runif** ()
- epiworld_double **rnorm** ()
- epiworld_double **rnorm** (epiworld_double mean, epiworld_double sd)
- epiworld_double **rgamma** ()
- epiworld_double **rgamma** (epiworld_double alpha, epiworld_double beta)
- const size_t **get_n_samples** ()
- @]*
- const size_t **get_n_statistics** ()
- const size_t **get_n_parameters** ()
- const epiworld_double **get_epsilon** ()
- const **VEC** (epiworld_double) &get_params_now()
- const **VEC** (epiworld_double) &get_params_prev()
- const **VEC** (epiworld_double) &get_params_init()
- const **VEC** (epiworld_double) &get_statistics_obs()
- const **VEC** (epiworld_double) &get_statistics_hist()
- const **VEC** (bool) &get_statistics_accepted()
- const **VEC** (epiworld_double) &get_posterior_lf_prob()
- const **VEC** (epiworld_double) &get_acceptance_prob()
- const **VEC** (epiworld_double) &get_drawn_prob()
- **VEC** (TData) *get_sampled_data()

4.12.1 Detailed Description

```
template<typename TData>
class LFMCMC< TData >
```

Likelihood-Free Markov Chain Monte Carlo.

Template Parameters

<i>TData</i>	Type of data that is generated
--------------	--------------------------------

4.12.2 Member Function Documentation

4.12.2.1 set_rand_engine()

```
template<typename TData >
void LFMCMC< TData >::set_rand_engine (
    std::mt19937 & eng ) [inline]
```

Random number generation.

Parameters

<i>eng</i>	@[
------------	----

The documentation for this class was generated from the following file:

- include/epiworld/math/lfmcmc.hpp

4.13 Location< TSeq > Class Template Reference

Public Member Functions

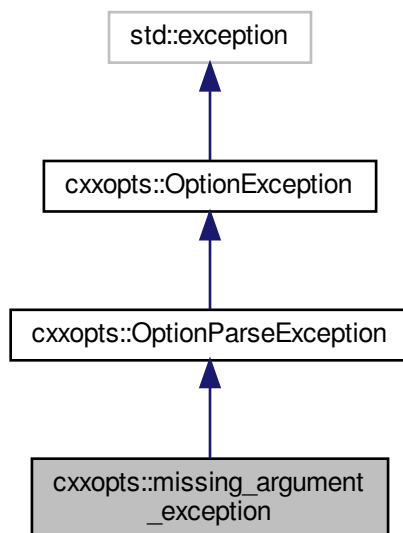
- [add_person](#) ([Person](#)< TSeq > &p)
@[
- [add_person](#) ([Person](#)< TSeq > *p)
- [size_t count](#) () const
- void [reset](#) ()

The documentation for this class was generated from the following file:

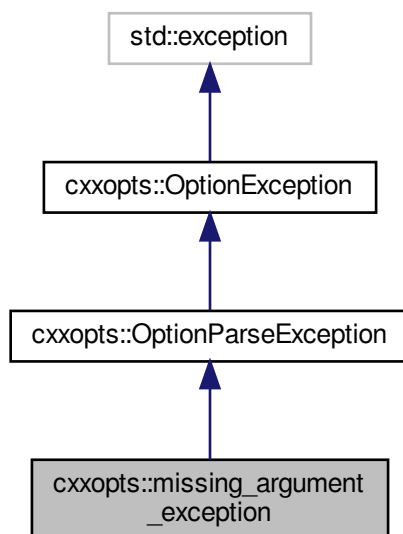
- include/epiworld/location-bones.hpp

4.14 cxxopts::missing_argument_exception Class Reference

Inheritance diagram for cxxopts::missing_argument_exception:



Collaboration diagram for cxxopts::missing_argument_exception:



Public Member Functions

- **missing_argument_exception** (const std::string &option)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.15 Model< TSeq > Class Template Reference

Public Member Functions

- **Model** (const Model< TSeq > &m)
- **Model** (Model< TSeq > &&m)
- Model< TSeq > & **operator=** (const Model< TSeq > &m)
- void **clone_population** (std::vector< Person< TSeq > > &p, std::map< int, int > &p_ids, bool &d, Model< TSeq > *m=nullptr) const
- void **clone_population** (const Model< TSeq > &m)
- void **set_backup** ()
Set the backup object.
- void **restore_backup** ()
- DataBase< TSeq > & **get_db** ()
@]
- epiworld_double & **operator()** (std::string pname)
- size_t **size** () const
- void **set_rand_engine** (std::mt19937 &eng)
Random number generation.
- std::mt19937 * **get_rand_engine** ()
- void **seed** (unsigned int s)
- void **set_rand_gamma** (epiworld_double alpha, epiworld_double beta)
- epiworld_double **runif** ()
- epiworld_double **rnorm** ()
- epiworld_double **rnorm** (epiworld_double mean, epiworld_double sd)
- epiworld_double **rgamma** ()
- epiworld_double **rgamma** (epiworld_double alpha, epiworld_double beta)
- void **add_virus** (Virus< TSeq > v, epiworld_double preval)
@]
- void **add_virus_n** (Virus< TSeq > v, unsigned int preval)
- void **add_tool** (Tool< TSeq > t, epiworld_double preval)
- void **add_tool_n** (Tool< TSeq > t, unsigned int preval)
- void **pop_from_adjlist** (std::string fn, int skip=0, bool directed=false, int min_id=-1, int max_id=-1)
Accessing population of the model.
- void **pop_from_adjlist** (AdjList al)
- bool **is_directed** () const
- std::vector< Person< TSeq > > * **get_population** ()
- void **pop_from_random** (unsigned int n=1000, unsigned int k=5, bool d=false, epiworld_double p=.01)
- void **init** (unsigned int ndays, unsigned int seed)
@]
- void **update_status** ()
- void **mutate_variant** ()
- void **next** ()

- void **run** ()
- void **run_multiple** (unsigned int nexperiments, std::function< void(Model< TSeq > *)> fun, bool **reset**, bool verbose)
- void **record_variant** (Virus< TSeq > *v)
 - @]
- int **get_nvariants** () const
- unsigned int **get_ndays** () const
- unsigned int **get_n_replicates** () const
- void **set_ndays** (unsigned int ndays)
- bool **get_verbose** () const
- void **verbose_off** ()
- void **verbose_on** ()
- int **today** () const
- void **set_rewire_fun** (std::function< void(std::vector< Person< TSeq >> *, Model< TSeq > *, epiworld_double)> fun)
 - Rewire the network preserving the degree sequence.*
- void **set_rewire_prop** (epiworld_double prop)
- epiworld_double **get_rewire_prop** () const
- void **rewire** ()
- void **set_update_susceptible** (UpdateFun< TSeq > fun)
 - @]
- void **set_update_exposed** (UpdateFun< TSeq > fun)
- void **set_update_removed** (UpdateFun< TSeq > fun)
- void **write_data** (std::string fn_variant_info, std::string fn_variant_hist, std::string fn_total_hist, std::string fn_transmission, std::string fn_transition) const
 - Wrapper of `DataBase::write_data`*
- void **write_edgelist** (std::string fn) const
 - Export the network data in edgelist form.*
- void **write_edgelist** (std::vector< unsigned int > &source, std::vector< unsigned int > &target) const
- std::map< std::string, epiworld_double > & **params** ()
 - @]
- void **reset** ()
 - Reset the model.*
- void **print** () const
- Model< TSeq > && **clone** () const
- void **add_status_susceptible** (epiworld_fast_uint s, std::string lab)
 - Adds extra statuses to the model.*
- void **add_status_exposed** (epiworld_fast_uint s, std::string lab)
- void **add_status_removed** (epiworld_fast_uint s, std::string lab)
- void **add_status_susceptible** (std::string lab)
- void **add_status_exposed** (std::string lab)
- void **add_status_removed** (std::string lab)
- const std::vector< epiworld_fast_uint > & **get_status_susceptible** () const
- const std::vector< epiworld_fast_uint > & **get_status_exposed** () const
- const std::vector< epiworld_fast_uint > & **get_status_removed** () const
- const std::vector< std::string > & **get_status_susceptible_labels** () const
- const std::vector< std::string > & **get_status_exposed_labels** () const
- const std::vector< std::string > & **get_status_removed_labels** () const
- void **print_status_codes** () const
- epiworld_fast_uint **get_default_susceptible** () const
- epiworld_fast_uint **get_default_exposed** () const
- epiworld_fast_uint **get_default_removed** () const
- void **reset_status_codes** (std::vector< epiworld_fast_uint > codes, std::vector< std::string > names, bool verbose=true)

@]

- `epiworld_double` [add_param](#) (`epiworld_double` initial_val, `std::string` pname)

Setting and accessing parameters from the model.

- `epiworld_double` **set_param** (`std::string` pname)
- `epiworld_double` **get_param** (`unsigned int` k)
- `epiworld_double` **get_param** (`std::string` pname)
- `epiworld_double` **par** (`unsigned int` k)
- `epiworld_double` **par** (`std::string` pname)
- `void` [get_elapsed](#) (`std::string` unit="auto", `epiworld_double` *last_elapsed=nullptr, `epiworld_double` *total_ ← elapsed=nullptr, `std::string` *unit_abbr=nullptr, `bool` print=true) `const`

@]

- `void` [set_user_data](#) (`std::vector`< `std::string` > names)

Set the user data object.

- `void` **add_user_data** (`unsigned int` j, `epiworld_double` x)
- `void` **add_user_data** (`std::vector`< `epiworld_double` > x)
- [UserData](#)< TSeq > & **get_user_data** ()
- `void` [add_global_action](#) (`std::function`< `void`([Model](#)< TSeq > *)> fun, `int` date)

@]

- `void` **run_global_actions** ()
- `void` **clear_status_set** ()
- `void` **toggle_visited** ()
- `void` **queuing_on** ()
- `void` **queuing_off** ()
- `bool` **is_queuing_on** () `const`
- [Queue](#)< TSeq > & **get_queue** ()

Public Attributes

- `std::vector`< `epiworld_double` > **array_double_tmp**
- `std::vector`< [Virus](#)< TSeq > * > **array_virus_tmp**
- `epiworld_double` * **p0**
- `epiworld_double` * **p1**
- `epiworld_double` * **p2**
- `epiworld_double` * **p3**
- `epiworld_double` * **p4**
- `epiworld_double` * **p5**
- `epiworld_double` * **p6**
- `epiworld_double` * **p7**
- `epiworld_double` * **p8**
- `epiworld_double` * **p9**
- `epiworld_double` * **p10**
- `epiworld_double` * **p11**
- `epiworld_double` * **p12**
- `epiworld_double` * **p13**
- `epiworld_double` * **p14**
- `epiworld_double` * **p15**
- `epiworld_double` * **p16**
- `epiworld_double` * **p17**
- `epiworld_double` * **p18**
- `epiworld_double` * **p19**
- `epiworld_double` * **p20**
- `epiworld_double` * **p21**
- `epiworld_double` * **p22**

- `epiworld_double * p23`
- `epiworld_double * p24`
- `epiworld_double * p25`
- `epiworld_double * p26`
- `epiworld_double * p27`
- `epiworld_double * p28`
- `epiworld_double * p29`
- `epiworld_double * p30`
- `epiworld_double * p31`
- `epiworld_double * p32`
- `epiworld_double * p33`
- `epiworld_double * p34`
- `epiworld_double * p35`
- `epiworld_double * p36`
- `epiworld_double * p37`
- `epiworld_double * p38`
- `epiworld_double * p39`
- `unsigned int npar_used = 0u`

Friends

- `class Person< TSeq >`
- `class DataBase< TSeq >`
- `class Queue< TSeq >`

4.15.1 Member Function Documentation

4.15.1.1 `add_param()`

```
template<typename TSeq = bool>
epiworld_double Model< TSeq >::add_param (
    epiworld_double initial_val,
    std::string pname )
```

Setting and accessing parameters from the model.

Tools can incorporate parameters included in the model. Internally, parameters in the tool are stored as pointers to an `std::map<>` of parameters in the model. Using the `unsigned int` method directly fetches the parameters in the order these were added to the tool. Accessing parameters via the `std::string` method involves searching the parameter directly in the `std::map<>` member of the model (so it is not recommended.)

The function `set_param()` can be used when the parameter already exists in the model.

The `par()` function members are aliases for `get_param()`.

Parameters

<i>initial_val</i>	
<i>pname</i>	Name of the parameter to add or to fetch

Returns

The current value of the parameter in the model. @]

4.15.1.2 add_status_susceptible()

```
template<typename TSeq = bool>
void Model< TSeq >::add_status_susceptible (
    epiworld_fast_uint s,
    std::string lab )
```

Adds extra statuses to the model.

Adding values of *s* that are already present in the model will result in an error.

The functions `get_status_*` return the current values for the statuses included in the model.

Parameters

<i>s</i>	unsigned int Code of the status
<i>lab</i>	std::string Name of the status.

Returns

`add_status*` returns nothing.

`get_status_*` returns a vector of pairs with the statuses and their labels. @]

4.15.1.3 init()

```
template<typename TSeq = bool>
void Model< TSeq >::init (
    unsigned int ndays,
    unsigned int seed )
```

@]

Functions to run the model

Parameters

<i>seed</i>	Seed to be used for Pseudo-RNG.
<i>ndays</i>	Number of days (steps) of the simulation.
<i>fun</i>	In the case of <code>run_multiple</code> , a function that is called after each experiment. @]

4.15.1.4 pop_from_adjlist()

```
template<typename TSeq = bool>
void Model< TSeq >::pop_from_adjlist (
    std::string fn,
    int skip = 0,
    bool directed = false,
    int min_id = -1,
    int max_id = -1 )
```

Accessing population of the model.

Parameters

<i>fn</i>	std::string Filename of the edgelist file.
<i>skip</i>	int Number of lines to skip in fn.
<i>directed</i>	bool Whether the graph is directed or not.
<i>min_id</i>	int Minimum id number (if negative, the program will try to guess from the data.)
<i>max_id</i>	int Maximum id number (if negative, the program will try to guess from the data.)
<i>al</i>	AdjList to read into the model. @[

4.15.1.5 reset()

```
template<typename TSeq = bool>
void Model< TSeq >::reset ( )
```

Reset the model.

Resetting the model will:

- clear the database
- restore the population (if [set_backup\(\)](#) was called before)
- re-distribute tools
- re-distribute viruses
- set the date to 0

4.15.1.6 reset_status_codes()

```
template<typename TSeq = bool>
void Model< TSeq >::reset_status_codes (
    std::vector< epiworld_fast_uint > codes,
    std::vector< std::string > names,
    bool verbose = true )
```

@]

Reset all the status codes of the model

The default values are those specified in the enum STATUS.

Parameters

<i>codes</i>	In the following order: Susceptible, Infected, Removed
<i>names</i>	Names matching the codes
<i>verbose</i>	When <code>true</code> , it will print the new mappings.

4.15.1.7 set_backup()

```
template<typename TSeq = bool>
void Model< TSeq >::set_backup ( )
```

Set the backup object.

`backup` can be used to restore the entire object after a run. This can be useful if the user wishes to have individuals start with the same network from the beginning. @[

4.15.1.8 set_rand_engine()

```
template<typename TSeq = bool>
void Model< TSeq >::set_rand_engine (
    std::mt19937 & eng )
```

Random number generation.

Parameters

<i>eng</i>	@[
------------	----

4.15.1.9 set_rewire_fun()

```
template<typename TSeq = bool>
void Model< TSeq >::set_rewire_fun (
    std::function< void(std::vector< Person< TSeq >> *, Model< TSeq > *, epiworld↵
_double)> fun )
```

Rewire the network preserving the degree sequence.

This implementation assumes an undirected network, thus if $\{(i,j), (k,l)\} \rightarrow \{(i,l), (k,j)\}$, the reciprocal is also true, i.e., $\{(j,i), (l,k)\} \rightarrow \{(j,k), (l,i)\}$.

Parameters

<i>proportion</i>	Proportion of ties to be rewired.
-------------------	-----------------------------------

Returns

A rewired version of the network. @

4.15.1.10 set_user_data()

```
template<typename TSeq = bool>
void Model< TSeq >::set_user_data (
    std::vector< std::string > names )
```

Set the user data object.

Parameters

<i>names</i>	@
--------------	---

4.15.1.11 write_data()

```
template<typename TSeq = bool>
void Model< TSeq >::write_data (
    std::string fn_variant_info,
    std::string fn_variant_hist,
    std::string fn_total_hist,
    std::string fn_transmission,
    std::string fn_transition ) const
```

Wrapper of [DataBase::write_data](#)

Parameters

<i>fn_variant_info</i>	Filename. Information about the variant.
<i>fn_variant_hist</i>	Filename. History of the variant.
<i>fn_total_hist</i>	Filename. Aggregated history (status)
<i>fn_transmission</i>	Filename. Transmission history.
<i>fn_transition</i>	Filename. Markov transition history.

4.15.1.12 write_edgelist()

```
template<typename TSeq = bool>
void Model< TSeq >::write_edgelist (
    std::string fn ) const
```

Export the network data in edgelist form.

Parameters

<i>fn</i>	std::string. File name.
<i>source</i>	Integer vector
<i>target</i>	Integer vector

When passing the source and target, the function will write the edgelist on those. [@

The documentation for this class was generated from the following files:

- include/epiworld/config.hpp
- include/epiworld/model-bones.hpp

4.16 cxxopts::Option Struct Reference

Public Member Functions

- **Option** (std::string opts, std::string desc, std::shared_ptr< const [Value](#) > value=:cxxopts::value< bool >(), std::string arg_help="")

Public Attributes

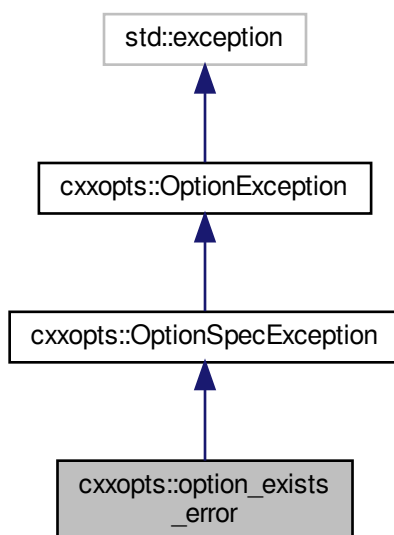
- std::string **opts_**
- std::string **desc_**
- std::shared_ptr< const [Value](#) > **value_**
- std::string **arg_help_**

The documentation for this struct was generated from the following file:

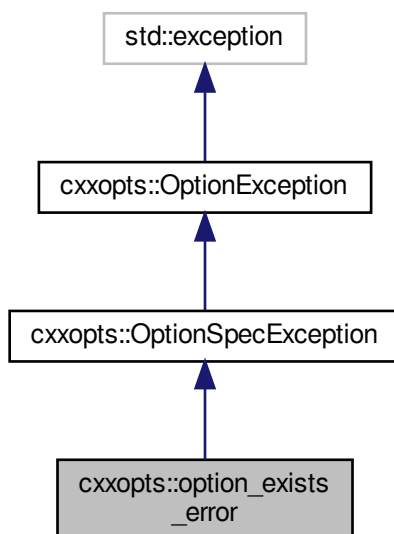
- include/cxxopts/cxxopts.hpp

4.17 cxxopts::option_exists_error Class Reference

Inheritance diagram for cxxopts::option_exists_error:



Collaboration diagram for cxxopts::option_exists_error:



Public Member Functions

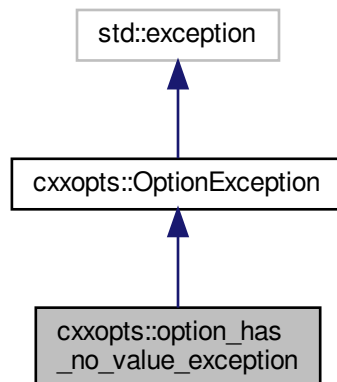
- **option_exists_error** (const std::string &option)

The documentation for this class was generated from the following file:

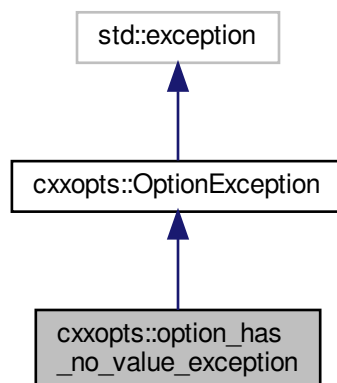
- include/cxxopts/cxxopts.hpp

4.18 cxxopts::option_has_no_value_exception Class Reference

Inheritance diagram for cxxopts::option_has_no_value_exception:



Collaboration diagram for cxxopts::option_has_no_value_exception:



Public Member Functions

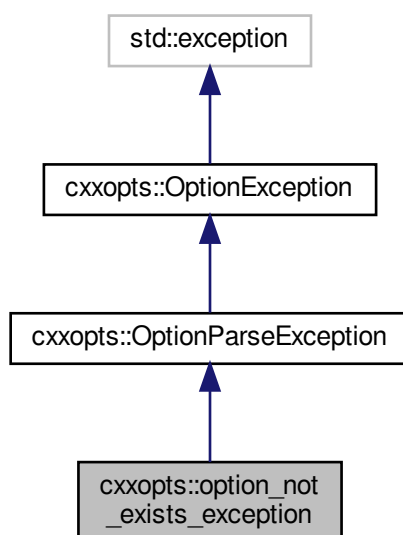
- **option_has_no_value_exception** (const std::string &option)

The documentation for this class was generated from the following file:

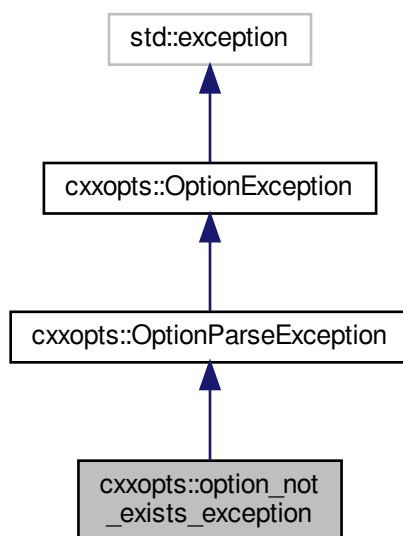
- include/cxxopts/cxxopts.hpp

4.19 cxxopts::option_not_exists_exception Class Reference

Inheritance diagram for cxxopts::option_not_exists_exception:



Collaboration diagram for `cxbopts::option_not_exists_exception`:



Public Member Functions

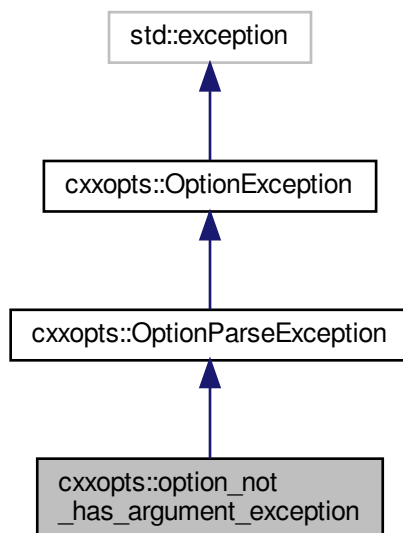
- **`option_not_exists_exception`** (const std::string &option)

The documentation for this class was generated from the following file:

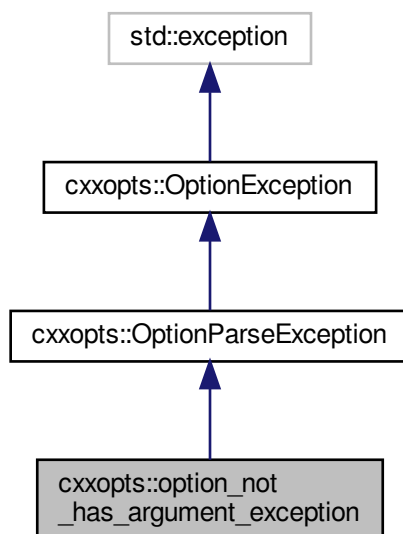
- `include/cxbopts/cxbopts.hpp`

4.20 cxxopts::option_not_has_argument_exception Class Reference

Inheritance diagram for cxxopts::option_not_has_argument_exception:



Collaboration diagram for cxxopts::option_not_has_argument_exception:



Public Member Functions

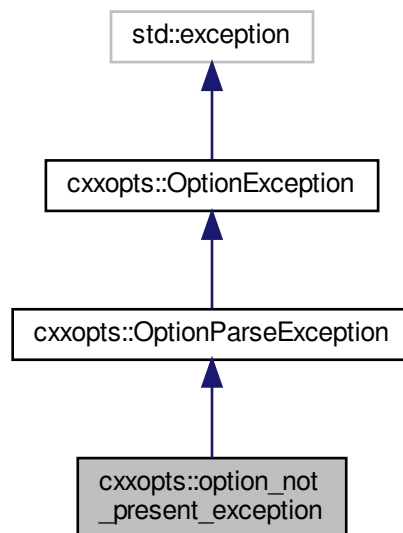
- **option_not_has_argument_exception** (const std::string &option, const std::string &arg)

The documentation for this class was generated from the following file:

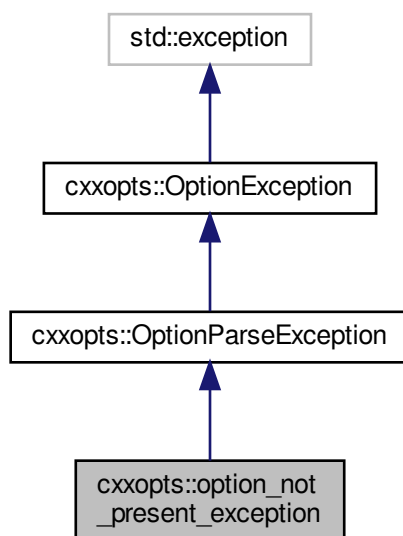
- include/cxxopts/cxxopts.hpp

4.21 cxxopts::option_not_present_exception Class Reference

Inheritance diagram for cxxopts::option_not_present_exception:



Collaboration diagram for cxxopts::option_not_present_exception:



Public Member Functions

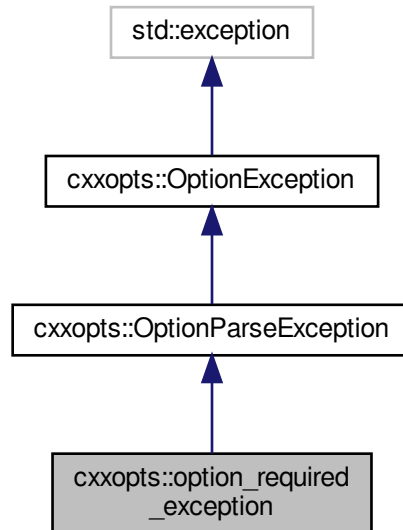
- **option_not_present_exception** (const std::string &option)

The documentation for this class was generated from the following file:

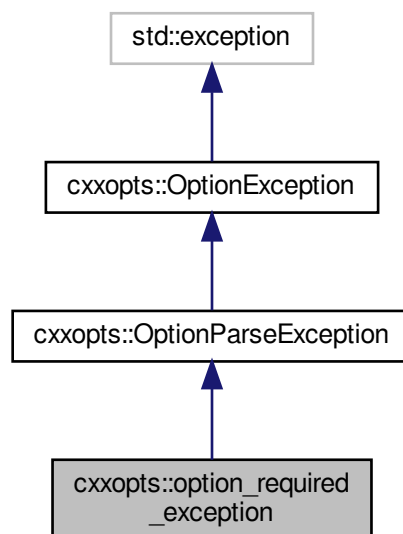
- include/cxxopts/cxxopts.hpp

4.22 cxxopts::option_required_exception Class Reference

Inheritance diagram for cxxopts::option_required_exception:



Collaboration diagram for cxxopts::option_required_exception:



Public Member Functions

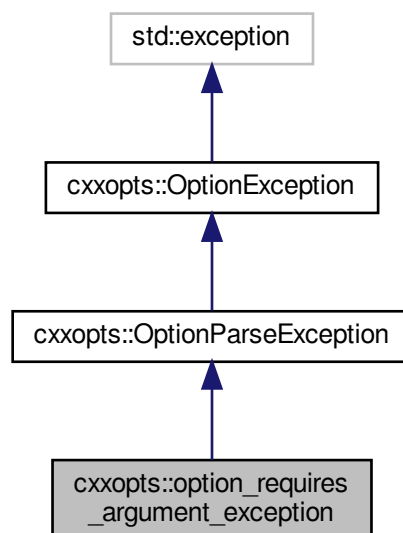
- **option_required_exception** (const std::string &option)

The documentation for this class was generated from the following file:

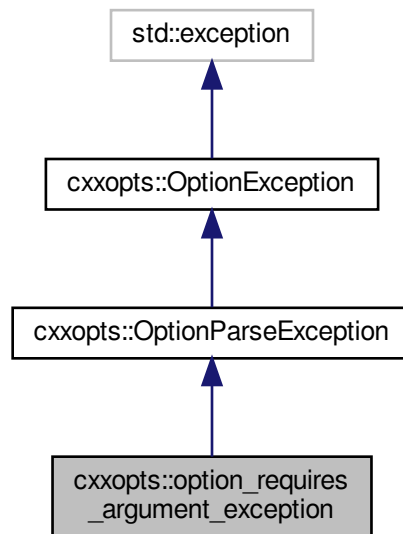
- include/cxxopts/cxxopts.hpp

4.23 cxxopts::option_requires_argument_exception Class Reference

Inheritance diagram for cxxopts::option_requires_argument_exception:



Collaboration diagram for `cxbopts::option_requires_argument_exception`:



Public Member Functions

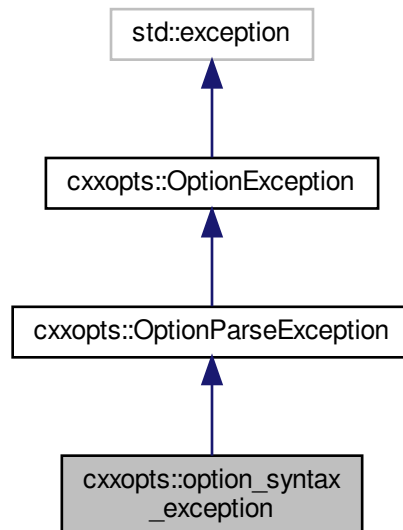
- **`option_requires_argument_exception`** (`const std::string &option`)

The documentation for this class was generated from the following file:

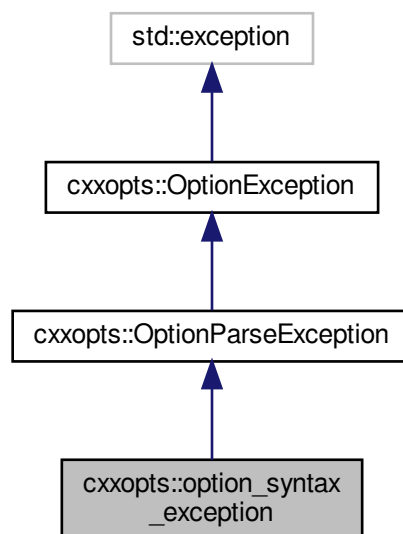
- `include/cxbopts/cxbopts.hpp`

4.24 cxxopts::option_syntax_exception Class Reference

Inheritance diagram for cxxopts::option_syntax_exception:



Collaboration diagram for cxxopts::option_syntax_exception:



Public Member Functions

- **option_syntax_exception** (const std::string &text)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.25 cxxopts::OptionAdder Class Reference

Public Member Functions

- **OptionAdder** ([Options](#) &options, std::string group)
- **OptionAdder & operator()** (const std::string &opts, const std::string &desc, const std::shared_ptr< const [Value](#) > &value=cxxopts::value< bool >(), std::string arg_help="")

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.26 cxxopts::OptionDetails Class Reference

Public Member Functions

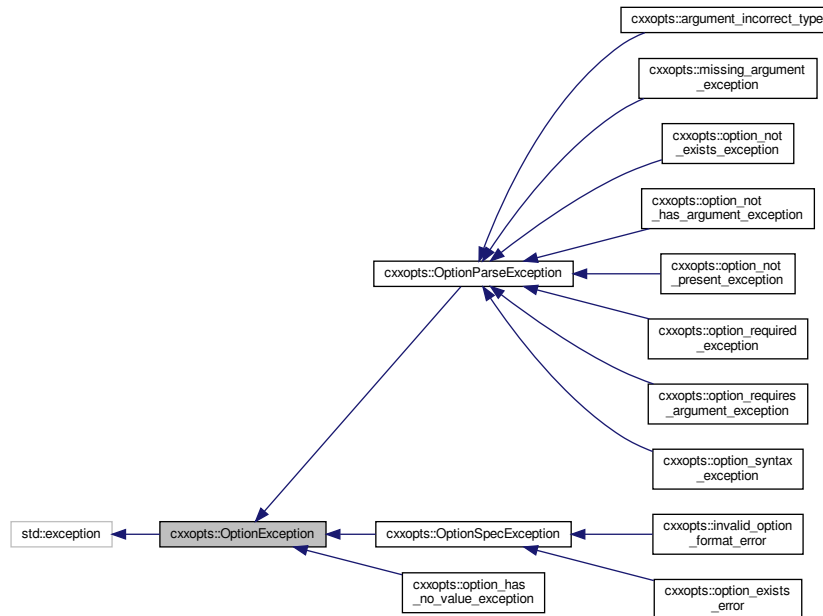
- **OptionDetails** (std::string short_, std::string long_, String desc, std::shared_ptr< const [Value](#) > val)
- **OptionDetails** (const [OptionDetails](#) &rhs)
- **OptionDetails** ([OptionDetails](#) &&rhs)=default
- CXXOPTS_NODISCARD const String & **description** () const
- CXXOPTS_NODISCARD const [Value](#) & **value** () const
- CXXOPTS_NODISCARD std::shared_ptr< [Value](#) > **make_storage** () const
- CXXOPTS_NODISCARD const std::string & **short_name** () const
- CXXOPTS_NODISCARD const std::string & **long_name** () const
- size_t **hash** () const

The documentation for this class was generated from the following file:

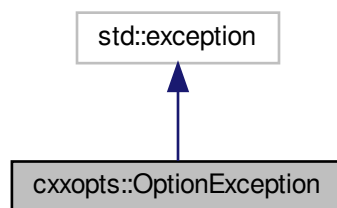
- include/cxxopts/cxxopts.hpp

4.27 cxxopts::OptionException Class Reference

Inheritance diagram for cxxopts::OptionException:



Collaboration diagram for cxxopts::OptionException:



Public Member Functions

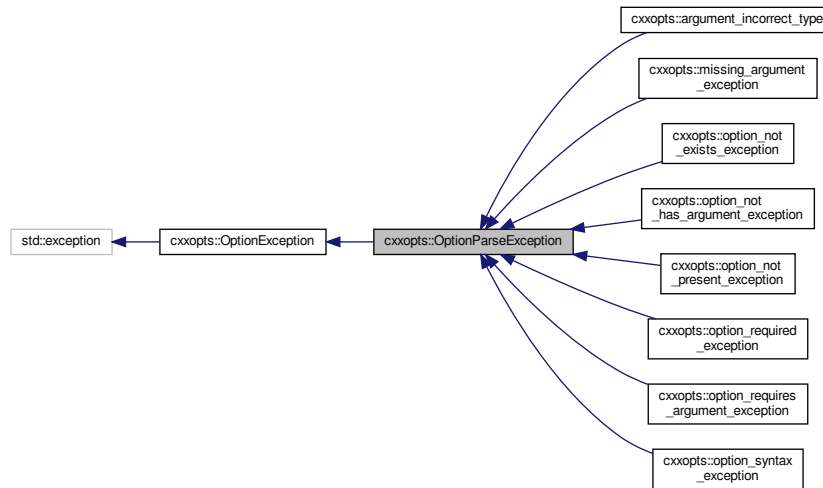
- **OptionException** (std::string message)
- CXXOPTS_NODISCARD const char * **what** () const noexcept override

The documentation for this class was generated from the following file:

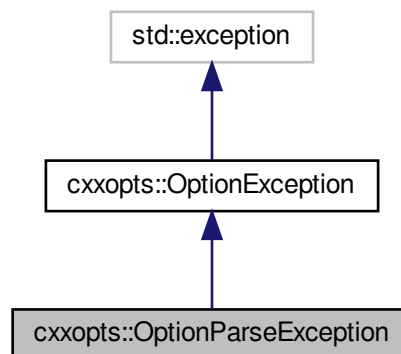
- include/cxxopts/cxxopts.hpp

4.28 cxxopts::OptionParseException Class Reference

Inheritance diagram for cxxopts::OptionParseException:



Collaboration diagram for cxxopts::OptionParseException:



Public Member Functions

- **OptionParseException** (const std::string &message)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.29 cxxopts::OptionParser Class Reference

Public Member Functions

- **OptionParser** (const OptionMap &options, const PositionalList &positional, bool allow_unrecognised)
- [ParseResult](#) **parse** (int argc, const char *const *argv)
- bool **consume_positional** (const std::string &a, PositionalListIterator &next)
- void **checked_parse_arg** (int argc, const char *const *argv, int ¤t, const std::shared_ptr<[OptionDetails](#) > &value, const std::string &name)
- void **add_to_option** (OptionMap::const_iterator iter, const std::string &option, const std::string &arg)
- void **parse_option** (const std::shared_ptr<[OptionDetails](#) > &value, const std::string &name, const std::string &arg="")
- void **parse_default** (const std::shared_ptr<[OptionDetails](#) > &details)
- void **parse_no_value** (const std::shared_ptr<[OptionDetails](#) > &details)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.30 cxxopts::Options Class Reference

Public Member Functions

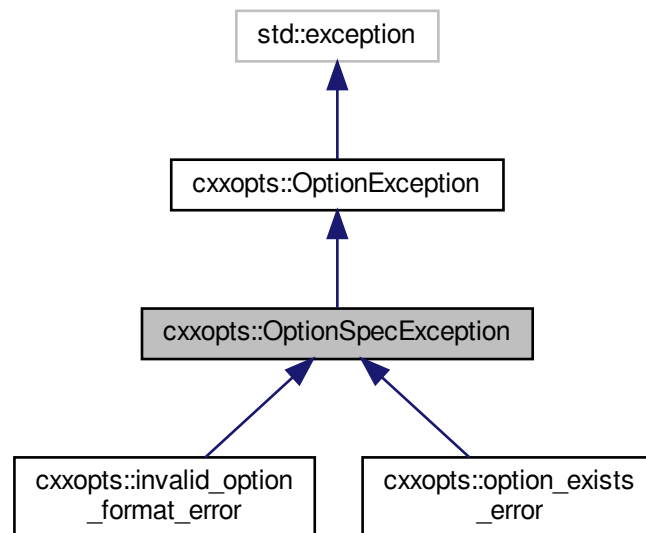
- **Options** (std::string program, std::string help_string="")
- [Options](#) & **positional_help** (std::string help_text)
- [Options](#) & **custom_help** (std::string help_text)
- [Options](#) & **show_positional_help** ()
- [Options](#) & **allow_unrecognised_options** ()
- [Options](#) & **set_width** (size_t width)
- [Options](#) & **set_tab_expansion** (bool expansion=true)
- [ParseResult](#) **parse** (int argc, const char *const *argv)
- [OptionAdder](#) **add_options** (std::string group="")
- void **add_options** (const std::string &group, std::initializer_list<[Option](#) > options)
- void **add_option** (const std::string &group, const [Option](#) &option)
- void **add_option** (const std::string &group, const std::string &s, const std::string &l, std::string desc, const std::shared_ptr< const [Value](#) > &value, std::string arg_help)
- void **parse_positional** (std::string option)
- void **parse_positional** (std::vector< std::string > options)
- void **parse_positional** (std::initializer_list< std::string > options)
- template<typename Iterator >
void **parse_positional** (Iterator begin, Iterator end)
- std::string **help** (const std::vector< std::string > &groups={}) const
- std::vector< std::string > **groups** () const
- const [HelpGroupDetails](#) & **group_help** (const std::string &group) const
- const std::string & **program** () const

The documentation for this class was generated from the following file:

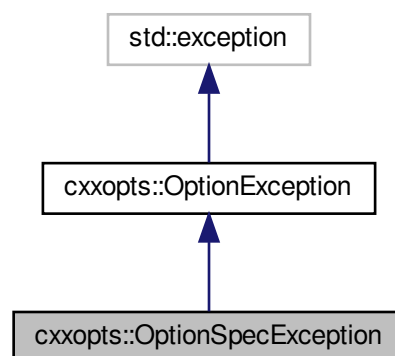
- include/cxxopts/cxxopts.hpp

4.31 cxxopts::OptionSpecException Class Reference

Inheritance diagram for cxxopts::OptionSpecException:



Collaboration diagram for cxxopts::OptionSpecException:



Public Member Functions

- **OptionSpecException** (const std::string &message)

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.32 cxxopts::OptionValue Class Reference

Public Member Functions

- void **parse** (const std::shared_ptr< const [OptionDetails](#) > &details, const std::string &text)
- void **parse_default** (const std::shared_ptr< const [OptionDetails](#) > &details)
- void **parse_no_value** (const std::shared_ptr< const [OptionDetails](#) > &details)
- CXXOPTS_NODISCARD size_t **count** () const noexcept
- CXXOPTS_NODISCARD bool **has_default** () const noexcept
- template<typename T >
const T & **as** () const

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.33 cxxopts::ParseResult Class Reference

Classes

- class [Iterator](#)

Public Member Functions

- **ParseResult** (const [ParseResult](#) &)=default
- **ParseResult** (NameHashMap &&keys, ParsedHashMap &&values, std::vector< [KeyValue](#) > sequential, std::vector< [KeyValue](#) > default_opts, std::vector< std::string > &&unmatched_args)
- [ParseResult](#) & **operator=** ([ParseResult](#) &)=default
- [ParseResult](#) & **operator=** (const [ParseResult](#) &)=default
- [Iterator](#) **begin** () const
- [Iterator](#) **end** () const
- size_t **count** (const std::string &o) const
- const [OptionValue](#) & **operator[]** (const std::string &option) const
- const std::vector< [KeyValue](#) > & **arguments** () const
- const std::vector< std::string > & **unmatched** () const
- const std::vector< [KeyValue](#) > & **defaults** () const
- const std::string **arguments_string** () const

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.34 Person< TSeq > Class Template Reference

Public Member Functions

- void **init** (epiworld_fast_uint baseline_status)
- void **add_tool** (int d, [Tool](#)< TSeq > tool)
- void **add_virus** ([Virus](#)< TSeq > *virus)
- void **rm_virus** ([Virus](#)< TSeq > *virus)
- epiworld_double **get_susceptibility_reduction** ([Virus](#)< TSeq > *v)
- epiworld_double **get_transmission_reduction** ([Virus](#)< TSeq > *v)
- epiworld_double **get_recovery_enhancer** ([Virus](#)< TSeq > *v)
- epiworld_double **get_death_reduction** ([Virus](#)< TSeq > *v)
- int **get_id** () const
- unsigned int **get_index** () const
- std::mt19937 * **get_rand_engine** ()
- [Model](#)< TSeq > * **get_model** ()
- [Virus](#)< TSeq > & **get_virus** (int i)
- [PersonViruses](#)< TSeq > & **get_viruses** ()
- [Tool](#)< TSeq > & **get_tool** (int i)
- [PersonTools](#)< TSeq > & **get_tools** ()
- void **mutate_variant** ()
- void **add_neighbor** ([Person](#)< TSeq > *p, bool check_source=true, bool check_target=true)
- std::vector< [Person](#)< TSeq > * > & **get_neighbors** ()
- void **update_status** ()
- void **update_status** (epiworld_fast_uint new_status)
- const epiworld_fast_uint & **get_status** () const
- const epiworld_fast_uint & **get_status_next** () const
- void **reset** ()
- void **set_update_susceptible** (UpdateFun< TSeq > fun)
- void **set_update_exposed** (UpdateFun< TSeq > fun)
- void **set_update_removed** (UpdateFun< TSeq > fun)
- bool **has_tool** (unsigned int t) const
- bool **has_tool** (std::string name) const
- bool **has_virus** (unsigned int t) const
- bool **has_virus** (std::string name) const
- bool **visited** () const
- void **toggle_visited** ()

Friends

- class **Model**< TSeq >
- class **Tool**< TSeq >
- class **Queue**< TSeq >

The documentation for this class was generated from the following files:

- include/epiworld/config.hpp
- include/epiworld/person-bones.hpp
- include/epiworld/person-meat.hpp

4.35 PersonTools< TSeq > Class Template Reference

List of tools available for the individual to.

```
#include <persontools-bones.hpp>
```

Public Member Functions

- void [add_tool](#) (int date, [Tool](#)< TSeq > tool)
@]
- epiworld_double [get_susceptibility_reduction](#) ([Virus](#)< TSeq > *v)
- epiworld_double [get_transmission_reduction](#) ([Virus](#)< TSeq > *v)
- epiworld_double [get_recovery_enhancer](#) ([Virus](#)< TSeq > *v)
- epiworld_double [get_death_reduction](#) ([Virus](#)< TSeq > *v)
- void [set_susceptibility_reduction_mixer](#) (MixerFun< TSeq > fun)
- void [set_transmission_reduction_mixer](#) (MixerFun< TSeq > fun)
- void [set_recovery_enhancer_mixer](#) (MixerFun< TSeq > fun)
- void [set_death_reduction_mixer](#) (MixerFun< TSeq > fun)
- size_t [size](#) () const
- [Tool](#)< TSeq > & [operator](#)() (int i)
- [Person](#)< TSeq > * [get_person](#) ()
- [Model](#)< TSeq > * [get_model](#) ()
- void [reset](#) ()
- bool [has_tool](#) (unsigned int t) const
- bool [has_tool](#) (std::string name) const

Friends

- class [Person](#)< TSeq >
- class [Model](#)< TSeq >

4.35.1 Detailed Description

```
template<typename TSeq = bool>
class PersonTools< TSeq >
```

List of tools available for the individual to.

Template Parameters

<i>TSeq</i>	
-------------	--

The documentation for this class was generated from the following files:

- include/epiworld/config.hpp
- include/epiworld/persontools-bones.hpp
- include/epiworld/persontools-meat.hpp

4.36 PersonViruses< TSeq > Class Template Reference

Set of viruses in host.

```
#include <personviruses-bones.hpp>
```

Public Member Functions

- void **add_virus** (epiworld_fast_uint new_status, [Virus](#)< TSeq > v)
- size_t **size** () const
- int **size_active** () const
- [Virus](#)< TSeq > & **operator()** (int i)
- void **mutate** ()
- void **reset** ()
- void **deactivate** ([Virus](#)< TSeq > &v)
- [Person](#)< TSeq > * **get_host** ()
- bool **has_virus** (unsigned int v) const
- bool **has_virus** (std::string vname) const

Friends

- class [Person](#)< TSeq >
- class [Model](#)< TSeq >

4.36.1 Detailed Description

```
template<typename TSeq = bool>
class PersonViruses< TSeq >
```

Set of viruses in host.

Template Parameters

<i>TSeq</i>	
-------------	--

The documentation for this class was generated from the following files:

- include/epiworld/person-bones.hpp
- include/epiworld/personviruses-bones.hpp
- include/epiworld/personviruses-meat.hpp

4.37 Progress Class Reference

A simple progress bar.

```
#include <progress.hpp>
```

Public Member Functions

- **Progress** (int n_, int width_)
- void **start** ()
- void **next** ()
- void **end** ()

4.37.1 Detailed Description

A simple progress bar.

The documentation for this class was generated from the following file:

- include/epiworld/progress.hpp

4.38 Queue< TSeq > Class Template Reference

Controls which agents are verified at each step.

```
#include <queue-bones.hpp>
```

Public Member Functions

- void **operator+=** ([Person](#)< TSeq > *p)
- void **operator-=** ([Person](#)< TSeq > *p)
- epiworld_fast_int **operator[]** (unsigned int i) const
- void **set_model** ([Model](#)< TSeq > *m)
- void **update** ()

4.38.1 Detailed Description

```
template<typename TSeq = bool>
class Queue< TSeq >
```

Controls which agents are verified at each step.

The idea is that only agents who are either in an infected state or have an infected neighbor should be checked. Otherwise it makes no sense (no chance to recover or capture the disease).

Template Parameters

<i>TSeq</i>	
-------------	--

The documentation for this class was generated from the following files:

- include/epiworld/model-bones.hpp
- include/epiworld/queue-bones.hpp

4.39 RandGraph Class Reference

Public Member Functions

- **RandGraph** (int N_)
- void **init** (int s)
- void **set_rand_engine** (std::mt19937 &e)
- epiworld_double **runif** ()

The documentation for this class was generated from the following file:

- include/epiworld/random_graph.hpp

4.40 cxxopts::values::detail::SignedCheck< T, B > Struct Template Reference

The documentation for this struct was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.41 cxxopts::values::detail::SignedCheck< T, false > Struct Template Reference

Public Member Functions

- template<typename U >
void **operator()** (bool, U, const std::string &) const

The documentation for this struct was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.42 cxxopts::values::detail::SignedCheck< T, true > Struct Template Reference

Public Member Functions

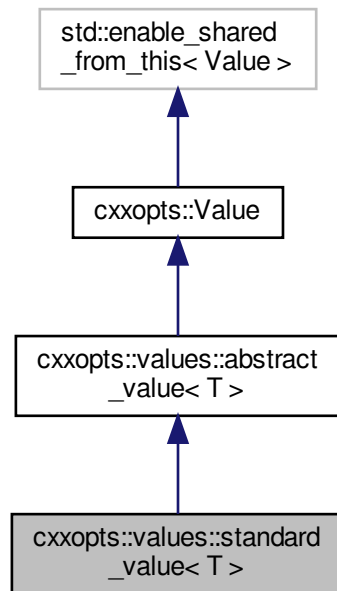
- template<typename U >
void **operator()** (bool negative, U u, const std::string &text)

The documentation for this struct was generated from the following file:

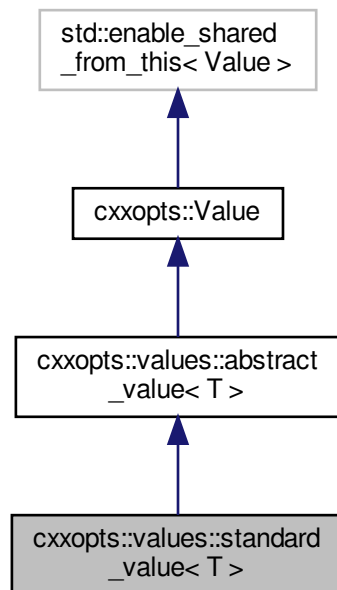
- include/cxxopts/cxxopts.hpp

4.43 cxxopts::values::standard_value< T > Class Template Reference

Inheritance diagram for cxxopts::values::standard_value< T >:



Collaboration diagram for `cxxopts::values::standard_value< T >`:



Public Member Functions

- CXXOPTS_NODISCARD `std::shared_ptr< Value >` **clone** () const override

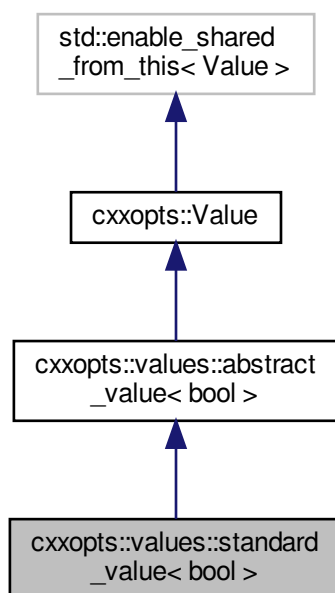
Additional Inherited Members

The documentation for this class was generated from the following file:

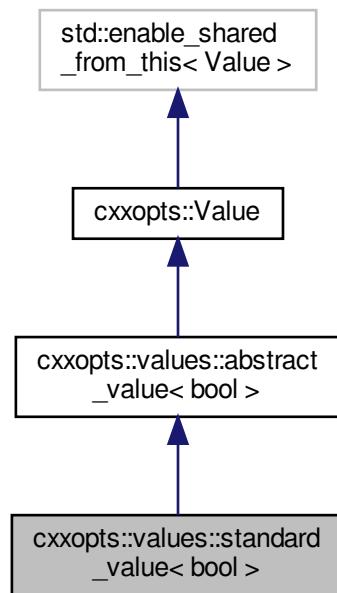
- `include/cxxopts/cxxopts.hpp`

4.44 cxxopts::values::standard_value< bool > Class Reference

Inheritance diagram for cxxopts::values::standard_value< bool >:



Collaboration diagram for `cxxopts::values::standard_value< bool >`:



Public Member Functions

- **standard_value** (bool *b)
- `std::shared_ptr< Value > clone ()` const override

Additional Inherited Members

The documentation for this class was generated from the following file:

- `include/cxxopts/cxxopts.hpp`

4.45 Tool< TSeq > Class Template Reference

Tools for defending the host against the virus.

```
#include <tools-bones.hpp>
```

Public Member Functions

- **Tool** (std::string name="unknown tool")
- void **set_sequence** (TSeq d)
- void **set_sequence_unique** (TSeq d)
- void **set_sequence** (std::shared_ptr< TSeq > d)
- std::shared_ptr< TSeq > **get_sequence** ()
- TSeq & **get_sequence_unique** ()
- epiworld_double **get_susceptibility_reduction** (Virus< TSeq > *v)
Get and set the tool functions.
- epiworld_double **get_transmission_reduction** (Virus< TSeq > *v)
- epiworld_double **get_recovery_enhancer** (Virus< TSeq > *v)
- epiworld_double **get_death_reduction** (Virus< TSeq > *v)
- void **set_susceptibility_reduction_fun** (ToolFun< TSeq > fun)
- void **set_transmission_reduction_fun** (ToolFun< TSeq > fun)
- void **set_recovery_enhancer_fun** (ToolFun< TSeq > fun)
- void **set_death_reduction_fun** (ToolFun< TSeq > fun)
- void **set_susceptibility_reduction** (epiworld_double *prob)
- void **set_transmission_reduction** (epiworld_double *prob)
- void **set_recovery_enhancer** (epiworld_double *prob)
- void **set_death_reduction** (epiworld_double *prob)
- void **set_susceptibility_reduction** (epiworld_double prob)
- void **set_transmission_reduction** (epiworld_double prob)
- void **set_recovery_enhancer** (epiworld_double prob)
- void **set_death_reduction** (epiworld_double prob)
- void **set_name** (std::string name)
- @]
- std::string **get_name** () const
- Person< TSeq > * **get_person** ()
- unsigned int **get_id** () const

Friends

- class **PersonTools**< TSeq >
- class **Person**< TSeq >
- class **Model**< TSeq >

4.45.1 Detailed Description

```
template<typename TSeq = bool>
class Tool< TSeq >
```

Tools for defending the host against the virus.

Template Parameters

<i>TSeq</i>	Type of sequence
-------------	------------------

4.45.2 Member Function Documentation

4.45.2.1 `get_susceptibility_reduction()`

```
template<typename TSeq >
epiworld_double Tool< TSeq >::get_susceptibility_reduction (
    Virus< TSeq > * v ) [inline]
```

Get and set the tool functions.

Parameters

<i>v</i>	The virus over which to operate
<i>fun</i>	the function to be used

Returns

`epiworld_double @[]`

The documentation for this class was generated from the following files:

- `include/epiworld/config.hpp`
- `include/epiworld/tools-bones.hpp`
- `include/epiworld/tools-meat.hpp`

4.46 `cxxopts::values::type_is_container< T >` Struct Template Reference

Static Public Attributes

- static constexpr bool **value** = false

The documentation for this struct was generated from the following file:

- `include/cxxopts/cxxopts.hpp`

4.47 `cxxopts::values::type_is_container< std::vector< T > >` Struct Template Reference

Static Public Attributes

- static constexpr bool **value** = true

The documentation for this struct was generated from the following file:

- `include/cxxopts/cxxopts.hpp`

4.48 UserData< TSeq > Class Template Reference

Public Member Functions

- **UserData** (std::vector< std::string > names)
- void **add** (std::vector< epiworld_double > x)
- void **add** (unsigned int j, epiworld_double x)
- epiworld_double & **operator()** (unsigned int i, unsigned int j)
- epiworld_double & **operator()** (unsigned int i, std::string name)
- std::vector< std::string > & **get_names** ()
- std::vector< int > & **get_dates** ()
- std::vector< epiworld_double > & **get_data** ()
- void **get_all** (std::vector< std::string > *names=nullptr, std::vector< int > *date=nullptr, std::vector< epiworld_double > *data=nullptr)
- unsigned int **nrow** () const
- unsigned int **ncol** () const
- void **write** (std::string fn)
- void **print** () const

Friends

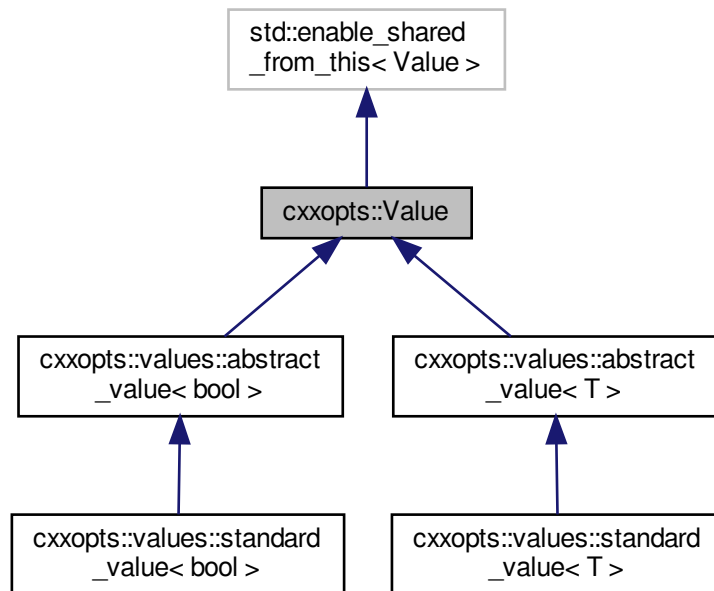
- class **Model**< TSeq >
- class **DataBase**< TSeq >

The documentation for this class was generated from the following files:

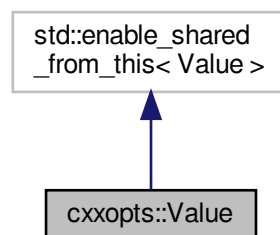
- include/epiworld/database-bones.hpp
- include/epiworld/userdata-bones.hpp
- include/epiworld/userdata-meat.hpp

4.49 cxxopts::Value Class Reference

Inheritance diagram for cxxopts::Value:



Collaboration diagram for cxxopts::Value:



Public Member Functions

- virtual `std::shared_ptr< Value > clone ()` const =0
- virtual void **parse** (const `std::string` &text) const =0
- virtual void **parse** () const =0
- virtual bool **has_default** () const =0

- virtual bool **is_container** () const =0
- virtual bool **has_implicit** () const =0
- virtual std::string **get_default_value** () const =0
- virtual std::string **get_implicit_value** () const =0
- virtual std::shared_ptr< [Value](#) > **default_value** (const std::string &value)=0
- virtual std::shared_ptr< [Value](#) > **implicit_value** (const std::string &value)=0
- virtual std::shared_ptr< [Value](#) > **no_implicit_value** ()=0
- virtual bool **is_boolean** () const =0

The documentation for this class was generated from the following file:

- include/cxxopts/cxxopts.hpp

4.50 vecHasher< T > Struct Template Reference

Public Member Functions

- std::size_t **operator()** (std::vector< T > const &dat) const noexcept

The documentation for this struct was generated from the following file:

- include/epiworld/misc.hpp

4.51 Virus< TSeq > Class Template Reference

[Virus](#).

```
#include <virus-bones.hpp>
```

Public Member Functions

- **Virus** (std::string name="unknown virus")
- void **mutate** ()
- void **set_mutation** (MutFun< TSeq > fun)
- const TSeq * **get_sequence** ()
- void **set_sequence** (TSeq sequence)
- [Person](#)< TSeq > * **get_host** ()
- [Model](#)< TSeq > * **get_model** ()
- void **set_date** (int d)
- int **get_date** () const
- void **set_id** (int idx)
- int **get_id** () const
- bool **is_active** () const
- void **deactivate** ()
- epiworld_double [get_prob_infecting](#) ()
- *Get and set the tool functions.*
- epiworld_double **get_prob_recovery** ()

- `epiworld_double` **get_prob_death** ()
 - `void` **post_recovery** ()
 - `void` **set_post_recovery** (PostRecoveryFun< TSeq > fun)
 - `void` **set_post_immunity** (epiworld_double prob)
 - `void` **set_post_immunity** (epiworld_double *prob)
 - `void` **set_prob_infecting_fun** (VirusFun< TSeq > fun)
 - `void` **set_prob_recovery_fun** (VirusFun< TSeq > fun)
 - `void` **set_prob_death_fun** (VirusFun< TSeq > fun)
 - `void` **set_prob_infecting** (epiworld_double *prob)
 - `void` **set_prob_recovery** (epiworld_double *prob)
 - `void` **set_prob_death** (epiworld_double *prob)
 - `void` **set_prob_infecting** (epiworld_double prob)
 - `void` **set_prob_recovery** (epiworld_double prob)
 - `void` **set_prob_death** (epiworld_double prob)
 - `void` **set_name** (std::string name)
- @]
- `std::string` **get_name** () const
 - `std::vector< epiworld_double >` & **get_data** ()

Friends

- class **Person**< TSeq >
- class **Model**< TSeq >
- class **PersonViruses**< TSeq >
- class **DataBase**< TSeq >

4.51.1 Detailed Description

```
template<typename TSeq = bool>
class Virus< TSeq >
```

[Virus](#).

Template Parameters

<i>TSeq</i>	
-------------	--

Raw transmissibility of a virus should be a function of its genetic sequence. Nonetheless, transmissibility can be reduced as a result of having one or more tools to fight the virus. Because of this, transmissibility should be a function of the host.

4.51.2 Member Function Documentation

4.51.2.1 get_prob_infecting()

```
template<typename TSeq >
epiworld_double Virus< TSeq >::get_prob_infecting [inline]
```

Get and set the tool functions.

Parameters

<i>v</i>	The virus over which to operate
<i>fun</i>	the function to be used

Returns

epiworld_double @[

The documentation for this class was generated from the following files:

- include/epiworld/config.hpp
- include/epiworld/virus-bones.hpp
- include/epiworld/virus-meat.hpp

Index

`add_param`
 `Model< TSeq >`, [23](#)
`add_status_susceptible`
 `Model< TSeq >`, [24](#)
`AdjList`, [9](#)
 `AdjList`, [9](#)

`cxxopts::argument_incorrect_type`, [10](#)
`cxxopts::HelpGroupDetails`, [13](#)
`cxxopts::HelpOptionDetails`, [14](#)
`cxxopts::invalid_option_format_error`, [15](#)
`cxxopts::KeyValue`, [16](#)
`cxxopts::missing_argument_exception`, [19](#)
`cxxopts::Option`, [28](#)
`cxxopts::option_exists_error`, [29](#)
`cxxopts::option_has_no_value_exception`, [30](#)
`cxxopts::option_not_exists_exception`, [31](#)
`cxxopts::option_not_has_argument_exception`, [33](#)
`cxxopts::option_not_present_exception`, [34](#)
`cxxopts::option_required_exception`, [36](#)
`cxxopts::option_requires_argument_exception`, [37](#)
`cxxopts::option_syntax_exception`, [39](#)
`cxxopts::OptionAdder`, [40](#)
`cxxopts::OptionDetails`, [40](#)
`cxxopts::OptionException`, [41](#)
`cxxopts::OptionParseException`, [42](#)
`cxxopts::OptionParser`, [43](#)
`cxxopts::Options`, [43](#)
`cxxopts::OptionSpecException`, [44](#)
`cxxopts::OptionValue`, [45](#)
`cxxopts::ParseResult`, [45](#)
`cxxopts::ParseResult::Iterator`, [16](#)
`cxxopts::Value`, [58](#)
`cxxopts::values::abstract_value< T >`, [7](#)
`cxxopts::values::detail::SignedCheck< T, B >`, [50](#)
`cxxopts::values::detail::SignedCheck< T, false >`, [50](#)
`cxxopts::values::detail::SignedCheck< T, true >`, [50](#)
`cxxopts::values::parser_tool::ArguDesc`, [10](#)
`cxxopts::values::parser_tool::IntegerDesc`, [14](#)
`cxxopts::values::standard_value< bool >`, [53](#)
`cxxopts::values::standard_value< T >`, [51](#)
`cxxopts::values::type_is_container< std::vector< T > >`, [56](#)
`cxxopts::values::type_is_container< T >`, [56](#)

`DataBase< TSeq >`, [11](#)
 `get_today_total`, [12](#)
 `record_variant`, [13](#)

`get_prob_infecting`
 `Virus< TSeq >`, [60](#)
`get_susceptibility_reduction`
 `Tool< TSeq >`, [56](#)
`get_today_total`
 `DataBase< TSeq >`, [12](#)

`init`
 `Model< TSeq >`, [24](#)

`LFMCMC< TData >`, [17](#)
 `set_rand_engine`, [18](#)
`Location< TSeq >`, [18](#)

`Model< TSeq >`, [20](#)
 `add_param`, [23](#)
 `add_status_susceptible`, [24](#)
 `init`, [24](#)
 `pop_from_adjlist`, [24](#)
 `reset`, [25](#)
 `reset_status_codes`, [25](#)
 `set_backup`, [26](#)
 `set_rand_engine`, [26](#)
 `set_rewire_fun`, [26](#)
 `set_user_data`, [27](#)
 `write_data`, [27](#)
 `write_edgelist`, [27](#)

`Person< TSeq >`, [46](#)
`PersonTools< TSeq >`, [47](#)
`PersonViruses< TSeq >`, [48](#)
`pop_from_adjlist`
 `Model< TSeq >`, [24](#)
`Progress`, [48](#)

`Queue< TSeq >`, [49](#)

`RandGraph`, [50](#)
`record_variant`
 `DataBase< TSeq >`, [13](#)
`reset`
 `Model< TSeq >`, [25](#)
`reset_status_codes`
 `Model< TSeq >`, [25](#)

`set_backup`
 `Model< TSeq >`, [26](#)
`set_rand_engine`
 `LFMCMC< TData >`, [18](#)
 `Model< TSeq >`, [26](#)
`set_rewire_fun`
 `Model< TSeq >`, [26](#)

set_user_data
 Model< TSeq >, [27](#)

Tool< TSeq >, [54](#)
 get_susceptibility_reduction, [56](#)

UserData< TSeq >, [57](#)

vecHasher< T >, [59](#)

Virus< TSeq >, [59](#)
 get_prob_infecting, [60](#)

write_data
 Model< TSeq >, [27](#)

write_edgelist
 Model< TSeq >, [27](#)